Welcome to E-XFL.COM

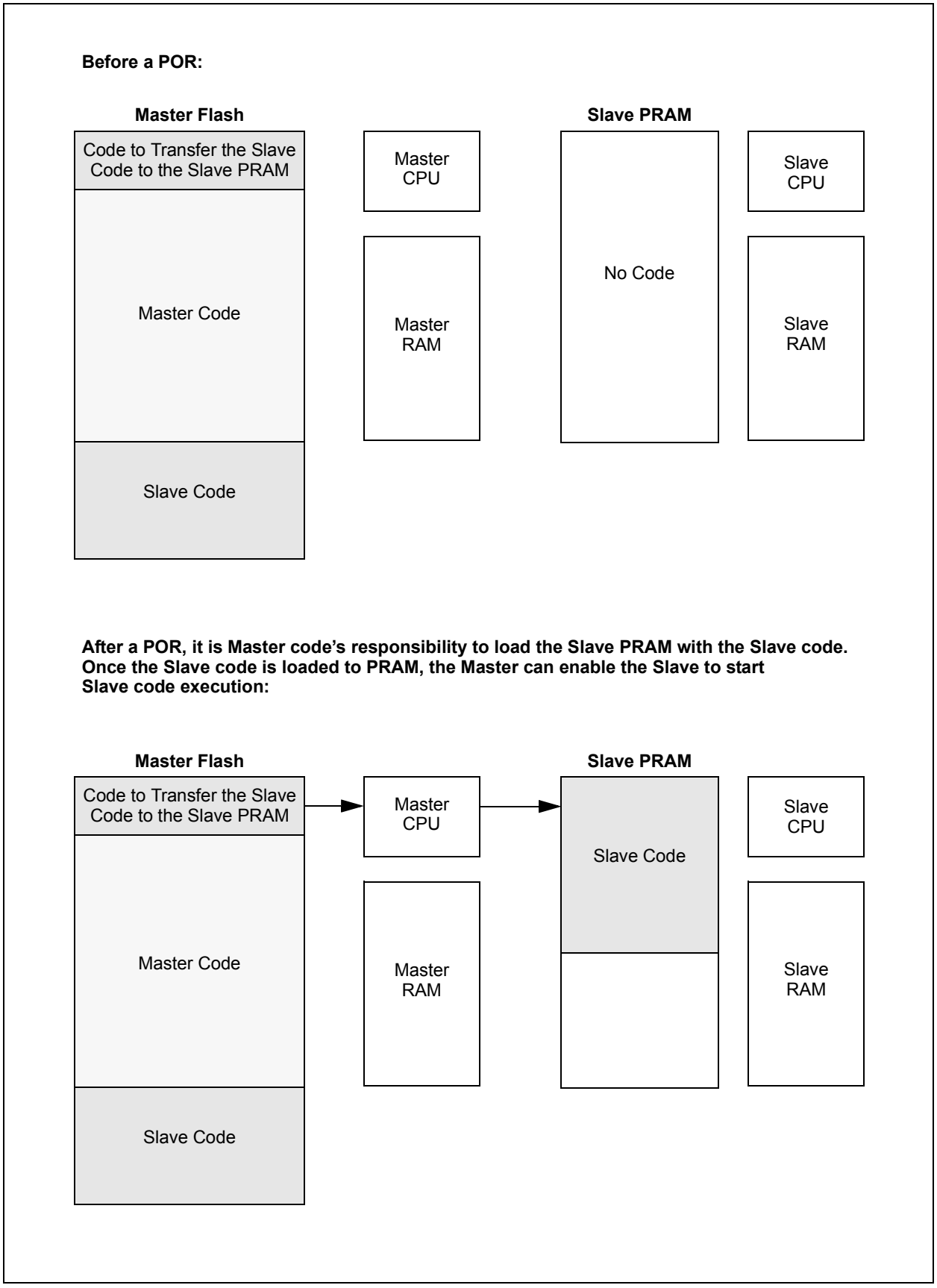
What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit Dual-Core
Speed	180MHz, 200MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, QEI, WDT
Number of I/O	39
Program Memory Size	152KB (152K x 8)
Program Memory Type	FLASH, PRAM
EEPROM Size	-
RAM Size	20K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 31x12b; D/A 4x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-TQFP
Supplier Device Package	48-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33ch128mp205-i-pt

FIGURE 1-1: SLAVE CORE CODE TRANSFER BLOCK DIAGRAM



REGISTER 3-11: ECCADDRL: ECC FAULT INJECT ADDRESS COMPARE REGISTER LOW

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCADDR<15:8>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCADDR<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **ECCADDR<15:0>:** ECC Fault Injection NVM Address Match Compare bits

REGISTER 3-12: ECCADDRH: