AN617

Fixed Point Routines

Author: *Frank J. Testa*
Design Consultant

INTRODUCTION

This application note presents an implementation of the following fixed point math routines for the PIC16/17 microcontroller families:

- Multiplication
- Division

Routines for the PIC16/17 families are provided in a variety of fixed point formats, including both unsigned and signed two's complement arithmetic.

FIXED POINT ARITHMETIC

Unsigned fixed point binary numbers A, can be represented in the form

$$A = \sum_{k=0}^{n-1} a(k) \bullet 2^{k-r} = 2^{-r} \sum_{k=0}^{n-1} a(k) \bullet 2k$$

where n is the number of bits, a(k) is the kth bit with a(0) = LSb, and r indicates the location of the radix point. For example, in the case where A is an integer, r = 0 and when A is a fraction less than one, r = n. The value of r only affects the interpretation of the numbers in a fixed point calculation, with the actual binary representation of the numbers independent of the value of r. Factoring out of the above sum, it simply locates the radix point of the representation and is analogous to an exponent in a floating point system. Using the notation Qi.j to denote a fixed point binary number with i bits to the left of the radix point and j to the right, the above n-bit format is in Qn-r.r. With care, fixed point calculations can be performed on operands in different Q formats. Although the radix point must be aligned for addition or subtraction, multiplication provides an illustrative example of the simple interpretive nature of r. Consider the unsigned product of a Q20.4 number with a Q8.8. After calling the appropriate unsigned 24•16 bit multiply for these fixed point arguments, the 40-bit fixed point result is in Q28.12, where the arguments of the Q notation are summed respectively. Similar arguments can be made for two's complement arithmetic, where the negative representation of a positive number is obtained by reversing the value of each bit and incrementing the result by one. Producing a unique representation of zero, and covering the range -2^{n-1} to $2^{n-1} - 1$, this is more easily applied in addition and subtraction operations and is therefore the most commonly used method of representing positive and negative numbers in fixed point arithmetic.

The above analysis in Q notation can be employed to build dedicated fixed point algorithms, leading to improved performance over floating point methods in cases where the size of the arguments required for the range and precision of the calculations is not large enough to destroy gains made by fixed point methods.

FIXED POINT FORMATS

The fixed point library routines supports 8-,16-, 24- and 32-bit formats in the following combinations:

Division Library Names	Format	Multiplication Library Names	Format
PIC16C5X/PIC16CXX Routines			
FXD0808S, FXD0808U, FXD0807U, FXD0707U	8/8	FXM0808S, FXM0808U, FXM0807U	8•8
FXD1608S, FXD1608U, FXD1607U, FXD1507U	16/8	FXM1608S, FXM1608U, FXM1607U, FXM1507U	16•8
FXD1616S, FXD1616U, FXD1515U	16/16	FXM1616S, FXM1616U, FXM1515U	16•16
FXD2416S, FXD2416U, FXD2315U	24/16	FXM2416S, FXM2416U, FXM2315U	24•16
FXD2424S, FXD2424U, FXD2323U	24/24	FXM2424S, FXM2424U, FXM2323U	24•24
FXD3216S, FXD3216U, FXD3115U	32/16	FXM3216S, FXM3216U, FXM3115U	32•16
FXD3224S, FXD3224U, FXD3123U	32/24	FXM3224S, FXM3224U, FXM3123U	32•24
FXD3232S, FXD3232U, FXD3131U	32/32	FXM3232S, FXM3232U, FXM3131U	32•32
PIC17CXX Functions			
FXD0808S, FXD0808U, FXD0807U, FXD0707U	8/8	FXM0808S, FXM0808U, FXM0807U	8•8
FXD1608S, FXD1608U, FXD1607U, FXD1507U	16/8	FXM1608S, FXM1608U, FXM1507U	16•8
FXD1616S, FXD1616U, FXD1615U, FXD1515U	16/16	FXM1616S, FXM1616U, FXM1515U	16•16
FXD2416S, FXD2416U, FXD2415U, FXD2315U	24/16	FXM2416S, FXM2416U, FXM2315U	24•16
FXD2424S, FXD2424U, FXD2423U, FXD2323U	24/24	FXM2424S, FXM2424U, FXM2323U	24•24
FXD3216S, FXD3216U, FXD3215U, FXD3115U	32/16	FXM3216S, FXM3216U, FXM3115U	32•16
FXD3224S, FXD3224U, FXD3223U, FXD3123U	32/24	FXM3224S, FXM3224U, FXM3123U	32•24
FXD3232S, FXD3232U, FXD3231U, FXD3131U	32/32	FXM3232S, FXM3232U, FXM3131U	32•32

Note: U - unsigned math operation, S - signed math operation

These general format combinations are implemented in both signed and unsigned versions. Additional unsigned routines are implemented with arguments reduced by one bit to accommodate the case of operations on signed numbers, with arguments known to be nonnegative, thereby, resulting in some performance improvement.

DATA RAM REQUIREMENTS

The following contiguous data ram locations are used by the library:

```
ACCB7    =   REMB3   =   AEXP   =   EXP          AARG and ACC exponent
ACCB6    =   REMB2   =   BEXP                    BARG exponent
ACCB5    =   REMB1
ACCB4    =   REMB0                               remainder
ACCB3    =   AARGB3
ACCB2    =   AARGB2
ACCB1    =   AARGB1
ACCB0    =   AARGB0   =  ACC                     AARG and ACC
SIGN                                             sign in MSB
FPFLAGS                                          exception flags and option bits
BARGB3
BARGB2
BARGB1
BARGB0                                           BARG
TEMPB3
TEMPB2
TEMPB1
TEMPB0                                           temporary storage
```

These definitions are identical with those used by the IEEE 754 compliant floating point library[6], AN575.

USAGE

Multiplication assumes the multiplicand in AARG, multiplier in BARG, and produces the result in ACC. Division assumes a dividend in AARG, divisor in BARG, and quotient in ACC with remainder in REM.

ADDITION/SUBTRACTION

Because of the generally trivial nature of addition and subtraction, the call and return overhead outweighs the need for explicit routines and so they are not included in the library. However, the PIC16C5X/PIC16CXX families do not have an add with carry or subtract with borrow instruction, leading to subtleties regarding production of a correct carry-out in a multiple byte add or subtract. In the case of a two byte add or subtract, the most elegant solution to these difficulties, requiring 6 cycles, appears to be given by the following code in Example 1.

EXAMPLE 1: TWO BYTE ADDITION/SUBTRACTION ROUTINES

```
ADD     MOVF    AARGB1,W
        ADDWF   BARGB1

        MOVF    AARGB0,W
        BTFSC   _C
        INCFSZ  AARGB0,W
        ADDWF   BARGB0

SUB     MOVF     AARGB1,W
        SUBWF   BARGB1

        MOVF    AARGB0,W
        BTFSS   _C
        INCFSZ  AARGB0,W
        SUBWF   BARGB0
```

The four instructions after the initial add/subtract, can be easily concatenated for operations involving more than two bytes. Because addition and subtraction are required in standard algorithms for multiplication and division, these issues permeate the implementation of both fixed and floating point algorithms for the PIC16C5X/PIC16CXX families.

MULTIPLICATION

The fixed point multiply routine FXPMxxyy, takes an xx-bit multiplicand in AARG, a yy-bit multiplier in BARG and returns the (xx+yy)-bit product in ACC. The implementation uses a standard sequential add-shift algorithm, negating both factors if BARG < 0, to produce the positive multiplier required by the method. Analogous to simple longhand binary multiplication, the multiplier bits are sequentially tested, with one indicating an add-shift and zero simply a shift. The shift

is required to align the partial product for the next possible add[1]. Several examples are shown in Example 2.

EXAMPLE 2: MULTIPLICATION EXAMPLES

FXM2416S(0xC11682,0x608B)

= FXM2416S(-4123006,24715)

= 0xE84647F896

= -101900093290

FXM1616U(0x0458,0x822C)

= FXM1616U(1112,33324)

= 0x02356F20

= 37056288

TABLE 1: PIC17CXX FIXED POINT MULTIPLY PERFORMANCE DATA: APPENDIX E

Routine	Max Cyc	Min Cyc	PM	DM	Page
FXM0808S	50/53	26	65	5	2-375
FXM0808U	39	23	53	3	2-375
FXM0707U	37	21	49	3	2-375
FXM1608S	74/79	35	100	6	2-376
FXM1608U	75	24	90	6	2-376
FXM1507U	69	24	82	6	2-376
FXM1616S	168/175	24	197	8	2-377
FXM1616U	156	41	191	8	2-377
FXM1515U	150	39	185	8	2-377
FXM2416S	213/223	43	253	10	2-378
FXM2416U	203	43	239	10	2-378
FXM2315U	194	41	229	10	2-378
FXM2424S	334/346	60	392	12	2-379
FXM2424U	316	60	364	12	2-379
FXM2323U	308	58	354	12	2-380
FXM3216S	258/270	46	301	13	2-380
FXM3216U	265	44	298	13	2-381
FXM3115U	254	42	285	13	2-381
FXM3224S	403/417	62	464	15	2-381
FXM3224U	410	61	459	15	2-382
FXM3123U	399	59	446	15	2-382
FXM3232S	556/572	79	635	16	2-383
FXM3232U	563	78	628	16	2-383
FXM3131U	543	76	606	16	2-384

Legend: PM - Program memory, DM - Data Memory

TABLE 2: PIC16C5X/PIC16CXX FIXED POINT MULTIPLY PERFORMANCE DATA: APPENDIX C

Routine	Max Cyc	Min Cyc	PM	DM	Page
FXM0808S	79/82	55	33	5	2-252
FXM0808U	73	54	21	4	2-252
FXM0707U	67	48	23	4	2-252
FXM1608S	122/128	55	44	7	2-246
FXM1608U	126	59	31	7	2-247
FXM1507U	114	52	35	7	2-247
FXM1616S	260/269	105	74	9	2-240
FXM1616U	256	107	58	9	2-241
FXM1515U	244	103	63	9	2-241
FXM2416S	334/346	108	92	12	2-231
FXM2416U	334	110	70	12	2-232
FXM2315U	319	104	76	12	2-232
FXM2424S	520/535	157	126	13	2-221
FXM2424U	512	159	98	13	2-222
FXM2323U	497	154	107	13	2-222
FXM3216S	408/423	111	98	9	2-208
FXM3216U	412	114	84	9	2-208
FXM3115U	392	106	91	9	2-209
FXM3224S	634/652	160	152	15	2-196
FXM3224U	630	162	151	15	2-197
FXM3123U	610	157	129	15	2-197
FXM3232S	868/889	207	189	17	2-181
FXM3232U	856	209	168	17	2-182
FXM3131U	836	204	168	17	2-182

Legend: PM - Program memory, DM - Data Memory

DIVISION

The fixed point divide routine FXPDxxyy, takes an xx-bit dividend in AARG, a yy-bit divisor in BARG and returns the xx-bit quotient in ACC and yy-bit remainder in REM. Unlike multiplication, division is not deterministic, requiring a trial-and-error sequential shift and subtract process. Binary division is less complicated than decimal division because the possible quotient digits are only zero or one. If the divisor is less than the partial remainder, the corresponding quotient bit is set to one followed by a shift and subtract. Otherwise, the divisor is greater than the partial remainder, the quotient bit is set to zero and only a shift is performed. The intermediate partial remainder may be restored at each stage as in restoring division, or corrected at the end as in nonrestoring division. Implementation dependent trade-offs between worst case versus average performance affect the choice between these two approaches, and therefore, macros for each method are provided.

Note:	A test for divide by zero exception is not performed and must be explicitly provided by the user.

The results of the division process for AARG/BARG, satisfy the relation

AARG = BARG • QUOTIENT + REMAINDER,

where the remainder has the same sign as the quotient, and represents the fraction of the result in units of the denominator BARG. Some simple examples are given in Example 3.

EXAMPLE 3: DIVISION EXAMPLES

FXD1608S(0xC116,0x60) = 0xFF59, 0xB6

FXD1616U(0x9543,0x4AA1) = 0x0002, 0x0001

TABLE 3: PIC17CXX FIXED POINT DIVIDE PERFORMANCE DATA: APPENDIX F

Routine	Max Cyc	Min Cyc	PM	DM	Page
FXD0808S	71/77	71/77	77	4	2-449
FXD0808U	75	67	74	3	2-449
FXD0807U	66	66	65	3	2-450
FXD0707U	61	61	60	3	2-450
FXD1608S	135/146	135/146	146	5	2-448
FXD1608U	196	156	195	4	2-448
FXD1607U	130	130	129	4	2-448
FXD1507U	125	125	124	4	2-449
FXD1616S	201/214	187/200	241	7	2-446
FXD1616U	244	180	243	6	2-447
FXD1615U	197	182	216	6	2-447
FXD1515U	191	177	218	6	2-447
FXD2416S	297/314	272/289	353	8	2-444
FXD2416U	365	339	453	8	2-445
FXD2415U	294	268	339	8	2-445
FXD2315U	287	262	330	8	2-446
FXD2424S	371/390	344/363	482	10	2-475
FXD2424U	440	412	577	10	2-476
FXD2423U	369	341	460	9	2-476
FXD2323U	361	334	448	9	2-476
FXD3216S	393/414	363/384	476	9	2-473
FXD3216U	485	459	608	9	2-474
FXD3215U	390	359	451	8	2-474
FXD3115U	383	353	442	8	2-475
FXD3224S	491/514	456/479	639	11	2-419
FXD3224U	584	548	769	11	2-420
FXD3223U	489	453	612	10	2-421
FXD3123U	481	446	600	10	2-421
FXD3232S	589/614	552/577	800	13	2-418
FXD3232U	683	645	930	13	2-419
FXD3231U	588	550	773	12	2-419
FXD3131U	579	542	758	12	2-419

Legend: PM - Program memory, DM - Data Memory

TABLE 4: PIC16C5X/PIC16CXX FIXED POINT DIVIDE PERFORMANCE DATA: APPENDIX D

Routine	Max Cyc	Min Cyc	PM	DM	Page
FXD0808S	90/96	90/96	41	5	2-344
FXD0808U	100	92	15	4	2-345
FXD0807U	88	88	21	4	2-345
FXD0707U	80	80	44	4	2-345
FXD1608S	176/188	176/188	67	6	2-338
FXD1608U	294	230	41	7	2-339
FXD1607U	174	174	41	5	2-339
FXD1507U	166	166	44	5	2-340
FXD1616S	304/319	269/284	74	8	2-330
FXD1616U	373	277	27	7	2-330
FXD1515U	294	274	45	7	2-331
FXD2416S	417/438	389/410	140	8	2-321
FXD2416U	529	501	172	8	2-322
FXD2315U	407	379	120	7	2-322
FXD2424S	541/565	509/533	253	12	2-311
FXD2424U	676	644	226	13	2-312
FXD2323U	531	499	211	12	2-313
FXD3216S	551/578	515/551	201	10	2-299
FXD3216U	703	667	243	9	2-300
FXD3115U	541	505	160	9	2-300
FXD3224S	715/742	675/702	280	11	2-287
FXD3224U	867	827	299	11	2-288
FXD3123U	705	665	232	10	2-288
FXD3232S	879/912	835/868	357	13	2-271
FXD3232U	1031	987	364	13	2-272
FXD3131U	869	825	304	13	2-272

Legend: PM - Program memory, DM - Data Memory

REFERENCES

1. Cavanagh, J.J.F., “Digital Computer Arithmetic,” McGraw-Hill,1984.
2. Hwang, K., “Computer Arithmetic,” John Wiley & Sons, 1979.
3. Scott, N.R., “Computer Number Systems & Arithmetic,” Prentice Hall, 1985.
4. IEEE Standards Board, “IEEE Standard for Floating-Point Arithmetic,” ANSI/IEEE Std 754-1985, IEEE, 1985.
5. Knuth, D.E., “The Art of Computer Programming, Volume 2,” Addison-Wesley, 1981.
6. F.J.Testa, “IEEE 754 Compliant Floating Point Routines,” AN575, Embedded Control Handbook, Microchip Technology Inc., 1995.

APPENDIX A: ALGORITHMS

Several algorithms for decimal to binary conversion are given below. The integer and fractional conversion algorithms are useful in both native assembly as well as high level languages.

A.1 Integer conversion algorithm[3]:

Given an integer I, where d(k) are the bit values of its n bit binary representation with d(0) = LSB,

$$I = \sum_{k=0}^{n-1} d(k) \bullet 2^{k}$$

k=0

I(k) = I

while I(k) = ! 0

d(k) = remainder of I(k)/2

I(k+1) = ⌈ I(k)/2 ⌉

k = k + 1

endw

where ⌈ ⌉ denotes the greatest integer function (or ceiling function).

A.2 Fractional conversion algorithm[3]:

Given a fraction F, where d(k) are the bit values of its n bit binary representation with d(1)=MSB,

$$F = \sum_{k=1}^{n} d(k) \bullet 2^{-k}$$

k=0

F(k) = F

while k <= n

d(k) = ⌈ F(k)•2 ⌉

F(k+1) = fractional part of F(k)•2

k = k + 1

endw

APPENDIX B: FLOWCHARTS

FIGURE B-1: MULTIPLICATION FLOWCHART

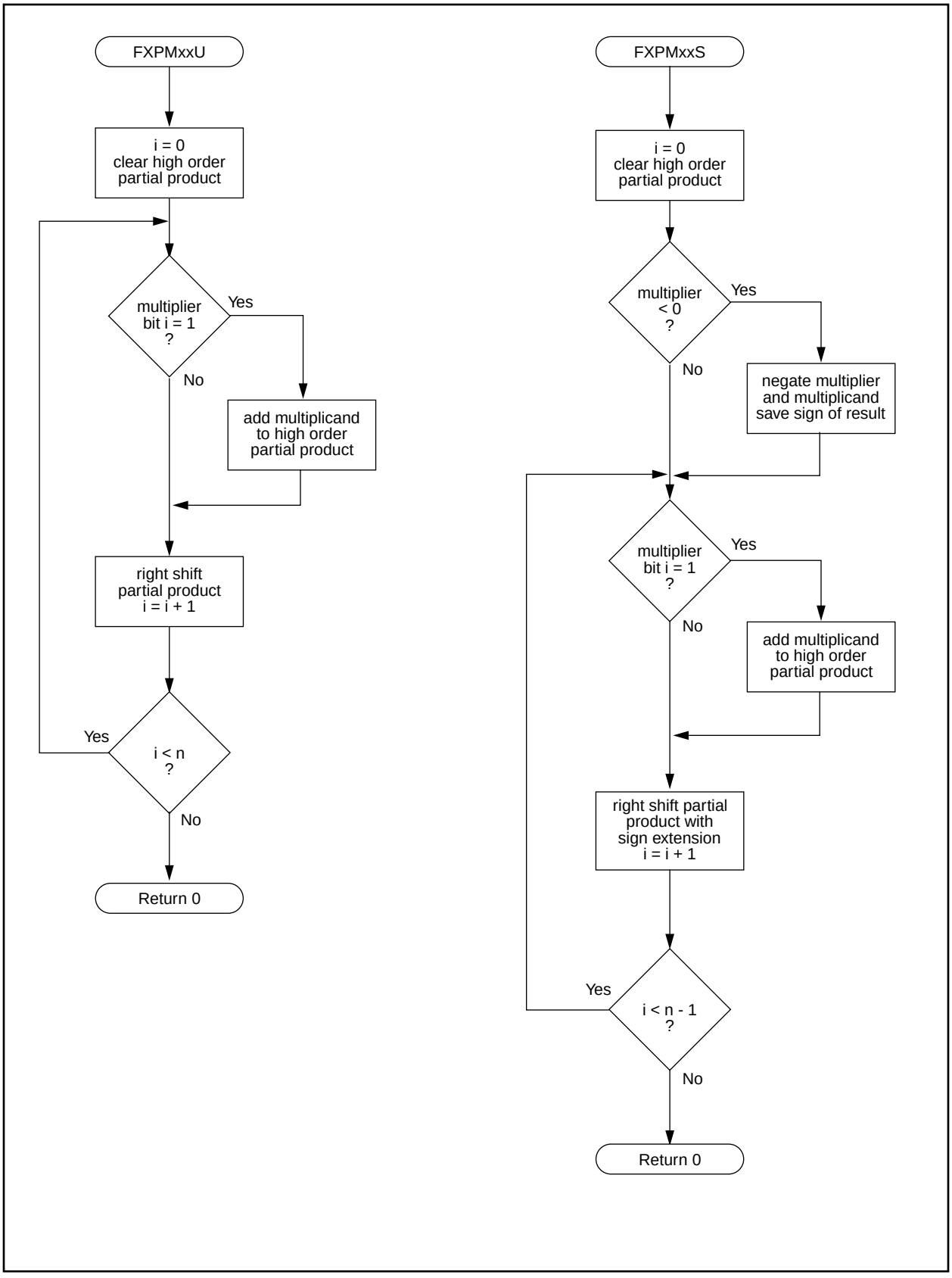
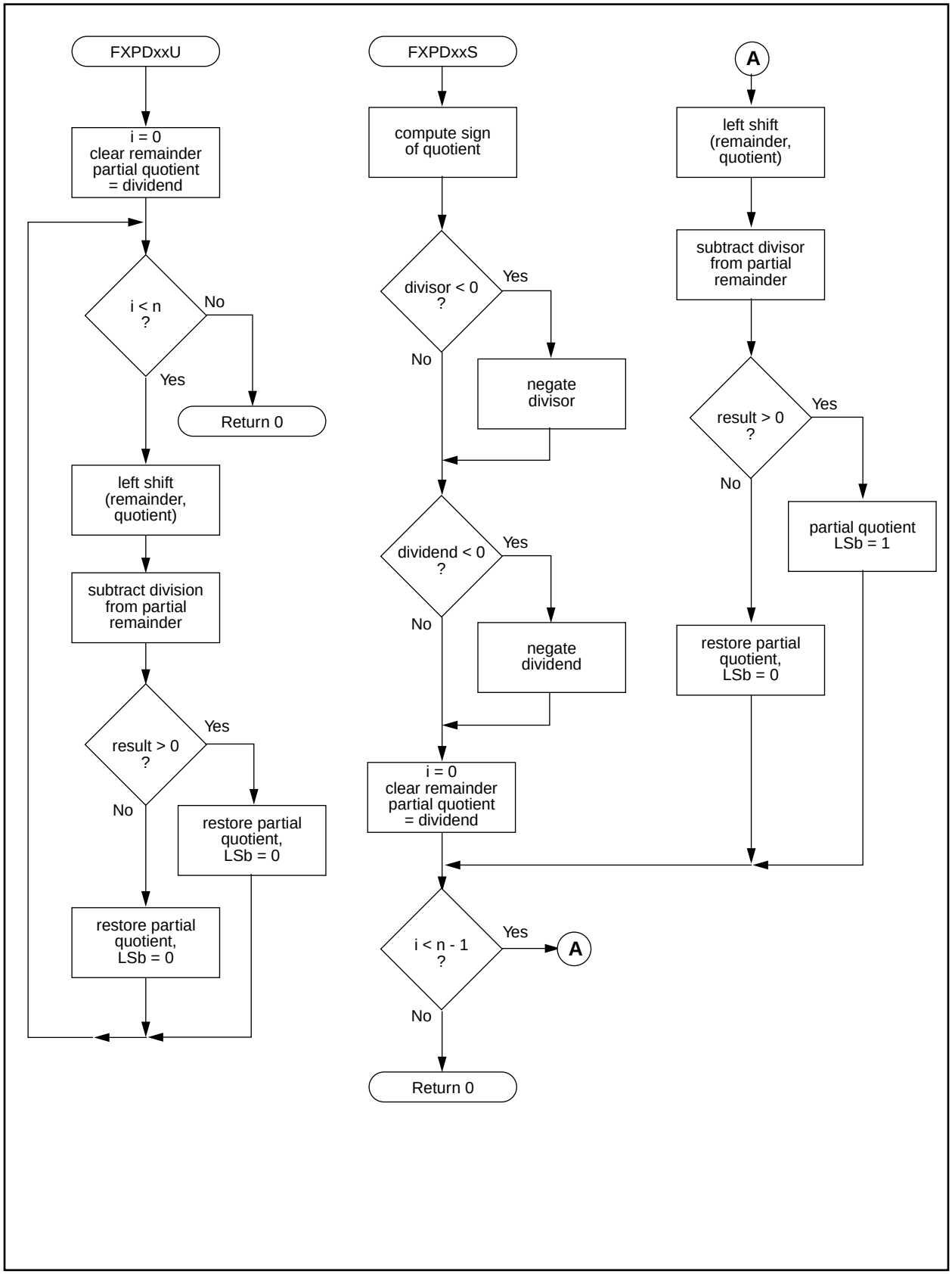


FIGURE B-2: DIVISION FLOWCHART



Please check the Microchip BBS for the latest version of the source code. For BBS access information, see Section 6, Microchip Bulletin Board Service information, page 6-3.

APPENDIX C: MULTIPLY ROUTINES FOR THE PIC16C5X/PIC16CXX

Table of Contents for Appendix C

C.1 32x32 PIC16C5X/PIC16CXX Fixed Point Multiply Routines

```
;       32x32 PIC16 FIXED POINT MULTIPLY ROUTINES       VERSION 1.2
;       Input:  fixed point arguments in AARG and BARG
;       Output: product AARGxBARG in AARG
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in performance.
;         Routine            Clocks     Function
;       FXM3232S     889        32x32 -> 64 bit signed fixed point multiply
;       FXM3232U     856        32x32 -> 64 bit unsigned fixed point multiply
;       FXM3131U     836        31x31 -> 62 bit unsigned fixed point multiply
;       The above timings are based on the looped macros. If space permits,
;       approximately 128-168 clocks can be saved by using the unrolled macros.
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number generator registers
RANDLO          equ     0x1B
DATA            equ     0x20    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 32x32 bit fixed point multiply algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            TFXM3232
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
```

```
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI
                RETEW           0
;       Test suite for FXM3232
TFXM3232
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
                BCF             BARGB0,MSB
                MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB2
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB3
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
                BCF             AARGB0,MSB
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB2
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB3
                MOVWF           INDF
                INCF            FSR
                CALL            FXM3131U
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB4,W
                MOVWF           INDF
```

```
                INCF            FSR
                MOVF            AARGB5,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB6,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB7,W
                MOVWF           INDF
                INCF            FSR
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       32x32 Bit Multiplication Macros
SMUL3232L       macro
;       Max Timing:     2+13+6*26+25+2+7*27+26+2+7*28+27+2+6*29+28+9 = 851 clks
;       Min Timing:     2+7*6+5+1+7*6+5+1+7*6+5+2+6*6+5+6 = 192 clks
;       PM: 31+25+2+26+2+27+2+28+9 = 152            DM: 17
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPSM3232A
                RRF             BARGB3
                BTFSC           _C
                GOTO            ALSM3232NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM3232A
                MOVWF           LOOPCOUNT
LOOPSM3232B
                RRF             BARGB2
                BTFSC           _C
                GOTO            BLSM3232NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM3232B
                MOVWF           LOOPCOUNT
LOOPSM3232C
                RRF             BARGB1
                BTFSC           _C
                GOTO            CLSM3232NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM3232C
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPSM3232D
                RRF             BARGB0
                BTFSC           _C
                GOTO            DLSM3232NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM3232D
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00
ALOOPSM3232
                RRF             BARGB3
                BTFSS           _C
                GOTO            ALSM3232NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
```

```
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALSM3232NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPSM3232
                MOVLW           0x8
                MOVWF           LOOPCOUNT
BLOOPSM3232
                RRF             BARGB2
                BTFSS           _C
                GOTO            BLSM3232NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLSM3232NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPSM3232
                MOVLW           0x8
                MOVWF           LOOPCOUNT
CLOOPSM3232
                RRF             BARGB1
                BTFSS           _C
                GOTO            CLSM3232NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
CLSM3232NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
```

```
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                DECFSZ          LOOPCOUNT
                GOTO            CLOOPSM3232
                MOVLW           0x7
                MOVWF           LOOPCOUNT
DLOOPSM3232
                RRF             BARGB0
                BTFSS           _C
                GOTO            DLSM3232NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
DLSM3232NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                RRF             ACCB7
                DECFSZ          LOOPCOUNT
                GOTO            DLOOPSM3232
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                RRF             ACCB7
                endm
UMUL3232L       macro
;       Max Timing:     2+15+6*25+24+2+7*26+25+2+7*27+26+2+7*28+27 = 842 clks
;       Min Timing:     2+7*6+5+1+7*6+5+1+7*6+5+1+7*6+5+6 = 197 clks
;       PM: 38+24+2+25+2+26+2+27+9 = 155            DM: 17
                MOVLW           0x08
                MOVWF           LOOPCOUNT
LOOPUM3232A
                RRF             BARGB3
                BTFSC           _C
                GOTO            ALUM3232NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3232A
                MOVWF           LOOPCOUNT
LOOPUM3232B
                RRF             BARGB2
                BTFSC           _C
                GOTO            BLUM3232NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3232B
```

```
                MOVWF           LOOPCOUNT
LOOPUM3232C
                RRF             BARGB1
                BTFSC           _C
                GOTO            CLUM3232NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3232C
                MOVWF           LOOPCOUNT
LOOPUM3232D
                RRF             BARGB0
                BTFSC           _C
                GOTO            DLUM3232NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3232D
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00
ALUM3232NAP     BCF             _C
                GOTO            ALUM3232NA
BLUM3232NAP     BCF             _C
                GOTO            BLUM3232NA
CLUM3232NAP     BCF             _C
                GOTO            CLUM3232NA
DLUM3232NAP     BCF             _C
                GOTO            DLUM3232NA
ALOOPUM3232
                RRF             BARGB3
                BTFSS           _C
                GOTO            ALUM3232NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALUM3232NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM3232
                MOVLW           0x08
                MOVWF           LOOPCOUNT
BLOOPUM3232
                RRF             BARGB2
                BTFSS           _C
                GOTO            BLUM3232NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
```

```
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM3232NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM3232
                MOVLW           0x08
                MOVWF           LOOPCOUNT
CLOOPUM3232
                RRF             BARGB1
                BTFSS           _C
                GOTO            CLUM3232NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
CLUM3232NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                DECFSZ          LOOPCOUNT
                GOTO            CLOOPUM3232
                MOVLW           0x08
                MOVWF           LOOPCOUNT
DLOOPUM3232
                RRF             BARGB0
                BTFSS           _C
                GOTO            DLUM3232NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
```

```
DLUM3232NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                RRF             ACCB7
                DECFSZ          LOOPCOUNT
                GOTO            DLOOPUM3232
                endm
UMUL3131L       macro
;       Max Timing:     2+15+6*25+24+2+7*26+25+2+7*27+26+2+6*28+27+8 = 822 clks
;       Min Timing:     2+7*6+5+1+7*6+5+1+7*6+5+2+6*6+5+6 = 192 clks
;       PM: 39+24+2+25+2+26+2+27+8 = 155            DM: 17
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPUM3131A
                RRF             BARGB3
                BTFSC           _C
                GOTO            ALUM3131NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3131A
                MOVWF           LOOPCOUNT
LOOPUM3131B
                RRF             BARGB2
                BTFSC           _C
                GOTO            BLUM3131NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3131B
                MOVWF           LOOPCOUNT
LOOPUM3131C
                RRF             BARGB1
                BTFSC           _C
                GOTO            CLUM3131NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3131C
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPUM3131D
                RRF             BARGB0
                BTFSC           _C
                GOTO            DLUM3131NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3131D
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

ALUM3131NAP     BCF             _C
                GOTO            ALUM3131NA

BLUM3131NAP     BCF             _C
                GOTO            BLUM3131NA

CLUM3131NAP     BCF             _C
                GOTO            CLUM3131NA

DLUM3131NAP     BCF             _C
                GOTO            DLUM3131NA
ALOOPUM3131
                RRF             BARGB3
                BTFSS           _C
```

```
                GOTO            ALUM3131NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALUM3131NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM3131
                MOVLW           0x08
                MOVWF           LOOPCOUNT
BLOOPUM3131
                RRF             BARGB2
                BTFSS           _C
                GOTO            BLUM3131NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM3131NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM3131
                MOVLW           0x08
                MOVWF           LOOPCOUNT
CLOOPUM3131
                RRF             BARGB1
                BTFSS           _C
                GOTO            CLUM3131NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
CLUM3131NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                DECFSZ          LOOPCOUNT
                GOTO            CLOOPUM3131
                MOVLW           0x07
                MOVWF           LOOPCOUNT
DLOOPUM3131
                RRF             BARGB0
                BTFSS           _C
                GOTO            DLUM3131NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
DLUM3131NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                RRF             ACCB7
                DECFSZ          LOOPCOUNT
                GOTO            DLOOPUM3131
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                RRF             ACCB7
                endm

SMUL3232        macro
;       Max Timing:     9+7*22+8*23+8*24+7*25+9 = 723 clks
;       Min Timing:     62+6 = 68 clks
;       PM: 68+6+7*22+8*23+8*24+7*25+9 = 788           DM: 16
                variable i
                i = 0

                while i < 8
```

```
                BTFSC           BARGB3,i
                GOTO            SM3232NA#v(i)
                i = i + 1
                endw
                i = 8

                while i < 16

                BTFSC           BARGB2,i-8
                GOTO            SM3232NA#v(i)
                i = i + 1
                endw
                i = 16

                while i < 24

                BTFSC           BARGB1,i-16
                GOTO            SM3232NA#v(i)
                i = i + 1
                endw
                i = 24

                while i < 31

                BTFSC           BARGB0,i-24
                GOTO            SM3232NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0          ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETEW           0
SM3232NA0       RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = 1
                while   i < 8
                BTFSS           BARGB3,i
                GOTO            SM3232NA#v(i)
SM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM3232NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 8
```

```
                while   i < 16
                BTFSS           BARGB2,i-8
                GOTO            SM3232NA#v(i)
SM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM3232NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                i = i + 1
                endw
                i = 16
                while   i < 24
                BTFSS           BARGB1,i-16
                GOTO            SM3232NA#v(i)
SM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM3232NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                i = i + 1
                endw
                i = 24
                while   i < 31
                BTFSS           BARGB0,i-24
                GOTO            SM3232NA#v(i)
SM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM3232NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                RRF             ACCB7
                i = i + 1
                endw
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                RRF             ACCB7
                endm
UMUL3232        macro
;       Max Timing:     9+8*21+8*22+8*23+8*24 = 729 clks
;       Min Timing:     63+6 = 69 clks
;       PM: 69+6+8*21+8*22+8*23+8*24 = 795            DM: 16
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB3,i
                GOTO            UM3232NA#v(i)
                i = i + 1
                endw
                i = 8

                while i < 16

                BTFSC           BARGB2,i-8
                GOTO            UM3232NA#v(i)
                i = i + 1
                endw
                i = 16

                while i < 24

                BTFSC           BARGB1,i-16
                GOTO            UM3232NA#v(i)
                i = i + 1
                endw
                i = 24

                while i < 32

                BTFSC           BARGB0,i-24
                GOTO            UM3232NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
```

```
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETEW           0
UM3232NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = 1
                while   i < 8
                BTFSS           BARGB3,i
                GOTO            UM3232NA#v(i)
UM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3232NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 8
                while   i < 16
                BTFSS           BARGB2,i-8
                GOTO            UM3232NA#v(i)
UM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3232NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                i = i + 1
                endw
                i = 16
                while   i < 24
                BTFSS           BARGB1,i-16
                GOTO            UM3232NA#v(i)
UM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
```

```
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3232NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                i = i + 1
                endw
                i = 24
                while   i < 32
                BTFSS           BARGB0,i-24
                GOTO            UM3232NA#v(i)
UM3232A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3232NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                RRF             ACCB7
                i = i + 1
                endw
                endm
UMUL3131        macro
;       Max Timing:     9+7*21+8*22+8*23+7*24+9 = 693 clks
;       Min Timing:     62+6 = 68 clks
;       PM: 68+6+7*22+8*23+8*24+7*25+9 = 788            DM: 16
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                while i < 8

                BTFSC           BARGB3,i
                GOTO            UM3131NA#v(i)
                i = i + 1
                endw
                i = 8
                while i < 16
```

```
                BTFSC           BARGB2,i-8
                GOTO            UM3131NA#v(i)
                i = i + 1
                endw
                i = 16
                while i < 24

                BTFSC           BARGB1,i-16
                GOTO            UM3131NA#v(i)
                i = i + 1
                endw
                i = 24
                while i < 31

                BTFSC           BARGB0,i-24
                GOTO            UM3131NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETEW           0
UM3131NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = 1
                while   i < 8
                BTFSS           BARGB3,i
                GOTO            UM3131NA#v(i)
UM3131A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3131NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 8
                while   i < 16
                BTFSS           BARGB2,i-8
                GOTO            UM3131NA#v(i)
UM3131A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
```

```
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3131NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                i = i + 1
                endw
                i = 16
                while   i < 24
                BTFSS           BARGB1,i-16
                GOTO            UM3131NA#v(i)
UM3131A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3131NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                i = i + 1
                endw
                i = 24
                while   i < 31
                BTFSS           BARGB0,i-24
                GOTO            UM3131NA#v(i)
UM3131A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3131NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                RRF             ACCB7
```

```
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                RRF             ACCB7
                endm

;*****************************************************************************************
;*****************************************************************************************
;       32x32 Bit Signed Fixed Point Multiply 32x32 -> 64
;       Input:  32 bit signed fixed point multiplicand in AARGB0
;                       32 bit signed fixed point multiplier in BARGB0
;       Use:    CALL    FXM3232S
;       Output: 64 bit signed fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     15+851+2 = 868 clks                B > 0
;                       36+851+2 = 889 clks                B < 0
;       Min Timing:     15+192 = 207 clks
;       PM: 36+152+1 = 189              DM: 17
FXM3232S        BTFSS           BARGB0,MSB
                GOTO            M3232SOK
                COMF            BARGB3          ; make multiplier BARG > 0
                INCF            BARGB3
                BTFSC           _Z
                DECF            BARGB2
                COMF            BARGB2
                BTFSC           _Z
                DECF            BARGB1
                COMF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
                COMF            AARGB3
                INCF            AARGB3
                BTFSC           _Z
                DECF            AARGB2
                COMF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
M3232SOK        CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                CLRF            ACCB7
                CLRF            ACCB6
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3
                SMUL3232L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       32x32 Bit Unsigned Fixed Point Multiply 32x32 -> 64
;       Input:  32 bit unsigned fixed point multiplicand in AARGB0
```

```
;                       32 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM3232U
;       Output: 64 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     12+842+2 = 856 clks
;       Min Timing:     12+197 = 209 clks
;       PM: 12+155+1 = 168              DM: 17
FXM3232U
                CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                CLRF            ACCB7
                CLRF            ACCB6
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3
                UMUL3232L
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************

;       31x31 Bit Unsigned Fixed Point Divide 31x31 -> 62
;       Input:  31 bit unsigned fixed point multiplicand in AARGB0
;                       31 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM3131U
;       Output: 62 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     12+822+2 = 836 clks
;       Min Timing:     12+192 = 204 clks
;       PM: 12+155+1 = 168              DM: 17
FXM3131U
                CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                CLRF            ACCB7
                CLRF            ACCB6
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3
                UMUL3131L
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************
                END
```

C.2 32x24 PIC16C5X/PIC16CXX Fixed Point Multiply Routines

```
;       32x24 PIC16 FIXED POINT MULTIPLY ROUTINES       VERSION 1.2
;       Input:  fixed point arguments in AARG and BARG
;       Output: product AARGxBARG in AARG
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine           Clocks     Function
;       FXM3224S     652        32x24 -> 56 bit signed fixed point multiply
;       FXM3224U     630        32x24 -> 56 bit unsigned fixed point multiply
;       FXM3123U     610        31x23 -> 54 bit unsigned fixed point multiply
```

```
;       The above timings are based on the looped macros. If space permits,
;       approximately 80-97 clocks can be saved by using the unrolled macros.
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number generator registers
RANDLO          equ     0x1B
DATA            equ     0x20    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 32x24 bit fixed point multiply algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            TFXM3224
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETEW           0
;       Test suite for FXM3224
TFXM3224
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
;               BCF             BARGB0,MSB
;               MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
```

```
                MOVF            RANDHI,W
                MOVWF           BARGB2
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
;               BCF             AARGB0,MSB
;               MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB2
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB3
                MOVWF           INDF
                INCF            FSR
                CALL            FXM3224S
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB3,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB4,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB5,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB6,W
                MOVWF           INDF
                INCF            FSR
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************
;       32x24 Bit Multiplication Macros
SMUL3224L       macro
;       Max Timing:     2+13+6*26+25+2+7*27+26+2+6*28+27+8 = 618 clks
;       Min Timing:     2+7*6+5+1+7*6+5+2+6*6+5+6 = 146 clks
;       PM: 25+25+2+26+2+27+8 = 115            DM: 15
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPSM3224A
                RRF             BARGB2
                BTFSC           _C
                GOTO            ALSM3224NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM3224A
                MOVWF           LOOPCOUNT
LOOPSM3224B
                RRF             BARGB1
                BTFSC           _C
```

```
                GOTO            BLSM3224NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM3224B
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPSM3224C
                RRF             BARGB0
                BTFSC           _C
                GOTO            CLSM3224NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM3224C
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00
ALOOPSM3224
                RRF             BARGB2
                BTFSS           _C
                GOTO            ALSM3224NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALSM3224NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPSM3224
                MOVLW           0x8
                MOVWF           LOOPCOUNT
BLOOPSM3224
                RRF             BARGB1
                BTFSS           _C
                GOTO            BLSM3224NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLSM3224NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
```

```
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPSM3224
                MOVLW           0x7
                MOVWF           LOOPCOUNT
CLOOPSM3224
                RRF             BARGB0
                BTFSS           _C
                GOTO            CLSM3224NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
CLSM3224NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                DECFSZ          LOOPCOUNT
                GOTO            CLOOPSM3224
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                endm
UMUL3224L       macro
;       Max Timing:     2+15+6*25+24+2+7*26+25+2+7*27+26 = 617 clks
;       Min Timing:     2+7*6+5+1+7*6+5+1+7*6+5+6 = 151 clks
;       PM: 31+24+2+25+2+26+2+27 = 139            DM: 15
                MOVLW           0x08
                MOVWF           LOOPCOUNT
LOOPUM3224A
                RRF             BARGB2
                BTFSC           _C
                GOTO            ALUM3224NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3224A
                MOVWF           LOOPCOUNT
LOOPUM3224B
                RRF             BARGB1
                BTFSC           _C
                GOTO            BLUM3224NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3224B
                MOVWF           LOOPCOUNT
LOOPUM3224C
                RRF             BARGB0
```

```
                BTFSC           _C
                GOTO            CLUM3224NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3224C
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

ALUM3224NAP     BCF             _C
                GOTO            ALUM3224NA

BLUM3224NAP     BCF             _C
                GOTO            BLUM3224NA

CLUM3224NAP     BCF             _C
                GOTO            CLUM3224NA
ALOOPUM3224
                RRF             BARGB2
                BTFSS           _C
                GOTO            ALUM3224NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALUM3224NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM3224
                MOVLW           0x08
                MOVWF           LOOPCOUNT
BLOOPUM3224
                RRF             BARGB1
                BTFSS           _C
                GOTO            BLUM3224NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM3224NA
                RRF             ACCB0
```

```
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM3224
                MOVLW           0x08
                MOVWF           LOOPCOUNT
CLOOPUM3224
                RRF             BARGB0
                BTFSS           _C
                GOTO            CLUM3224NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
CLUM3224NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                DECFSZ          LOOPCOUNT
                GOTO            CLOOPUM3224
                endm
UMUL3123L       macro
;       Max Timing:     2+15+6*25+24+2+7*26+25+2+6*27+26+7 = 597 clks
;       Min Timing:     2+7*6+5+1+7*6+5+2+6*6+5+6 = 146 clks
;       PM: 31+24+2+25+2+26+7 = 117            DM: 15
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPUM3123A
                RRF             BARGB2
                BTFSC           _C
                GOTO            ALUM3123NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3123A
                MOVWF           LOOPCOUNT
LOOPUM3123B
                RRF             BARGB1
                BTFSC           _C
                GOTO            BLUM3123NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3123B
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPUM3123C
                RRF             BARGB0
                BTFSC           _C
                GOTO            CLUM3123NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3123C
                CLRF            AARGB0
```

```
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00

ALUM3123NAP     BCF             _C
                GOTO            ALUM3123NA

BLUM3123NAP     BCF             _C
                GOTO            BLUM3123NA

CLUM3123NAP     BCF             _C
                GOTO            CLUM3123NA
ALOOPUM3123
                RRF             BARGB2
                BTFSS           _C
                GOTO            ALUM3123NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALUM3123NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM3123
                MOVLW           0x08
                MOVWF           LOOPCOUNT
BLOOPUM3123
                RRF             BARGB1
                BTFSS           _C
                GOTO            BLUM3123NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM3123NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
```

```
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM3123
                MOVLW           0x07
                MOVWF           LOOPCOUNT
CLOOPUM3123
                RRF             BARGB0
                BTFSS           _C
                GOTO            CLUM3123NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
CLUM3123NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                DECFSZ          LOOPCOUNT
                GOTO            CLOOPUM3123
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                endm

SMUL3224        macro
;       Max Timing:     9+7*22+8*23+7*24+8 = 523 clks
;       Min Timing:     40+6 = 46 clks
;       PM: 46+6+7*22+8*23+7*24+8 = 566            DM: 14
                variable i
                i = 0

                while i < 8

                BTFSC           BARGB2,i
                GOTO            SM3224NA#v(i)
                i = i + 1
                endw
                i = 8

                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            SM3224NA#v(i)
                i = i + 1
                endw
                i = 16

                while i < 23
```

```
                BTFSC           BARGB0,i-16
                GOTO            SM3224NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETEW           0
SM3224NA0       RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = 1
                while   i < 8
                BTFSS           BARGB2,i
                GOTO            SM3224NA#v(i)
SM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM3224NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 8
                while   i < 16
                BTFSS           BARGB1,i-8
                GOTO            SM3224NA#v(i)
SM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM3224NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
```

```
                i = i + 1
                endw
                i = 16
                while   i < 23
                BTFSS           BARGB0,i-16
                GOTO            SM3224NA#v(i)
SM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM3224NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                i = i + 1
                endw
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                endm
UMUL3224        macro
;       Max Timing:     9+8*21+8*22+8*23 = 537 clks
;       Min Timing:     41+6 = 47 clks
;       PM: 47+6+8*21+8*22+8*23 = 581           DM: 14
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB2,i
                GOTO            UM3224NA#v(i)
                i = i + 1
                endw
                i = 8

                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            UM3224NA#v(i)
                i = i + 1
                endw
                i = 16

                while i < 24

                BTFSC           BARGB0,i-16
```

```
                GOTO            UM3224NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETEW           0
UM3224NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = 1
                while   i < 8
                BTFSS           BARGB2,i
                GOTO            UM3224NA#v(i)
UM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3224NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 8
                while   i < 16
                BTFSS           BARGB1,i-8
                GOTO            UM3224NA#v(i)
UM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3224NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                i = i + 1
                endw
                i = 16
                while   i < 24
```

```
                BTFSS           BARGB0,i-16
                GOTO            UM3224NA#v(i)
UM3224A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3224NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                i = i + 1
                endw
                endm
UMUL3123        macro
;       Max Timing:     9+7*21+8*22+7*23+7 = 500 clks
;       Min Timing:     41+6 = 47 clks
;       PM: 47+5+7*22+8*23+7*24+7 = 565            DM: 14
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                while i < 8

                BTFSC           BARGB2,i
                GOTO            UM3123NA#v(i)
                i = i + 1
                endw
                i = 8
                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            UM3123NA#v(i)
                i = i + 1
                endw
                i = 16
                while i < 23

                BTFSC           BARGB0,i-16
                GOTO            UM3123NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETEW           0
UM3123NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = 1
                while   i < 8
                BTFSS           BARGB2,i
```

```
                GOTO            UM3123NA#v(i)
UM3123A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3123NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 8
                while   i < 16
                BTFSS           BARGB1,i-8
                GOTO            UM3123NA#v(i)
UM3123A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3123NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                i = i + 1
                endw
                i = 16
                while   i < 23
                BTFSS           BARGB0,i-16
                GOTO            UM3123NA#v(i)
UM3123A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
```

```
UM3123NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                RRF             ACCB6
                endm

;*****************************************************************************************
;*****************************************************************************************

;       32x24 Bit Signed Fixed Point Multiply 32x24 -> 56
;       Input:  32 bit signed fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               24 bit signed fixed point multiplier in BARGB0, BARGB1,
;               BARGB2
;       Use:    CALL    FXM3224S
;       Output: 56 bit signed fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     14+618+2 = 634 clks               B > 0
;                       32+618+2 = 652 clks               B < 0
;       Min Timing:     14+146 = 160 clks
;       PM: 36+115+1 = 152              DM: 15
FXM3224S        BTFSS           BARGB0,MSB
                GOTO            M3224SOK
                COMF            BARGB2          ; make multiplier BARG > 0
                INCF            BARGB2
                BTFSC           _Z
                DECF            BARGB1
                COMF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
                COMF            AARGB3
                INCF            AARGB3
                BTFSC           _Z
                DECF            AARGB2
                COMF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
M3224SOK        CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                CLRF            ACCB6
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3
                SMUL3224L
```

```
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32x24 Bit Unsigned Fixed Point Multiply 32x24 -> 56
;       Input:  32 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               24 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2
;       Use:    CALL    FXM3224U
;       Output: 56 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     11+617+2 = 630 clks
;       Min Timing:     11+151 = 162 clks
;       PM: 11+139+1 = 151              DM: 15
FXM3224U
                CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                CLRF            ACCB6
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3
                UMUL3224L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       31x23 Bit Unsigned Fixed Point Divide 31x23 -> 54
;       Input:  31 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               23 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2
;       Use:    CALL    FXM3123U
;       Output: 54 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     11+597+2 = 610 clks
;       Min Timing:     11+146 = 157 clks
;       PM: 11+117+1 = 129              DM: 15
FXM3123U
                CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                CLRF            ACCB6
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3
                UMUL3123L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

C.3 32x16 PIC16C5X/PIC16CXX Fixed Point Multiply Routines

```
;       32x16 PIC16 FIXED POINT MULTIPLY ROUTINES       VERSION 1.2
;       Input:  fixed point arguments in AARG and BARG
;       Output: product AARGxBARG in AARG
;       All timings are worst case cycle counts
```

```
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXM3216S     423        32x16 -> 48 bit signed fixed point multiply
;       FXM3216U     412        32x16 -> 48 bit unsigned fixed point multiply
;       FXM3115U     392        31x15 -> 46 bit unsigned fixed point multiply
;       The above timings are based on the looped macros. If space permits,
;       approximately 65-88 clocks can be saved by using the unrolled macros.
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number generator registers
RANDLO          equ     0x1B
TESTCOUNT       equ     0x20    ; counter
DATA            equ     0x21    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 32x16 bit fixed point multiply algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            TFXM3216
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETEW           0
;       Test suite for FXM3216
TFXM3216        MOVLW           1
                MOVWF           TESTCOUNT
M3216LOOP
                CALL            RANDOM16
```

```
                MOVF            RANDHI,W
                MOVWF           BARGB0
                BCF             BARGB0,MSB
                MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
                BCF             AARGB0,MSB
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB2
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB3
                MOVWF           INDF
                INCF            FSR
                CALL            FXM3115U
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB3,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB4,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB5,W
                MOVWF           INDF
                INCF            FSR
                DECFSZ          TESTCOUNT
                GOTO            M3216LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       32x16 Bit Multiplication Macros
SMUL3216L       macro
;       Max Timing:     2+13+6*26+25+2+6*27+26+7 = 393 clks
;       Min Timing:     2+7*6+5+2+6*6+5+6 = 98 clks
;       PM: 19+60 = 79            DM: 11
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPSM3216A
                RRF             BARGB1
                BTFSC           _C
                GOTO            ALSM3216NA
                DECFSZ          LOOPCOUNT
```

```
                GOTO            LOOPSM3216A
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPSM3216B
                RRF             BARGB0
                BTFSC           _C
                GOTO            BLSM3216NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM3216B
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00
ALOOPSM3216
                RRF             BARGB1
                BTFSS           _C
                GOTO            ALSM3216NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALSM3216NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPSM3216
                MOVLW           0x7
                MOVWF           LOOPCOUNT
BLOOPSM3216
                RRF             BARGB0
                BTFSS           _C
                GOTO            BLSM3216NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLSM3216NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
```

```
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPSM3216
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                endm
UMUL3216L       macro
;       Max Timing:     2+15+6*25+24+2+7*26+25 = 400 clks
;       Min Timing:     2+7*6+5+1+7*6+5+6 = 103 clks
;       PM: 73            DM: 11
                MOVLW           0x08
                MOVWF           LOOPCOUNT
LOOPUM3216A
                RRF             BARGB1
                BTFSC           _C
                GOTO            ALUM3216NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3216A
                MOVWF           LOOPCOUNT
LOOPUM3216B
                RRF             BARGB0
                BTFSC           _C
                GOTO            BLUM3216NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3216B
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00
BLUM3216NAP
                BCF             _C
                GOTO            BLUM3216NA
ALUM3216NAP
                BCF             _C
                GOTO            ALUM3216NA
ALOOPUM3216
                RRF             BARGB1
                BTFSS           _C
                GOTO            ALUM3216NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALUM3216NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
```

```
                GOTO            ALOOPUM3216
                MOVLW           0x08
                MOVWF           LOOPCOUNT
BLOOPUM3216
                RRF             BARGB0
                BTFSS           _C
                GOTO            BLUM3216NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM3216NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM3216
                endm
UMUL3115L       macro
;       Max Timing:     2+15+6*25+24+2+6*26+25+6 = 380 clks
;       Min Timing:     2+7*6+5+2+6*6+5+6 = 96 clks
;       PM: 80            DM: 11
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPUM3115A
                RRF             BARGB1
                BTFSC           _C
                GOTO            ALUM3115NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3115A
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPUM3115B
                RRF             BARGB0
                BTFSC           _C
                GOTO            BLUM3115NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM3115B
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                CLRF            AARGB3
                RETLW           0x00
BLUM3115NAP
                BCF             _C
                GOTO            BLUM3115NA
ALUM3115NAP
                BCF             _C
                GOTO            ALUM3115NA
ALOOPUM3115
                RRF             BARGB1
                BTFSS           _C
                GOTO            ALUM3115NA
```

```
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALUM3115NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM3115
                MOVLW           0x07
                MOVWF           LOOPCOUNT
BLOOPUM3115
                RRF             BARGB0
                BTFSS                   _C
                GOTO            BLUM3115NA
                MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM3115NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM3115
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                endm

SMUL3216        macro
;       Max Timing:     5+8+7*20+7*21+5 = 305 clks
;       Min Timing:     5+24+21+7 = 57 clks
;       PM: 5+24+21+6+5+7*20+7*21+5 = 353           DM: 10
                variable i
                i = 0
                BTFSC           AARGB0,MSB
```

```
                COMF            ACCB4
                MOVF            ACCB4,W
                MOVWF           ACCB5
                RLF             ACCB0,W

                while i < 8

                BTFSC           BARGB1,i
                GOTO            SM3216NA#v(i)
                BCF             ACCB4,7-i
                i = i + 1
                endw
                i = 8

                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            SM3216NA#v(i)
                BCF             ACCB5,15-i
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                CLRF            ACCB5
                RETEW           0
SM3216NA0
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = 1
                while   i < 8
                BTFSS           BARGB1,i
                GOTO            SM3216NA#v(i)
SM3216A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM3216NA#v(i)
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 8
                while   i < 15
                BTFSS           BARGB0,i-8
                GOTO            SM3216NA#v(i)
SM3216A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
```

```
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM3216NA#v(i)
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                endm
UMUL3216        macro
;       Max Timing:     1+8+7*21+8*22 = 332 clks
;       Min Timing:     1+2*8+2*8+6 = 39 clks
;       PM: 1+2*8+2*8+6+7*21+8*22 = 362            DM: 10
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM3216NA#v(i)
                i = i + 1
                endw
                i = 8

                while i < 16

                BTFSC           BARGB0,i-8
                GOTO            UM3216NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETEW           0
UM3216NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = 1
                while   i < 8
                BTFSS           BARGB1,i
                GOTO            UM3216NA#v(i)
UM3216A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
```

```
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3216NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 8
                while   i < 16
                BTFSS           BARGB0,i-8
                GOTO            UM3216NA#v(i)
UM3216A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3216NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                i = i + 1
                endw
                endm
UMUL3115        macro
;       Max Timing:     9+7*21+7*22+6 = 316 clks
;       Min Timing:     1+30+6 = 37 clks
;       PM: 1+30+10+7*21+7*22+6 = 348           DM: 10
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM3115NA#v(i)
                i = i + 1
                endw
                i = 8
                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            UM3115NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
```

```
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETEW           0
UM3115NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = 1
                while   i < 8
                BTFSS           BARGB1,i
                GOTO            UM3115NA#v(i)
UM3115A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3115NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 8
                while   i < 15
                BTFSS           BARGB0,i-8
                GOTO            UM3115NA#v(i)
UM3115A#v(i)    MOVF            TEMPB3,W
                ADDWF           ACCB3
                MOVF            TEMPB2,W
                BTFSC           _C
                INCFSZ          TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM3115NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                endm
```

```
;*****************************************************************************************
;*****************************************************************************************

;       32x16 Bit Signed Fixed Point Multiply 32x16 -> 32
;       Input:  16 bit signed fixed point multiplicand in AARGB0
;                       16 bit signed fixed point multiplier in BARGB0
;       Use:    CALL    FXM3216S
;       Output: 32 bit signed fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     13+393+2 = 408 clks               B > 0
;                       28+393+2 = 423 clks               B < 0
;       Min Timing:     13+98 = 111 clks
;       PM: 18+79+1 = 98             DM: 9
FXM3216S        BTFSS           BARGB0,MSB
                GOTO            M3216SOK
                COMF            BARGB1          ; make multiplier BARG > 0
                INCF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
                COMF            AARGB3
                INCF            AARGB3
                BTFSC           _Z
                DECF            AARGB2
                COMF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
M3216SOK        CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3
                SMUL3216L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32x16 Bit Unsigned Fixed Point Multiply 32x16 -> 32
;       Input:  16 bit unsigned fixed point multiplicand in AARGB0
;                       16 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM3216U
;       Output: 32 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     10+400+2 = 412 clks
;       Min Timing:     10+104 = 114 clks
;       PM: 10+73+1 = 84             DM: 9
FXM3216U
                CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
```

```
                MOVF            AARGB3,W
                MOVWF           TEMPB3
                UMUL3216L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       31x15 Bit Unsigned Fixed Point Divide 31x15 -> 30
;       Input:  15 bit unsigned fixed point multiplicand in AARGB0
;                       15 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM3115U
;       Output: 30 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     10+380+2 = 392 clks
;       Min Timing:     10+96 = 106 clks
;       PM: 10+80+1 = 91              DM: 9
FXM3115U
                CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                MOVF            AARGB3,W
                MOVWF           TEMPB3
                UMUL3115L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

C.4 24x24 PIC16C5X/PIC16CXX Fixed Point Multiply Routines

```
;       24x24 PIC16 FIXED POINT MULTIPLY ROUTINES       VERSION 1.2
;       Input:  fixed point arguments in AARG and BARG
;       Output: product AARGxBARG in AARG
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXM2424S     535        24x24 -> 48 bit signed fixed point multiply
;       FXM2424U     512        24x24 -> 48 bit unsigned fixed point multiply
;       FXM2323U     497        23x23 -> 46 bit unsigned fixed point multiply
;       The above timings are based on the looped macros. If space permits,
;       approximately 61-95 clocks can be saved by using the unrolled macros.
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number generator registers
RANDLO          equ     0x1B
DATA            equ     0x20    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 24x24 bit fixed point multiply algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
```

```
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            TFXM2424
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETEW           0
;       Test suite for FXM2424
TFXM2424
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
                BCF             BARGB0,MSB
                MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB2
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
                BCF             AARGB0,MSB
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB2
                MOVWF           INDF
                INCF            FSR
```

```
                CALL            FXM2323U
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB3,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB4,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB5,W
                MOVWF           INDF
                INCF            FSR
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       24x24 Bit Multiplication Macros
SMUL2424L       macro
;       Max Timing:     2+12+6*21+20+2+7*22+21+2+6*23+22+7 = 506 clks
;       Min Timing:     2+7*6+5+1+7*6+5+2+6*6+5+5 = 145 clks
;       PM: 24+20+2+21+2+22+7 = 98           DM: 13
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPSM2424A
                RRF             BARGB2
                BTFSC           _C
                GOTO            ALSM2424NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM2424A
                MOVWF           LOOPCOUNT
LOOPSM2424B
                RRF             BARGB1
                BTFSC           _C
                GOTO            BLSM2424NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM2424B
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPSM2424C
                RRF             BARGB0
                BTFSC           _C
                GOTO            CLSM2424NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM2424C
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00
ALOOPSM2424
                RRF             BARGB2
                BTFSS           _C
                GOTO            ALSM2424NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
```

```
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALSM2424NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPSM2424
                MOVLW           0x8
                MOVWF           LOOPCOUNT
BLOOPSM2424
                RRF             BARGB1
                BTFSS           _C
                GOTO            BLSM2424NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLSM2424NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPSM2424
                MOVLW           0x7
                MOVWF           LOOPCOUNT
CLOOPSM2424
                RRF             BARGB0
                BTFSS           _C
                GOTO            CLSM2424NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
CLSM2424NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            CLOOPSM2424
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                endm
```

```
UMUL2424L       macro
;       Max Timing:     2+14+6*20+19+2+7*21+20+2+7*22+21 = 501 clks
;       Min Timing:     2+7*6+5+1+7*6+5+1+7*6+5+5 = 150 clks
;       PM: 23+20+2+21+2+22 = 88            DM: 13
                MOVLW           0x08
                MOVWF           LOOPCOUNT
LOOPUM2424A
                RRF             BARGB2
                BTFSC           _C
                GOTO            ALUM2424NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM2424A
                MOVWF           LOOPCOUNT
LOOPUM2424B
                RRF             BARGB1
                BTFSC           _C
                GOTO            BLUM2424NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM2424B
                MOVWF           LOOPCOUNT
LOOPUM2424C
                RRF             BARGB0
                BTFSC           _C
                GOTO            CLUM2424NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM2424C
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00
CLUM2424NAP
                BCF             _C
                GOTO            CLUM2424NA
BLUM2424NAP
                BCF             _C
                GOTO            BLUM2424NA
ALUM2424NAP
                BCF             _C
                GOTO            ALUM2424NA
ALOOPUM2424
                RRF             BARGB2
                BTFSS           _C
                GOTO            ALUM2424NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALUM2424NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM2424
                MOVLW           0x08
                MOVWF           LOOPCOUNT
BLOOPUM2424
                RRF             BARGB1
                BTFSS           _C
                GOTO            BLUM2424NA
```

```
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM2424NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM2424
                MOVLW           0x08
                MOVWF           LOOPCOUNT
CLOOPUM2424
                RRF             BARGB0
                BTFSS           _C
                GOTO            CLUM2424NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
CLUM2424NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            CLOOPUM2424
                endm
UMUL2323L       macro
;       Max Timing:     2+15+6*20+19+2+7*21+20+2+6*22+21+6 = 486 clks
;       Min Timing:     2+7*6+5+1+7*6+5+2+6*6+5+5 = 145 clks
;       PM: 24+20+2+21+2+22+6 = 97           DM: 13
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPUM2323A
                RRF             BARGB2
                BTFSC           _C
                GOTO            ALUM2323NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM2323A
                MOVWF           LOOPCOUNT
LOOPUM2323B
                RRF             BARGB1
                BTFSC           _C
                GOTO            BLUM2323NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM2323B
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPUM2323C
```

```
                RRF             BARGB0
                BTFSC           _C
                GOTO            CLUM2323NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM2323C
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00
CLUM2323NAP
                BCF             _C
                GOTO            CLUM2323NA
BLUM2323NAP
                BCF            _C
                GOTO            BLUM2323NA
ALUM2323NAP
                BCF            _C
                GOTO            ALUM2323NA
ALOOPUM2323
                RRF             BARGB2
                BTFSS           _C
                GOTO            ALUM2323NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALUM2323NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM2323
                MOVLW           0x08
                MOVWF           LOOPCOUNT
BLOOPUM2323
                RRF             BARGB1
                BTFSS           _C
                GOTO            BLUM2323NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM2323NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM2323
                MOVLW           0x07
                MOVWF           LOOPCOUNT
CLOOPUM2323
```

```
                RRF             BARGB0
                BTFSS           _C
                GOTO            CLUM2323NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
CLUM2323NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                DECFSZ          LOOPCOUNT
                GOTO            CLOOPUM2323
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                endm

SMUL2424        macro
;       Max Timing:     8+7*17+8*18+7*19+7 = 411 clks
;       Min Timing:     46+5 = 51 clks
;       PM: 51+4+7*17+8*18+7*19+7 = 466           DM: 12
                variable i
                i = 0

                while i < 8

                BTFSC           BARGB2,i
                GOTO            SM2424NA#v(i)
                i = i + 1
                endw
                i = 8

                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            SM2424NA#v(i)
                i = i + 1
                endw
                i = 16

                while i < 23

                BTFSC           BARGB0,i-16
                GOTO            SM2424NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0          ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETEW           0
SM2424NA0       RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
```

```
                RRF             ACCB2
                RRF             ACCB3
                i = 1
                while   i < 8
                BTFSS           BARGB2,i
                GOTO            SM2424NA#v(i)
SM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM2424NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = i + 1
                endw
                i = 8
                while   i < 16
                BTFSS           BARGB1,i-8
                GOTO            SM2424NA#v(i)
SM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM2424NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 16
                while   i < 23
                BTFSS           BARGB0,i-16
                GOTO            SM2424NA#v(i)
SM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM2424NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
```

```
                i = i + 1
                endw
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                endm
UMUL2424        macro
;       Max Timing:     8+8*17+8*18+8*19 = 440 clks
;       Min Timing:     49+5 = 54 clks
;       PM: 54+4+8*17+8*18+8*19 = 490            DM: 12
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB2,i
                GOTO            UM2424NA#v(i)
                i = i + 1
                endw
                i = 8

                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            UM2424NA#v(i)
                i = i + 1
                endw
                i = 16

                while i < 24

                BTFSC           BARGB0,i-16
                GOTO            UM2424NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETEW           0
UM2424NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = 1
                while   i < 8
                BTFSS           BARGB2,i
                GOTO            UM2424NA#v(i)
UM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM2424NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
```

```
                i = i + 1
                endw
                i = 8
                while   i < 16
                BTFSS           BARGB1,i-8
                GOTO            UM2424NA#v(i)
UM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM2424NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 16
                while   i < 24
                BTFSS           BARGB0,i-16
                GOTO            UM2424NA#v(i)
UM2424A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM2424NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                i = i + 1
                endw
                endm
UMUL2323        macro
;       Max Timing:     8+7*17+8*18+7*19+7 = 411 clks
;       Min Timing:     46+5 = 51 clks
;       PM: 51+4+7*17+8*18+7*19+7 = 466            DM: 12
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                while i < 8

                BTFSC           BARGB2,i
                GOTO            UM2323NA#v(i)
                i = i + 1
                endw
                i = 8
                while i < 16

                BTFSC           BARGB1,i-8
                GOTO            UM2323NA#v(i)
                i = i + 1
                endw
```

```
                i = 16
                while i < 23

                BTFSC           BARGB0,i-16
                GOTO            UM2323NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETEW           0
UM2323NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = 1
                while   i < 8
                BTFSS           BARGB2,i
                GOTO            UM2323NA#v(i)
UM2323A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM2323NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = i + 1
                endw
                i = 8
                while   i < 16
                BTFSS           BARGB1,i-8
                GOTO            UM2323NA#v(i)
UM2323A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM2323NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                i = 16
                while   i < 23
                BTFSS           BARGB0,i-16
                GOTO            UM2323NA#v(i)
UM2323A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
```

```
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM2323NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                RRF             ACCB5
                endm

;*****************************************************************************************
;*****************************************************************************************

;       24x24 Bit Signed Fixed Point Multiply 24x24 -> 48
;       Input:  24 bit signed fixed point multiplicand in AARGB0
;                       24 bit signed fixed point multiplier in BARGB0
;       Use:    CALL    FXM2424S
;       Output: 48 bit signed fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     12+506+2 = 520 clks                B > 0
;                       27+506+2 = 535 clks                B < 0
;       Min Timing:     12+145 = 157 clks
;       PM: 27+98+1 = 126               DM: 13
FXM2424S        BTFSS           BARGB0,MSB
                GOTO            M2424SOK
                COMF            BARGB2          ; make multiplier BARG > 0
                INCF            BARGB2
                BTFSC           _Z
                DECF            BARGB1
                COMF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
                COMF            AARGB2
                INCF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
M2424SOK        CLRF            ACCB3           ; clear partial product
                CLRF            ACCB4
                CLRF            ACCB5
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                SMUL2424L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
```

```
;       24x24 Bit Unsigned Fixed Point Multiply 24x24 -> 48
;       Input:  24 bit unsigned fixed point multiplicand in AARGB0
;                       24 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM2424U
;       Output: 48 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     9+501+2 = 512 clks
;       Min Timing:     9+150 = 159 clks
;       PM: 9+88+1 = 98              DM: 13
FXM2424U
                CLRF            ACCB3           ; clear partial product
                CLRF            ACCB4
                CLRF            ACCB5
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                UMUL2424L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       23x23 Bit Unsigned Fixed Point Divide 23x23 -> 46
;       Input:  23 bit unsigned fixed point multiplicand in AARGB0
;                       23 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM2323U
;       Output: 46 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     9+486+2 = 497 clks
;       Min Timing:     9+145 = 154 clks
;       PM: 9+97+1 = 107              DM: 13
FXM2323U
                CLRF            ACCB3           ; clear partial product
                CLRF            ACCB4
                CLRF            ACCB5
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                UMUL2323L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

C.5 24x16 PIC16C5X/PIC16CXX Fixed Point Multiply Routines

```
;       24x16 PIC16 FIXED POINT MULTIPLY ROUTINES       VERSION 1.2
;       Input:  fixed point arguments in AARG and BARG
;       Output: product AARGxBARG in AARG
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXM2416S     346        24x16 -> 40 bit signed fixed point multiply
;       FXM2416U     334        24x16 -> 40 bit unsigned fixed point multiply
;       FXM2315U     319        23x15 -> 38 bit unsigned fixed point multiply
;       The above timings are based on the looped macros. If space permits,
;       approximately 36-62 clocks can be saved by using the unrolled macros.
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
```

```
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number generator registers
RANDLO          equ     0x1B
DATA            equ     0x20    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 24x16 bit fixed point multiply algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            TFXM2416
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETEW           0
;       Test suite for FXM2416
TFXM2416
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
;               BCF             BARGB0,MSB
;               MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
;               BCF             AARGB0,MSB
;               MOVF            AARGB0,W
```

```
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB2
                MOVWF           INDF
                INCF            FSR
                CALL            FXM2416S
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB3,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB4,W
                MOVWF           INDF
                INCF            FSR
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************
;       24x16 Bit Multiplication Macros
SMUL2416L       macro
;       Max Timing:     2+12+6*21+20+2+6*22+21+6 = 321 clks
;       Min Timing:     2+7*6+5+2+6*6+5+5 = 97 clks
;       PM: 19+20+2+21+6 = 68           DM: 12
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPSM2416A
                RRF             BARGB1
                BTFSC           _C
                GOTO            ALSM2416NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM2416A
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPSM2416B
                RRF             BARGB0
                BTFSC           _C
                GOTO            BLSM2416NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM2416B
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00
ALOOPSM2416
                RRF             BARGB1
                BTFSS           _C
                GOTO            ALSM2416NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
```

```
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALSM2416NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPSM2416
                MOVLW           0x7
                MOVWF           LOOPCOUNT
BLOOPSM2416
                RRF             BARGB0
                BTFSS           _C
                GOTO            BLSM2416NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLSM2416NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPSM2416
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                endm
UMUL2416L       macro
;       Max Timing:     2+14+6*20+19+2+7*21+20 = 324 clks
;       Min Timing:     2+7*6+5+1+7*6+5+5 = 102 clks
;       PM: 18+20+2+21 = 61             DM: 12
                MOVLW           0x08
                MOVWF           LOOPCOUNT
LOOPUM2416A
                RRF             BARGB1
                BTFSC           _C
                GOTO            ALUM2416NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM2416A
                MOVWF           LOOPCOUNT
LOOPUM2416B
                RRF             BARGB0
                BTFSC           _C
                GOTO            BLUM2416NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM2416B
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00
BLUM2416NAP
                BCF             _C
```

```
                GOTO            BLUM2416NA
ALUM2416NAP
                BCF             _C
                GOTO            ALUM2416NA
ALOOPUM2416
                RRF             BARGB1
                BTFSS           _C
                GOTO            ALUM2416NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALUM2416NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM2416
                MOVLW           0x08
                MOVWF           LOOPCOUNT
BLOOPUM2416
                RRF             BARGB0
                BTFSS           _C
                GOTO            BLUM2416NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM2416NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM2416
                endm
UMUL2315L       macro
;       Max Timing:     2+15+6*20+19+2+6*21+20+5 = 309 clks
;       Min Timing:     2+7*6+5+1+6*6+5+5 = 96 clks
;       PM: 19+20+2+21+5 = 67           DM: 12
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPUM2315A
                RRF             BARGB1
                BTFSC           _C
                GOTO            ALUM2315NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM2315A
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPUM2315B
                RRF             BARGB0
```

```
                BTFSC           _C
                GOTO            BLUM2315NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM2315B
                CLRF            AARGB0
                CLRF            AARGB1
                CLRF            AARGB2
                RETLW           0x00
BLUM2315NAP
                BCF            _C
                GOTO            BLUM2315NA
ALUM2315NAP
                BCF            _C
                GOTO            ALUM2315NA
ALOOPUM2315
                RRF             BARGB1
                BTFSS           _C
                GOTO            ALUM2315NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALUM2315NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM2315
                MOVLW           0x07
                MOVWF           LOOPCOUNT
BLOOPUM2315
                RRF             BARGB0
                BTFSS           _C
                GOTO            BLUM2315NA
                MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM2315NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM2315
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                endm
```

```
SMUL2416        macro
;       Max Timing:     8+7*17+7*18+6 = 259 clks
;       Min Timing:     30+5 = 35 clks
;       PM: 30+4+7*17+7*18+6 = 285            DM: 11
                variable i
                i = 0

                while i < 8

                BTFSC           BARGB1,i
                GOTO            SM2416NA#v(i)
                i = i + 1
                endw
                i = 8

                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            SM2416NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0          ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETEW           0
SM2416NA0       RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = 1
                while   i < 8
                BTFSS           BARGB1,i
                GOTO            SM2416NA#v(i)
SM2416A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM2416NA#v(i)   RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = i + 1
                endw
                i = 8
                while   i < 15
                BTFSS           BARGB0,i-8
                GOTO            SM2416NA#v(i)
SM2416A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM2416NA#v(i)   RLF             TEMPB0,W
```

```
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                endm
UMUL2416        macro
;       Max Timing:     8+8*17+8*18 = 288 clks
;       Min Timing:     33+5 = 38 clks
;       PM: 37+4+8*17+8*18 = 321            DM: 11
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM2416NA#v(i)
                i = i + 1
                endw
                i = 8

                while i < 16

                BTFSC           BARGB0,i-8
                GOTO            UM2416NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETEW           0
UM2416NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = 1
                while   i < 8
                BTFSS           BARGB1,i
                GOTO            UM2416NA#v(i)
UM2416A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM2416NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = i + 1
                endw
                i = 8
                while   i < 16
```

```
                BTFSS           BARGB0,i-8
                GOTO            UM2416NA#v(i)
UM2416A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM2416NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                endm
UMUL2315        macro
;       Max Timing:     8+7*17+7*18+6 = 259 clks
;       Min Timing:     31+5 = 36 clks
;       PM: 35+4+7*17+7*18+6 = 290            DM: 11
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM2315NA#v(i)
                i = i + 1
                endw
                i = 8
                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            UM2315NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETEW           0
UM2315NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = 1
                while   i < 8
                BTFSS           BARGB1,i
                GOTO            UM2315NA#v(i)
UM2315A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM2315NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
```

```
                i = i + 1
                endw
                i = 8
                while   i < 15
                BTFSS           BARGB0,i-8
                GOTO            UM2315NA#v(i)
UM2315A#v(i)    MOVF            TEMPB2,W
                ADDWF           ACCB2
                MOVF            TEMPB1,W
                BTFSC           _C
                INCFSZ          TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM2315NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                RRF             ACCB4
                endm

;*****************************************************************************************
;*****************************************************************************************

;       24x16 Bit Signed Fixed Point Multiply 24x16 -> 40
;       Input:  24 bit signed fixed point multiplicand in AARGB0
;               16 bit signed fixed point multiplier in BARGB0
;       Use:    CALL    FXM2416S
;       Output: 40 bit signed fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     11+321+2 = 334 clks               B > 0
;                       23+321+2 = 346 clks               B < 0
;       Min Timing:     11+97 = 108 clks
;       PM: 23+68+1 = 92              DM: 12
FXM2416S        BTFSS           BARGB0,MSB
                GOTO            M2416SOK
                COMF            BARGB1          ; make multiplier BARG > 0
                INCF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
                COMF            AARGB2
                INCF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
M2416SOK        CLRF            ACCB3           ; clear partial product
                CLRF            ACCB4
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
```

```
                SMUL2416L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       24x16 Bit Unsigned Fixed Point Multiply 24x16 -> 40
;       Input:  24 bit unsigned fixed point multiplicand in AARGB0
;               16 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM2416U
;       Output: 40 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     8+324+2 = 334 clks
;       Min Timing:     8+102 = 110 clks
;       PM: 8+61+1 = 70             DM: 12
FXM2416U
                CLRF            ACCB3           ; clear partial product
                CLRF            ACCB4
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                UMUL2416L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       23x15 Bit Unsigned Fixed Point Divide 23x15 -> 38
;       Input:  23 bit unsigned fixed point multiplicand in AARGB0
;               15 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM2315U
;       Output: 38 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     8+309+2 = 319 clks
;       Min Timing:     8+96 = 104 clks
;       PM: 8+67+1 = 76             DM: 12
FXM2315U
                CLRF            ACCB3           ; clear partial product
                CLRF            ACCB4
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                MOVF            AARGB2,W
                MOVWF           TEMPB2
                UMUL2315L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

C.6 16x16 PIC16C5X/PIC16CXX Fixed Point Multiply Routines

```
;       16x16 PIC16 FIXED POINT MULTIPLY ROUTINES       VERSION 1.2
;       Input:  fixed point arguments in AARG and BARG
;       Output: product AARGxBARG in AARG
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXM1616S     269        16x16 -> 32 bit signed fixed point multiply
;       FXM1616U     256        16x16 -> 32 bit unsigned fixed point multiply
;       FXM1515U     244        15x15 -> 30 bit unsigned fixed point multiply
;       The above timings are based on the looped macros. If space permits,
```

```
;       approximately 64-73 clocks can be saved by using the unrolled macros.
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number generator registers
RANDLO          equ     0x1B
TESTCOUNT       equ     0x20    ; counter
DATA            equ     0x21    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 16x16 bit fixed point multiply algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            TFXM1616
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETEW           0
;       Test suite for FXM1616
TFXM1616        MOVLW           1
                MOVWF           TESTCOUNT
M1616LOOP
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
                BCF             BARGB0,MSB
                MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
```

```
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
                BCF             AARGB0,MSB
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            FXM1515U
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB3,W
                MOVWF           INDF
                INCF            FSR
                DECFSZ          TESTCOUNT
                GOTO            M1616LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       16x16 Bit Multiplication Macros
SMUL1616L       macro
;       Max Timing:     2+11+6*16+15+2+6*17+16+5 = 249 clks
;       Min Timing:     2+7*6+5+2+6*6+5+4 = 96 clks
;       PM: 55            DM: 9
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPSM1616A
                RRF             BARGB1
                BTFSC           _C
                GOTO            ALSM1616NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM1616A
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPSM1616B
                RRF             BARGB0
                BTFSC           _C
                GOTO            BLSM1616NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM1616B
                CLRF            AARGB0
                CLRF            AARGB1
                RETLW           0x00
ALOOPSM1616
                RRF             BARGB1
                BTFSS           _C
                GOTO            ALSM1616NA
                MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
ALSM1616NA      RLF             TEMPB0,W
                RRF             ACCB0
```

```
                RRF             ACCB1
                RRF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPSM1616
                MOVLW           0x7
                MOVWF           LOOPCOUNT
BLOOPSM1616
                RRF             BARGB0
                BTFSS           _C
                GOTO            BLSM1616NA
                MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLSM1616NA      RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPSM1616
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                endm
UMUL1616L       macro
;       Max Timing:     2+13+6*15+14+2+7*16+15 = 248 clks
;       Min Timing:     2+7*6+5+1+7*6+5+4 = 101 clks
;       PM: 51            DM: 9
                MOVLW           0x08
                MOVWF           LOOPCOUNT
LOOPUM1616A
                RRF             BARGB1
                BTFSC           _C
                GOTO            ALUM1616NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM1616A
                MOVWF           LOOPCOUNT
LOOPUM1616B
                RRF             BARGB0
                BTFSC           _C
                GOTO            BLUM1616NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM1616B
                CLRF            AARGB0
                CLRF            AARGB1
                RETLW           0x00
BLUM1616NAP
                BCF             _C
                GOTO            BLUM1616NA
ALUM1616NAP
                BCF             _C
                GOTO            ALUM1616NA
ALOOPUM1616
                RRF             BARGB1
                BTFSS           _C
                GOTO            ALUM1616NA
                MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
```

```
                ADDWF           ACCB0
ALUM1616NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM1616
                MOVLW           0x08
                MOVWF           LOOPCOUNT
BLOOPUM1616
                RRF             BARGB0
                BTFSS           _C
                GOTO            BLUM1616NA
                MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM1616NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM1616
                endm
UMUL1515L       macro
;       Max Timing:     2+13+6*15+14+2+6*16+15+4 = 236 clks
;       Min Timing:     2+7*6+5+2+6*6+5+4 = 97 clks
;       PM: 56            DM: 9
                MOVLW           0x8
                MOVWF           LOOPCOUNT
LOOPUM1515A
                RRF             BARGB1
                BTFSC           _C
                GOTO            ALUM1515NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM1515A
                MOVLW           0x7
                MOVWF           LOOPCOUNT
LOOPUM1515B
                RRF             BARGB0
                BTFSC           _C
                GOTO            BLUM1515NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM1515B
                CLRF            AARGB0
                CLRF            AARGB1
                RETLW           0x00
BLUM1515NAP
                BCF             _C
                GOTO            BLUM1515NA
ALUM1515NAP
                BCF             _C
                GOTO            ALUM1515NA
ALOOPUM1515
                RRF             BARGB1
                BTFSS           _C
                GOTO            ALUM1515NA
                MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
```

```
ALUM1515NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            ALOOPUM1515
                MOVLW           0x07
                MOVWF           LOOPCOUNT
BLOOPUM1515
                RRF             BARGB0
                BTFSS           _C
                GOTO            BLUM1515NA
                MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
BLUM1515NA
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            BLOOPUM1515

                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                endm

SMUL1616        macro
;       Max Timing:     5+6+7*11+7*12+4 = 176 clks
;       Min Timing:     5+24+21+5 = 55 clks
;       PM: 5+3*8+3*7+6+7*11+7*12+4 = 221            DM: 8
                variable i
                i = 0
                BTFSC           AARGB0,MSB
                COMF            ACCB2
                MOVF            ACCB2,W
                MOVWF           ACCB3
                RLF             ACCB0,W

                while i < 8

                BTFSC           BARGB1,i
                GOTO            SM1616NA#v(i)
                BCF             ACCB2,7-i
                i = i + 1
                endw
                i = 8

                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            SM1616NA#v(i)
                BCF             ACCB2,15-i
                i = i + 1
                endw
                CLRF            ACCB0          ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB3
                RETEW           0
SM1616NA0
                RRF             ACCB0
```

```
                RRF             ACCB1
                RRF             ACCB2
                i = 1
                while   i < 8
                BTFSS           BARGB1,i
                GOTO            SM1616NA#v(i)
SM1616A#v(i)    MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM1616NA#v(i)
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                i = i + 1
                endw
                i = 8
                while   i < 15
                BTFSS           BARGB0,i-8
                GOTO            SM1616NA#v(i)
SM1616A#v(i)    MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM1616NA#v(i)
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                endm
UMUL1616        macro
;       Max Timing:     1+6+7*11+8*12 = 180 clks
;       Min Timing:     1+2*8+2*8+4 = 37 clks
;       PM: 1+2*8+2*8+4+7*11+8*12 = 210           DM: 8
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM1616NA#v(i)
                i = i + 1
                endw
                i = 8

                while i < 16

                BTFSC           BARGB0,i-8
                GOTO            UM1616NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                RETEW           0
UM1616NA0       RRF             ACCB0
```

```
                RRF             ACCB1
                RRF             ACCB2
                i = 1
                while   i < 8
                BTFSS           BARGB1,i
                GOTO            UM1616NA#v(i)
UM1616A#v(i)    MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM1616NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                i = i + 1
                endw
                i = 8
                while   i < 16
                BTFSS           BARGB0,i-8
                GOTO            UM1616NA#v(i)
UM1616A#v(i)    MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM1616NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = i + 1
                endw
                endm
UMUL1515        macro
;       Max Timing:     7+7*11+7*12+4 = 172 clks
;       Min Timing:     1+16+14+4 = 35 clks
;       PM: 1+2*8+2*7+6+7*11+7*12+4 = 202           DM: 8
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                while i < 8

                BTFSC           BARGB1,i
                GOTO            UM1515NA#v(i)
                i = i + 1
                endw
                i = 8
                while i < 15

                BTFSC           BARGB0,i-8
                GOTO            UM1515NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                RETEW           0
UM1515NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                i = 1
                while   i < 8
                BTFSS           BARGB1,i
                GOTO            UM1515NA#v(i)
UM1515A#v(i)    MOVF            TEMPB1,W
                ADDWF           ACCB1
```

```
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM1515NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                i = i + 1
                endw
                i = 8
                while   i < 15
                BTFSS           BARGB0,i-8
                GOTO            UM1515NA#v(i)
UM1515A#v(i)    MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM1515NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                RRF             ACCB3
                endm

;*****************************************************************************************
;*****************************************************************************************

;       16x16 Bit Signed Fixed Point Multiply 16x16 -> 32
;       Input:  16 bit signed fixed point multiplicand in AARGB0
;                       16 bit signed fixed point multiplier in BARGB0
;       Use:    CALL    FXM1616S
;       Output: 32 bit signed fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     9+249+2 = 260 clks                B > 0
;                       18+249+2 = 269 clks               B < 0
;       Min Timing:     9+96 = 105 clks
;       PM: 18+55+1 = 74              DM: 9
FXM1616S        BTFSS           BARGB0,MSB
                GOTO            M1616SOK
                COMF            BARGB1          ; make multiplier BARG > 0
                INCF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
                COMF            AARGB1
                INCF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
M1616SOK        CLRF            ACCB2           ; clear partial product
                CLRF            ACCB3
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                SMUL1616L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
```

```
;       16x16 Bit Unsigned Fixed Point Multiply 16x16 -> 32
;       Input:  16 bit unsigned fixed point multiplicand in AARGB0
;                       16 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM1616U
;       Output: 32 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     6+248+2 = 256 clks
;       Min Timing:     6+101 = 107 clks
;       PM: 6+51+1 = 58              DM: 9
FXM1616U
                CLRF            ACCB2           ; clear partial product
                CLRF            ACCB3
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                UMUL1616L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       15x15 Bit Unsigned Fixed Point Divide 15x15 -> 30
;       Input:  15 bit unsigned fixed point multiplicand in AARGB0
;                       15 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM1515U
;       Output: 30 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     6+236+2 = 244 clks
;       Min Timing:     6+97 = 103 clks
;       PM: 6+56+1 = 63              DM: 9
FXM1515U
                CLRF            ACCB2           ; clear partial product
                CLRF            ACCB3
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                UMUL1515L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

C.7 16x8 PIC16C5X/PIC16CXX Fixed Point Multiply Routines

```
;       16x8 PIC16 FIXED POINT MULTIPLY ROUTINES        VERSION 1.2
;       Input:  fixed point arguments in AARG and BARG
;       Output: product AARGxBARG in AARG
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXM1608S     128        16x08 -> 24 bit signed fixed point multiply
;       FXM1608U     126        16x08 -> 24 bit unsigned fixed point multiply
;       FXM1507U     114        15x07 -> 22 bit unsigned fixed point multiply
;       The above timings are based on the looped macros. If space permits,
;       approximately 24-35 clocks can be saved by using the unrolled macros.
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number generator registers
```

```
RANDLO          equ     0x1B
TESTCOUNT       equ     0x20    ; counter
DATA            equ     0x21    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 16x8 bit fixed point multiply algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            TFXM1608
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETEW           0
;       Test suite for FXM1608
TFXM1608        MOVLW           2
                MOVWF           TESTCOUNT
M1608LOOP
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
                BCF             BARGB0,MSB
                MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
                BCF             AARGB0,MSB
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
```

```
                CALL            FXM1507U
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                DECFSZ          TESTCOUNT
                GOTO            M1608LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       16x08 Bit Multiplication Macros
SMUL1608L       macro
;       Max Timing:     2+11+5*16+15+4 = 112 clks
;       Min Timing:     2+6*6+5+4 = 47 clks
;       PM: 29            DM: 7
                MOVLW           0x07
                MOVWF           LOOPCOUNT
LOOPSM1608A
                RRF             BARGB0
                BTFSC           _C
                GOTO            LSM1608NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM1608A
                CLRF            AARGB0
                CLRF            AARGB1
                RETLW           0x00
LOOPSM1608
                RRF             BARGB0
                BTFSS           _C
                GOTO            LSM1608NA
                MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
LSM1608NA       RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM1608
                RLF             TEMPB0,W
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                endm
UMUL1608L       macro
;       Max Timing:     2+13+6*15+14 = 119 clks
;       Min Timing:     2+7*6+5+4 = 54 clks
;       PM: 26            DM: 7
                MOVLW           0x08
                MOVWF           LOOPCOUNT
LOOPUM1608A
                RRF             BARGB0
                BTFSC           _C
                GOTO            LUM1608NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM1608A
                CLRF            AARGB0
                CLRF            AARGB1
```

```
                RETLW           0x00
LUM1608NAP
                BCF             _C
                GOTO            LUM1608NA
LOOPUM1608
                RRF             BARGB0
                BTFSS           _C
                GOTO            LUM1608NA
                MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
LUM1608NA       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM1608
                endm
UMUL1507L       macro
;       Max Timing:     2+13+5*15+14+3 = 107 clks
;       Min Timing:     2+6*6+5+4 = 47 clks
;       PM: 29            DM: 7
                MOVLW           0x07
                MOVWF           LOOPCOUNT
LOOPUM1507A
                RRF             BARGB0
                BTFSC           _C
                GOTO            LUM1507NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM1507A
                CLRF            AARGB0
                CLRF            AARGB1
                RETLW           0x00
LUM1507NAP
                BCF             _C
                GOTO            LUM1507NA
LOOPUM1507
                RRF             BARGB0
                BTFSS           _C
                GOTO            LUM1507NA
                MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
LUM1507NA       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM1507
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                endm

SMUL1608        macro
;       Max Timing:     3+6+6*11+3 = 78 clks
;       Min Timing:     3+21+5 = 29 clks
;       PM: 3+3*7+7+6*11+3 = 100            DM: 6
                variable i
                i = 0
                BTFSC           AARGB0,MSB
                COMF            ACCB2
```

```
                RLF             ACCB0,W

                while i < 7

                BTFSC           BARGB0,i
                GOTO            SM1608NA#v(i)
                BCF             ACCB2,7-i
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETEW           0
SM1608NA0
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                i = 1
                while   i < 7
                BTFSS           BARGB0,i
                GOTO            SM1608NA#v(i)
SM1608A#v(i)    MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
SM1608NA#v(i)
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                endm
UMUL1608        macro
;       Max Timing:     1+6+7*11 = 84 clks
;       Min Timing:     1+2*8+4 = 21 clks
;       PM: 1+2*8+4+6*7 = 63            DM: 4
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift

                while i < 8

                BTFSC           BARGB0,i
                GOTO            UM1608NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                RETEW           0
UM1608NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                i = 1
                while   i < 8
                BTFSS           BARGB0,i
                GOTO            UM1608NA#v(i)
UM1608A#v(i)    MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
```

```
                ADDWF           ACCB0
UM1608NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                i = i + 1
                endw
                endm
UMUL1507        macro
;       Max Timing:     7+6*12+4 = 83 clks
;       Min Timing:     14+3 = 17 clks
;       PM: 2*7+7+6*12+4 = 97           DM: 6
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                while i < 7

                BTFSC           BARGB0,i
                GOTO            UM1507NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                RETEW           0
UM1507NA0       RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                i = 1
                while   i < 7
                BTFSS           BARGB0,i
                GOTO            UM1507NA#v(i)
UM1507A#v(i)    MOVF            TEMPB1,W
                ADDWF           ACCB1
                MOVF            TEMPB0,W
                BTFSC           _C
                INCFSZ          TEMPB0,W
                ADDWF           ACCB0
UM1507NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                RRF             ACCB2
                endm

;*************************************************************************************************
;*************************************************************************************************

;       16x8 Bit Signed Fixed Point Multiply 16x8 -> 24
;       Input:  16 bit signed fixed point multiplicand in AARGB0
;                       8 bit signed fixed point multiplier in BARGB0
;       Use:    CALL    FXM1608S
;       Output: 24 bit signed fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     8+112+2 = 122 clks              B > 0
;                       14+112+2 = 128 clks              B < 0
;       Min Timing:     8+47 = 55 clks
;       PM: 14+29+1 = 44             DM: 7
FXM1608S        BTFSS           BARGB0,MSB
                GOTO            M1608SOK
                COMF            BARGB0          ; make multiplier BARG > 0
                INCF            BARGB0
                COMF            AARGB1
                INCF            AARGB1
                BTFSC           _Z
```

```
                DECF            AARGB0
                COMF            AARGB0
M1608SOK        CLRF            ACCB2           ; clear partial product
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                SMUL1608L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16x8 Bit Unsigned Fixed Point Multiply 16x8 -> 24
;       Input:  16 bit unsigned fixed point multiplicand in AARGB0
;                       8 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM1608U
;       Output: 24 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     5+119+2 = 126 clks
;       Min Timing:     5+54 = 59 clks
;       PM: 5+26+1 = 31             DM: 7
FXM1608U        CLRF            ACCB2           ; clear partial product
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                UMUL1608L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       15x7 Bit Unsigned Fixed Point Divide 15x7 -> 22
;       Input:  15 bit unsigned fixed point multiplicand in AARGB0
;                       7 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM1507U
;       Output: 22 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     5+107+2 = 114 clks
;       Min Timing:     5+47 = 52 clks
;       PM: 5+29+1 = 35             DM: 7
FXM1507U        CLRF            ACCB2           ; clear partial product
                MOVF            AARGB0,W
                MOVWF           TEMPB0
                MOVF            AARGB1,W
                MOVWF           TEMPB1
                UMUL1507
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

C.8 8x8 PIC16C5X/PIC16CXX Fixed Point Multiply Routines

```
;       8x8 PIC16 FIXED POINT MULTIPLY ROUTINES VERSION 1.2
;       Input:  fixed point arguments in AARG and BARG
;       Output: product AARGxBARG in AARG
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed multiply application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine           Clocks     Function
;       FXM0808S     82 08x08 -> 16 bit signed fixed point multiply
;       FXM0808U     73 08x08 -> 16 bit unsigned fixed point multiply
;       FXM0707U     67 07x07 -> 14 bit unsigned fixed point multiply
;       The above timings are based on the looped macros. If space permits,
;       approximately 29-35 clocks can be saved by using the unrolled macros.
```

```
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number generator registers
RANDLO          equ     0x1B
TESTCOUNT       equ     0x20    ; counter
DATA            equ     0x21    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 8x8 bit fixed point multiply algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            TFXM0808
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETEW           0
;       Test suite for FXM0808
TFXM0808        MOVLW           3
                MOVWF           TESTCOUNT
M0808LOOP
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
                BCF             BARGB0,MSB
                MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
                BCF             AARGB0,MSB
```

```
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                CALL            FXM0707U
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                DECFSZ          TESTCOUNT
                GOTO            M0808LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       08x08 Bit Multiplication Macros
SMUL0808L       macro
;       Max Timing:     7+10+5*9+8+3 = 73 clks
;       Min Timing:     7+6*6+5+3 = 51 clks
;       PM: 25            DM: 5
                MOVLW           0x07
                MOVWF           LOOPCOUNT
                CLRW
                BTFSC           AARGB0,MSB
                MOVLW           0xFF
                MOVWF           SIGN
                MOVF            AARGB0,W
LOOPSM0808A
                RRF             BARGB0
                BTFSC           _C
                GOTO            LSM0808NA
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM0808A
                CLRF            AARGB0
                RETLW           0x00
LOOPSM0808
                RRF             BARGB0
                BTFSC           _C
                ADDWF           ACCB0
LSM0808NA       RLF             SIGN
                RRF             ACCB0
                RRF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPSM0808
                RLF             SIGN
                RRF             ACCB0
                RRF             ACCB1
                endm
UMUL0808L       macro
;       Max Timing:     3+12+6*8+7 = 70 clks
;       Min Timing:     3+7*6+5+3 = 53 clks
;       PM: 19            DM: 4
                MOVLW           0x08
                MOVWF           LOOPCOUNT
                MOVF            AARGB0,W
LOOPUM0808A
                RRF             BARGB0
                BTFSC           _C
                GOTO            LUM0808NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM0808A
                CLRF            AARGB0
                RETLW           0x00
LUM0808NAP
                BCF             _C
                GOTO            LUM0808NA
```

```
LOOPUM0808
                RRF             BARGB0
                BTFSC           _C
                ADDWF           ACCB0
LUM0808NA       RRF             ACCB0
                RRF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM0808
                endm
UMUL0707L       macro
;       Max Timing:     3+12+5*8+7+2 = 64 clks
;       Min Timing:     3+6*6+5+3 = 47 clks
;       PM: 21            DM: 4
                MOVLW           0x07
                MOVWF           LOOPCOUNT
                MOVF            AARGB0,W
LOOPUM0707A
                RRF             BARGB0
                BTFSC           _C
                GOTO            LUM0707NAP
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM0707A
                CLRF            AARGB0
                RETLW           0x00
LUM0707NAP
                BCF             _C
                GOTO            LUM0707NA
LOOPUM0707
                RRF             BARGB0
                BTFSC           _C
                ADDWF           ACCB0
LUM0707NA       RRF             ACCB0
                RRF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPUM0707
                RRF             ACCB0
                RRF             ACCB1
                endm

SMUL0808        macro
;       Max Timing:     5+6+6*5+3 = 44 clks
;       Min Timing:     5+14+3 = 22 clks
;       PM: 5+2*7+5+6*5+3 = 57            DM: 5
                variable i
                i = 0
                CLRW
                BTFSC           AARGB0,MSB
                MOVLW           0xFF
                MOVWF           SIGN
                MOVF            AARGB0,W

                while i < 7

                BTFSC           BARGB0,i
                GOTO            SM0808NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0          ; if we get here, BARG = 0
                RETEW           0
SM0808NA0       RLF             SIGN
                RRF             ACCB0
                RRF             ACCB1
                i = 1
                while   i < 7
                BTFSC           BARGB0,i
                ADDWF           ACCB0
```

```
SM0808NA#v(i)   RLF             SIGN
                RRF             ACCB0
                RRF             ACCB1
                i = i + 1
                endw
                RLF             SIGN
                RRF             ACCB0
                RRF             ACCB1
                endm
UMUL0808        macro
;       Max Timing:     2+5+7*4 = 35 clks
;       Min Timing:     2+16+3 = 21 clks
;       PM: 2+2*8+4+7*4 = 50            DM: 3
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                MOVF            AARGB0,W

                while i < 8

                BTFSC           BARGB0,i
                GOTO            UM0808NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                RETEW           0
UM0808NA0       RRF             ACCB0
                RRF             ACCB1
                i = 1
                while   i < 8
                BTFSC           BARGB0,i
                ADDWF           ACCB0
UM0808NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                i = i + 1
                endw
                endm
UMUL0707        macro
;       Max Timing:     2+5+6*4+2 = 33 clks
;       Min Timing:     2+14+3 = 19 clks
;       PM: 2+2*7+4+6*4+2 = 46            DM: 3
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                MOVF            AARGB0,W
                while i < 7

                BTFSC           BARGB0,i
                GOTO            UM0707NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                RETEW           0
UM0707NA0       RRF             ACCB0
                RRF             ACCB1
                i = 1
                while   i < 7
                BTFSC           BARGB0,i
                ADDWF           ACCB0
UM0707NA#v(i)   RRF             ACCB0
                RRF             ACCB1
                i = i + 1
                endw
                RRF             ACCB0
                RRF             ACCB1
                endm
```

```
;*****************************************************************************************
;*****************************************************************************************

;       8x8 Bit Signed Fixed Point Multiply 8x8 -> 16
;       Input:  8 bit signed fixed point multiplicand in AARGB0
;                       8 bit signed fixed point multiplier in BARGB0
;       Use:    CALL    FXM0808S
;       Output: 16 bit signed fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     4+73+2 = 79 clks                B > 0
;                       7+73+2 = 82 clks                B < 0
;       Min Timing:     4+51 = 55 clks
;       PM: 7+25+1 = 33             DM: 5
FXM0808S        BTFSS           BARGB0,MSB
                GOTO            M0808SOK
                COMF            BARGB0          ; make multiplier BARG > 0
                INCF            BARGB0
                COMF            AARGB0
                INCF            AARGB0
M0808SOK        CLRF            ACCB1           ; clear partial product
                SMUL0808L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       8x8 Bit Unsigned Fixed Point Multiply 8x8 -> 16
;       Input:  8 bit unsigned fixed point multiplicand in AARGB0
;               8 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM0808U
;       Output: 8 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     1+70+2 = 73 clks
;       Min Timing:     1+53 = 54 clks
;       PM: 1+19+1 = 21             DM: 4
FXM0808U        CLRF            ACCB1           ; clear partial product
                UMUL0808L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       7x7 Bit Unsigned Fixed Point Divide 7x7 -> 14
;       Input:  7 bit unsigned fixed point multiplicand in AARGB0
;                       7 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM0707U
;       Output: 14 bit unsigned fixed point product in AARGB0
;       Result: AARG  <--  AARG x BARG
;       Max Timing:     1+64+2 = 67 clks
;       Min Timing:     1+47 = 48 clks
;       PM: 1+21+1 = 23             DM: 4
FXM0707U        CLRF            ACCB1           ; clear partial product
                UMUL0707L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

NOTES:

Please check the Microchip BBS for the latest version of the source code. For BBS access information, see Section 6, Microchip Bulletin Board Service information, page 6-3.

APPENDIX D: PIC16C5X/PIC16CXX DIVIDE ROUTINES

Table of Contents for Appendix D

D.1 32/32 PIC16C5X/PIC16CXX Fixed Point Divide Routines

```
;       32/32 PIC16 FIXED POINT DIVIDE ROUTINES VERSION 1.7
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in performance.
;         Routine   Clocks     Function
;       FXD3232S    912         32 bit/32 bit -> 32.32 signed fixed point divide
;       FXD3232U    1031        32 bit/32 bit -> 32.32 unsigned fixed point divide
;       FXD3131U    869         31 bit/31 bit -> 31.31 unsigned fixed point divide
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;
;       Test suite storage
RANDHI          equ     0x1E    ; random number senerator registers
RANDLO          equ     0x1F
DATA            equ     0x20
;*****************************************************************************************
;*****************************************************************************************
;
;       Test suite for 32/32 bit fixed point divide algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
;               BCF             BARGB0,MSB
;               MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
```

```
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB2
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB3
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
;               BCF             AARGB0,MSB
;               MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16

                MOVF            RANDHI,W
                MOVWF           AARGB2
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB3
                MOVWF           INDF
                INCF            FSR
                CALL            FXD3232S
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB3,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB3,W
                MOVWF           INDF
                INCF            FSR
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
```

```
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETLW           0
;*****************************************************************************************
;*****************************************************************************************
;       32/32 Bit Division Macros
SDIV3232L       macro
;       Max Timing:     17+6*27+26+26+6*27+26+26+6*27+26+26+6*27+26+16 = 863 clks
;       Min Timing:     17+6*26+25+25+6*26+25+25+6*26+25+25+6*26+25+3 = 819 clks
;       PM: 17+7*38+16 = 299                                    DM: 13
                MOVF            BARGB3,W
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3232A      RLF             ACCB0,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB0,LSB
                GOTO            SADD22LA
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK22LA
SADD22LA        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
```

```
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK22LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3232A
                RLF             ACCB1,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB0,LSB
                GOTO            SADD22L8
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK22L8
SADD22L8        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK22L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3232B      RLF             ACCB1,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB1,LSB
                GOTO            SADD22LB
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
```

```
                SUBWF           REMB0
                GOTO            SOK22LB
SADD22LB        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK22LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3232B
                RLF             ACCB2,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB1,LSB
                GOTO            SADD22L16
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK22L16
SADD22L16       ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK22L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3232C      RLF             ACCB2,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB2,LSB
                GOTO            SADD22LC
```

```
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK22LC
SADD22LC        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK22LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3232C
                RLF             ACCB3,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB2,LSB
                GOTO            SADD22L24
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK22L24
SADD22L24       ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
```

```
SOK22L24        RLF             ACCB3
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3232D      RLF             ACCB3,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB3,LSB
                GOTO            SADD22LD
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK22LD
SADD22LD        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK22LD         RLF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3232D
                BTFSC           ACCB3,LSB
                GOTO            SOK22L
                MOVF            BARGB3,W
                ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCF            BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
SOK22L
                endm
UDIV3232L       macro
;       Max Timing:     24+6*32+31+31+6*32+31+31+6*32+31+31+6*32+31+16 = 1025 clks
;       Min Timing:     24+6*31+30+30+6*31+30+30+6*31+30+30+6*31+30+3 = 981 clks
;       PM: 359                                 DM: 13
                CLRF            TEMP
```

```
                RLF             ACCB0,W
                RLF             REMB3
                MOVF            BARGB3,W
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3232A      RLF             ACCB0,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB3,W
                BTFSS           ACCB0,LSB
                GOTO            UADD22LA
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK22LA
UADD22LA        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
```

```
                ADDWF           TEMP

UOK22LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3232A
                RLF             ACCB1,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB3,W
                BTFSS           ACCB0,LSB
                GOTO            UADD22L8
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK22L8
UADD22L8        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK22L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3232B      RLF             ACCB1,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB3,W
                BTFSS           ACCB1,LSB
                GOTO            UADD22LB
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
```

```
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK22LB
UADD22LB        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK22LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3232B
                RLF             ACCB2,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB3,W
                BTFSS           ACCB1,LSB
                GOTO            UADD22L16
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK22L16
UADD22L16       ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
```

```
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK22L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3232C      RLF             ACCB2,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB3,W
                BTFSS           ACCB2,LSB
                GOTO            UADD22LC
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK22LC
UADD22LC        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK22LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3232C
                RLF             ACCB3,W
```

```
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB3,W
                BTFSS           ACCB2,LSB
                GOTO            UADD22L24
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK22L24
UADD22L24       ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK22L24        RLF             ACCB3
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3232D      RLF             ACCB3,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB3,W
                BTFSS           ACCB3,LSB
                GOTO            UADD22LD
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
```

```
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK22LD
UADD22LD        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK22LD         RLF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3232D
                BTFSC           ACCB3,LSB
                GOTO            UOK22L
                MOVF            BARGB3,W
                ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCF            BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
UOK22L
                endm
UDIV3131L       macro
;       Max Timing:     17+6*27+26+26+6*27+26+26+6*27+26+26+6*27+26+16 = 863 clks
;       Min Timing:     17+6*26+25+25+6*26+25+25+6*26+25+25+6*26+25+3 = 819 clks
;       PM: 17+7*38+16 = 299                                    DM: 13
                MOVF            BARGB3,W
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
```

```
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3131A      RLF             ACCB0,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB0,LSB
                GOTO            UADD11LA
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK11LA
UADD11LA        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK11LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3131A
                RLF             ACCB1,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB0,LSB
                GOTO            UADD11L8
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK11L8
UADD11L8        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
```

```
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK11L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3131B      RLF             ACCB1,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB1,LSB
                GOTO            UADD11LB
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK11LB
UADD11LB        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK11LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3131B
                RLF             ACCB2,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB1,LSB
                GOTO            UADD11L16
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
```

```
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK11L16
UADD11L16       ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK11L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3131C      RLF             ACCB2,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB2,LSB
                GOTO            UADD11LC
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK11LC
UADD11LC        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK11LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3131C
                RLF             ACCB3,W
```

```
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB2,LSB
                GOTO            UADD11L24
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK11L24
UADD11L24       ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK11L24        RLF             ACCB3
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3131D      RLF             ACCB3,W
                RLF             REMB3
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB3,W
                BTFSS           ACCB3,LSB
                GOTO            UADD11LD
                SUBWF           REMB3
                MOVF            BARGB2,W
                BTFSS           _C
                INCFSZ          BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK11LD
UADD11LD        ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCFSZ          BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
```

```
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK11LD         RLF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3131D
                BTFSC           ACCB3,LSB
                GOTO            UOK11L
                MOVF            BARGB3,W
                ADDWF           REMB3
                MOVF            BARGB2,W
                BTFSC           _C
                INCF            BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
UOK11L
                endm
;*****************************************************************************************
;*****************************************************************************************
;
;       32/32 Bit Signed Fixed Point Divide 32/32 -> 32.32
;       Input:  32 bit fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3
;       Use:    CALL    FXD3232S
;       Output: 32 bit fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit fixed point remainder in REMB0, REMB1, REMB2, REMB3
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     13+863+3 = 879 clks             A > 0, B > 0
;                       23+863+26 = 912 clks            A > 0, B < 0
;                       23+863+26 = 912 clks            A < 0, B > 0
;                       33+863+3 = 899 clks             A < 0, B < 0
;       Min Timing:     13+819+3 = 835 clks             A > 0, B > 0
;                       23+819+26 = 868 clks            A > 0, B < 0
;                       23+819+26 = 868 clks            A < 0, B > 0
;                       33+819+3 = 855 clks             A < 0, B < 0
;       PM: 33+299+25 = 357                             DM: 13
FXD3232S        MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           SIGN
                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA3232S
                COMF            BARGB3
                INCF            BARGB3
                BTFSC           _Z
                DECF            BARGB2
                COMF            BARGB2
                BTFSC           _Z
                DECF            BARGB1
                COMF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
CA3232S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C3232S
                COMF            AARGB3
```

```
                INCF            AARGB3
                BTFSC           _Z
                DECF            AARGB2
                COMF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
C3232S          CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                CLRF            REMB3
                SDIV3232L
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB3
                INCF            AARGB3
                BTFSC           _Z
                DECF            AARGB2
                COMF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
                COMF            REMB3
                INCF            REMB3
                BTFSC           _Z
                DECF            REMB2
                COMF            REMB2
                BTFSC           _Z
                DECF            REMB1
                COMF            REMB1
                BTFSC           _Z
                DECF            REMB0
                COMF            REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32/32 Bit Unsigned Fixed Point Divide 32/32 -> 32.32
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3
;       Use:    CALL    FXD3232U
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit unsigned fixed point remainder in REMB0, REMB1, REMB2, REMB3
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     4+1025+2 = 1031 clks
;       Max Timing:     4+981+2 = 987 clks
;       PM: 4+359+1 = 364               DM: 13
FXD3232U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                CLRF            REMB3
                UDIV3232L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       31/31 Bit Unsigned Fixed Point Divide 31/31 -> 31.31
;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point divisor in BARGB0, BARGB1, BARBB2, BARGB3
;       Use:    CALL    FXD3131U
```

```
;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point remainder in REMB0, REMB1, REMB2, REMB3
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     4+863+2 = 869 clks
;       Min Timing:     4+819+2 = 825 clks
;       PM: 4+299+1 = 304               DM: 13
FXD3131U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                CLRF            REMB3
                UDIV3131L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

D.2 32/24 PIC16C5X/PIC16CXX Fixed Point Divide Routines

```
;       32/24 PIC16 FIXED POINT DIVIDE ROUTINES VERSION 1.7
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine   Clocks     Function
;       FXD3224S    742         32 bit/24 bit -> 32.24 signed fixed point divide
;       FXD3224U    867         32 bit/24 bit -> 32.24 unsigned fixed point divide
;       FXD3123U    705         31 bit/23 bit -> 31.23 unsigned fixed point divide
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1E    ; random number senerator registers
RANDLO          equ     0x1F
DATA            equ     0x20
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 32/24 bit fixed point divide algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
;               BCF             BARGB0,MSB
;               MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
```

```
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB2
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
;               BCF             AARGB0,MSB
;               MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16

                MOVF            RANDHI,W
                MOVWF           AARGB2
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB3
                MOVWF           INDF
                INCF            FSR
                CALL            FXD3224S
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB3,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB2,W
                MOVWF           INDF
                INCF            FSR
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
```

```
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETLW           0
;*****************************************************************************************
;*****************************************************************************************
;       32/24 Bit Division Macros
SDIV3224L       macro
;       Max Timing:     13+6*22+21+21+6*22+21+21+6*22+21+21+6*22+21+12 = 700 clks
;       Min Timing:     13+6*21+20+20+6*21+20+20+6*21+20+20+6*21+20+3 = 660 clks
;       PM: 11+3*58+43 = 228                                    DM: 10
                MOVF            BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3224A      RLF             ACCB0,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            SADD24LA
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK24LA
SADD24LA        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK24LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3224A
                RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            SADD24L8
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
```

```
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK24L8
SADD24L8        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK24L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3224B      RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            SADD24LB
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK24LB
SADD24LB        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK24LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3224B
                RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            SADD24L16
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK24L16
```

```
SADD24L16       ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK24L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3224C      RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB2,LSB
                GOTO            SADD24LC
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK24LC
SADD24LC        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK24LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3224C
                RLF             ACCB3,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB2,LSB
                GOTO            SADD24L24
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK24L24
SADD24L24       ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
```

```
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK24L24        RLF             ACCB3
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3224D      RLF             ACCB3,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB3,LSB
                GOTO            SADD24LD
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK24LD
SADD24LD        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK24LD         RLF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3224D
                BTFSC           ACCB3,LSB
                GOTO            SOK24L
                MOVF            BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
SOK24L
                endm
UDIV3224L       macro
;       Max Timing:     20+6*27+26+26+6*27+26+26+6*27+26+26+6*27+26+12 = 862 clks
;       Min Timing:     20+6*26+25+25+6*26+25+25+6*26+25+25+6*26+25+3 = 822 clks
;       PM: 18+3*75+40+12 = 295                                 DM: 11
                CLRF            TEMP
                RLF             ACCB0,W
                RLF             REMB2
                MOVF            BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
```

```
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3224A      RLF             ACCB0,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            UADD24LA
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK24LA
UADD24LA        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK24LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3224A
                RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            UADD24L8
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
```

```
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK24L8
UADD24L8        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK24L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3224B      RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            UADD24LB
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK24LB
UADD24LB        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK24LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3224B
                RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
```

```
                BTFSS           ACCB1,LSB
                GOTO            UADD24L16
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK24L16
UADD24L16       ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK24L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3224C      RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB2,LSB
                GOTO            UADD24LC
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK24LC
UADD24LC        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
```

```
                MOVLW           1
                ADDWF           TEMP

UOK24LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3224C
                RLF             ACCB3,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB2,LSB
                GOTO            UADD24L24
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK24L24
UADD24L24       ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK24L24        RLF             ACCB3
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3224D      RLF             ACCB3,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB3,LSB
                GOTO            UADD24LD
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
```

```
                GOTO            UOK24LD
UADD24LD        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK24LD         RLF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3224D
                BTFSC           ACCB3,LSB
                GOTO            UOK24L
                MOVF            BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
UOK24L
                endm
UDIV3123L       macro
;       Max Timing:     13+6*22+21+21+6*22+21+21+6*22+21+21+6*22+21+12 = 700 clks
;       Min Timing:     13+6*21+20+20+6*21+20+20+6*21+20+20+6*21+20+3 = 660 clks
;       PM: 11+3*58+43 = 228                                        DM: 10
                MOVF            BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3123A      RLF             ACCB0,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            UADD13LA
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK13LA
```

```
UADD13LA        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK13LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3123A
                RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            UADD13L8
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK13L8
UADD13L8        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK13L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3123B      RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            UADD13LB
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK13LB
UADD13LB        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
```

```
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK13LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3123B
                RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            UADD13L16
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK13L16
UADD13L16       ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK13L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3123C      RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB2,LSB
                GOTO            UADD13LC
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK13LC
UADD13LC        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK13LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
```

```
                GOTO            LOOPU3123C
                RLF             ACCB3,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB2,LSB
                GOTO            UADD13L24
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK13L24
UADD13L24       ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK13L24        RLF             ACCB3
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3123D      RLF             ACCB3,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB3,LSB
                GOTO            UADD13LD
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK13LD
UADD13LD        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK13LD         RLF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3123D
                BTFSC           ACCB3,LSB
                GOTO            UOK13L
                MOVF            BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
```

```
                BTFSC           _C
                INCF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
UOK13L
                endm
;*****************************************************************************************
;*****************************************************************************************

;       32/24 Bit Signed Fixed Point Divide 32/24 -> 32.24
;       Input:  32 bit fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD3224S
;       Output: 32 bit fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     12+700+3 = 715 clks             A > 0, B > 0
;                       19+700+23 = 742 clks            A > 0, B < 0
;                       19+700+23 = 742 clks            A < 0, B > 0
;                       29+700+3 = 732 clks             A < 0, B < 0
;       Min Timing:     12+660+3 = 675 clks             A > 0, B > 0
;                       19+660+23 = 702 clks            A > 0, B < 0
;                       19+660+23 = 702 clks            A < 0, B > 0
;                       29+660+3 = 692 clks             A < 0, B < 0
;       PM: 29+228+23 = 280                             DM: 11
FXD3224S        MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           SIGN
                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA3224S
                COMF            BARGB2
                INCF            BARGB2
                BTFSC           _Z
                DECF            BARGB1
                COMF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
CA3224S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C3224S
                COMF            AARGB3
                INCF            AARGB3
                BTFSC           _Z
                DECF            AARGB2
                COMF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
C3224S          CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                SDIV3224L
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB3
                INCF            AARGB3
                BTFSC           _Z
                DECF            AARGB2
                COMF            AARGB2
                BTFSC           _Z
```

```
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
                COMF            REMB2
                INCF            REMB2
                BTFSC           _Z
                DECF            REMB1
                COMF            REMB1
                BTFSC           _Z
                DECF            REMB0
                COMF            REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32/24 Bit Unsigned Fixed Point Divide 32/24 -> 32.24
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD3224U
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     3+862+2 = 867 clks
;       Min Timing:     3+822+2 = 827 clks
;       PM: 3+295+1 = 299               DM: 11
FXD3224U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                UDIV3224L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       31/23 Bit Unsigned Fixed Point Divide 31/23 -> 31.23
;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARBB2
;       Use:    CALL    FXD3123U
;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     3+700+2 = 705 clks
;       Min Timing:     3+660+2 = 665 clks
;       PM: 3+228+1 = 232               DM: 10
FXD3123U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                UDIV3123L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

D.3 32/16 PIC16C5X/PIC16CXX Fixed Point Divide Routines

```
;       32/16 PIC16 FIXED POINT DIVIDE ROUTINES VERSION 1.7
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXD3216S    578 32 bit/16 bit -> 32.16 signed fixed point divide
;       FXD3216U    703 32 bit/16 bit -> 32.16 unsigned fixed point divide
```

```
;       FXD3115U    541 31 bit/15 bit -> 31.15 unsigned fixed point divide
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1E    ; random number senerator registers
RANDLO          equ     0x1F
TDATA           equ     0x20
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 32/16 bit fixed point divide algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           TDATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            RANDOM16
;               SWAPF           RANDHI
;               SWAPF           RANDLO
                MOVF            RANDHI,W
                MOVWF           BARGB0
;               BCF             BARGB0,MSB
;               MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16

;               SWAPF           RANDHI
;               SWAPF           RANDLO
                MOVF            RANDHI,W
                MOVWF           AARGB0
;               BCF             AARGB0,MSB
;               MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16

                MOVF            RANDHI,W
                MOVWF           AARGB2
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB3
```

```
                MOVWF           INDF
                INCF            FSR
                CALL            FXD3216S
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB3,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB1,W
                MOVWF           INDF
                INCF            FSR
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETLW           0
;*****************************************************************************************
;*****************************************************************************************
;       32/16 Bit Division Macros
SDIV3216L       macro
;       Max Timing:     9+6*17+16+16+6*17+16+16+6*17+16+16+6*17+16+8 = 537 clks
;       Min Timing:     9+6*16+15+15+6*16+15+15+6*16+15+15+6*16+15+3 = 501 clks
;       PM: 157                                 DM: 9
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3216A      RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            SADD26LA
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
```

```
                SUBWF           REMB0
                GOTO            SOK26LA
SADD26LA        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK26LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3216A
                RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            SADD26L8
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK26L8
SADD26L8        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK26L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3216B      RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            SADD26LB
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK26LB
SADD26LB        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK26LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3216B
                RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            SADD26L16
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK26L16
SADD26L16       ADDWF           REMB1
```

```
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK26L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3216C      RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB2,LSB
                GOTO            SADD26LC
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK26LC
SADD26LC        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK26LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3216C
                RLF             ACCB3,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB2,LSB
                GOTO            SADD26L24
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK26L24
SADD26L24       ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK26L24        RLF             ACCB3
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS3216D      RLF             ACCB3,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB3,LSB
                GOTO            SADD26LD
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK26LD
SADD26LD        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
```

```
                ADDWF           REMB0

SOK26LD         RLF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS3216D
                BTFSC           ACCB3,LSB
                GOTO            SOK26L
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
SOK26L
                endm
UDIV3216L       macro
;       Max Timing:     16+6*22+21+21+6*22+21+21+6*22+21+21+6*22+21+8 = 699 clks
;       Min Timing:     16+6*21+20+20+6*21+20+20+6*21+20+20+6*21+20+3 = 663 clks
;       PM: 240                                 DM: 9
                CLRF            TEMP
                RLF             ACCB0,W
                RLF             REMB1
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3216A      RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            UADD26LA
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK26LA
UADD26LA        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK26LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3216A
                RLF             ACCB1,W
```

```
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            UADD26L8
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK26L8
UADD26L8        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK26L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3216B      RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            UADD26LB
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK26LB
UADD26LB        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK26LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3216B
                RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            UADD26L16
```

```
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK26L16
UADD26L16       ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK26L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3216C      RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB2,LSB
                GOTO            UADD26LC
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK26LC
UADD26LC        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK26LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3216C
                RLF             ACCB3,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB2,LSB
                GOTO            UADD26L24
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
```

```
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK26L24
UADD26L24       ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK26L24        RLF             ACCB3
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3216D      RLF             ACCB3,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB3,LSB
                GOTO            UADD26LD
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK26LD
UADD26LD        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK26LD         RLF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3216D
                BTFSC           ACCB3,LSB
                GOTO            UOK26L
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
UOK26L
                endm
UDIV3115L       macro
;       Max Timing:     9+6*17+16+16+6*17+16+16+6*17+16+16+6*17+16+8 = 537 clks
;       Min Timing:     9+6*16+15+15+6*16+15+15+6*16+15+15+6*16+15+3 = 501 clks
;       PM: 157                                 DM: 9
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
```

```
                SUBWF           REMB0
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3115A      RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            UADD15LA
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK15LA
UADD15LA        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK15LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3115A
                RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            UADD15L8
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK15L8
UADD15L8        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK15L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3115B      RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            UADD15LB
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK15LB
UADD15LB        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK15LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
```

```
                GOTO            LOOPU3115B
                RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            UADD15L16
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK15L16
UADD15L16       ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK15L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3115C      RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB2,LSB
                GOTO            UADD15LC
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK15LC
UADD15LC        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK15LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3115C
                RLF             ACCB3,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB2,LSB
                GOTO            UADD15L24
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK15L24
UADD15L24       ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK15L24        RLF             ACCB3
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU3115D      RLF             ACCB3,W
                RLF             REMB1
```

```
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB3,LSB
                GOTO            UADD15LD
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK15LD
UADD15LD        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK15LD         RLF             ACCB3
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU3115D
                BTFSC           ACCB3,LSB
                GOTO            UOK15L
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
UOK15L
                endm

;*****************************************************************************************************
;*****************************************************************************************************

;       32/16 Bit Signed Fixed Point Divide 32/16 -> 32.16
;       Input:  32 bit fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD3216S
;       Output: 32 bit fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     11+537+3 = 551 clks             A > 0, B > 0
;                       15+537+20 = 572 clks            A > 0, B < 0
;                       21+537+20 = 578 clks            A < 0, B > 0
;                       25+537+3 = 565 clks             A < 0, B < 0
;       Min Timing:     11+501+3 = 515 clks             A > 0, B > 0
;                       15+501+20 = 536 clks            A > 0, B < 0
;                       21+501+20 = 542 clks            A < 0, B > 0
;                       25+501+3 = 529 clks             A < 0, B < 0
;       PM: 25+157+19 = 201             DM: 10
FXD3216S        MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           SIGN
                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA3216S
                COMF            BARGB1
                INCF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
CA3216S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C3216S
                COMF            AARGB3
                INCF            AARGB3
                BTFSC           _Z
                DECF            AARGB2
                COMF            AARGB2
```

```
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
C3216S          CLRF            REMB0
                CLRF            REMB1
                SDIV3216L
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB3
                INCF            AARGB3
                BTFSC           _Z
                DECF            AARGB2
                COMF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
                COMF            REMB1
                INCF            REMB1
                BTFSC           _Z
                DECF            REMB0
                COMF            REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32/16 Bit Unsigned Fixed Point Divide 32/16 -> 32.16
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD3216U
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+699+2 = 703 clks
;       Max Timing:     2+663+2 = 667 clks
;       PM: 2+240+1 = 243               DM: 9
FXD3216U        CLRF            REMB0
                CLRF            REMB1
                UDIV3216L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       31/15 Bit Unsigned Fixed Point Divide 31/15 -> 31.15
;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD3115U
;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+537+2 = 541 clks
;       Min Timing:     2+501+2 = 505 clks
;       PM: 2+157+1 = 160               DM: 9
FXD3115U        CLRF            REMB0
                CLRF            REMB1
                UDIV3115L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

D.4 24/24 PIC16C5X/PIC16CXX Fixed Point Divide Routines

```
;       24/24 PIC16 FIXED POINT DIVIDE ROUTINES VERSION 1.7
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine               Clocks     Function
;       FXD2424S        565     24 bit/24 bit -> 24.24 signed fixed point divide
;       FXD2424U        676    24 bit/24 bit -> 24.24 unsigned fixed point divide
;       FXD2323U        531     23 bit/23 bit -> 23.23 unsigned fixed point divide
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number senerator registers
RANDLO          equ     0x1B
DATA            equ     0x20
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 24/24 bit fixed point divide algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
;               BCF             BARGB0,MSB
;               MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB2
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
;               BCF             AARGB0,MSB
;               MOVF            AARGB0,W
                MOVWF           INDF
```



```
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16

                MOVF            RANDHI,W
                MOVWF           AARGB2
                MOVWF           INDF
                INCF            FSR
                CALL            FXD2424S
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB2,W
                MOVWF           INDF
                INCF            FSR
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETLW           0
;*************************************************************************************
;*************************************************************************************
;       24/24 Bit Division Macros
SDIV2424L       macro
;       Max Timing:     13+6*22+21+21+6*22+21+21+6*22+21+12 = 526 clks
;       Min Timing:     13+6*21+20+20+6*21+20+20+6*21+20+3 = 494 clks
;       PM: 11+3*51+31+12 = 207                                 DM: 12
                MOVF            BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
```

```
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS2424A      RLF             ACCB0,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            SADD44LA
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK44LA
SADD44LA        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK44LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS2424A
                RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            SADD44L8
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK44L8
SADD44L8        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK44L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS2424B      RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
```

```
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            SADD44LB
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK44LB
SADD44LB        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK44LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS2424B
                RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            SADD44L16
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK44L16
SADD44L16       ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK44L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS2424C      RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB2,LSB
                GOTO            SADD44LC
                SUBWF           REMB2
                MOVF            BARGB1,W
```

```
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK44LC
SADD44LC        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK44LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS2424C
                BTFSC           ACCB2,LSB
                GOTO            SOK44L
                MOVF            BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
SOK44L
                endm
UDIV2424L       macro
;       Max Timing:     20+6*28+27+27+6*28+27+27+6*28+27+12 = 671 clks
;       Min Timing:     20+6*27+26+26+6*27+26+26+6*27+26+3 = 639 clks
;       PM: 18+2*76+40+12 = 222                                 DM: 13
                CLRF            TEMP
                RLF             ACCB0,W
                RLF             REMB2
                MOVF            BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2424A      RLF             ACCB0,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
```

```
                GOTO            UADD44LA
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK44LA
UADD44LA        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK44LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2424A
                RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            UADD44L8
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK44L8
UADD44L8        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
```

```
                ADDWF           TEMP

U0K44L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2424B      RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            UADD44LB
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            U0K44LB
UADD44LB        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

U0K44LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2424B
                RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            UADD44L16
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            U0K44L16
```

```
UADD44L16       ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK44L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2424C      RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB2,W
                BTFSS           ACCB2,LSB
                GOTO            UADD44LC
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK44LC
UADD44LC        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK44LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2424C
                BTFSC           ACCB2,LSB
                GOTO            UOK44L
                MOVF            BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
```

```
                INCF            BARGB0,W
                ADDWF           REMB0
U0K44L
                endm
UDIV2323L       macro
;       Max Timing:     13+6*22+21+21+6*22+21+21+6*22+21+12 = 526 clks
;       Min Timing:     13+6*21+20+20+6*21+20+20+6*21+20+3 = 494 clks
;       PM: 11+3*51+31+12 = 207                                 DM: 12
                MOVF            BARGB2,W
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2323A      RLF             ACCB0,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            UADD33LA
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            U0K33LA
UADD33LA        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

U0K33LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2323A
                RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB0,LSB
                GOTO            UADD33L8
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
```

```
                SUBWF           REMB0
                GOTO            UOK33L8
UADD33L8        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK33L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2323B      RLF             ACCB1,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            UADD33LB
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK33LB
UADD33LB        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK33LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2323B
                RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB1,LSB
                GOTO            UADD33L16
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK33L16
UADD33L16       ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
```

```
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

U0K33L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2323C      RLF             ACCB2,W
                RLF             REMB2
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB2,W
                BTFSS           ACCB2,LSB
                GOTO            UADD33LC
                SUBWF           REMB2
                MOVF            BARGB1,W
                BTFSS           _C
                INCFSZ          BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            U0K33LC
UADD33LC        ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCFSZ          BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

U0K33LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2323C
                BTFSC           ACCB2,LSB
                GOTO            U0K33L
                MOVF            BARGB2,W
                ADDWF           REMB2
                MOVF            BARGB1,W
                BTFSC           _C
                INCF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
U0K33L
                endm
;*****************************************************************************************************
;*****************************************************************************************************

;       24/24 Bit Signed Fixed Point Divide 24/24 -> 24.24
;       Input:  24 bit fixed point dividend in AARGB0, AARGB1,AARGB2
;               24 bit fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD2424S
;       Output: 24 bit fixed point quotient in AARGB0, AARGB1,AARGB2
;               24 bit fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     12+526+3 = 541 clks             A > 0, B > 0
;                       19+526+20 = 565 clks            A > 0, B < 0
;                       19+526+20 = 565 clks            A < 0, B > 0
```

```
;                       26+526+3 = 555 clks             A < 0, B < 0
;       Min Timing:     12+494+3 = 509 clks             A > 0, B > 0
;                       19+494+20 = 533 clks            A > 0, B < 0
;                       19+494+20 = 533 clks            A < 0, B > 0
;                       26+494+3 = 523 clks             A < 0, B < 0
;       PM: 26+207+20 = 253                             DM: 12
FXD2424S        MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           SIGN
                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA2424S
                COMF            BARGB2
                INCF            BARGB2
                BTFSC           _Z
                DECF            BARGB1
                COMF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
CA2424S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C2424S
                COMF            AARGB2
                INCF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
C2424S          CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                SDIV2424L
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB2
                INCF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
                COMF            REMB2
                INCF            REMB2
                BTFSC           _Z
                DECF            REMB1
                COMF            REMB1
                BTFSC           _Z
                DECF            REMB0
                COMF            REMB0
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************

;       24/24 Bit Unsigned Fixed Point Divide 24/24 -> 24.24
;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD2424U
;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2
;               24 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     3+671+2 = 676 clks
;       Max Timing:     3+639+2 = 644 clks
;       PM: 3+222+1 = 226               DM: 13
FXD2424U        CLRF            REMB0
```

```
                CLRF            REMB1
                CLRF            REMB2
                UDIV2424L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       23/23 Bit Unsigned Fixed Point Divide 23/23 -> 23.23
;       Input:  23 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARBB2
;       Use:    CALL    FXD2323U
;       Output: 23 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     3+526+2 = 531 clks
;       Min Timing:     3+494+2 = 499 clks
;       PM: 3+207+1 = 211               DM: 12
FXD2323U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                UDIV2323L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

D.5 24/16 PIC16C5X/PIC16CXX Fixed Point Divide Routines

```
;       24/16 PIC16 FIXED POINT DIVIDE ROUTINES VERSION 1.7
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;       Routine        Clocks     Function
;       FXD2416S        438     24 bit/16 bit -> 24.16 signed fixed point divide
;       FXD2416U        529     24 bit/16 bit -> 24.16 unsigned fixed point divide
;       FXD2315U        407     23 bit/15 bit -> 23.15 unsigned fixed point divide
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1E    ; random number senerator registers
RANDLO          equ     0x1F
DATA            equ     0x20
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 24/16 bit fixed point divide algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
```



```
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
                BCF             BARGB0,MSB
                MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
                BCF             AARGB0,MSB
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16

                MOVF            RANDHI,W
                MOVWF           AARGB2
                MOVWF           INDF
                INCF            FSR
                CALL            FXD2315U
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB2,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB1,W
                MOVWF           INDF
                INCF            FSR
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETLW           0
```

```
;*****************************************************************************************
;*****************************************************************************************
;       24/16 Bit Division Macros
SDIV2416L       macro
;       Max Timing:     9+6*17+16+16+6*17+16+16+6*17+16+8 = 403 clks
;       Min Timing:     9+6*16+15+15+6*16+15+15+6*16+15+3 = 375 clks
;       PM: 7+2*40+22+8 = 117                                   DM: 7
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS2416A      RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            SADD46LA
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK46LA
SADD46LA        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK46LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS2416A
                RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            SADD46L8
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK46L8
SADD46L8        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK46L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS2416B      RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            SADD46LB
                SUBWF           REMB1
                MOVF            BARGB0,W
```

```
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK46LB
SADD46LB        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK46LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS2416B
                RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            SADD46L16
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK46L16
SADD46L16       ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK46L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS2416C      RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB2,LSB
                GOTO            SADD46LC
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK46LC
SADD46LC        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

SOK46LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS2416C
                BTFSC           ACCB2,LSB
                GOTO            SOK46L
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
SOK46L
                endm
UDIV2416L       macro
```

```
;       Max Timing:     16+6*22+21+21+6*22+21+21+6*22+21+8 = 525 clks
;       Min Timing:     16+6*21+20+20+6*21+20+20+6*21+20+3 = 497 clks
;       PM: 14+31+27+31+27+31+8 = 169                   DM: 8
                CLRF            TEMP
                RLF             ACCB0,W
                RLF             REMB1
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2416A      RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            UADD46LA
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK46LA
UADD46LA        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK46LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2416A
                RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            UADD46L8
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK46L8
```

```
UADD46L8        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK46L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2416B      RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            UADD46LB
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK46LB
UADD46LB        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

UOK46LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2416B
                RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            UADD46L16
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            UOK46L16
UADD46L16       ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
```

```
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

U0K46L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2416C      RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB1,W
                BTFSS           ACCB2,LSB
                GOTO            UADD46LC
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                CLRW
                BTFSS           _C
                MOVLW           1
                SUBWF           TEMP
                GOTO            U0K46LC
UADD46LC        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                CLRW
                BTFSC           _C
                MOVLW           1
                ADDWF           TEMP

U0K46LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2416C
                BTFSC           ACCB2,LSB
                GOTO            U0K46L
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
U0K46L
                endm
UDIV2315L       macro
;       Max Timing:     9+6*17+16+16+6*17+16+16+6*17+16+8 = 403 clks
;       Min Timing:     9+6*16+15+15+6*16+15+15+6*16+15+3 = 375 clks
;       PM: 7+2*40+22+8 = 117                                   DM: 7
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2315A      RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            UADD35LA
```

```
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK35LA
UADD35LA        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK35LA         RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2315A
                RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            UADD35L8
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK35L8
UADD35L8        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK35L8         RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2315B      RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            UADD35LB
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK35LB
UADD35LB        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK35LB         RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2315B
                RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            UADD35L16
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
```

```
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK35L16
UADD35L16       ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK35L16        RLF             ACCB2
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU2315C      RLF             ACCB2,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB2,LSB
                GOTO            UADD35LC
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK35LC
UADD35LC        ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0

UOK35LC         RLF             ACCB2
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU2315C
                BTFSC           ACCB2,LSB
                GOTO            UOK35L
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
UOK35L
                endm
;*****************************************************************************************
;*****************************************************************************************

;       24/16 Bit Signed Fixed Point Divide 24/16 -> 24.16
;       Input:  24 bit fixed point dividend in AARGB0, AARGB1,AARGB2
;               16 bit fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD2416S
;       Output: 24 bit fixed point quotient in AARGB0, AARGB1,AARGB2
;               16 bit fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     11+403+3 = 417 clks             A > 0, B > 0
;                       15+403+17 = 435 clks            A > 0, B < 0
;                       18+403+17 = 438 clks            A < 0, B > 0
;                       22+403+3 = 428 clks             A < 0, B < 0
;       Min Timing:     11+375+3 = 389 clks             A > 0, B > 0
;                       15+375+17 = 407 clks            A > 0, B < 0
;                       18+375+17 = 410 clks            A < 0, B > 0
;                       22+375+3 = 400 clks             A < 0, B < 0
;       PM: 22+117+16 = 140                             DM: 8
FXD2416S        MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           SIGN
                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
```

```
                GOTO            CA2416S
                COMF            BARGB1
                INCF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
CA2416S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C2416S
                COMF            AARGB2
                INCF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
C2416S          CLRF            REMB0
                CLRF            REMB1
                SDIV2416L
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB2
                INCF            AARGB2
                BTFSC           _Z
                DECF            AARGB1
                COMF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
                COMF            REMB1
                INCF            REMB1
                BTFSC           _Z
                DECF            REMB0
                COMF            REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       24/16 Bit Unsigned Fixed Point Divide 24/16 -> 24.16
;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD2416U
;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2
;               16 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+525+2 = 529 clks
;       Max Timing:     2+497+2 = 501 clks
;       PM: 2+169+1 = 172               DM: 8
FXD2416U        CLRF            REMB0
                CLRF            REMB1
                UDIV2416L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       23/15 Bit Unsigned Fixed Point Divide 23/15 -> 23.15
;       Input:  23 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD2315U
;       Output: 23 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+403+2 = 407 clks
;       Min Timing:     2+375+2 = 379 clks
;       PM: 2+117+1 = 120               DM: 7
FXD2315U        CLRF            REMB0
```

```
                CLRF            REMB1
                UDIV2315L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

D.6 16/16 PIC16C5X/PIC16CXX Fixed Point Divide Routines

```
;       16/16 PIC16 FIXED POINT DIVIDE ROUTINES VERSION 1.7
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXD1616S    319 16 bit/16 bit -> 16.16 signed fixed point divide
;       FXD1616U    373 16 bit/16 bit -> 16.16 unsigned fixed point divide
;       FXD1515U    294 15 bit/15 bit -> 15.15 unsigned fixed point divide
;       The above timings are based on the looped macros. If space permits,
;       approximately 65-69 clocks can be saved by using the unrolled macros.
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number senerator registers
RANDLO          equ     0x1B
TESTCOUNT       equ     0x20    ; counter
DATA            equ     0x21    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 16/16 bit fixed point divide algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            TFXD1616
                MOVLW           0x99
                MOVWF           AARGB0
                MOVLW           0xAB
                MOVWF           AARGB1
                MOVLW           0x00
                MOVWF           BARGB0
                MOVLW           0xFF
                MOVWF           BARGB1
                CALL            FXD1616S
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
```

```
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETLW           0
;       Test suite for FXD1616
TFXD1616        MOVLW           2
                MOVWF           TESTCOUNT
D1616LOOP
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
;               BCF             BARGB0,MSB
;               MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           BARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
;               BCF             AARGB0,MSB
;               MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            FXD1616S
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB1,W
                MOVWF           INDF
                INCF            FSR
                DECFSZ          TESTCOUNT
                GOTO            D1616LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       16/16 Bit Division Macros
SDIV1616L       macro
;       Max Timing:     13+14*18+17+8 = 290 clks
;       Min Timing:     13+14*16+15+3 = 255 clks
;       PM: 42                                  DM: 7
                RLF             ACCB0,W
```

```
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB1
                RLF             ACCB0
                MOVLW           15
                MOVWF           LOOPCOUNT
LOOPS1616       RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            SADD66L
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK66LL
SADD66L         ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
SOK66LL         RLF             ACCB1
                RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS1616
                BTFSC           ACCB1,LSB
                GOTO            SOK66L
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
SOK66L
                endm
UDIV1616L       macro
;       restore = 23 clks,  nonrestore = 17 clks
;       Max Timing:     2+15*23+22 = 369 clks
;       Min Timing:     2+15*17+16 = 273 clks
;       PM: 24                                  DM: 7
                MOVLW           16
                MOVWF           LOOPCOUNT
LOOPU1616       RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                BTFSC           _C
                GOTO            UOK66LL
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
```

```
                ADDWF           REMB0
                BCF             _C
UOK66LL         RLF             ACCB1
                RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU1616
                endm
UDIV1515L       macro
;       Max Timing:     13+14*18+17+8 = 290 clks
;       Min Timing:     13+14*17+16+3 = 270 clks
;       PM: 42                                  DM: 7
                RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB1
                RLF             ACCB0
                MOVLW           15
                MOVWF           LOOPCOUNT
LOOPU1515       RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            UADD55L
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK55LL
UADD55L         ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
UOK55LL         RLF             ACCB1
                RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU1515
                BTFSC           ACCB1,LSB
                GOTO            UOK55L
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
UOK55L
                endm
SDIV1616        macro
;       Max Timing:     7+10+6*14+14+7*14+8 = 221 clks
;       Min Timing:     7+10+6*13+13+7*13+3 = 202 clks
;       PM: 7+10+6*18+18+7*18+8 = 277   DM: 6
                variable i
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
```

```
                RLF             ACCB0
                RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                RLF             ACCB0
                i = 2
                while i < 8
                RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            SADD66#v(i)
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK66#v(i)
SADD66#v(i)     ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
SOK66#v(i)      RLF             ACCB0
                i=i+1
                endw
                RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            SADD668
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK668
SADD668         ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
SOK668          RLF             ACCB1
                i = 9
                while i < 16
                RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            SADD66#v(i)
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            SOK66#v(i)
SADD66#v(i)     ADDWF           REMB1
```

```
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
SOK66#v(i)      RLF             ACCB1
                i=i+1
                endw
                BTFSC           ACCB1,LSB
                GOTO            SOK66
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
SOK66
                endm
UDIV1616        macro
;       restore = 20 clks,  nonrestore = 14 clks
;       Max Timing: 16*20 = 320 clks
;       Min Timing: 16*14 = 224 clks
;       PM: 16*20 = 320         DM: 6
                variable         i
                i = 0
                while i < 16
                RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                BTFSC           _C
                GOTO            UOK66#v(i)
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                BCF             _C
UOK66#v(i)      RLF             ACCB1
                RLF             ACCB0
                i=i+1
                endw
                endm
UDIV1515        macro
;       Max Timing:     7+10+6*14+14+7*14+8 = 221 clks
;       Min Timing:     7+10+6*13+13+7*13+3 = 202 clks
;       PM:     7+10+6*18+18+7*18+8 = 277       DM: 6
                variable i
                MOVF            BARGB1,W
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
```

```
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
                RLF             ACCB0
                i = 2
                while i < 8
                RLF             ACCB0,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            UADD55#v(i)
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK55#v(i)
UADD55#v(i)     ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
UOK55#v(i)      RLF             ACCB0
                i=i+1
                endw
                RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB0,LSB
                GOTO            UADD558
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK558
UADD558         ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
UOK558          RLF             ACCB1
                i = 9
                while i < 16
                RLF             ACCB1,W
                RLF             REMB1
                RLF             REMB0
                MOVF            BARGB1,W
                BTFSS           ACCB1,LSB
                GOTO            UADD55#v(i)
                SUBWF           REMB1
                MOVF            BARGB0,W
                BTFSS           _C
                INCFSZ          BARGB0,W
                SUBWF           REMB0
                GOTO            UOK55#v(i)
UADD55#v(i)     ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCFSZ          BARGB0,W
                ADDWF           REMB0
UOK55#v(i)      RLF             ACCB1
                i=i+1
                endw
```

```
                BTFSC           ACCB1,LSB
                GOTO            UOK55
                MOVF            BARGB1,W
                ADDWF           REMB1
                MOVF            BARGB0,W
                BTFSC           _C
                INCF            BARGB0,W
                ADDWF           REMB0
UOK55
                endm
;*****************************************************************************************
;*****************************************************************************************

;       16/16 Bit Signed Fixed Point Divide 16/16 -> 16.16
;       Input:  16 bit fixed point dividend in AARGB0, AARGB1
;               16 bit fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD1616S
;       Output: 16 bit fixed point quotient in AARGB0, AARGB1
;               16 bit fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     11+290+3 = 304 clks             A > 0, B > 0
;                       15+290+14 = 319 clks            A > 0, B < 0
;                       15+290+14 = 319 clks            A < 0, B > 0
;                       19+290+3 = 312 clks             A < 0, B < 0
;       Min Timing:     11+255+3 = 269 clks             A > 0, B > 0
;                       15+255+14 = 284 clks            A > 0, B < 0
;                       15+255+14 = 284 clks            A < 0, B > 0
;                       19+255+3 = 277 clks             A < 0, B < 0
;       PM: 19+42+13 = 74               DM: 8
FXD1616S        MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           SIGN
                BTFSS           BARGB0,MSB        ; if MSB set, negate BARG
                GOTO            CA1616S
                COMF            BARGB1
                INCF            BARGB1
                BTFSC           _Z
                DECF            BARGB0
                COMF            BARGB0
CA1616S         BTFSS           AARGB0,MSB        ; if MSB set, negate AARG
                GOTO            C1616S
                COMF            AARGB1
                INCF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
C1616S          CLRF            REMB0
                CLRF            REMB1
                SDIV1616L
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB1
                INCF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
                COMF            REMB1
                INCF            REMB1
                BTFSC           _Z
                DECF            REMB0
                COMF            REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16/16 Bit Unsigned Fixed Point Divide 16/16 -> 16.16
```

```
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD1616U
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               16 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+369+2 = 373 clks
;       Min Timing:     2+273+2 = 277 clks
;       PM: 2+24+1 = 27         DM: 7
FXD1616U        CLRF            REMB0
                CLRF            REMB1
                UDIV1616L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       15/15 Bit Unsigned Fixed Point Divide 15/15 -> 15.15
;       Input:  15 bit unsigned fixed point dividend in AARGB0, AARGB1
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD1515U
;       Output: 15 bit unsigned fixed point quotient in AARGB0, AARGB1
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+290+2 = 294 clks
;       Min Timing:     2+270+2 = 274 clks
;       PM: 2+42+1 = 45         DM: 7
FXD1515U        CLRF            REMB0
                CLRF            REMB1
                UDIV1515L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

D.7 16/8 PIC16C5X/PIC16CXX Fixed Point Divide Routines

```
;       16/8 PIC16 FIXED POINT DIVIDE ROUTINES  VERSION 1.7
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXD1608S    188 16 bit/8 bit -> 16.08 signed fixed point divide
;       FXD1608U    294 16 bit/8 bit -> 16.08 unsigned fixed point divide
;       FXD1607U    174 16 bit/7 bit -> 16.07 unsigned fixed point divide
;       FXD1507U    166 15 bit/7 bit -> 15.07 unsigned fixed point divide
;       The above timings are based on the looped macros. If space permits,
;       approximately 41-50 clocks can be saved by using the unrolled macros.
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number generator registers
RANDLO          equ     0x1B
TESTCOUNT       equ     0x20    ; counter
DATA            equ     0x21    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 16/8 bit fixed point divide algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
```

```
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
                CALL            TFXD1608
                MOVLW           0x99
                MOVWF           AARGB0
                MOVLW           0xAB
                MOVWF           AARGB1
                MOVLW           0xFF
                MOVWF           BARGB0
                CALL            FXD1608U
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETLW           0
;       Test suite for FXD1608
TFXD1608        MOVLW           3
                MOVWF           TESTCOUNT
D1608LOOP
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
;               BCF             BARGB0,MSB
;               MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
;               BCF             AARGB0,MSB
;               MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            RANDLO,W
                MOVWF           AARGB1
                MOVWF           INDF
                INCF            FSR
                CALL            FXD1608S
                MOVF            AARGB0,W
```

```
                MOVWF           INDF
                INCF            FSR
                MOVF            AARGB1,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB0,W
                MOVWF           INDF
                INCF            FSR
                DECFSZ          TESTCOUNT
                GOTO            D1608LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       16/08 BIT Division Macros
SDIV1608L       macro
;       Max Timing:     3+5+2+5*11+10+10+6*11+10+2 = 163 clks
;       Min Timing:     3+5+2+5*11+10+10+6*11+10+2 = 163 clks
;       PM: 42                                  DM: 5
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                ADDWF           REMB0
                RLF             ACCB0
                MOVLW           6
                MOVWF           LOOPCOUNT
LOOPS1608A      RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS1608A
                RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPS1608B      RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB1,LSB
                SUBWF           REMB0
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS1608B
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                endm
UDIV1608L       macro
;       Max Timing: 2+7*12+11+3+7*24+23 = 291 clks
;       Min Timing: 2+7*11+10+3+7*17+16 = 227 clks
;       PM: 39                                  DM: 7
                MOVLW           8
```

```
                MOVWF           LOOPCOUNT
LOOPU1608A      RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                SUBWF           REMB0
                BTFSC           _C
                GOTO            UOK68A
                ADDWF           REMB0
                BCF             _C
UOK68A          RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU1608A
                CLRF            TEMP
                MOVLW           8
                MOVWF           LOOPCOUNT
LOOPU1608B      RLF             ACCB1,W
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB0,W
                SUBWF           REMB0
                CLRF            ACCB5
                CLRW
                BTFSS           _C
                INCFSZ          ACCB5,W
                SUBWF           TEMP
                BTFSC           _C
                GOTO            UOK68B
                MOVF            BARGB0,W
                ADDWF           REMB0
                CLRF            ACCB5
                CLRW
                BTFSC           _C
                INCFSZ          ACCB5,W
                ADDWF           TEMP
                BCF             _C
UOK68B          RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU1608B
                endm
UDIV1607L       macro
;       Max Timing:     7+6*11+10+10+6*11+10+2 = 171 clks
;       Min Timing:     7+6*11+10+10+6*11+10+2 = 171 clks
;       PM: 39                                  DM: 5
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU1607A      RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU1607A
                RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
```

```
                ADDWF           REMB0
                RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU1607B      RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB1,LSB
                SUBWF           REMB0
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU1607B
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                endm
UDIV1507L       macro
;       Max Timing:     3+5+2+5*11+10+10+6*11+10+2 = 163 clks
;       Min Timing:     3+5+2+5*11+10+10+6*11+10+2 = 163 clks
;       PM: 42                                  DM: 5
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                ADDWF           REMB0
                RLF             ACCB0
                MOVLW           6
                MOVWF           LOOPCOUNT
LOOPU1507A      RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU1507A
                RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB1
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU1507B      RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB1,LSB
                SUBWF           REMB0
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                RLF             ACCB1
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU1507B
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                endm
SDIV1608        macro
;       Max Timing:     3+5+14*8+2 = 122 clks
```

```
;       Min Timing:     3+5+14*8+2 = 122 clks
;       PM: 122                                 DM: 4
                variable i
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                ADDWF           REMB0
                RLF             ACCB0
                i = 2
                while i < 8
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                i=i+1
                endw
                RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB1
                i = 9
                while i < 16
                RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB1,LSB
                SUBWF           REMB0
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                RLF             ACCB1
                i=i+1
                endw
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                endm
UDIV1608        macro
;       restore = 9/21 clks,  nonrestore = 8/14 clks
;       Max Timing: 8*9+1+8*21 = 241 clks
;       Min Timing: 8*8+1+8*14 = 177 clks
;       PM: 241                                 DM: 6
                variable         i
                i = 0
                while i < 8
                RLF             ACCB0,W
                RLF              REMB0
                MOVF            BARGB0,W
                SUBWF           REMB0
                BTFSC           _C
                GOTO            UOK68#v(i)
                ADDWF           REMB0
                BCF             _C
UOK68#v(i)      RLF             ACCB0
                i=i+1
                endw
                CLRF            TEMP
```

```
                i = 8
                while i < 16
                RLF             ACCB1,W
                RLF             REMB0
                RLF             TEMP
                MOVF            BARGB0,W
                SUBWF           REMB0
                CLRF            ACCB5
                CLRW
                BTFSS           _C
                INCFSZ          ACCB5,W
                SUBWF           TEMP
                BTFSC           _C
                GOTO            UOK68#v(i)
                MOVF            BARGB0,W
                ADDWF           REMB0
                CLRF            ACCB5
                CLRW
                BTFSC           _C
                INCFSZ          ACCB5,W
                ADDWF           TEMP
                BCF             _C
UOK68#v(i)      RLF             ACCB1
                i=i+1
                endw
                endm
UDIV1607        macro
;       Max Timing:     5+15*8+2 = 127 clks
;       Min Timing:     5+15*8+2 = 127 clks
;       PM: 127                                 DM: 4
                variable i
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                i = 1
                while i < 8
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                i=i+1
                endw
                RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB1
                i = 9
                while i < 16
                RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB1,LSB
                SUBWF           REMB0
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                RLF             ACCB1
```

```
                i=i+1
                endw
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                endm
UDIV1507        macro
;       Max Timing:     3+5+14*8+2 = 122 clks
;       Min Timing:     3+5+14*8+2 = 122 clks
;       PM: 122                                 DM: 4
                variable i
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                ADDWF           REMB0
                RLF             ACCB0
                i = 2
                while i < 8
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                i=i+1
                endw
                RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB1
                i = 9
                while i < 16
                RLF             ACCB1,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB1,LSB
                SUBWF           REMB0
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                RLF             ACCB1
                i=i+1
                endw
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                endm

;*****************************************************************************************************
;*****************************************************************************************************

;       16/8 Bit Signed Fixed Point Divide 16/8 -> 16.08
;       Input:  16 bit signed fixed point dividend in AARGB0, AARGB1
;               8 bit signed fixed point divisor in BARGB0
;       Use:    CALL    FXD1608S
;       Output: 16 bit signed fixed point quotient in AARGB0, AARGB1
;               8 bit signed fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     10+163+3 = 176 clks             A > 0, B > 0
;                       11+163+11 = 185 clks            A > 0, B < 0
```

```
;                       14+163+11 = 188 clks            A < 0, B > 0
;                       15+163+3 = 181 clks             A < 0, B < 0
;       Min Timing:     10+163+3 = 176 clks             A > 0, B > 0
;                       11+163+11 = 185 clks            A > 0, B < 0
;                       14+163+11 = 188 clks            A < 0, B > 0
;                       15+163+3 = 181 clks             A < 0, B < 0
;       PM: 15+42+10 = 67               DM: 6
FXD1608S        MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           SIGN
                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA1608S
                COMF            BARGB0
                INCF            BARGB0
CA1608S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C1608S
                COMF            AARGB1
                INCF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
C1608S          CLRF            REMB0
                SDIV1608L
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB1
                INCF            AARGB1
                BTFSC           _Z
                DECF            AARGB0
                COMF            AARGB0
                COMF            REMB0
                INCF            REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16/8 Bit Unsigned Fixed Point Divide 16/8 -> 16.08
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               8 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD1608U
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               8 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+291+2 = 294 clks
;       Min Timing:     1+227+2 = 230 clks
;       PM: 1+39+1 = 41         DM: 7
FXD1608U        CLRF            REMB0
                UDIV1608L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16/7 Bit Unsigned Fixed Point Divide 16/7 -> 16.07
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               7 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD1607U
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               7 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+171+2 = 174 clks
;       Min Timing:     1+171+2 = 174 clks
;       PM: 1+39+1 = 41         DM: 5
FXD1607U        CLRF            REMB0
                UDIV1607L
                RETLW           0x00
;*****************************************************************************************
```

```
;*****************************************************************************************

;       15/7 Bit Unsigned Fixed Point Divide 15/7 -> 15.07
;       Input:  15 bit unsigned fixed point dividend in AARGB0, AARGB1
;               7 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD1507U
;       Output: 15 bit unsigned fixed point quotient in AARGB0, AARGB1
;               7 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+163+2 = 166 clks
;       Min Timing:     1+163+2 = 166 clks
;       PM: 1+42+1 = 44         DM: 5
FXD1507U        CLRF            REMB0
                UDIV1507L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

D.8 8/8 PIC16C5X/PIC16CXX Fixed Point Divide Routines

```
;       8/8 PIC16 FIXED POINT DIVIDE ROUTINES   VERSION 1.7
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG in AARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXD0808S     96 8 bit/8 bit -> 08.08 signed fixed point divide
;       FXD0808U    100 8 bit/8 bit -> 08.08 unsigned fixed point divide
;       FXD0807U     88 8 bit/7 bit -> 08.07 unsigned fixed point divide
;       FXD0707U     80 7 bit/7 bit -> 07.07 unsigned fixed point divide
;       The above timings are based on the looped macros. If space permits,
;       approximately 19-25 clocks can be saved by using the unrolled macros.
                list    r=dec,x=on,t=off
                include <PIC16.INC>     ; general PIC16 definitions
                include <MATH16.INC>    ; PIC16 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x1A    ; random number generator registers
RANDLO          equ     0x1B
TESTCOUNT       equ     0x20    ; counter
DATA            equ     0x21    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for 8/8 bit fixed point divide algorithms
                org             0x0005
MAIN            MOVLW           RAMSTART
                MOVWF           FSR
MEMLOOP         CLRF            INDF
                INCF            FSR
                MOVLW           RAMSTOP
                SUBWF           FSR,W
                BTFSS           _Z
                GOTO            MEMLOOP
                MOVLW           0x45                    ; seed for random numbers
                MOVWF           RANDLO
                MOVLW           0x30
                MOVWF           RANDHI
                MOVLW           DATA
                MOVWF           FSR
                BCF             _RP0
                BCF             _RP1
                BCF             _IRP
```

```
                CALL            TFXD0808
SELF            GOTO            SELF
RANDOM16        RLF             RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                MOVWF           TEMPB0
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                MOVWF           TEMPB1
                RLF             TEMPB1,W
                RLF             TEMPB1
                MOVF            TEMPB1,W
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLF             TEMPB0
                RLF             RANDLO
                XORWF           RANDLO
                RLF             RANDHI

                RETLW           0
;       Test suite for FXD0808
TFXD0808        MOVLW           3
                MOVWF           TESTCOUNT
D0808LOOP
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           BARGB0
;               BCF             BARGB0,MSB
;               MOVF            BARGB0,W
                MOVWF           INDF
                INCF            FSR
                CALL            RANDOM16
                MOVF            RANDHI,W
                MOVWF           AARGB0
;               BCF             AARGB0,MSB
;               MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                CALL            FXD0808S
                MOVF            AARGB0,W
                MOVWF           INDF
                INCF            FSR
                MOVF            REMB0,W
                MOVWF           INDF
                INCF            FSR
                DECFSZ          TESTCOUNT
                GOTO            D0808LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       08/08 BIT Division Macros
SDIV0808L       macro
;       Max Timing:     3+5+2+5*11+10+2 = 77 clks
;       Min Timing:     3+5+2+5*11+10+2 = 77 clks
;       PM: 22                                  DM: 4
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                ADDWF           REMB0
                RLF             ACCB0
                MOVLW           6
                MOVWF           LOOPCOUNT
LOOPS0808A      RLF             ACCB0,W
```

```
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPS0808A
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                endm
UDIV0808L       macro
;       Max Timing: 2+7*12+11 = 97 clks
;       Min Timing: 2+7*11+10 = 89 clks
;       PM: 13                                  DM: 4
                MOVLW           8
                MOVWF           LOOPCOUNT
LOOPU0808A      RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                SUBWF           REMB0
                BTFSC           _C
                GOTO            UOK88A
                ADDWF           REMB0
                BCF             _C
UOK88A          RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU0808A
                endm
UDIV0807L       macro
;       Max Timing:     7+6*11+10+2 = 85 clks
;       Min Timing:     7+6*11+10+2 = 85 clks
;       PM: 19                                  DM: 4
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                MOVLW           7
                MOVWF           LOOPCOUNT
LOOPU0807       RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU0807
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                endm
UDIV0707L       macro
;       Max Timing:     3+5+2+5*11+10+2 = 77 clks
;       Min Timing:     3+5+2+5*11+10+2 = 77 clks
;       PM: 22                                  DM: 4
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                ADDWF           REMB0
                RLF             ACCB0
```

```
                MOVLW           6
                MOVWF           LOOPCOUNT
LOOPU0707       RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                DECFSZ          LOOPCOUNT
                GOTO            LOOPU0707
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                endm
SDIV0808        macro
;       Max Timing:     3+5+6*8+2 = 58 clks
;       Min Timing:     3+5+6*8+2 = 58 clks
;       PM: 58                                  DM: 3
                variable i
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                ADDWF           REMB0
                RLF             ACCB0
                i = 2
                while i < 8
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                i=i+1
                endw
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                endm
UDIV0808        macro
;       restore = 9 clks,  nonrestore = 8 clks
;       Max Timing: 8*9 = 72 clks
;       Min Timing: 8*8 = 64 clks
;       PM: 72                                  DM: 3
                variable         i
                i = 0
                while i < 8
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                SUBWF           REMB0
                BTFSC           _C
                GOTO            UOK88#v(i)
                ADDWF           REMB0
                BCF             _C
UOK88#v(i)      RLF             ACCB0
                i=i+1
                endw
                endm
UDIV0807        macro
;       Max Timing:     5+7*8+2 = 63 clks
;       Min Timing:     5+7*8+2 = 63 clks
```

```
;       PM: 63                                  DM: 3
                variable i
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                i = 1
                while i < 8
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                i=i+1
                endw
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                endm
UDIV0707        macro
;       Max Timing:     3+5+6*8+2 = 58 clks
;       Min Timing:     3+5+6*8+2 = 58 clks
;       PM: 58                                  DM: 3
                variable i
                MOVF            BARGB0,W
                SUBWF           REMB0
                RLF             ACCB0
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                ADDWF           REMB0
                RLF             ACCB0
                i = 2
                while i < 8
                RLF             ACCB0,W
                RLF             REMB0
                MOVF            BARGB0,W
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLF             ACCB0
                i=i+1
                endw
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                endm

;*************************************************************************************************
;*************************************************************************************************

;       8/8 Bit Signed Fixed Point Divide 8/8 -> 08.08
;       Input:  8 bit signed fixed point dividend in AARGB0
;               8 bit signed fixed point divisor in BARGB0
;       Use:    CALL    FXD0808S
;       Output: 8 bit signed fixed point quotient in AARGB0
;               8 bit signed fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     10+77+3 = 90 clks               A > 0, B > 0
;                       11+77+8 = 96 clks               A > 0, B < 0
;                       11+77+8 = 96 clks               A < 0, B > 0
;                       12+77+3 = 92 clks               A < 0, B < 0
;       Min Timing:     10+77+3 = 90 clks               A > 0, B > 0
```

```
;                       11+77+8 = 96 clks               A > 0, B < 0
;                       11+77+8 = 96 clks               A < 0, B > 0
;                       12+77+3 = 92 clks               A < 0, B < 0
;       PM: 12+22+7 = 41                DM: 5
FXD0808S        MOVF            AARGB0,W
                XORWF           BARGB0,W
                MOVWF           SIGN
                BTFSS           BARGB0,MSB              ; if MSB set, negate BARG
                GOTO            CA0808S
                COMF            BARGB0
                INCF            BARGB0
CA0808S         BTFSS           AARGB0,MSB              ; if MSB set, negate AARG
                GOTO            C0808S
                COMF            AARGB0
                INCF            AARGB0
C0808S          CLRF            REMB0
                SDIV0808L
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB0
                INCF            AARGB0
                COMF            REMB0
                INCF            REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       8/8 Bit Unsigned Fixed Point Divide 8/8 -> 08.08
;       Input:  8 bit unsigned fixed point dividend in AARGB0
;               8 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD0808U
;       Output: 8 bit unsigned fixed point quotient in AARGB0
;               8 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+97+2 = 100 clks
;       Min Timing:     1+89+2 = 92 clks
;       PM: 1+13+1 = 15         DM: 4
FXD0808U        CLRF            REMB0
                UDIV0808L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       8/7 Bit Unsigned Fixed Point Divide 8/7 -> 08.07
;       Input:  8 bit unsigned fixed point dividend in AARGB0
;               7 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD0807U
;       Output: 8 bit unsigned fixed point quotient in AARGB0
;               7 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+85+2 = 88 clks
;       Min Timing:     1+85+2 = 88 clks
;       PM: 1+19+1 = 21         DM: 4
FXD0807U        CLRF            REMB0
                UDIV0807L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       7/7 Bit Unsigned Fixed Point Divide 7/7 -> 07.07
;       Input:  7 bit unsigned fixed point dividend in AARGB0
;               7 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD0707U
;       Output: 7 bit unsigned fixed point quotient in AARGB0
;               7 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
```

```
;       Max Timing:     1+77+2 = 80 clks
;       Min Timing:     1+77+2 = 80 clks
;       PM: 1+22+1 = 44         DM: 4
FXD0707U        CLRF            REMB0
                UDIV0707L
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

NOTES:

Please check the Microchip BBS for the latest version of the source code. For BBS access information, see Section 6, Microchip Bulletin Board Service information, page 6-3.

APPENDIX E: PIC17CXX MULTIPLY ROUTINES

```
;       PIC17 FIXED POINT MULTIPLY ROUTINES       VERSION 1.3
;       Input:  fixed point arguments in AARG and BARG
;       Output: product AARG*BARG in AARG
;       All timings are worst case cycle counts
;       It is useful to note that the additional routines FXM3115U, FXM1515U, and
;       FXM1507U can be called in a signed multiply application in the special case
;       where AARG > 0 and BARG > 0, thereby offering some improvement in
;       performance.
;       Routine     Clocks     Function
;       FXM0808S      53       08x08 -> 16 bit signed fixed point multiply
;       FXM0808U      39       08x08 -> 16 bit unsigned fixed point multiply
;       FXM0707U      37       07x07 -> 14 bit unsigned fixed point multiply
;       FXM1608S      79       16x08 -> 24 bit signed fixed point multiply
;       FXM1608U      75       16x08 -> 24 bit unsigned fixed point multiply
;       FXM1507U      69       15x07 -> 22 bit unsigned fixed point multiply
;       FXM1616S     175       16x16 -> 32 bit signed fixed point multiply
;       FXM1616U     156       16x16 -> 32 bit unsigned fixed point multiply
;       FXM1515U     150       15x15 -> 30 bit unsigned fixed point multiply
;       FXM2416S     223       24x16 -> 40 bit signed fixed point multiply
;       FXM2416U     203       24x16 -> 40 bit unsigned fixed point multiply
;       FXM2315U     194       23x15 -> 38 bit unsigned fixed point multiply
;       FXM2424S     346       24x24 -> 48 bit signed fixed point multiply
;       FXM2424U     316       24x24 -> 48 bit unsigned fixed point multiply
;       FXM2323U     308       23x23 -> 46 bit unsigned fixed point multiply
;       FXM3216S     270       32x16 -> 48 bit signed fixed point multiply
;       FXM3216U     265       32x16 -> 48 bit unsigned fixed point multiply
;       FXM3115U     254       31x15 -> 46 bit unsigned fixed point multiply
;       FXM3224S     417       32x24 -> 56 bit signed fixed point multiply
;       FXM3224U     410       32x24 -> 56 bit unsigned fixed point multiply
;       FXM3123U     399       31x23 -> 54 bit unsigned fixed point multiply
;       FXM3232S     572       32x32 -> 64 bit signed fixed point multiply
;       FXM3232U     563       32x32 -> 64 bit unsigned fixed point multiply
;       FXM3131U     543       31x31 -> 62 bit unsigned fixed point multiply
                list    r=dec,x=on,t=off,p=17C42
                include <PIC17.INC>     ; general PIC17 definitions

                include <MATH17.INC>    ; PIC17 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x2B    ; random number senerator registers
RANDLO          equ     0x2C
TESTCODE        equ     0x2D    ; integer code labeling test contained in following data
NUMTESTS        equ     0x2E    ; number of tests contained in following data
TESTCOUNT       equ     0x2F    ; counter
DATA            equ     0x30    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for fixed point multiplication algorithms
                org             0x0021
MAIN            MOVLW           RAMSTART
                MOVPF           WREG,FSR0
MEMLOOP         CLRF            INDF0
                INCFSZ          FSR0
                GOTO            MEMLOOP
                BSF             RTCSTA,5
;               MOVPF           RTCCH,WREG
                MOVLW           0x45                    ; seed for random numbers
                MOVPF           WREG,RANDLO
;               MOVPF           RTCCL,WREG
                MOVLW           0x30
                MOVPF           WREG,RANDHI
```

```
                MOVLW           0x30
                MOVPF           WREG,FSR0
                BCF             _FS1
                BSF             _FS0
;               CALL            TFXM0808
;               CALL            TFXM1608
                CALL            TFXM1616
;               CALL            TFXM2416
;               CALL            TFXM2424
;               CALL            TFXM3216
;               CALL            TFXM3224
;               CALL            TFXM3232
SELF            GOTO            SELF
RANDOM16        RLCF            RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                RLCF            WREG
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                RLNCF           WREG
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLCF            RANDLO
                XORWF           RANDLO
                RLCF            RANDHI

                RETLW           0
;       Test suite for FXM0808
TFXM0808        MOVLW           52
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           1
                MOVPF           WREG,TESTCODE
M0808LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
;               BCF             BARGB0,MSB

                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB0
;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           BARGB0,INDF0
                CALL            FXM0808S
;               CALL            FXM0808U
;               CALL            FXM0707U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                DECFSZ          TESTCOUNT
                GOTO            M0808LOOP
                RETLW           0x00
;       Test suite for FXM1608
TFXM1608        MOVLW           34
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           2
                MOVPF           WREG,TESTCODE
M1608LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
;               BCF             BARGB0,MSB
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
```

```
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           BARGB0,INDF0
                CALL            FXM1608S
;               CALL            FXM1608U
;               CALL            FXM1507U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                DECFSZ          TESTCOUNT
                GOTO            M1608LOOP
                RETLW           0x00
;       Test suite for FXM1616
TFXM1616        MOVLW           26
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           3
                MOVPF           WREG,TESTCODE
M1616LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
;               BCF             BARGB0,MSB
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
;               CALL            FXM1616S
                CALL            FXM1616U
;               CALL            FXM1515U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                DECFSZ          TESTCOUNT
                GOTO            M1616LOOP
                RETLW           0x00
;       Test suite for FXM2416
TFXM2416        MOVLW           20
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           4
                MOVPF           WREG,TESTCODE
M2416LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
;               BCF             BARGB0,MSB
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
```

```
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB2
;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
                CALL            FXM2416S
;               CALL            FXM2416U
;               CALL            FXM2315U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                MOVFP           AARGB4,INDF0
                DECFSZ          TESTCOUNT
                GOTO            M2416LOOP
                RETLW           0x00
;       Test suite for FXM2424
TFXM2424        MOVLW           17
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           5
                MOVPF           WREG,TESTCODE
M2424LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB2
                BCF             BARGB0,MSB
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB0
                CALL            RANDOM16

                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB1
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB2
                BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
                MOVFP           BARGB2,INDF0
;               CALL            FXM2424S
;               CALL            FXM2424U
                CALL            FXM2323U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                MOVFP           AARGB4,INDF0
                MOVFP           AARGB5,INDF0
                DECFSZ          TESTCOUNT
                GOTO            M2424LOOP
                RETLW           0x00
;       Test suite for FXM3216
TFXM3216        MOVLW           17
                MOVPF           WREG,TESTCOUNT
```

```
                MOVPF           WREG,NUMTESTS
                MOVLW           6
                MOVPF           WREG,TESTCODE
M3216LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
                BCF             BARGB0,MSB
                CALL            RANDOM16

                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB2
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB3
                BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
;               CALL            FXM3216S
;               CALL            FXM3216U
                CALL            FXM3115U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                MOVFP           AARGB4,INDF0
                MOVFP           AARGB5,INDF0
                DECFSZ          TESTCOUNT
                GOTO            M3216LOOP
                RETLW           0x00
;       Test suite for FXM3224
TFXM3224        MOVLW           14
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           6
                MOVPF           WREG,TESTCODE
M3224LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB2
;               BCF             BARGB0,MSB
                CALL            RANDOM16

                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB2
                MOVFP           RANDLO,WREG
```

```
                MOVPF           WREG,AARGB3
;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
                MOVFP           BARGB2,INDF0
                CALL            FXM3224S
;               CALL            FXM3224U
;               CALL            FXM3123U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                MOVFP           AARGB4,INDF0
                MOVFP           AARGB5,INDF0
                MOVFP           AARGB6,INDF0
                DECFSZ          TESTCOUNT
                GOTO            M3224LOOP
                RETLW           0x00
;       Test suite for FXM3232
TFXM3232        MOVLW           13
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           7
                MOVPF           WREG,TESTCODE
M3232LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB2
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB3
                BCF             BARGB0,MSB
                CALL            RANDOM16

                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB2
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB3
                BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
                MOVFP           BARGB2,INDF0
                MOVFP           BARGB3,INDF0
;               CALL            FXM3232S
;               CALL            FXM3232U
                CALL            FXM3131U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
```

```
                MOVFP           AARGB3,INDF0
                MOVFP           AARGB4,INDF0
                MOVFP           AARGB5,INDF0
                MOVFP           AARGB6,INDF0
                MOVFP           AARGB7,INDF0
                DECFSZ          TESTCOUNT
                GOTO            M3232LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;      Multiplication Macros
SMUL0808        macro
;       Max Timing:     4+6+6*5+3 = 43 clks
;       Min Timing:     4+14+3 = 21 clks
;       PM: 4+2*7+5+6*5+3 = 56            DM: 5
                variable i
                i = 0
                CLRF            SIGN
                BTFSC           AARGB0,MSB
                COMF            SIGN
                MOVPF           AARGB0,WREG

                while i < 7

                BTFSC           BARGB0,i
                GOTO            SM0808NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                RETLW           0
SM0808NA0       RLCF            SIGN
                RRCF            ACCB0
                RRCF            ACCB1
                i = 1
                while   i < 7
                BTFSC           BARGB0,i
                ADDWF           ACCB0
SM0808NA#v(i)   RLCF            SIGN
                RRCF            ACCB0
                RRCF            ACCB1
                i = i + 1
                endw
                RLCF            SIGN
                RRCF            ACCB0
                RRCF            ACCB1
                endm
UMUL0808        macro
;       Max Timing:     2+5+7*4 = 35 clks
;       Min Timing:     2+16+3 = 21 clks
;       PM: 2+2*8+4+7*4 = 50            DM: 3
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                MOVPF           AARGB0,WREG

                while i < 8

                BTFSC           BARGB0,i
                GOTO            UM0808NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                RETLW           0
UM0808NA0       RRCF            ACCB0
                RRCF            ACCB1
                i = 1
```

```
                while   i < 8
                BTFSC           BARGB0,i
                ADDWF           ACCB0
UM0808NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                i = i + 1
                endw
                endm
UMUL0707        macro
;       Max Timing:     2+5+6*4+2 = 33 clks
;       Min Timing:     2+14+3 = 19 clks
;       PM: 2+2*7+4+6*4+2 = 46          DM: 3
                variable i
                i = 0
                BCF             _C              ; clear carry for first right shift
                MOVPF           AARGB0,WREG
                while i < 7

                BTFSC           BARGB0,i
                GOTO            UM0707NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                RETLW           0
UM0707NA0       RRCF            ACCB0
                RRCF            ACCB1
                i = 1
                while   i < 7
                BTFSC           BARGB0,i
                ADDWF           ACCB0
UM0707NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                i = i + 1
                endw
                RRCF            ACCB0
                RRCF            ACCB1
                endm
;-------------------------------------------------------------------------------
SSMUL1608       macro
;       Max Timing:     3+6+6*9+3 = 66 clks
;       Min Timing:     3+21+5 = 29 clks
;       PM: 3+3*7+7+6*9+3 = 88          DM: 6
                variable i
                i = 0
                BTFSC           AARGB0,MSB
                COMF            ACCB2
                RLCF            ACCB0,W

                while i < 7

                BTFSC           BARGB0,i
                GOTO            SSM1608NA#v(i)
                BCF             ACCB2,7-i
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETLW           0
SSM1608NA0
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = 1
                while   i < 7
                BTFSS           BARGB0,i
```

```
                GOTO            SSM1608NA#v(i)
SSM1608A#v(i)   MOVFP           TEMPB1,WREG
                ADDWF           ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
SSM1608NA#v(i)
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = i + 1
                endw
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                endm
SMUL1608        macro
;       Max Timing:     7+6*10+4 = 71 clks
;       Min Timing:     7*2+4 = 18 clks
;       PM: 7*2+7+6*10+4 = 85           DM: 6
                variable i
                i = 0
                while i < 7

                BTFSC           BARGB0,i
                GOTO            SM1608NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                RETLW           0
SM1608NA0       RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = 1
                while   i < 7
                BTFSS           BARGB0,i
                GOTO            SM1608NA#v(i)
SM1608A#v(i)    MOVFP           TEMPB1,WREG
                ADDWF           ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
SM1608NA#v(i)   RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = i + 1
                endw
                RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                endm
UMUL1608        macro
;       Max Timing:     1+6+7*9 = 70 clks
;       Min Timing:     1+8*2+4 = 21 clks
;       PM: 1+8*2+6+7*9 = 86            DM: 6
                variable i

                i = 0

                BCF             _C
                while i < 8

                BTFSC           BARGB0,i
                GOTO            UM1608NA#v(i)
```

```
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                RETLW           0
UM1608NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = 1
                while   i < 8
                BTFSS           BARGB0,i
                GOTO            UM1608NA#v(i)
UM1608A#v(i)    MOVFP           TEMPB1,WREG
                ADDWF           ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM1608NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = i + 1
                endw
                endm
UMUL1507        macro
;       Max Timing:     1+6+6*9+3 = 64 clks
;       Min Timing:     1+7*2+4 = 19 clks
;       PM: 1+7*2+6+6*9+3 = 78            DM: 6
                variable i
                i = 0
                BCF             _C
                while i < 7

                BTFSC           BARGB0,i
                GOTO            UM1507NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0           ; if we get here, BARG = 0
                CLRF            ACCB1
                RETLW           0
UM1507NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = 1
                while   i < 7
                BTFSS           BARGB0,i
                GOTO            UM1507NA#v(i)
UM1507A#v(i)    MOVFP           TEMPB1,WREG
                ADDWF           ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM1507NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = i + 1
                endw
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                endm
;-------------------------------------------------------------------------------
SMUL1616        macro
;       Max Timing:     7+7*10+7*11+5 = 159 clks
;       Min Timing:     15*2+4 = 34 clks
;       PM: 15*2+7+7*10+7*11+5 = 193           DM: 8
                variable i
                i = 0
                while i < 15
```

```
                if i < 8
                BTFSC           BARGB1,i
                else
                BTFSC           BARGB0,i-8
                endif
                GOTO            SM1616NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                RETLW           0
SM1616NA0       RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = 1
                while   i < 15
                if i < 8
                BTFSS           BARGB1,i
                else
                BTFSS           BARGB0,i-8
                endif
                GOTO            SM1616NA#v(i)
SM1616A#v(i)    MOVFP           TEMPB1,WREG
                ADDWF           ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
SM1616NA#v(i)   RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                if i > 7
                RRCF            ACCB3
                endif
                i = i + 1
                endw
                RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                endm
UMUL1616        macro
;       Max Timing:     1+6+7*9+8*10 = 150 clks
;       Min Timing:     1+16*2+4 = 37 clks
;       PM: 1+16*2+4+6+7*9+8*10 = 186           DM: 8
                variable i

                i = 0

                BCF             _C
                while i < 16

                if i < 8
                BTFSC           BARGB1,i
                else
                BTFSC           BARGB0,i-8
                endif

                GOTO            UM1616NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                RETLW           0
```

```
UM1616NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = 1
                while   i < 16
                if i < 8
                BTFSS           BARGB1,i
                else
                BTFSS           BARGB0,i-8
                endif
                GOTO            UM1616NA#v(i)
UM1616A#v(i)    MOVFP           TEMPB1,WREG
                ADDWF           ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM1616NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                if i > 7
                RRCF            ACCB3
                endif
                i = i + 1
                endw
                endm
UMUL1515        macro
;       Max Timing:     1+6+7*9+7*10+4 = 144 clks
;       Min Timing:     1+15*2+4 = 35 clks
;       PM: 1+16*2+4+6+7*9+7*10+4 = 180           DM: 8
                variable i

                i = 0

                BCF             _C
                while i < 15

                if i < 8
                BTFSC           BARGB1,i
                else
                BTFSC           BARGB0,i-8
                endif

                GOTO            UM1515NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                RETLW           0
UM1515NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                i = 1
                while   i < 15
                if i < 8
                BTFSS           BARGB1,i
                else
                BTFSS           BARGB0,i-8
                endif
                GOTO            UM1515NA#v(i)
UM1515A#v(i)    MOVFP           TEMPB1,WREG
                ADDWF           ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM1515NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                if i > 7
```

```
                RRCF            ACCB3
                endif
                i = i + 1
                endw
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                endm
;-------------------------------------------------------------------------
SMUL2416        macro
;       Max Timing:     8+7*13+7*14+6 = 203 clks
;       Min Timing:     15*2+5 = 35 clks
;       PM: 15*2+9+7*13+7*14+6 = 234            DM: 10
                variable i
                i = 0
                while i < 15

                if i < 8
                BTFSC           BARGB1,i
                else

                BTFSC           BARGB0,i-8
                endif
                GOTO            SM2416NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETLW           0
SM2416NA0       RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                i = 1
                while   i < 15
                if i < 8
                BTFSS           BARGB1,i
                else
                BTFSS           BARGB0,i-8
                endif
                GOTO            SM2416NA#v(i)
SM2416A#v(i)    MOVFP           TEMPB2,WREG
                ADDWF           ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
SM2416NA#v(i)   RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                if i > 7
                RRCF            ACCB4
                endif
                i = i + 1
                endw
                RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
```

```
                endm
UMUL2416        macro
;       Max Timing:     1+7+7*12+8*13 = 196 clks
;       Min Timing:     1+16*2+5 = 38 clks
;       PM: 1+16*2+5+7+7*12+8*13 = 233            DM: 10
                variable i

                i = 0

                BCF             _C
                while i < 16

                if i < 8
                BTFSC           BARGB1,i
                else
                BTFSC           BARGB0,i-8
                endif

                GOTO            UM2416NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETLW           0
UM2416NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                i = 1
                while   i < 16
                if i < 8
                BTFSS           BARGB1,i
                else
                BTFSS           BARGB0,i-8
                endif
                GOTO            UM2416NA#v(i)
UM2416A#v(i)    MOVFP           TEMPB2,WREG
                ADDWF           ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM2416NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                if i > 7
                RRCF            ACCB4
                endif
                i = i + 1
                endw
                endm
UMUL2315        macro
;       Max Timing:     1+7+7*12+7*13+5 = 188 clks
;       Min Timing:     1+15*2+5 = 36 clks
;       PM: 1+15*2+5+7+7*2+7*13+5 = 223           DM: 10
                variable i

                i = 0

                BCF             _C
                while i < 15

                if i < 8
                BTFSC           BARGB1,i
```

```
                else
                BTFSC           BARGB0,i-8
                endif

                GOTO            UM2315NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETLW           0
UM2315NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                i = 1
                while   i < 15
                if i < 8
                BTFSS           BARGB1,i
                else
                BTFSS           BARGB0,i-8
                endif
                GOTO            UM2315NA#v(i)
UM2315A#v(i)    MOVFP           TEMPB2,WREG
                ADDWF           ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM2315NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                if i > 7
                RRCF            ACCB4
                endif
                i = i + 1
                endw
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                endm
;-------------------------------------------------------------------------------
SMUL2424        macro
;       Max Timing:     8+7*13+8*14+7*15+7 = 323 clks
;       Min Timing:     23*2+5 = 51 clks
;       PM: 23*2+9+7*13+8*14+7*15+7 = 370            DM: 12
                variable i
                i = 0
                while i < 23

                if i < 8
                BTFSC           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSC           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSC           BARGB0,i-16
                endif
                GOTO            SM2424NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
```

```
                CLRF            ACCB1
                CLRF            ACCB2
                RETLW           0
SM2424NA0       RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                i = 1
                while   i < 23
                if i < 8
                BTFSS           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSS           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSS           BARGB0,i-16
                endif
                GOTO            SM2424NA#v(i)
SM2424A#v(i)    MOVFP           TEMPB2,WREG
                ADDWF           ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
SM2424NA#v(i)   RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                if i > 7
                RRCF            ACCB4
                endif
                if i > 15
                RRCF            ACCB5
                endif
                i = i + 1
                endw
                RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                RRCF            ACCB5
                endm
UMUL2424        macro
;       Max Timing:     1+7+7*12+8*13+8*14 = 308 clks
;       Min Timing:     1+24*2+5 = 54 clks
;       PM: 1+24*2+8+7*12+8*13+8*14 = 357           DM: 12
                variable i
                i = 0

                BCF             _C
                while i < 24

                if i < 8
                BTFSC           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSC           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSC           BARGB0,i-16
                endif
```

```
                GOTO            UM2424NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                RETLW           0
UM2424NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                i = 1
                while   i < 24
                if i < 8
                BTFSS           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSS           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSS           BARGB0,i-16
                endif
                GOTO            UM2424NA#v(i)
UM2424A#v(i)    MOVFP           TEMPB2,WREG
                ADDWF           ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM2424NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                if i > 7
                RRCF            ACCB4
                endif
                if i > 15
                RRCF            ACCB5
                endif
                i = i + 1
                endw
                endm
UMUL2323        macro
;       Max Timing:     1+7+7*12+8*13+7*14+6 = 300 clks
;       Min Timing:     1+23*2+5 = 52 clks
;       PM: 1+23*2+8+7*12+8*13+7*14+6 = 347            DM: 12
                variable i
                i = 0

                BCF             _C
                while i < 23

                if i < 8
                BTFSC           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSC           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSC           BARGB0,i-16
                endif
                GOTO            UM2323NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
```

```
                CLRF            ACCB2
                RETLW           0
UM2323NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                i = 1
                while   i < 23
                if i < 8
                BTFSS           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSS           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSS           BARGB0,i-16
                endif
                GOTO            UM2323NA#v(i)
UM2323A#v(i)    MOVFP           TEMPB2,WREG
                ADDWF           ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM2323NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                if i > 7
                RRCF            ACCB4
                endif
                if i > 15
                RRCF            ACCB5
                endif
                i = i + 1
                endw
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                RRCF            ACCB5
                endm
;-------------------------------------------------------------------------------
SMUL3216        macro
;       Max Timing:     9+7*16+7*17+7 = 247 clks
;       Min Timing:     15*2+6 = 36 clks
;       PM: 15*2+11+7*16+7*17+7 = 279           DM: 13
                variable i
                i = 0
                while i < 15

                if i < 8
                BTFSC           BARGB1,i
                endif
                if (i >= 8)
                BTFSC           BARGB0,i-8
                endif
                GOTO            SM3216NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETLW           0
```

```
SM3216NA0       RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                i = 1
                while   i < 15
                if i < 8
                BTFSS           BARGB1,i
                endif
                if (i >= 8)
                BTFSS           BARGB0,i-8
                endif
                GOTO            SM3216NA#v(i)
SM3216A#v(i)    MOVFP           TEMPB3,WREG
                ADDWF           ACCB3
                MOVFP           TEMPB2,WREG
                ADDWFC          ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
SM3216NA#v(i)   RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                if i > 7
                RRCF            ACCB5
                endif
                i = i + 1
                endw
                RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                RRCF            ACCB5
                endm
UMUL3216        macro
;       Max Timing:     1+8+7*16+8*17 = 257 clks
;       Min Timing:     16*2+6 = 38 clks
;       PM: 1+16*2+10+7*16+8*17 = 291           DM: 13
                variable i
                i = 0
                BCF             _C
                while i < 16

                if i < 8
                BTFSC           BARGB1,i
                endif
                if (i >= 8)
                BTFSC           BARGB0,i-8
                endif
                GOTO            UM3216NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETLW           0
UM3216NA0       RRCF            ACCB0
```

```
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                i = 1
                while   i < 16
                if i < 8
                BTFSS           BARGB1,i
                endif
                if (i >= 8)
                BTFSS           BARGB0,i-8
                endif
                GOTO            UM3216NA#v(i)
UM3216A#v(i)    MOVFP           TEMPB3,WREG
                ADDWF           ACCB3
                MOVFP           TEMPB2,WREG
                ADDWFC          ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM3216NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                if i > 7
                RRCF            ACCB5
                endif
                i = i + 1
                endw
                endm
UMUL3115        macro
;       Max Timing:     1+8+7*16+7*17+6 = 246 clks
;       Min Timing:     15*2+6 = 36 clks
;       PM: 1+15*2+10+7*16+7*17+6 = 278           DM: 13
                variable i
                i = 0
                BCF             _C
                while i < 15

                if i < 8
                BTFSC           BARGB1,i
                endif
                if (i >= 8)
                BTFSC           BARGB0,i-8
                endif
                GOTO            UM3115NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETLW           0
UM3115NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                i = 1
                while   i < 15
                if i < 8
                BTFSS           BARGB1,i
                endif
                if (i >= 8)
```

```
                BTFSS           BARGB0,i-8
                endif
                GOTO            UM3115NA#v(i)
UM3115A#v(i)    MOVFP           TEMPB3,WREG
                ADDWF           ACCB3
                MOVFP           TEMPB2,WREG
                ADDWFC          ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM3115NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                if i > 7
                RRCF            ACCB5
                endif
                i = i + 1
                endw
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                RRCF            ACCB5
                endm
;-------------------------------------------------------------------------
SMUL3224        macro
;       Max Timing:     9+7*16+8*17+7*18+8 = 391 clks
;       Min Timing:     23*2+6 = 52 clks
;       PM: 23*2+11+7*16+8*17+7*18+8 = 439            DM: 15
                variable i
                i = 0
                while i < 23

                if i < 8
                BTFSC           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSC           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSC           BARGB0,i-16
                endif
                GOTO            SM3224NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETLW           0
SM3224NA0       RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                i = 1
                while   i < 23
                if i < 8
                BTFSS           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
```

```
                BTFSS           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSS           BARGB0,i-16
                endif
                GOTO            SM3224NA#v(i)
SM3224A#v(i)    MOVFP           TEMPB3,WREG
                ADDWF           ACCB3
                MOVFP           TEMPB2,WREG
                ADDWFC          ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
SM3224NA#v(i)   RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                if i > 7
                RRCF            ACCB5
                endif
                if i > 15
                RRCF            ACCB6
                endif
                i = i + 1
                endw
                RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                RRCF            ACCB5
                RRCF            ACCB6
                endm
UMUL3224        macro
;       Max Timing:     1+8+7*16+8*17+8*18 = 401 clks
;       Min Timing:     24*2+6 = 54 clks
;       PM: 1+24*2+10+7*16+8*17+8*18 = 451           DM: 15
                variable i
                i = 0
                BCF             _C
                while i < 24

                if i < 8
                BTFSC           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSC           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSC           BARGB0,i-16
                endif
                GOTO            UM3224NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETLW           0
UM3224NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
```

```
                RRCF            ACCB3
                RRCF            ACCB4
                i = 1
                while   i < 24
                if i < 8
                BTFSS           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSS           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSS           BARGB0,i-16
                endif
                GOTO            UM3224NA#v(i)
UM3224A#v(i)    MOVFP           TEMPB3,WREG
                ADDWF           ACCB3
                MOVFP           TEMPB2,WREG
                ADDWFC          ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM3224NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                if i > 7
                RRCF            ACCB5
                endif
                if i > 15
                RRCF            ACCB6
                endif
                i = i + 1
                endw
                endm
UMUL3123        macro
;       Max Timing:     1+8+7*16+8*17+7*18+7 = 390 clks
;       Min Timing:     23*2+6 = 52 clks
;       PM: 1+23*2+10+7*16+8*17+7*18+7 = 438           DM: 15
                variable i
                i = 0
                BCF             _C
                while i < 23

                if i < 8
                BTFSC           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSC           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSC           BARGB0,i-16
                endif
                GOTO            UM3123NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETLW           0
UM3123NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
```

```
                RRCF            ACCB4
                i = 1
                while   i < 23
                if i < 8
                BTFSS           BARGB2,i
                endif
                if (i >= 8) && (i < 16)
                BTFSS           BARGB1,i-8
                endif
                if (i >= 16)
                BTFSS           BARGB0,i-16
                endif
                GOTO            UM3123NA#v(i)
UM3123A#v(i)    MOVFP           TEMPB3,WREG
                ADDWF           ACCB3
                MOVFP           TEMPB2,WREG
                ADDWFC          ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM3123NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                if i > 7
                RRCF            ACCB5
                endif
                if i > 15
                RRCF            ACCB6
                endif
                i = i + 1
                endw
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                RRCF            ACCB5
                RRCF            ACCB6
                endm
;-------------------------------------------------------------------------------
SMUL3232        macro
;       Max Timing:     9+7*16+8*17+8*18+7*19+9 = 543 clks
;       Min Timing:     31*2+6 = 68 clks
;       PM: 31*2+11+7*16+8*17+8*18+7*19+9 = 607            DM: 16
                variable i
                i = 0
                while i < 31

                if i < 8
                BTFSC           BARGB3,i
                endif
                if (i >= 8) && (i < 16)
                BTFSC           BARGB2,i-8
                endif
                if (i >= 16) && (i < 24)
                BTFSC           BARGB1,i-16
                endif
                if (i >= 24)
                BTFSC           BARGB0,i-24
                endif
                GOTO            SM3232NA#v(i)
                i = i + 1
                endw
```

```
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETLW           0
SM3232NA0       RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                i = 1
                while   i < 31
                if i < 8
                BTFSS           BARGB3,i
                endif
                if (i >= 8) && (i < 16)
                BTFSS           BARGB2,i-8
                endif
                if (i >= 16) && (i < 24)
                BTFSS           BARGB1,i-16
                endif
                if (i >= 24)
                BTFSS           BARGB0,i-24
                endif
                GOTO            SM3232NA#v(i)
SM3232A#v(i)    MOVFP           TEMPB3,WREG
                ADDWF           ACCB3
                MOVFP           TEMPB2,WREG
                ADDWFC          ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
SM3232NA#v(i)   RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                if i > 7
                RRCF            ACCB5
                endif
                if i > 15
                RRCF            ACCB6
                endif
                if i > 23
                RRCF            ACCB7
                endif
                i = i + 1
                endw
                RLCF            TEMPB0,W
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                RRCF            ACCB5
                RRCF            ACCB6
                RRCF            ACCB7
                endm
UMUL3232        macro
;       Max Timing:     1+8+7*16+8*17+8*18+8*19 = 553 clks
;       Min Timing:     32*2+6 = 70 clks
;       PM: 1+32*2+10+7*16+8*17+8*18+8*19 = 619           DM: 16
                variable i
```

```
                i = 0
                BCF             _C
                while i < 32

                if i < 8
                BTFSC           BARGB3,i
                endif
                if (i >= 8) && (i < 16)
                BTFSC           BARGB2,i-8
                endif
                if (i >= 16) && (i < 24)
                BTFSC           BARGB1,i-16
                endif
                if (i >= 24)
                BTFSC           BARGB0,i-24
                endif
                GOTO            UM3232NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETLW           0
UM3232NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                i = 1
                while   i < 32
                if i < 8
                BTFSS           BARGB3,i
                endif
                if (i >= 8) && (i < 16)
                BTFSS           BARGB2,i-8
                endif
                if (i >= 16) && (i < 24)
                BTFSS           BARGB1,i-16
                endif
                if (i >= 24)
                BTFSS           BARGB0,i-24
                endif
                GOTO            UM3232NA#v(i)
UM3232A#v(i)    MOVFP           TEMPB3,WREG
                ADDWF           ACCB3
                MOVFP           TEMPB2,WREG
                ADDWFC          ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM3232NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                if i > 7
                RRCF            ACCB5
                endif
                if i > 15
                RRCF            ACCB6
                endif
                if i > 23
                RRCF            ACCB7
                endif
```

```
                i = i + 1
                endw
                endm
UMUL3131        macro
;       Max Timing:     1+8+7*16+8*17+8*18+7*19+9 = 533 clks
;       Min Timing:     31*2+6 = 68 clks
;       PM: 1+31*2+10+7*16+8*17+8*18+7*19+9 = 597            DM: 16
                variable i
                i = 0
                BCF             _C
                while i < 31

                if i < 8
                BTFSC           BARGB3,i
                endif
                if (i >= 8) && (i < 16)
                BTFSC           BARGB2,i-8
                endif
                if (i >= 16) && (i < 24)
                BTFSC           BARGB1,i-16
                endif
                if (i >= 24)
                BTFSC           BARGB0,i-24
                endif
                GOTO            UM3131NA#v(i)
                i = i + 1
                endw
                CLRF            ACCB0         ; if we get here, BARG = 0
                CLRF            ACCB1
                CLRF            ACCB2
                CLRF            ACCB3
                RETLW           0
UM3131NA0       RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                i = 1
                while   i < 31
                if i < 8
                BTFSS           BARGB3,i
                endif
                if (i >= 8) && (i < 16)
                BTFSS           BARGB2,i-8
                endif
                if (i >= 16) && (i < 24)
                BTFSS           BARGB1,i-16
                endif
                if (i >= 24)
                BTFSS           BARGB0,i-24
                endif
                GOTO            UM3131NA#v(i)
UM3131A#v(i)    MOVFP           TEMPB3,WREG
                ADDWF           ACCB3
                MOVFP           TEMPB2,WREG
                ADDWFC          ACCB2
                MOVFP           TEMPB1,WREG
                ADDWFC          ACCB1
                MOVFP           TEMPB0,WREG
                ADDWFC          ACCB0
UM3131NA#v(i)   RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                if i > 7
```

```
                RRCF            ACCB5
                endif
                if i > 15
                RRCF            ACCB6
                endif
                if i > 23
                RRCF            ACCB7
                endif
                i = i + 1
                endw
                RRCF            ACCB0
                RRCF            ACCB1
                RRCF            ACCB2
                RRCF            ACCB3
                RRCF            ACCB4
                RRCF            ACCB5
                RRCF            ACCB6
                RRCF            ACCB7
                endm
;*****************************************************************************************
;*****************************************************************************************

;       8x8 Bit Signed Fixed Point Multiply 08 x 08 -> 16
;       Input:  8 bit signed fixed point multiplicand in AARGB0
;               8 bit signed fixed point multiplier in BARGB0
;       Use:    CALL    FXM0808S
;       Output: 16 bit signed fixed point product in AARGB0, AARGB1
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     5+43+2 = 50 clks                B > 0
;                       8+43+2 = 53 clks                B < 0
;       Min Timing:     5+21 = 26 clks
;       PM: 8+56+1 = 65             DM: 5
FXM0808S        BTFSS           BARGB0,MSB
                GOTO            M0808SOK
                COMF            BARGB0          ; make multiplier BARG > 0
                INCF            BARGB0
                COMF            AARGB0
                INCF            AARGB0
M0808SOK        CLRF            ACCB1           ; clear partial product
                MOVPF           AARGB0,TEMPB0
                SMUL0808
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       8x8 Bit Unsigned Fixed Point Multiply 08 x 08 -> 16
;       Input:  8 bit unsigned fixed point multiplicand in AARGB0
;               8 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM0808U
;       Output: 16 bit unsigned fixed point product in AARGB0, AARGB1
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     2+35+2 = 39 clks
;       Min Timing:     2+21 = 23 clks
;       PM: 2+50+1 = 53             DM: 3
FXM0808U        CLRF            ACCB1           ; clear partial product
                MOVPF           AARGB0,TEMPB0
                UMUL0808
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       7x7 Bit Unsigned Fixed Point Multiply 07 x 07 -> 14
;       Input:  7 bit unsigned fixed point multiplicand in AARGB0
;               7 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM0707U
;       Output: 14 bit unsigned fixed point product in AARGB0, AARGB1
```

```
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     2+33+2 = 37 clks
;       Min Timing:     2+19 = 21 clks
;       PM: 2+46+1 = 49             DM: 3
FXM0707U        CLRF            ACCB1           ; clear partial product
                MOVPF           AARGB0,TEMPB0
                UMUL0707
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16x8 Bit Signed Fixed Point Multiply 16 x 08 -> 24
;       Input:  16 bit signed fixed point multiplicand in AARGB0
;               8 bit signed fixed point multiplier in BARGB0
;       Use:    CALL    FXM1608S
;       Output: 24 bit signed fixed point product in AARGB0, AARGB1
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     6+66+2 = 74 clks                B > 0
;                       11+66+2 = 79 clks                B < 0
;       Min Timing:     6+29 = 35 clks
;       PM: 11+88+1 = 100               DM: 6
FXM1608S        BTFSS           BARGB0,MSB
                GOTO            M1608SOK
                COMF            BARGB0          ; make multiplier BARG > 0
                INCF            BARGB0
                COMF            AARGB0
                COMF            AARGB1
                INFSNZ          AARGB1
                INCF            AARGB0
M1608SOK        CLRF            ACCB2           ; clear partial product
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                SSMUL1608
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16x8 Bit Unsigned Fixed Point Multiply 16 x 08 -> 24
;       Input:  16 bit unsigned fixed point multiplicand in AARGB0
;               8 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM1608U
;       Output: 24 bit unsigned fixed point product in AARGB0, AARGB1, AARGB2
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     3+70+2 = 75 clks
;       Min Timing:     3+21 = 24 clks
;       PM: 3+86+1 = 90             DM: 6
FXM1608U        CLRF            ACCB2
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                UMUL1608
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       15x7 Bit Unsigned Fixed Point Multiply 15 x 07 -> 22
;       Input:  15 bit unsigned fixed point multiplicand in AARGB0,AARGB1
;               7 bit unsigned fixed point multiplier in BARGB0
;       Use:    CALL    FXM1507U
;       Output: 22 bit unsigned fixed point product in AARGB0, AARGB1, AARGB2
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     3+64+2 = 69 clks
;       Min Timing:     3+21 = 24 clks
;       PM: 3+78+1 = 82             DM: 6
FXM1507U        CLRF            ACCB2
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
```

```
                UMUL1507
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16x16 Bit Signed Fixed Point Multiply 16 x 16 -> 32
;       Input:  16 bit signed fixed point multiplicand in AARGB0, AARGB1
;               16 bit signed fixed point multiplier in BARGB0, BARGB1
;       Use:    CALL    FXM1616S
;       Output: 32 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     7+159+2 = 168 clks               B > 0
;                       14+159+2 = 175 clks               B < 0
;       Min Timing:     6+34 = 24 clks
;       PM: 11+193+1 = 97               DM: 8
FXM1616S        BTFSS           BARGB0,MSB
                GOTO            M1616SOK
                COMF            BARGB0          ; make multiplier BARG > 0
                COMF            BARGB1
                INFSNZ          BARGB1
                INCF            BARGB0
                COMF            AARGB0
                COMF            AARGB1
                INFSNZ          AARGB1
                INCF            AARGB0
M1616SOK        CLRF            ACCB2           ; clear partial product
                CLRF            ACCB3
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                SMUL1616
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16x16 Bit Unsigned Fixed Point Multiply 16 x 16 -> 32
;       Input:  16 bit unsigned fixed point multiplicand in AARGB0, AARGB1
;               16 bit unsigned fixed point multiplier in BARGB0, BARGB1
;       Use:    CALL    FXM1616U
;       Output: 32 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     4+150+2 = 156 clks
;       Min Timing:     4+37 = 41 clks
;       PM: 4+186+1 = 191               DM: 8
FXM1616U        CLRF            ACCB2
                CLRF            ACCB3
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                UMUL1616
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       15x15 Bit Unsigned Fixed Point Multiply 15 x 15 -> 30
;       Input:  15 bit unsigned fixed point multiplicand in AARGB0, AARGB1
;               15 bit unsigned fixed point multiplier in BARGB0, BARGB1
;       Use:    CALL    FXM1515U
;       Output: 30 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     4+144+2 = 150 clks
;       Min Timing:     4+35 = 39 clks
;       PM: 4+180+1 = 185               DM: 8
FXM1515U        CLRF            ACCB2
                CLRF            ACCB3
```

```
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                UMUL1515
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       24x16 Bit Signed Fixed Point Multiply 24 x 16 -> 40
;       Input:  24 bit signed fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2
;               16 bit signed fixed point multiplier in BARGB0, BARGB1
;       Use:    CALL    FXM2416S
;       Output: 40 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     8+203+2 = 213 clks               B > 0
;                       18+203+2 = 223 clks               B < 0
;       Min Timing:     8+35 = 43 clks
;       PM: 18+234+1 = 253              DM: 10
FXM2416S        BTFSS           BARGB0,MSB
                GOTO            M2416SOK
                COMF            BARGB0          ; make multiplier BARG > 0
                COMF            BARGB1
                INFSNZ          BARGB1
                INCF            BARGB0
                CLRF            WREG
                COMF            AARGB0
                COMF            AARGB1
                COMF            AARGB2
                INCF            AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
M2416SOK        CLRF            ACCB3           ; clear partial product
                CLRF            ACCB4
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                SMUL2416
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       24x16 Bit Unsigned Fixed Point Multiply 24 x 16 -> 40
;       Input:  24 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2
;               16 bit unsigned fixed point multiplier in BARGB0, BARGB1
;       Use:    CALL    FXM2416U
;       Output: 40 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     5+196+2 = 203 clks
;       Min Timing:     5+38 = 43 clks
;       PM: 5+233+1 = 239               DM: 10
FXM2416U        CLRF            ACCB3
                CLRF            ACCB4
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                UMUL2416
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       23x15 Bit Unsigned Fixed Point Multiply 23 x 15 -> 38
;       Input:  23 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2
```

```
;               15 bit unsigned fixed point multiplier in BARGB0, BARGB1
;       Use:    CALL    FXM2315U
;       Output: 38 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     4+188+2 = 194 clks
;       Min Timing:     5+36 = 41 clks
;       PM: 5+223+1 = 229             DM: 10
FXM2315U        CLRF            ACCB3
                CLRF            ACCB4
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                UMUL2315
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************

;       24x24 Bit Signed Fixed Point Multiply 24 x 24 -> 48
;       Input:  24 bit signed fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2
;               24 bit signed fixed point multiplier in BARGB0, BARGB1,
;               BARGB2
;       Use:    CALL    FXM2424S
;       Output: 48 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     9+323+2 = 334 clks               B > 0
;                       21+323+2 = 346 clks               B < 0
;       Min Timing:     9+51 = 60 clks
;       PM: 21+370+1 = 392              DM: 12
FXM2424S        BTFSS           BARGB0,MSB
                GOTO            M2424SOK
                CLRF            WREG
                COMF            BARGB0          ; make multiplier BARG > 0
                COMF            BARGB1
                COMF            BARGB2
                INCF            BARGB2
                ADDWFC          BARGB1
                ADDWFC          BARGB0
                COMF            AARGB0
                COMF            AARGB1
                COMF            AARGB2
                INCF            AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
M2424SOK        CLRF            ACCB3           ; clear partial product
                CLRF            ACCB4
                CLRF            ACCB5
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                SMUL2424
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************

;       24x24 Bit Unsigned Fixed Point Multiply 24 x 24 -> 48
;       Input:  24 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2
;               24 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2
;       Use:    CALL    FXM2424U
;       Output: 48 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5
;       Result: AARG  <--  AARG * BARG
```

```
;       Max Timing:     6+308+2 = 316 clks
;       Min Timing:     6+54 = 60 clks
;       PM: 6+357+1 = 364              DM: 12
FXM2424U        CLRF            ACCB3
                CLRF            ACCB4
                CLRF            ACCB5
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                UMUL2424
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************

;       23x23 Bit Unsigned Fixed Point Multiply 23 x 23 -> 46
;       Input:  23 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2
;               23 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2
;       Use:    CALL    FXM2323U
;       Output: 46 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     6+300+2 = 308 clks
;       Min Timing:     6+52 = 58 clks
;       PM: 6+347+1 = 354              DM: 12
FXM2323U        CLRF            ACCB3
                CLRF            ACCB4
                CLRF            ACCB5
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                UMUL2323
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************

;       32x16 Bit Signed Fixed Point Multiply 32 x 16 -> 48
;       Input:  32 bit signed fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               16 bit signed fixed point multiplier in BARGB0, BARGB1
;       Use:    CALL    FXM3216S
;       Output: 48 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     9+247+2 = 258 clks               B > 0
;                       21+247+2 = 270 clks               B < 0
;       Min Timing:     10+36 = 46 clks
;       PM: 21+279+1 = 301              DM: 13
FXM3216S        BTFSS           BARGB0,MSB
                GOTO            M3216SOK
                CLRF            WREG
                COMF            BARGB0          ; make multiplier BARG > 0
                COMF            BARGB1
                INCF            BARGB1
                ADDWFC          BARGB0
                COMF            AARGB0
                COMF            AARGB1
                COMF            AARGB2
                COMF            AARGB3
                INCF            AARGB3
                ADDWFC          AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
M3216SOK        CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
```

```
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3
                SMUL3216
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32x16 Bit Unsigned Fixed Point Multiply 32 x 16 -> 48
;       Input:  32 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               16 bit unsigned fixed point multiplier in BARGB0, BARGB1
;       Use:    CALL    FXM3216U
;       Output: 48 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     6+257+2 = 265 clks
;       Min Timing:     6+38 = 44 clks
;       PM: 6+291+1 = 298               DM: 13
FXM3216U        CLRF            ACCB4
                CLRF            ACCB5
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3
                UMUL3216
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       31x15 Bit Unsigned Fixed Point Multiply 31 x 15 -> 46
;       Input:  31 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               15 bit unsigned fixed point multiplier in BARGB0, BARGB1
;       Use:    CALL    FXM3115U
;       Output: 46 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     6+246+2 = 254 clks
;       Min Timing:     6+36 = 42 clks
;       PM: 6+278+1 = 285               DM: 13
FXM3115U        CLRF            ACCB4
                CLRF            ACCB5
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3
                UMUL3115
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32x24 Bit Signed Fixed Point Multiply 32 x 24 -> 56
;       Input:  32 bit signed fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               24 bit signed fixed point multiplier in BARGB0, BARGB1,
;               BARGB2
;       Use:    CALL    FXM3224S
;       Output: 56 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     10+391+2 = 403 clks             B > 0
;                       24+391+2 = 417 clks             B < 0
;       Min Timing:     10+52 = 62 clks
;       PM: 24+439+1 = 464              DM: 15
```

```
FXM3224S        BTFSS           BARGB0,MSB
                GOTO            M3224SOK
                CLRF            WREG
                COMF            BARGB0          ; make multiplier BARG > 0
                COMF            BARGB1
                COMF            BARGB2
                INCF            BARGB2
                ADDWFC          BARGB1
                ADDWFC          BARGB0
                COMF            AARGB0
                COMF            AARGB1
                COMF            AARGB2
                COMF            AARGB3
                INCF            AARGB3
                ADDWFC          AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
M3224SOK        CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                CLRF            ACCB6
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3
                SMUL3224
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32x24 Bit Unsigned Fixed Point Multiply 32 x 24 -> 56
;       Input:  32 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               24 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2
;       Use:    CALL    FXM3224U
;       Output: 56 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     7+401+2 = 410 clks
;       Min Timing:     7+54 = 61 clks
;       PM: 7+451+1 = 459               DM: 15
FXM3224U        CLRF            ACCB4
                CLRF            ACCB5
                CLRF            ACCB6
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3
                UMUL3224
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       31x23 Bit Unsigned Fixed Point Multiply 31 x 23 -> 54
;       Input:  31 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               23 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2
;       Use:    CALL    FXM3123U
;       Output: 54 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     7+390+2 = 399 clks
;       Min Timing:     7+52 = 59 clks
;       PM: 7+438+1 = 446               DM: 15
FXM3123U        CLRF            ACCB4
```

```
                CLRF            ACCB5
                CLRF            ACCB6
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3
                UMUL3123
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32x32 Bit Signed Fixed Point Multiply 32 x 32 -> 64
;       Input:  32 bit signed fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               32 bit signed fixed point multiplier in BARGB0, BARGB1,
;               BARGB2, BARGB3
;       Use:    CALL    FXM3232S
;       Output: 64 bit signed fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6, AARGB7
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     11+543+2 = 556 clks               B > 0
;                       27+543+2 = 572 clks               B < 0
;       Min Timing:     11+68 = 79 clks
;       PM: 27+607+1 = 635              DM: 16
FXM3232S        BTFSS           BARGB0,MSB
                GOTO            M3232SOK
                CLRF            WREG
                COMF            BARGB0          ; make multiplier BARG > 0
                COMF            BARGB1
                COMF            BARGB2
                COMF            BARGB3
                INCF            BARGB3
                ADDWFC          BARGB2
                ADDWFC          BARGB1
                ADDWFC          BARGB0
                COMF            AARGB0
                COMF            AARGB1
                COMF            AARGB2
                COMF            AARGB3
                INCF            AARGB3
                ADDWFC          AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
M3232SOK        CLRF            ACCB4           ; clear partial product
                CLRF            ACCB5
                CLRF            ACCB6
                CLRF            ACCB7
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3
                SMUL3232
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32x32 Bit Unsigned Fixed Point Multiply 32 x 32 -> 64
;       Input:  32 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               32 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2, BARGB3
;       Use:    CALL    FXM3232U
;       Output: 64 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6, AARGB7
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     8+553+2 = 563 clks
```

```
;       Min Timing:     8+70 = 78 clks
;       PM: 8+619+1 = 628               DM: 16
FXM3232U        CLRF            ACCB4
                CLRF            ACCB5
                CLRF            ACCB6
                CLRF            ACCB7
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3
                UMUL3232
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************

;       31x31 Bit Unsigned Fixed Point Multiply 31 x 31 -> 62
;       Input:  31 bit unsigned fixed point multiplicand in AARGB0, AARGB1,
;               AARGB2, AARGB3
;               31 bit unsigned fixed point multiplier in BARGB0, BARGB1,
;               BARGB2, BARGB3
;       Use:    CALL    FXM3131U
;       Output: 62 bit unsigned fixed point product in AARGB0, AARGB1,
;               AARGB2, AARGB3, AARGB4, AARGB5, AARGB6, AARGB7
;       Result: AARG  <--  AARG * BARG
;       Max Timing:     8+533+2 = 543 clks
;       Min Timing:     8+68 = 76 clks
;       PM: 8+597+1 = 606               DM: 16
FXM3131U        CLRF            ACCB4
                CLRF            ACCB5
                CLRF            ACCB6
                CLRF            ACCB7
                MOVPF           AARGB0,TEMPB0
                MOVPF           AARGB1,TEMPB1
                MOVPF           AARGB2,TEMPB2
                MOVPF           AARGB3,TEMPB3
                UMUL3131
                RETLW           0x00
;*************************************************************************************************
;*************************************************************************************************
                END
```

NOTES:

Please check the Microchip BBS for the latest version of the source code. For BBS access information, see Section 6, Microchip Bulletin Board Service information, page 6-3.

APPENDIX F: PIC17CXX DIVIDE ROUTINES

Table of Contents for Appendix F

F.1 PIC17CXX Fixed Point Divide Routines A

```
;       PIC17 FIXED POINT DIVIDE ROUTINES A     VERSION 1.8
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXD3232S    614 32 bit/32 bit -> 32.32 signed fixed point divide
;       FXD3232U    683 32 bit/32 bit -> 32.32 unsigned fixed point divide
;       FXD3231U    588 32 bit/31 bit -> 32.31 unsigned fixed point divide
;       FXD3131U    579 31 bit/31 bit -> 31.31 unsigned fixed point divide
;       FXD3224S    514 32 bit/24 bit -> 32.24 signed fixed point divide
;       FXD3224U    584 32 bit/24 bit -> 32.24 unsigned fixed point divide
;       FXD3223U    489 32 bit/23 bit -> 32.23 unsigned fixed point divide
;       FXD3123U    481 31 bit/23 bit -> 31.23 unsigned fixed point divide
                list    r=dec,x=on,t=off,p=17C42
                include <PIC17.INC>     ; general PIC17 definitions

                include <MATH17.INC>    ; PIC17 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x2B    ; random number senerator registers
RANDLO          equ     0x2C
TESTCODE        equ     0x2D    ; integer code labeling test contained in following data
NUMTESTS        equ     0x2E    ; number of tests contained in following data
TESTCOUNT       equ     0x2F    ; counter
DATA            equ     0x30    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for fixed point divide algorithms
                org             0x0021
MAIN            MOVLW           RAMSTART
                MOVPF           WREG,FSR0
MEMLOOP         CLRF            INDF0
                INCFSZ          FSR0
                GOTO            MEMLOOP
                BSF             RTCSTA,5
;               MOVPF           RTCCH,WREG
                MOVLW           0x45                    ; seed for random numbers
                MOVPF           WREG,RANDLO
;               MOVPF           RTCCL,WREG
                MOVLW           0x30
                MOVPF           WREG,RANDHI
                MOVLW           0x30
                MOVPF           WREG,FSR0
                BCF             _FS1
                BSF             _FS0
                CALL            TFXD3232
;               CALL            TFXD3224
SELF            GOTO            SELF
RANDOM16        RLCF            RANDHI,W                ; random number generator
```

```
                XORWF           RANDHI,W
                RLCF            WREG
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                RLNCF           WREG
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLCF            RANDLO
                XORWF           RANDLO
                RLCF            RANDHI

                RETLW           0
;       Test suite for FXD3232
TFXD3232        MOVLW           13
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           5
                MOVPF           WREG,TESTCODE
D3232LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB2
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB3
;               BCF             BARGB0,MSB
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
                MOVFP           BARGB2,INDF0
                MOVFP           BARGB3,INDF0
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB2
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB3

;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
;               CALL            FXD3232S
                CALL            FXD3232U
;               CALL            FXD3231U
;               CALL            FXD3131U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                MOVFP           REMB0,INDF0
                MOVFP           REMB1,INDF0
                MOVFP           REMB2,INDF0
                MOVFP           REMB3,INDF0
                DECFSZ          TESTCOUNT
                GOTO            D3232LOOP
                RETLW           0x00
```

```
;       Test suite for FXD3224
TFXD3224        MOVLW           14
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           5
                MOVPF           WREG,TESTCODE
D3224LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB2
;               BCF             BARGB0,MSB
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
                MOVFP           BARGB2,INDF0
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB2
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB3

;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                CALL            FXD3224S
;               CALL            FXD3224U
;               CALL            FXD3223U
;               CALL            FXD3123U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                MOVFP           REMB0,INDF0
                MOVFP           REMB1,INDF0
                MOVFP           REMB2,INDF0
                DECFSZ          TESTCOUNT
                GOTO            D3224LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       32/32 Bit Division Macros
SDIV3232        macro
;       Max Timing:     9+14+30*18+10 = 573 clks
;       Min Timing:     9+14+30*17+3 = 536 clks
;       PM: 9+14+30*24+10 = 753         DM: 12
                variable i
                MOVFP           BARGB3,WREG
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
```

```
                RLCF            ACCB0,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD22#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK22#v(i)
SADD22#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK22#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD228
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK228
SADD228         ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK228          RLCF            ACCB1
```

```
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB1,LSB
                GOTO            SADD22#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK22#v(i)
SADD22#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK22#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB1,LSB
                GOTO            SADD2216
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK2216
SADD2216        ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK2216         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB2,LSB
                GOTO            SADD22#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
```

```
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK22#v(i)
SADD22#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK22#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB2,LSB
                GOTO            SADD2224
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK2224
SADD2224        ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK2224         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB3,LSB
                GOTO            SADD22#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK22#v(i)
SADD22#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK22#v(i)      RLCF            ACCB3
                i=i+1
                endw
```

```
                BTFSC           ACCB3,LSB
                GOTO            SOK22
                MOVFP           BARGB3,WREG
                ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK22
                endm
UDIV3232        macro
;       restore = 25/30 clks,  nonrestore = 17/20 clks
;       Max Timing: 16*25+1+16*30 = 881 clks
;       Min Timing: 16*17+1+16*20 = 593 clks
;       PM:  16*25+1+16*30 = 881                DM: 13
                variable         i
                i = 0
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK22#v(i)
                MOVFP           BARGB3,WREG
                ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK22#v(i)      RLCF            ACCB0
                i=i+1
                endw
                i = 8
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK22#v(i)
                MOVFP           BARGB3,WREG
                ADDWF           REMB3
```

```
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK22#v(i)      RLCF            ACCB1
                i=i+1
                endw
                CLRF            TEMP
                i = 16
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB3,WREG
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                BTFSC           _C
                GOTO            UOK22#v(i)
                MOVFP           BARGB3,WREG
                ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP
                BCF             _C
UOK22#v(i)      RLCF            ACCB2
                i=i+1
                endw
                i = 24
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB3,WREG
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                BTFSC           _C
                GOTO            UOK22#v(i)
                MOVFP           BARGB3,WREG
```

```
                ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP
                BCF             _C
UOK22#v(i)      RLCF            ACCB3
                i=i+1
                endw
                endm
NDIV3232        macro
;       Max Timing:     16+31*21+10 = 677 clks
;       Min Timing: 16+31*20+3 = 639 clks
;       PM: 16+31*29+10 = 925           DM: 13
                variable i
                RLCF            ACCB0,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            TEMP,W
                SUBWFB          TEMP
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB3,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD22#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK22#v(i)
NADD22#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP
```

```
NOK22#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB3,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD228
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK228
NADD228         ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK228          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB3,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD22#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK22#v(i)
NADD22#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK22#v(i)      RLCF            ACCB1
                i=i+1
```

```
                endw
                RLCF            ACCB2,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB3,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD2216
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK2216
NADD2216        ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK2216         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB3,WREG
                BTFSS           ACCB2,LSB
                GOTO            NADD22#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK22#v(i)
NADD22#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK22#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
```

```
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB3,WREG
                BTFSS           ACCB2,LSB
                GOTO            NADD2224
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK2224
NADD2224        ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK2224         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB3,WREG
                BTFSS           ACCB3,LSB
                GOTO            NADD22#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK22#v(i)
NADD22#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK22#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            NOK22
                MOVFP           BARGB3,WREG
```

```
                ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
NOK22
                endm
UDIV3231        macro
;       Max Timing:     14+31*18+10 = 582 clks
;       Min Timing:     14+31*17+3 = 544 clks
;       PM: 14+31*24+10 = 768                   DM: 12
                variable i
                RLCF            ACCB0,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD21#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK21#v(i)
UADD21#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK21#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD218
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
```

```
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK218
UADD218         ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK218          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD21#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK21#v(i)
UADD21#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK21#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD2116
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK2116
UADD2116        ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK2116         RLCF            ACCB2
```

```
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD21#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK21#v(i)
UADD21#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK21#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD2124
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK2124
UADD2124        ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK2124         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB3,LSB
                GOTO            UADD21#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
```

```
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK21#v(i)
UADD21#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK21#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            UOK21
                MOVFP           BARGB3,WREG
                ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK21
                endm
UDIV3131        macro
;       Max Timing:     9+14+30*18+10 = 573 clks
;       Min Timing:     9+14+30*17+3 = 536 clks
;       PM: 9+14+30*24+10 = 753         DM: 12
                variable i
                MOVFP           BARGB3,WREG
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD11#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
```

```
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK11#v(i)
UADD11#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK11#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD118
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK118
UADD118         ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK118          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD11#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK11#v(i)
UADD11#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK11#v(i)      RLCF            ACCB1
```

```
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD1116
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK1116
UADD1116        ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK1116         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD11#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK11#v(i)
UADD11#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK11#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD1124
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
```

```
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK1124
UADD1124        ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK1124         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB3
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB3,WREG
                BTFSS           ACCB3,LSB
                GOTO            UADD11#v(i)
                SUBWF           REMB3
                MOVFP           BARGB2,WREG
                SUBWFB          REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK11#v(i)
UADD11#v(i)     ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK11#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            UOK11
                MOVFP           BARGB3,WREG
                ADDWF           REMB3
                MOVFP           BARGB2,WREG
                ADDWFC          REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK11
                endm
;*************************************************************************************************
;*************************************************************************************************
;       32/24 Bit Division Macros
SDIV3224        macro
;       Max Timing:     7+11+30*15+8 = 476 clks
;       Min Timing:     7+11+30*14+3 = 441 clks
;       PM: 7+11+30*19+8 = 596          DM: 10
                variable i
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
```

```
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD24#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK24#v(i)
SADD24#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK24#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD248
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK248
SADD248         ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK248          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            SADD24#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
```

```
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK24#v(i)
SADD24#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK24#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            SADD2416
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK2416
SADD2416        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK2416         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            SADD24#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK24#v(i)
SADD24#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK24#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            SADD2424
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK2424
```

```
SADD2424        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK2424         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB3,LSB
                GOTO            SADD24#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK24#v(i)
SADD24#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK24#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            SOK24
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK24
                endm
UDIV3224        macro
;       restore = 20/25 clks,  nonrestore = 14/17 clks
;       Max Timing: 16*20+1+16*25 = 721 clks
;       Min Timing: 16*14+1+16*17 = 497 clks
;       PM:  16*20+1+16*25 = 721                DM: 11
                variable         i
                i = 0
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK24#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
```

```
UOK24#v(i)      RLCF            ACCB0
                i=i+1
                endw
                i = 8
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK24#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK24#v(i)      RLCF            ACCB1
                i=i+1
                endw
                CLRF            TEMP
                i = 16
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                BTFSC           _C
                GOTO            UOK24#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP
                BCF             _C
UOK24#v(i)      RLCF            ACCB2
                i=i+1
                endw
                i = 24
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
```

```
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                BTFSC           _C
                GOTO            UOK24#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP
                BCF             _C
UOK24#v(i)      RLCF            ACCB3
                i=i+1
                endw
                endm
NDIV3224        macro
;       Max Timing:     13+31*18+8 = 579 clks
;       Min Timing: 13+31*17+3 = 543 clks
;       PM: 13+31*24+8 = 765            DM: 11
                variable i
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            TEMP,W
                SUBWFB          TEMP
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD24#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK24#v(i)
NADD24#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK24#v(i)      RLCF            ACCB0
```

```
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD248
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK248
NADD248         ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK248          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD24#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK24#v(i)
NADD24#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK24#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD2416
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
```

```
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK2416
NADD2416        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK2416         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            NADD24#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK24#v(i)
NADD24#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK24#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            NADD2424
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK2424
NADD2424        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
```

```
                ADDWFC          TEMP

NOK2424         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB3,LSB
                GOTO            NADD24#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK24#v(i)
NADD24#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK24#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            NOK24
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
NOK24
                endm
UDIV3223        macro
;       Max Timing:     11+31*15+8 = 484 clks
;       Min Timing:     11+31*14+3 = 448 clks
;       PM: 11+31*19+8 = 608                    DM: 10
                variable i
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
```

```
                GOTO            UADD23#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK23#v(i)
UADD23#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK23#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD238
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK238
UADD238         ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK238          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD23#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK23#v(i)
UADD23#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK23#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD2316
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
```

```
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK2316
UADD2316        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK2316         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD23#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK23#v(i)
UADD23#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK23#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD2324
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK2324
UADD2324        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK2324         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB3,LSB
                GOTO            UADD23#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
```

```
                GOTO            UOK23#v(i)
UADD23#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK23#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            UOK23
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK23
                endm
UDIV3123        macro
;       Max Timing:     7+11+30*15+8 = 476 clks
;       Min Timing:     7+11+30*14+3 = 441 clks
;       PM: 7+11+30*19+8 = 596          DM: 10
                variable i
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD13#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK13#v(i)
UADD13#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK13#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
```

```
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD138
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK138
UADD138         ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK138          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD13#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK13#v(i)
UADD13#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK13#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD1316
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK1316
UADD1316        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK1316         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
```

```
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD13#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK13#v(i)
UADD13#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK13#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD1324
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK1324
UADD1324        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK1324         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB3,LSB
                GOTO            UADD13#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK13#v(i)
UADD13#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK13#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            UOK13
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
```

```
                ADDWFC          REMB0
UOK13
                endm
;*****************************************************************************************
;*****************************************************************************************

;       32/32 Bit Signed Fixed Point Divide 32/32 -> 32.32
;       Input:  32 bit signed fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3
;       Use:    CALL    FXD3232S
;       Output: 32 bit signed fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit fixed point remainder in REMB0, REMB1, REMB2, REMB3
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     13+573+3 = 589 clks             A > 0, B > 0
;                       20+573+21 = 614 clks            A > 0, B < 0
;                       20+573+21 = 614 clks            A < 0, B > 0
;                       27+573+3 = 603 clks             A < 0, B < 0
;       Min Timing:     13+536+3 = 552 clks             A > 0, B > 0
;                       20+536+21 = 577 clks            A > 0, B < 0
;                       20+536+21 = 577 clks            A < 0, B > 0
;                       27+536+3 = 566 clks             A < 0, B < 0
;       PM: 27+753+20 = 800             DM: 13
FXD3232S        MOVFP           AARGB0,WREG
                XORWF           BARGB0,W
                MOVWF           SIGN
                CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                CLRF            REMB3,W
                BTFSS           BARGB0,MSB        ; if MSB set, negate BARG
                GOTO            CA3232S
                COMF            BARGB3
                COMF            BARGB2
                COMF            BARGB1
                COMF            BARGB0
                INCF            BARGB3
                ADDWFC          BARGB2
                ADDWFC          BARGB1
                ADDWFC          BARGB0
CA3232S         BTFSS           AARGB0,MSB        ; if MSB set, negate AARG
                GOTO            C3232S
                COMF            AARGB3
                COMF            AARGB2
                COMF            AARGB1
                COMF            AARGB0
                INCF            AARGB3
                ADDWFC          AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
C3232S          SDIV3232
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB3
                COMF            AARGB2
                COMF            AARGB1
                COMF            AARGB0
                CLRF            WREG
                INCF            AARGB3
                ADDWFC          AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
                COMF            REMB3
                COMF            REMB2
                COMF            REMB1
                COMF            REMB0
                INCF            REMB3
```

```
                ADDWFC          REMB2
                ADDWFC          REMB1
                ADDWFC          REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       32/32 Bit Unsigned Fixed Point Divide 32/32 -> 32.32
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               32 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3
;       Use:    CALL    FXD3232U
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1AARGB2,AARGB3
;               32 bit unsigned fixed point remainder in REMB0, REMB1, REMB2, REMB3
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     4+677+2 = 683 clks
;       Min Timing:     4+639+2 = 645 clks
;       PM: 4+925+1 = 930               DM: 13
FXD3232U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                CLRF            REMB3
                NDIV3232
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32/31 Bit Unsigned Fixed Point Divide 32/31 -> 32.31
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3
;       Use:    CALL    FXD3231U
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point remainder in REMB0, REMB1, REMB2, REMB3
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     4+582+2 = 588 clks
;       Min Timing:     4+544+2 = 550 clks
;       PM: 4+768+1 = 773               DM: 12
FXD3231U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                CLRF            REMB3
                UDIV3231
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       31/31 Bit Unsigned Fixed Point Divide 31/31 -> 31.31
;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2, BARGB3
;       Use:    CALL    FXD3131U
;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               31 bit unsigned fixed point remainder in REMB0, REMB1, REMB2, REMB3
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     4+573+2 = 579 clks
;       Min Timing:     4+536+2 = 542 clks
;       PM: 4+753+1 = 758               DM: 12
FXD3131U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                CLRF            REMB3
                UDIV3131
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32/24 Bit Signed Fixed Point Divide 32/24 -> 32.24
;       Input:  32 bit signed fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
```

```
;       Use:    CALL    FXD3224S
;       Output: 32 bit signed fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     12+476+3 = 491 clks             A > 0, B > 0
;                       17+476+19 = 512 clks            A > 0, B < 0
;                       19+476+19 = 514 clks            A < 0, B > 0
;                       24+476+3 = 503 clks             A < 0, B < 0
;       Min Timing:     12+441+3 = 456 clks             A > 0, B > 0
;                       17+441+19 = 477 clks            A > 0, B < 0
;                       19+441+19 = 479 clks            A < 0, B > 0
;                       24+441+3 = 468 clks             A < 0, B < 0
;       PM: 24+596+19 = 639             DM: 11
FXD3224S        MOVFP           AARGB0,WREG
                XORWF           BARGB0,W
                MOVWF           SIGN
                CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2,W
                BTFSS           BARGB0,MSB        ; if MSB set, negate BARG
                GOTO            CA3224S
                COMF            BARGB2
                COMF            BARGB1
                COMF            BARGB0
                INCF            BARGB2
                ADDWFC          BARGB1
                ADDWFC          BARGB0
CA3224S         BTFSS           AARGB0,MSB        ; if MSB set, negate AARG
                GOTO            C3224S
                COMF            AARGB3
                COMF            AARGB2
                COMF            AARGB1
                COMF            AARGB0
                INCF            AARGB3
                ADDWFC          AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
C3224S          SDIV3224
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB3
                COMF            AARGB2
                COMF            AARGB1
                COMF            AARGB0
                CLRF            WREG
                INCF            AARGB3
                ADDWFC          AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
                COMF            REMB2
                COMF            REMB1
                COMF            REMB0
                INCF            REMB2
                ADDWFC          REMB1
                ADDWFC          REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       32/24 Bit Unsigned Fixed Point Divide 32/24 -> 32.24
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD3224U
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1AARGB2,AARGB3
;               24 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     3+579+2 = 584 clks
```

```
;       Min Timing:     3+543+2 = 548 clks
;       PM: 3+765+1 = 769               DM: 11
FXD3224U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                NDIV3224
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32/23 Bit Unsigned Fixed Point Divide 32/23 -> 32.23
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD3223U
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     3+484+2 = 489 clks
;       Min Timing:     3+448+2 = 453 clks
;       PM: 3+608+1 = 612               DM: 10
FXD3223U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                UDIV3223
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       31/23 Bit Unsigned Fixed Point Divide 31/23 -> 31.23
;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD3123U
;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     3+476+2 = 481 clks
;       Min Timing:     3+441+2 = 446 clks
;       PM: 3+596+1 = 600               DM: 10
FXD3123U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                UDIV3123
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

F.2 PIC17CXX Fixed Point Divide Routines B

```
;       PIC17 FIXED POINT DIVIDE ROUTINES B     VERSION 1.8
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXD2416S    314 24 bit/16 bit -> 24.16 signed fixed point divide
;       FXD2416U    365 24 bit/16 bit -> 24.16 unsigned fixed point divide
;       FXD2415U    294 24 bit/15 bit -> 24.15 unsigned fixed point divide
;       FXD2315U    287 23 bit/15 bit -> 23.15 unsigned fixed point divide
;       FXD1616S    214 16 bit/16 bit -> 16.16 signed fixed point divide
;       FXD1616U    244 16 bit/16 bit -> 16.16 unsigned fixed point divide
;       FXD1615U    197 16 bit/15 bit -> 16.15 unsigned fixed point divide
;       FXD1515U    191 15 bit/15 bit -> 15.15 unsigned fixed point divide
;       FXD1608S    146 16 bit/08 bit -> 16.08 signed fixed point divide
```

```
;       FXD1608U    196 16 bit/08 bit -> 16.08 unsigned fixed point divide
;       FXD1607U    130 16 bit/07 bit -> 16.07 unsigned fixed point divide
;       FXD1507U    125 15 bit/07 bit -> 15.07 unsigned fixed point divide
;       FXD0808S     77 08 bit/08 bit -> 08.08 signed fixed point divide
;       FXD0808U     75 08 bit/08 bit -> 08.08 unsigned fixed point divide
;       FXD0807U     66 08 bit/07 bit -> 08.07 unsigned fixed point divide
;       FXD0707U     61 07 bit/07 bit -> 07.07 unsigned fixed point divide
                list    r=dec,x=on,t=off,p=17C42
                include <PIC17.INC>     ; general PIC17 definitions

                include <MATH17.INC>    ; PIC17 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x2B    ; random number senerator registers
RANDLO  equ     0x2C
TESTCODE        equ     0x2D    ; integer code labeling test contained in following data
NUMTESTS        equ     0x2E    ; number of tests contained in following data
TESTCOUNT       equ     0x2F    ; counter
DATA            equ     0x30    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for fixed point divide algorithms
                org             0x0021
MAIN            MOVLW           RAMSTART
                MOVPF           WREG,FSR0
MEMLOOP         CLRF            INDF0
                INCFSZ          FSR0
                GOTO            MEMLOOP
                BSF             RTCSTA,5
;               MOVPF           RTCCH,WREG
                MOVLW           0x45                    ; seed for random numbers
                MOVPF           WREG,RANDLO
;               MOVPF           RTCCL,WREG
                MOVLW           0x30
                MOVPF           WREG,RANDHI
                MOVLW           0x30
                MOVPF           WREG,FSR0
                BCF             _FS1
                BSF             _FS0
;               CALL            TFXD0808
;               CALL            TFXD1608
                CALL            TFXD1616
;               CALL            TFXD2416
                MOVLW           0xFF
                MOVPF           WREG,AARGB0
                MOVLW           0xFF
                MOVPF           WREG,AARGB1
                MOVLW           0xFF
                MOVPF           WREG,AARGB2
                MOVLW           0xFF
                MOVPF           WREG,AARGB3
                MOVLW           0xFF
                MOVPF           WREG,BARGB0
                MOVLW           0xFF
                MOVPF           WREG,BARGB1
                CALL            FXD1616U
SELF            GOTO            SELF
RANDOM16        RLCF            RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                RLCF            WREG
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                RLNCF           WREG
                XORWF           RANDHI,W
                SWAPF           RANDHI
```

```
                ANDLW           0x01
                RLCF            RANDLO
                XORWF           RANDLO
                RLCF            RANDHI

                RETLW           0
;       Test suite for FXD2416
TFXD2416        MOVLW           20
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           1
                MOVPF           WREG,TESTCODE
D2416LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
;               BCF             BARGB0,MSB
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB2
;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                CALL            FXD2416S
;               CALL            FXD2416U
;               CALL            FXD2415U
;               CALL            FXD2315U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           REMB0,INDF0
                MOVFP           REMB1,INDF0
                DECFSZ          TESTCOUNT
                GOTO            D2416LOOP
                RETLW           0x00
;       Test suite for FXD1616
TFXD1616        MOVLW           26
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           2
                MOVPF           WREG,TESTCODE
D1616LOOP
                CALL            RANDOM16
;               SWAPF           RANDHI
;               SWAPF           RANDLO
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
;               BCF             BARGB0,MSB
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
                CALL            RANDOM16

;               SWAPF           RANDHI
;               SWAPF           RANDLO
```

```
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                CALL            FXD1616U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           REMB0,INDF0
                MOVFP           REMB1,INDF0
                DECFSZ          TESTCOUNT
                GOTO            D1616LOOP
                RETLW           0x00
;       Test suite for FXD1608
TFXD1608        MOVLW           34
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           3
                MOVPF           WREG,TESTCODE
D1608LOOP
                CALL            RANDOM16
;               SWAPF           RANDHI
;               SWAPF           RANDLO
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                BCF             BARGB0,MSB
                MOVFP           BARGB0,INDF0
                CALL            RANDOM16

;               SWAPF           RANDHI
;               SWAPF           RANDLO
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                CALL            FXD1608S
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           REMB0,INDF0
                DECFSZ          TESTCOUNT
                GOTO            D1608LOOP
                RETLW           0x00
;       Test suite for FXD0808
TFXD0808        MOVLW           52
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           4
                MOVPF           WREG,TESTCODE
D0808LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
;               BCF             BARGB0,MSB
                MOVFP           BARGB0,INDF0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB0
;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                CALL            FXD0808S
                MOVFP           AARGB0,INDF0
                MOVFP           REMB0,INDF0
```

```
                DECFSZ          TESTCOUNT
                GOTO            D0808LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       24/16 Bit Division Macros
SDIV2416        macro
;       Max Timing:     5+8+22*12+6 = 283 clks
;       Min Timing:     5+8+22*11+3 = 258 clks
;       PM: 5+8+22*14+6 = 327           DM: 8
                variable i
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD46#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK46#v(i)
SADD46#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK46#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD468
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK468
SADD468         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK468          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            SADD46#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
```

```
                SUBWFB          REMB0
                GOTO            SOK46#v(i)
SADD46#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK46#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            SADD4616
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK4616
SADD4616        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK4616         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            SADD46#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK46#v(i)
SADD46#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK46#v(i)      RLCF            ACCB2
                i=i+1
                endw
                BTFSC           ACCB2,LSB
                GOTO            SOK46
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK46
                endm
UDIV2416        macro
;       restore = 15/20 clks,  nonrestore = 11/14 clks
;       Max Timing: 16*15+1+8*20 = 401 clks
;       Min Timing: 16*11+1+8*14 = 289 clks
;       PM:  16*15+1+8*20 = 401         DM: 8
                variable         i
                i = 0
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK46#v(i)
                MOVFP           BARGB1,WREG
```

```
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK46#v(i)      RLCF            ACCB0
                i=i+1
                endw
                i = 8
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK46#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK46#v(i)      RLCF            ACCB1
                i=i+1
                endw
                CLRF            TEMP
                i = 16
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                BTFSC           _C
                GOTO            UOK46#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP
                BCF             _C
UOK46#v(i)      RLCF            ACCB2
                i=i+1
                endw
                endm
NDIV2416        macro
;       Max Timing:     10+23*15+6 = 361 clks
;       Min Timing: 10+23*14+3 = 335 clks
;       PM: 10+23*19+6 = 450            DM: 8
                variable i
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            TEMP,W
                SUBWFB          TEMP
```

```
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD46#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK46#v(i)
NADD46#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK46#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD468
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK468
NADD468         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK468          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD46#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK46#v(i)
NADD46#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP
```

```
NOK46#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD4616
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK4616
NADD4616        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK4616         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            NADD46#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK46#v(i)
NADD46#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK46#v(i)      RLCF            ACCB2
                i=i+1
                endw
                BTFSC           ACCB2,LSB
                GOTO            NOK46
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
NOK46
                endm
UDIV2415        macro
;       Max Timing:     8+23*12+6 = 290 clks
;       Min Timing:     8+23*11+3 = 264 clks
;       PM: 8+23*14+6 = 336                     DM: 8
                variable i
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
```

```
                SUBWFB          REMB0
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD45#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK45#v(i)
UADD45#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK45#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD458
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK458
UADD458         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK458          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD45#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK45#v(i)
UADD45#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK45#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD4516
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK4516
UADD4516        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK4516         RLCF            ACCB2
```

```
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD45#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK45#v(i)
UADD45#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK45#v(i)      RLCF            ACCB2
                i=i+1
                endw
                BTFSC           ACCB2,LSB
                GOTO            UOK45
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK45
                endm
UDIV2315        macro
;       Max Timing:     5+8+22*12+6 = 283 clks
;       Min Timing:     5+8+22*11+3 = 258 clks
;       PM: 5+8+22*14+6 = 327           DM: 8
                variable i
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD35#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK35#v(i)
UADD35#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK35#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
```

```
                BTFSS           ACCB0,LSB
                GOTO            UADD358
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK358
UADD358         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK358          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD35#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK35#v(i)
UADD35#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK35#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD3516
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK3516
UADD3516        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK3516         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD35#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK35#v(i)
UADD35#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK35#v(i)      RLCF            ACCB2
                i=i+1
                endw
                BTFSC           ACCB2,LSB
                GOTO            UOK35
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
```

```
UOK35
                endm
;*****************************************************************************************
;*****************************************************************************************
;       16/16 Bit Division Macros
SDIV1616        macro
;       Max Timing:     5+8+14*12+6 = 187 clks
;       Min Timing:     5+8+14*11+6 = 173 clks
;       PM: 5+8+14*14+6 = 215           DM: 6
                variable i
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD66#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK66#v(i)
SADD66#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0

SOK66#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD668
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK668
SADD668         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0

SOK668          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            SADD66#v(i)
                SUBWF           REMB1
```

```
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK66#v(i)
SADD66#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0

SOK66#v(i)      RLCF            ACCB1
                i=i+1
                endw
                BTFSC           ACCB1,LSB
                GOTO            SOK66
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK66
                endm
UDIV1616        macro
;       restore = 15 clks,  nonrestore = 11 clks
;       Max Timing: 8*15+8*15 = 240 clks
;       Min Timing: 8*11+8*11 = 176 clks
;       PM: 8*15+8*15 = 240             DM: 6
                variable         i
                i = 0
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK66#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK66#v(i)      RLCF            ACCB0
                i=i+1
                endw
                i = 8
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK66#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK66#v(i)      RLCF            ACCB1
                i=i+1
                endw
                endm
NDIV1616        macro
;       Max Timing:     9+15*15+6 = 240 clks
;       Min Timing:     9+15*14+6 = 225 clks
```

```
;       PM: 9+15*19+6 = 300             DM: 7
                variable i
                RLCF            ACCB0,W
                RLCF            REMB1
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            TEMP,W
                SUBWFB          TEMP
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD66#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK66#v(i)
NADD66#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK66#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD668
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK668
NADD668         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK668          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD66#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
```

```
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK66#v(i)
NADD66#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK66#v(i)      RLCF            ACCB1
                i=i+1
                endw
                BTFSC           ACCB1,LSB
                GOTO            NOK66
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
NOK66
                endm
UDIV1615        macro
;       Max Timing:     7+15*12+6 = 193 clks
;       Min Timing:     7+15*11+6 = 178 clks
;       PM: 7+15*14+6 = 213             DM: 6
                variable i
                RLCF            ACCB0,W
                RLCF            REMB1
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD65#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK65#v(i)
UADD65#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0

UOK65#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD658
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK658
UADD658         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
```

```
UOK658          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD65#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK65#v(i)
UADD65#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0

UOK65#v(i)      RLCF            ACCB1
                i=i+1
                endw
                BTFSC           ACCB1,LSB
                GOTO            UOK65
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK65
                endm
UDIV1515        macro
;       Max Timing:     5+8+14*12+6 = 187 clks
;       Min Timing:     5+8+14*11+6 = 173 clks
;       PM: 5+8+14*14+6 = 215           DM: 6
                variable i
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD55#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK55#v(i)
UADD55#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0

UOK55#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
```

```
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD558
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK558
UADD558         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0

UOK558          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD55#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK55#v(i)
UADD55#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0

UOK55#v(i)      RLCF            ACCB1
                i=i+1
                endw
                BTFSC           ACCB1,LSB
                GOTO            UOK55
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK55
                endm
;_____________________________________________________________________________
;       Extra 16 Bit Divide Macros

DIV1616         macro
;       Timing: restore = 16 clks,  nonrestore = 13 clks        16*16 = 256 clks
                variable i
                i = 0
                while i < 16
                RLCF            AARGB1
                RLCF            AARGB0
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSS           _C
                GOTO            RS1616_#v( i )
                BSF             AARGB1,LSB
                GOTO            OK1616_#v( i )
RS1616_#v( i )  MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             AARGB1,LSB
```

```
OK1616_#v(i)
                i=i+1
                endw
                endm
DIVMAC          macro
;       Timing: restore = 19 clks,  nonrestore = 14 clks        16*19 = 304 clks
                variable i
                i = 0
                while i < 16
                RLCF            AARGB1
                RLCF            AARGB0
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                SUBWF           REMB0,W
                BTFSS           _Z
                GOTO            notz#v( i )
                MOVFP           BARGB1,WREG
                SUBWF           REMB1,W
notz#v( i )     BTFSS           _C
                GOTO            nosub#v( i )
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BSF             AARGB1,LSB
                GOTO            ok#v(i)
nosub#v(i)      BCF             AARGB1,LSB
ok#v(i)
                i=i+1
                endw
                endm

;*****************************************************************************************************
;*****************************************************************************************************
;       16/08 BIT Division Macros
SDIV1608        macro
;       Max Timing:     3+5+14*8+2 = 122 clks
;       Min Timing:     3+5+14*8+2 = 122 clks
;       PM: 3+5+14*8+2 = 122            DM: 4
                variable i
                MOVFP           BARGB0,WREG
                SUBWF           REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                ADDWF           REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
```

```
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB1,LSB
                SUBWF           REMB0
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                RLCF            ACCB1
                i=i+1
                endw
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                endm
UDIV1608        macro
;       restore = 9/15 clks,  nonrestore = 8/11 clks
;       Max Timing: 8*9+1+8*15 = 193 clks       max
;       Min Timing: 8*8+1+8*11 = 153 clks       min
;       PM: 8*9+1+8*15 = 193            DM: 4
                variable         i
                i = 0
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                SUBWF           REMB0
                BTFSC           _C
                GOTO            UOK68#v(i)
                ADDWF           REMB0
                BCF             _C
UOK68#v(i)      RLCF            ACCB0
                i=i+1
                endw
                CLRF            TEMP
                i = 8
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB0,WREG
                SUBWF           REMB0
                CLRF            WREG
                SUBWFB          TEMP
                BTFSC           _C
                GOTO            UOK68#v(i)
                MOVFP           BARGB0,WREG
                ADDWF           REMB0
                CLRF            WREG
                ADDWFC          TEMP
                BCF             _C
UOK68#v(i)      RLCF            ACCB1
                i=i+1
                endw
                endm
NDIV1608        macro
;       Max Timing:     7+15*12+3 = 190 clks
;       Min Timing: 7+15*11+3 = 175 clks
;       PM: 7+15*14+3 = 220             DM: 5
                variable i
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
```

```
                SUBWF           REMB0
                CLRF            TEMP,W
                SUBWFB          TEMP
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB0,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD68#v(i)
                SUBWF           REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK68#v(i)
NADD68#v(i)     ADDWF           REMB0
                CLRF            WREG
                ADDWFC          TEMP
NOK68#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB0,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD688
                SUBWF           REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK688
NADD688         ADDWF           REMB0
                CLRF            WREG
                ADDWFC          TEMP
NOK688          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB0,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD68#v(i)
                SUBWF           REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK68#v(i)
NADD68#v(i)     ADDWF           REMB0
                CLRF            WREG
                ADDWFC          TEMP
NOK68#v(i)      RLCF            ACCB1
                i=i+1
                endw
                BTFSS           ACCB1,LSB
                MOVFP           BARGB0,WREG
                ADDWF           REMB0
                endm
UDIV1607        macro
;       Max Timing:     5+15*8+2 = 127 clks
;       Min Timing:     5+15*8+2 = 127 clks
;       PM: 5+15*8+2 = 127              DM: 4
                variable i
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
```

```
                SUBWF           REMB0
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB1,LSB
                SUBWF           REMB0
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                RLCF            ACCB1
                i=i+1
                endw
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                endm
UDIV1507        macro
;       Max Timing:     3+5+14*8+2 = 122 clks
;       Min Timing:     3+5+14*8+2 = 122 clks
;       PM: 3+5+14*8+2 = 122            DM: 4
                variable i
                MOVFP           BARGB0,WREG
                SUBWF           REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                ADDWF           REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB0,LSB
```

```
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB1,LSB
                SUBWF           REMB0
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                RLCF            ACCB1
                i=i+1
                endw
                BTFSS           ACCB1,LSB
                ADDWF           REMB0
                endm

;*****************************************************************************************
;*****************************************************************************************
;       08/08 BIT Division Macros
SDIV0808        macro
;       Max Timing:     3+5+6*8+2 = 58 clks
;       Min Timing:     3+5+6*8+2 = 58 clks
;       PM: 3+5+6*8+2 = 58              DM: 3
                variable i
                MOVFP           BARGB0,WREG
                SUBWF           REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                ADDWF           REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLCF            ACCB0
                i=i+1
                endw
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                endm
UDIV0808        macro
;       restore = 9 clks,  nonrestore = 8 clks
;       Max Timing: 8*9 = 72 clks       max
;       Min Timing: 8*8 = 64 clks       min
;       PM: 8*9 = 72            DM: 3
                variable         i
                i = 0
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                SUBWF           REMB0
                BTFSC           _C
                GOTO            UOK88#v(i)
                ADDWF           REMB0
```

```
                BCF             _C
UOK88#v(i)      RLCF            ACCB0
                i=i+1
                endw
                endm
UDIV0807        macro
;       Max Timing:     5+7*8+2 = 63 clks
;       Min Timing:     5+7*8+2 = 63 clks
;       PM: 5+7*8+2 = 63                DM: 3
                variable i
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                SUBWF           REMB0
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLCF            ACCB0
                i=i+1
                endw
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                endm
UDIV0707        macro
;       Max Timing:     3+5+6*8+2 = 58 clks
;       Min Timing:     3+5+6*8+2 = 58 clks
;       PM: 3+5+6*8+2 = 58              DM: 3
                variable i
                MOVFP           BARGB0,WREG
                SUBWF           REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                ADDWF           REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB0
                MOVFP           BARGB0,WREG
                BTFSC           ACCB0,LSB
                SUBWF           REMB0
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                RLCF            ACCB0
                i=i+1
                endw
                BTFSS           ACCB0,LSB
                ADDWF           REMB0
                endm
;*****************************************************************************************
;*****************************************************************************************

;       24/16 Bit Signed Fixed Point Divide 24/16 -> 24.16
;       Input:  24 bit fixed point dividend in AARGB0, AARGB1, AARGB2
;               16 bit fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD2416S
;       Output: 24 bit fixed point quotient in AARGB0, AARGB1, AARGB2
```

```
;               16 bit fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     11+283+3 = 297 clks             A > 0, B > 0
;                       14+283+15 = 312 clks            A > 0, B < 0
;                       16+283+15 = 314 clks            A < 0, B > 0
;                       19+283+3 = 305 clks             A < 0, B < 0
;       Min Timing:     11+258+3 = 272 clks             A > 0, B > 0
;                       14+258+15 = 287 clks            A > 0, B < 0
;                       16+258+15 = 289 clks            A < 0, B > 0
;                       19+258+3 = 280 clks             A < 0, B < 0
;       PM: 14+327+12 = 353             DM: 8
FXD2416S        MOVFP           AARGB0,WREG
                XORWF           BARGB0,W
                MOVWF           SIGN
                CLRF            REMB0
                CLRF            REMB1,W
                BTFSS           BARGB0,MSB        ; if MSB set go & negate BARG
                GOTO            CA2416S
                COMF            BARGB1
                COMF            BARGB0
                INCF            BARGB1
                ADDWFC          BARGB0
CA2416S         BTFSS           AARGB0,MSB        ; if MSB set go & negate ACCa
                GOTO            C2416S
                COMF            AARGB2
                COMF            AARGB1
                COMF            AARGB0
                INCF            AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
C2416S          SDIV2416
                BTFSS           SIGN,MSB          ; negate (ACCc,ACCd)
                RETLW           0x00
                COMF            AARGB2
                COMF            AARGB1
                COMF            AARGB0
                CLRF            WREG
                INCF            AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
                COMF            REMB1
                COMF            REMB0
                INCF            REMB1
                ADDWFC          REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       24/16 Bit Unsigned Fixed Point Divide 24/16 -> 24.16
;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD2416U
;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               16 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+361+2 = 365 clks
;       Min Timing:     2+335+2 = 339 clks
;       PM: 2+450+1 = 453               DM: 8
FXD2416U        CLRF            REMB0
                CLRF            REMB1
                NDIV2416
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       24/15 Bit Unsigned Fixed Point Divide 24/15 -> 24.15
```

```
;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD2415U
;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+290+2 = 294 clks
;       Min Timing:     2+264+2 = 268 clks
;       PM: 2+336+1 = 339               DM: 8
FXD2415U        CLRF            REMB0
                CLRF            REMB1
                UDIV2415
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       23/15 Bit Unsigned Fixed Point Divide 23/15 -> 23.15
;       Input:  23 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD2315U
;       Output: 23 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+283+2 = 287 clks
;       Min Timing:     2+258+2 = 262 clks
;       PM: 2+327+1 = 330               DM: 8
FXD2315U        CLRF            REMB0
                CLRF            REMB1
                UDIV2315
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16/16 Bit Signed Fixed Point Divide 16/16 -> 16.16
;       Input:  16 bit fixed point dividend in AARGB0, AARGB1
;               16 bit fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD1616S
;       Output: 16 bit fixed point quotient in AARGB0, AARGB1
;               16 bit fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     11+187+3 = 201 clks             A > 0, B > 0
;                       14+187+13 = 214 clks            A > 0, B < 0
;                       14+187+13 = 214 clks            A < 0, B > 0
;                       17+187+3 = 207 clks             A < 0, B < 0
;       Min Timing:     11+173+3 = 187 clks             A > 0, B > 0
;                       14+173+13 = 200 clks            A > 0, B < 0
;                       14+173+13 = 200 clks            A < 0, B > 0
;                       17+173+3 = 193 clks             A < 0, B < 0
;       PM: 14+215+12 = 241             DM: 7
FXD1616S        MOVFP           AARGB0,WREG
                XORWF           BARGB0,W
                MOVWF           SIGN
                CLRF            REMB0
                CLRF            REMB1,W
                BTFSS           BARGB0,MSB        ; if MSB set go & negate BARG
                GOTO            CA1616S
                COMF            BARGB1
                COMF            BARGB0
                INCF            BARGB1
                ADDWFC          BARGB0
CA1616S         BTFSS           AARGB0,MSB        ; if MSB set go & negate ACCa
                GOTO            C1616S
                COMF            AARGB1
                COMF            AARGB0
                INCF            AARGB1
                ADDWFC          AARGB0
```

```
C1616S          SDIV1616
                BTFSS           SIGN,MSB        ; negate (ACCc,ACCd)
                RETLW           0x00
                COMF            AARGB1
                COMF            AARGB0
                CLRF            WREG
                INCF            AARGB1
                ADDWFC          AARGB0
                COMF            REMB1
                COMF            REMB0
                INCF            REMB1
                ADDWFC          REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16/16 Bit Unsigned Fixed Point Divide 16/16 -> 16.16
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD1616U
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               16 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+240+2 = 244 clks
;       Min Timing:     2+176+2 = 180 clks
;       PM: 2+240+1 = 243               DM: 6
FXD1616U        CLRF            REMB0
                CLRF            REMB1
                UDIV1616
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16/15 Bit Unsigned Fixed Point Divide 16/15 -> 16.15
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD1615U
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+193+2 = 197 clks
;       Min Timing:     2+178+2 = 182 clks
;       PM: 2+213+1 = 216               DM: 6
FXD1615U        CLRF            REMB0
                CLRF            REMB1
                UDIV1615
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       15/15 Bit Unsigned Fixed Point Divide 15/15 -> 15.15
;       Input:  15 bit unsigned fixed point dividend in AARGB0, AARGB1
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD1515U
;       Output: 15 bit unsigned fixed point quotient in AARGB0, AARGB1
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+187+2 = 191 clks
;       Min Timing:     2+173+2 = 177 clks
;       PM: 2+215+1 = 218               DM: 6
FXD1515U        CLRF            REMB0
                CLRF            REMB1
                UDIV1515
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
```

```
;       16/8 Bit Signed Fixed Point Divide 16/08 -> 16.08
;       Input:  16 bit fixed point dividend in AARGB0, AARGB1
;               8 bit fixed point divisor in BARGB0
;       Use:    CALL    FXD1608S
;       Output: 16 bit fixed point quotient in AARGB0, AARGB1
;               8 bit fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     10+122+3 = 135 clks             A > 0, B > 0
;                       11+122+11 = 144 clks            A > 0, B < 0
;                       13+122+11 = 146 clks            A < 0, B > 0
;                       14+122+3 = 139 clks             A < 0, B < 0
;       Min Timing:     10+122+3 = 135 clks             A > 0, B > 0
;                       11+122+11 = 144 clks            A > 0, B < 0
;                       13+122+11 = 146 clks            A < 0, B > 0
;                       14+122+3 = 139 clks             A < 0, B < 0
;       PM: 14+122+10 = 146             DM: 5
FXD1608S        MOVFP           AARGB0,WREG
                XORWF           BARGB0,W
                MOVWF           SIGN
                CLRF            REMB0,W
                BTFSS           BARGB0,MSB              ; if MSB set go & negate BARG
                GOTO            CA1608S
                COMF            BARGB0
                INCF            BARGB0
CA1608S         BTFSS           AARGB0,MSB              ; if MSB set go & negate ACCa
                GOTO            C1608S
                COMF            AARGB1
                COMF            AARGB0
                INCF            AARGB1
                ADDWFC          AARGB0
C1608S          SDIV1608
                BTFSS           SIGN,MSB                ; negate (ACCc,ACCd)
                RETLW           0x00
                COMF            AARGB1
                COMF            AARGB0
                CLRF            WREG
                INCF            AARGB1
                ADDWFC          AARGB0
                COMF            REMB0
                INCF            REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16/8 Bit Unsigned Fixed Point Divide 16/08 -> 16.08
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;                8 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD1608U
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
;               8 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+193+2 = 196 clks
;       Min Timing:     1+153+2 = 156 clks
;       PM: 1+193+1 = 195               DM: 4
FXD1608U        CLRF            REMB0
                UDIV1608
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       16/7 Bit Unsigned Fixed Point Divide 16/07 -> 16.07
;       Input:  16 bit unsigned fixed point dividend in AARGB0, AARGB1
;                7 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD1607U
;       Output: 16 bit unsigned fixed point quotient in AARGB0, AARGB1
```

```
;               7 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+127+2 = 130 clks
;       Min Timing:     1+127+2 = 130 clks
;       PM: 1+127+1 = 129               DM: 4
FXD1607U        CLRF            REMB0
                UDIV1607
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;
;       15/7 Bit Unsigned Fixed Point Divide 15/07 -> 15.07
;       Input:  15 bit unsigned fixed point dividend in AARGB0, AARGB1
;                7 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD1507U
;       Output: 15 bit unsigned fixed point quotient in AARGB0, AARGB1
;               7 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+122+2 = 125 clks
;       Min Timing:     1+122+2 = 125 clks
;       PM: 1+122+1 = 124               DM: 4
FXD1507U        CLRF            REMB0
                UDIV1507
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       8/8 Bit Signed Fixed Point Divide 08/08 -> 08.08
;       Input:  8 bit fixed point dividend in AARGB0
;               8 bit fixed point divisor in BARGB0
;       Use:    CALL    FXD0808S
;       Output: 8 bit fixed point quotient in AARGB0
;               8 bit fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     10+58+3 = 71 clks               A > 0, B > 0
;                       11+58+8 = 77 clks               A > 0, B < 0
;                       11+58+8 = 77 clks               A < 0, B > 0
;                       12+58+3 = 73 clks               A < 0, B < 0
;       Min Timing:     10+58+3 = 71 clks               A > 0, B > 0
;                       11+58+8 = 77 clks               A > 0, B < 0
;                       11+58+8 = 77 clks               A < 0, B > 0
;                       12+58+3 = 71 clks               A < 0, B < 0
;       PM: 12+58+7 = 77                DM: 4
FXD0808S        MOVFP           AARGB0,WREG
                XORWF           BARGB0,W
                MOVWF           SIGN
                CLRF            REMB0,W
                BTFSS           BARGB0,MSB
                GOTO            CA0808S
                COMF            BARGB0
                INCF            BARGB0
CA0808S         BTFSS           AARGB0,MSB
                GOTO            C0808S
                COMF            AARGB0
                INCF            AARGB0
C0808S          SDIV0808
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB0
                INCF            AARGB0
                COMF            REMB0
                INCF            REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       8/8 Bit Unsigned Fixed Point Divide 08/08 -> 08.08
```

```
;       Input:  8 bit unsigned fixed point dividend in AARGB0
;               8 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD0808U
;       Output: 8 bit unsigned fixed point quotient in AARGB0
;               8 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+72+2 = 75 clks
;       Min Timing:     1+64+2 = 67 clks
;       PM: 1+72+1 = 74         DM: 3
FXD0808U        CLRF            REMB0
                UDIV0808
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       8/7 Bit Unsigned Fixed Point Divide 08/07 -> 08.07
;       Input:  8 bit unsigned fixed point dividend in AARGB0
;               7 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD0807U
;       Output: 8 bit unsigned fixed point quotient in AARGB0
;               7 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+63+2 = 66 clks
;       Min Timing:     1+63+2 = 66 clks
;       PM: 1+63+1 = 65         DM: 3
FXD0807U        CLRF            REMB0
                UDIV0807
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       7/7 Bit Unsigned Fixed Point Divide 07/07 -> 07.07
;       Input:  7 bit unsigned fixed point dividend in AARGB0
;               7 bit unsigned fixed point divisor in BARGB0
;       Use:    CALL    FXD0707U
;       Output: 7 bit unsigned fixed point quotient in AARGB0
;               7 bit unsigned fixed point remainder in REMB0
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     1+58+2 = 61 clks
;       Min Timing:     1+58+2 = 61 clks
;       PM: 1+58+1 = 60         DM: 3
FXD0707U        CLRF            REMB0
                UDIV0707
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

F.3 PIC17CXX Fixed Point Divide Routines C

```
;       PIC17 FIXED POINT DIVIDE ROUTINES C     VERSION 1.8
;       Input:  fixed point arguments in AARG and BARG
;       Output: quotient AARG/BARG followed by remainder in REM
;       All timings are worst case cycle counts
;       It is useful to note that the additional unsigned routines requiring a non-power of two
;       argument can be called in a signed divide application where it is known that the
;       respective argument is nonnegative, thereby offering some improvement in
;       performance.
;         Routine            Clocks     Function
;       FXD3216S    414 32 bit/16 bit -> 32.16 signed fixed point divide
;       FXD3216U    485 32 bit/16 bit -> 32.16 unsigned fixed point divide
;       FXD3215U    390 32 bit/15 bit -> 32.15 unsigned fixed point divide
;       FXD3115U    383 31 bit/15 bit -> 31.15 unsigned fixed point divide
;       FXD2424S    390 24 bit/24 bit -> 24.24 signed fixed point divide
;       FXD2424U    440 24 bit/24 bit -> 24.24 unsigned fixed point divide
;       FXD2423U    369 24 bit/23 bit -> 24.23 unsigned fixed point divide
;       FXD2323U    361 23 bit/23 bit -> 23.23 unsigned fixed point divide
                list    r=dec,x=on,t=off,p=17C42
```

```
                include <PIC17.INC>     ; general PIC17 definitions

                include <MATH17.INC>    ; PIC17 math library definitions
;*****************************************************************************************
;*****************************************************************************************
;       Test suite storage
RANDHI          equ     0x2B    ; random number senerator registers
RANDLO  equ     0x2C
TESTCODE        equ     0x2D    ; integer code labeling test contained in following data
NUMTESTS        equ     0x2E    ; number of tests contained in following data
TESTCOUNT       equ     0x2F    ; counter
DATA            equ     0x30    ; beginning of test data
;*****************************************************************************************
;*****************************************************************************************
;       Test suite for fixed point divide algorithms
                org             0x0021
MAIN            MOVLW           RAMSTART
                MOVPF           WREG,FSR0
MEMLOOP         CLRF            INDF0
                INCFSZ          FSR0
                GOTO            MEMLOOP
                BSF             RTCSTA,5
;               MOVPF           RTCCH,WREG
                MOVLW           0x45                    ; seed for random numbers
                MOVPF           WREG,RANDLO
;               MOVPF           RTCCL,WREG
                MOVLW           0x30
                MOVPF           WREG,RANDHI
                MOVLW           0x30
                MOVPF           WREG,FSR0
                BCF             _FS1
                BSF             _FS0
;               CALL            TFXD3216
                CALL            TFXD2424
SELF            GOTO            SELF
RANDOM16        RLCF            RANDHI,W                ; random number generator
                XORWF           RANDHI,W
                RLCF            WREG
                SWAPF           RANDHI
                SWAPF           RANDLO,W
                RLNCF           WREG
                XORWF           RANDHI,W
                SWAPF           RANDHI
                ANDLW           0x01
                RLCF            RANDLO
                XORWF           RANDLO
                RLCF            RANDHI

                RETLW           0
;       Test suite for FXD3216
TFXD3216        MOVLW           17
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           1
                MOVPF           WREG,TESTCODE
D3216LOOP
                CALL            RANDOM16
;               SWAPF           RANDHI
;               SWAPF           RANDLO
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
;               BCF             BARGB0,MSB
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
```

```
                CALL            RANDOM16

;               SWAPF           RANDHI
;               SWAPF           RANDLO
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
;               BCF             AARGB0,MSB
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB2
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB3

                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                CALL            FXD3216U
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           AARGB3,INDF0
                MOVFP           REMB0,INDF0
                MOVFP           REMB1,INDF0
                DECFSZ          TESTCOUNT
                GOTO            D3216LOOP
                RETLW           0x00
;       Test suite for FXD2424
TFXD2424        MOVLW           17
                MOVPF           WREG,TESTCOUNT
                MOVPF           WREG,NUMTESTS
                MOVLW           6
                MOVPF           WREG,TESTCODE
D2424LOOP
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,BARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,BARGB2
;               BCF             BARGB0,MSB
                MOVFP           BARGB0,INDF0
                MOVFP           BARGB1,INDF0
                MOVFP           BARGB2,INDF0
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB0
                MOVFP           RANDLO,WREG
                MOVPF           WREG,AARGB1
                CALL            RANDOM16
                MOVFP           RANDHI,WREG
                MOVPF           WREG,AARGB2

;               BCF             AARGB0,MSB
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                CALL            FXD2424S
                MOVFP           AARGB0,INDF0
                MOVFP           AARGB1,INDF0
                MOVFP           AARGB2,INDF0
                MOVFP           REMB0,INDF0
```

```
                MOVFP           REMB1,INDF0
                MOVFP           REMB2,INDF0
                DECFSZ          TESTCOUNT
                GOTO            D2424LOOP
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       32/16 Bit Division Macros
SDIV3216        macro
;       Max Timing:     5+8+30*12+6 = 379 clks
;       Min Timing:     5+8+30*11+6 = 349 clks
;       PM: 5+8+30*14+6 = 439           DM: 8
                variable i
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD26#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK26#v(i)
SADD26#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK26#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD268
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK268
SADD268         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK268          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            SADD26#v(i)
```

```
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK26#v(i)
SADD26#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK26#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            SADD2616
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK2616
SADD2616        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK2616         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            SADD26#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK26#v(i)
SADD26#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK26#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            SADD2624
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK2624
SADD2624        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK2624         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB3,LSB
                GOTO            SADD26#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
```

```
                SUBWFB          REMB0
                GOTO            SOK26#v(i)
SADD26#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK26#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            SOK26
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK26
                endm
UDIV3216        macro
;       restore = 15/20 clks,  nonrestore = 11/14 clks
;       Max Timing: 16*15+1+16*20 = 561 clks
;       Min Timing: 16*11+1+16*14 = 401 clks
;       PM:  16*15+1+16*20 = 561                DM: 9
                variable         i
                i = 0
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK26#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK26#v(i)      RLCF            ACCB0
                i=i+1
                endw
                i = 8
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK26#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK26#v(i)      RLCF            ACCB1
                i=i+1
                endw
                CLRF            TEMP
                i = 16
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
```

```
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                BTFSC           _C
                GOTO            UOK26#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP
                BCF             _C
UOK26#v(i)      RLCF            ACCB2
                i=i+1
                endw
                i = 24
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                BTFSC           _C
                GOTO            UOK26#v(i)
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP
                BCF             _C
UOK26#v(i)      RLCF            ACCB3
                i=i+1
                endw
                endm
NDIV3216        macro
;       Max Timing:     10+31*15+6 = 481 clks
;       Min Timing: 10+31*14+6 = 450 clks
;       PM: 10+31*19+6 = 605            DM: 9
                variable i
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            TEMP,W
                SUBWFB          TEMP
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
```

```
                BTFSS           ACCB0,LSB
                GOTO            NADD26#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK26#v(i)
NADD26#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK26#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD268
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK268
NADD268         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK268          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD26#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK26#v(i)
NADD26#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK26#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
```

```
                BTFSS           ACCB1,LSB
                GOTO            NADD2616
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK2616
NADD2616        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK2616         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            NADD26#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK26#v(i)
NADD26#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK26#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            NADD2624
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK2624
NADD2624        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK2624         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB1,WREG
```

```
                BTFSS           ACCB3,LSB
                GOTO            NADD26#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK26#v(i)
NADD26#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK26#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            NOK26
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
NOK26
                endm
UDIV3215        macro
;       Max Timing:     8+31*12+6 = 386 clks
;       Min Timing:     8+31*11+6 = 355 clks
;       PM: 8+31*14+6 = 448             DM: 8
                variable i
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD25#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK25#v(i)
UADD25#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK25#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD258
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK258
```

```
UADD258         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK258          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD25#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK25#v(i)
UADD25#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK25#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD2516
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK2516
UADD2516        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK2516         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD25#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK25#v(i)
UADD25#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK25#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD2524
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK2524
UADD2524        ADDWF           REMB1
                MOVFP           BARGB0,WREG
```

```
                ADDWFC          REMB0
UOK2524         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB3,LSB
                GOTO            UADD25#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK25#v(i)
UADD25#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK25#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            UOK25
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK25
                endm
UDIV3115        macro
;       Max Timing:     5+8+30*12+6 = 379 clks
;       Min Timing:     5+8+30*11+6 = 349 clks
;       PM: 5+8+30*14+6 = 439           DM: 8
                variable i
                MOVFP           BARGB1,WREG
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD15#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK15#v(i)
UADD15#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK15#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB1
```

```
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD158
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK158
UADD158         ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK158          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD15#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK15#v(i)
UADD15#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK15#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD1516
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK1516
UADD1516        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK1516         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD15#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK15#v(i)
UADD15#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK15#v(i)      RLCF            ACCB2
                i=i+1
                endw
                RLCF            ACCB3,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
```

```
                BTFSS           ACCB2,LSB
                GOTO            UADD1524
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK1524
UADD1524        ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK1524         RLCF            ACCB3
                i = 25
                while i < 32
                RLCF            ACCB3,W
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB1,WREG
                BTFSS           ACCB3,LSB
                GOTO            UADD15#v(i)
                SUBWF           REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK15#v(i)
UADD15#v(i)     ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK15#v(i)      RLCF            ACCB3
                i=i+1
                endw
                BTFSC           ACCB3,LSB
                GOTO            UOK15
                MOVFP           BARGB1,WREG
                ADDWF           REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK15
                endm
;*****************************************************************************************
;*****************************************************************************************
;       24/24 Bit Division Macros
SDIV2424        macro
;       Max Timing:     7+11+22*15+8 = 356 clks
;       Min Timing:     7+11+22*14+3 = 329 clks
;       PM: 7+11+22*19+8 = 444          DM: 9
                variable i
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB2
```

```
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD44#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK44#v(i)
SADD44#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK44#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            SADD448
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK448
SADD448         ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK448          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            SADD44#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK44#v(i)
SADD44#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK44#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
```

```
                BTFSS           ACCB1,LSB
                GOTO            SADD4416
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK4416
SADD4416        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK4416         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            SADD44#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            SOK44#v(i)
SADD44#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK44#v(i)      RLCF            ACCB2
                i=i+1
                endw
                BTFSC           ACCB2,LSB
                GOTO            SOK44
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
SOK44
                endm
UDIV2424        macro
;       restore = 20/25 clks,  nonrestore = 14/17 clks
;       Max Timing: 16*20+1+8*25 = 521 clks
;       Min Timing: 16*14+1+8*17 = 361 clks
;       PM:  16*20+1+8*25 = 521         DM: 10
                variable         i
                i = 0
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
```

```
                GOTO            UOK44#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK44#v(i)      RLCF            ACCB0
                i=i+1
                endw
                i = 8
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                BTFSC           _C
                GOTO            UOK44#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                BCF             _C
UOK44#v(i)      RLCF            ACCB1
                i=i+1
                endw
                CLRF            TEMP
                i = 16
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                BTFSC           _C
                GOTO            UOK44#v(i)
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP
                BCF             _C
UOK44#v(i)      RLCF            ACCB2
                i=i+1
                endw
                endm
```

```
NDIV2424        macro
;       Max Timing:     13+23*18+8 = 435 clks
;       Min Timing: 13+23*17+3 = 407 clks
;       PM: 13+23*24+8 = 573            DM: 10
                variable i
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            TEMP,W
                SUBWFB          TEMP
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD44#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK44#v(i)
NADD44#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK44#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            NADD448
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK448
NADD448         ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
```

```
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK448          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD44#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK44#v(i)
NADD44#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK44#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            NADD4416
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK4416
NADD4416        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK4416         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
```

```
                RLCF            TEMP
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            NADD44#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                CLRF            WREG
                SUBWFB          TEMP
                GOTO            NOK44#v(i)
NADD44#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                CLRF            WREG
                ADDWFC          TEMP

NOK44#v(i)      RLCF            ACCB2
                i=i+1
                endw
                BTFSC           ACCB2,LSB
                GOTO            NOK44
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
NOK44
                endm
UDIV2423        macro
;       Max Timing:     11+23*15+8 = 364 clks
;       Min Timing:     11+23*14+3 = 336 clks
;       PM: 11+23*19+8 = 456                    DM: 9
                variable i
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                i = 1
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD43#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK43#v(i)
UADD43#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
```

```
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK43#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD438
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK438
UADD438         ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK438          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD43#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK43#v(i)
UADD43#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK43#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD4316
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK4316
UADD4316        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
```

```
UOK4316         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD43#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK43#v(i)
UADD43#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK43#v(i)      RLCF            ACCB2
                i=i+1
                endw
                BTFSC           ACCB2,LSB
                GOTO            UOK43
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK43
                endm
UDIV2323        macro
;       Max Timing:     7+11+22*15+8 = 356 clks
;       Min Timing:     7+11+22*14+3 = 329 clks
;       PM: 7+11+22*19+8 = 444          DM: 9
                variable i
                MOVFP           BARGB2,WREG
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                RLCF            ACCB0
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
                RLCF            ACCB0
                i = 2
                while i < 8
                RLCF            ACCB0,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD33#v(i)
```

```
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK33#v(i)
UADD33#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK33#v(i)      RLCF            ACCB0
                i=i+1
                endw
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB0,LSB
                GOTO            UADD338
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK338
UADD338         ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK338          RLCF            ACCB1
                i = 9
                while i < 16
                RLCF            ACCB1,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD33#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK33#v(i)
UADD33#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK33#v(i)      RLCF            ACCB1
                i=i+1
                endw
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB1,LSB
                GOTO            UADD3316
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
```

```
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK3316
UADD3316        ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK3316         RLCF            ACCB2
                i = 17
                while i < 24
                RLCF            ACCB2,W
                RLCF            REMB2
                RLCF            REMB1
                RLCF            REMB0
                MOVFP           BARGB2,WREG
                BTFSS           ACCB2,LSB
                GOTO            UADD33#v(i)
                SUBWF           REMB2
                MOVFP           BARGB1,WREG
                SUBWFB          REMB1
                MOVFP           BARGB0,WREG
                SUBWFB          REMB0
                GOTO            UOK33#v(i)
UADD33#v(i)     ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK33#v(i)      RLCF            ACCB2
                i=i+1
                endw
                BTFSC           ACCB2,LSB
                GOTO            UOK33
                MOVFP           BARGB2,WREG
                ADDWF           REMB2
                MOVFP           BARGB1,WREG
                ADDWFC          REMB1
                MOVFP           BARGB0,WREG
                ADDWFC          REMB0
UOK33
                endm
;*****************************************************************************************
;*****************************************************************************************

;       32/16 Bit Signed Fixed Point Divide 32/16 -> 32.16
;       Input:  32 bit signed fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD3216S
;       Output: 32 bit signed fixed point quotient in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     11+379+3 = 393 clks             A > 0, B > 0
;                       14+379+17 = 410 clks            A > 0, B < 0
;                       18+379+17 = 414 clks            A < 0, B > 0
;                       21+379+3 = 403 clks             A < 0, B < 0
;       Min Timing:     11+349+3 = 363 clks             A > 0, B > 0
;                       14+349+17 = 380 clks            A > 0, B < 0
;                       18+349+17 = 384 clks            A < 0, B > 0
;                       21+349+3 = 373 clks             A < 0, B < 0
;       PM: 21+439+16 = 476             DM: 9
FXD3216S        MOVFP           AARGB0,WREG
                XORWF           BARGB0,W
                MOVWF           SIGN
                CLRF            REMB0
                CLRF            REMB1,W
```

```
                BTFSS           BARGB0,MSB        ; if MSB set go & negate BARG
                GOTO            CA3216S
                COMF            BARGB1
                COMF            BARGB0
                INCF            BARGB1
                ADDWFC          BARGB0
CA3216S         BTFSS           AARGB0,MSB        ; if MSB set go & negate ACCa
                GOTO            C3216S
                COMF            AARGB3
                COMF            AARGB2
                COMF            AARGB1
                COMF            AARGB0
                INCF            AARGB3
                ADDWFC          AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
C3216S          SDIV3216
                BTFSS           SIGN,MSB          ; negate (ACCc,ACCd)
                RETLW           0x00
                COMF            AARGB3
                COMF            AARGB2
                COMF            AARGB1
                COMF            AARGB0
                CLRF            WREG
                INCF            AARGB3
                ADDWFC          AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
                COMF            REMB1
                COMF            REMB0
                INCF            REMB1
                ADDWFC          REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       32/16 Bit Unsigned Fixed Point Divide 32/16 -> 32.16
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               16 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD3216U
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1AARGB2,AARGB3
;               16 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+481+2 = 485 clks
;       Min Timing:     2+450+2 = 459 clks
;       PM: 2+605+1 = 608               DM: 9
FXD3216U        CLRF            REMB0
                CLRF            REMB1
                NDIV3216
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       32/15 Bit Unsigned Fixed Point Divide 32/15 -> 32.15
;       Input:  32 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD3215U
;       Output: 32 bit unsigned fixed point quotient in AARGB0, AARGB1
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+386+2 = 390 clks
;       Min Timing:     2+355+2 = 359 clks
;       PM: 2+448+1 = 451               DM: 8
FXD3215U        CLRF            REMB0
                CLRF            REMB1
                UDIV3215
                RETLW           0x00
```

```
;*****************************************************************************************
;*****************************************************************************************

;       31/15 Bit Unsigned Fixed Point Divide 31/15 -> 31.15
;       Input:  31 bit unsigned fixed point dividend in AARGB0, AARGB1,AARGB2,AARGB3
;               15 bit unsigned fixed point divisor in BARGB0, BARGB1
;       Use:    CALL    FXD3115U
;       Output: 31 bit unsigned fixed point quotient in AARGB0, AARGB1
;               15 bit unsigned fixed point remainder in REMB0, REMB1
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     2+379+2 = 383 clks
;       Min Timing:     2+349+2 = 353 clks
;       PM: 2+439+1 = 442               DM: 8
FXD3115U        CLRF            REMB0
                CLRF            REMB1
                UDIV3115
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       24/24 Bit Signed Fixed Point Divide 24/24 -> 24.24
;       Input:  24 bit signed fixed point dividend in AARGB0, AARGB1, AARGB2
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD2424S
;       Output: 24 bit signed fixed point quotient in AARGB0, AARGB1, AARGB2
;               24 bit fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     12+356+3 = 371 clks             A > 0, B > 0
;                       17+356+17 = 390 clks            A > 0, B < 0
;                       17+356+17 = 390 clks            A < 0, B > 0
;                       22+356+3 = 381 clks             A < 0, B < 0
;       Min Timing:     12+329+3 = 344 clks             A > 0, B > 0
;                       17+329+17 = 363 clks            A > 0, B < 0
;                       17+329+17 = 363 clks            A < 0, B > 0
;                       22+329+3 = 354 clks             A < 0, B < 0
;       PM: 22+444+16 = 482             DM: 10
FXD2424S        MOVFP           AARGB0,WREG
                XORWF           BARGB0,W
                MOVWF           SIGN
                CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2,W
                BTFSS           BARGB0,MSB      ; if MSB set, negate BARG
                GOTO            CA2424S
                COMF            BARGB2
                COMF            BARGB1
                COMF            BARGB0
                INCF            BARGB2
                ADDWFC          BARGB1
                ADDWFC          BARGB0
CA2424S         BTFSS           AARGB0,MSB      ; if MSB set, negate AARG
                GOTO            C2424S
                COMF            AARGB2
                COMF            AARGB1
                COMF            AARGB0
                INCF            AARGB2
                ADDWFC          AARGB1
                ADDWFC          AARGB0
C2424S          SDIV2424
                BTFSS           SIGN,MSB
                RETLW           0x00
                COMF            AARGB2
                COMF            AARGB1
                COMF            AARGB0
                CLRF            WREG
                INCF            AARGB2
```

```
                ADDWFC          AARGB1
                ADDWFC          AARGB0
                COMF            REMB2
                COMF            REMB1
                COMF            REMB0
                INCF            REMB2
                ADDWFC          REMB1
                ADDWFC          REMB0
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
;       24/24 Bit Unsigned Fixed Point Divide 24/24 -> 24.24
;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               24 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD2424U
;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               24 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     3+435+2 = 440 clks
;       Min Timing:     3+407+2 = 412 clks
;       PM: 3+573+1 = 577               DM: 10
FXD2424U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                NDIV2424
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       24/23 Bit Unsigned Fixed Point Divide 24/23 -> 24.23
;       Input:  24 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD2423U
;       Output: 24 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     3+364+2 = 369 clks
;       Min Timing:     3+336+2 = 341 clks
;       PM: 3+456+1 = 460               DM: 9
FXD2423U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                UDIV2423
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************

;       23/23 Bit Unsigned Fixed Point Divide 23/23 -> 23.23
;       Input:  23 bit unsigned fixed point dividend in AARGB0, AARGB1, AARGB2
;               23 bit unsigned fixed point divisor in BARGB0, BARGB1, BARGB2
;       Use:    CALL    FXD2323U
;       Output: 23 bit unsigned fixed point quotient in AARGB0, AARGB1, AARGB2
;               23 bit unsigned fixed point remainder in REMB0, REMB1, REMB2
;       Result: AARG, REM  <--  AARG / BARG
;       Max Timing:     3+356+2 = 361 clks
;       Min Timing:     3+329+2 = 334 clks
;       PM: 3+444+1 = 448               DM: 9
FXD2323U        CLRF            REMB0
                CLRF            REMB1
                CLRF            REMB2
                UDIV2323
                RETLW           0x00
;*****************************************************************************************
;*****************************************************************************************
                END
```

NOTES:

WORLDWIDE SALES & SERVICE

AMERICAS

Corporate Office
Microchip Technology Inc.
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 602 786-7200 Fax: 602 786-7277
Technical Support: 602 786-7627
Web: http://www.mchip.com/microchip

Atlanta
Microchip Technology Inc.
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770 640-0034 Fax: 770 640-0307

Boston
Microchip Technology Inc.
5 Mount Royal Avenue
Marlborough, MA 01752
Tel: 508 480-9990 Fax: 508 480-8575

Chicago
Microchip Technology Inc.
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 708 285-0071 Fax: 708 285-0075

Dallas
Microchip Technology Inc.
14651 Dallas Parkway, Suite 816
Dallas, TX 75240-8809
Tel: 214 991-7177 Fax: 214 991-8588

Dayton
Microchip Technology Inc.
35 Rockridge Road
Englewood, OH 45322
Tel: 513 832-2543 Fax: 513 832-2841

Los Angeles
Microchip Technology Inc.
18201 Von Karman, Suite 455
Irvine, CA 92715
Tel: 714 263-1888 Fax: 714 263-1338

New York
Microchip Technology Inc.
150 Motor Parkway, Suite 416
Hauppauge, NY 11788
Tel: 516 273-5305 Fax: 516 273-5335

San Jose
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408 436-7950 Fax: 408 436-7955

ASIA/PACIFIC

Hong Kong
Microchip Technology
Unit No. 3002-3004, Tower 1
Metroplaza
223 Hing Fong Road
Kwai Fong, N.T. Hong Kong
Tel: 852 2 401 1200 Fax: 852 2 401 3431

Korea
Microchip Technology
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku,
Seoul, Korea
Tel: 82 2 554 7200 Fax: 82 2 558 5934

Singapore
Microchip Technology
200 Middle Road
#10-03 Prime Centre
Singapore 188980
Tel: 65 334 8870 Fax: 65 334 8850

Taiwan
Microchip Technology
10F-1C 207
Tung Hua North Road
Taipei, Taiwan, ROC
Tel: 886 2 717 7175 Fax: 886 2 545 0139

EUROPE

United Kingdom
Arizona Microchip Technology Ltd.
Unit 6, The Courtyard
Meadow Bank, Furlong Road
Bourne End, Buckinghamshire SL8 5AJ
Tel: 44 0 1628 851077 Fax: 44 0 1628 850259

France
Arizona Microchip Technology SARL
2 Rue du Buisson aux Fraises
91300 Massy - France
Tel: 33 1 69 53 63 20 Fax: 33 1 69 30 90 79

Germany
Arizona Microchip Technology GmbH
Gustav-Heinemann-Ring 125
D-81739 Muenchen, Germany
Tel: 49 89 627 144 0 Fax: 49 89 627 144 44

Italy
Arizona Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Pegaso Ingresso No. 2
Via Paracelso 23, 20041
Agrate Brianza (MI) Italy
Tel: 39 039 689 9939 Fax: 39 039 689 9883

JAPAN

Microchip Technology Intl. Inc.
Benex S-1 6F
3-18-20, Shin Yokohama
Kohoku-Ku, Yokohama
Kanagawa 222 Japan
Tel: 81 45 471 6166 Fax: 81 45 471 6122

9/22/95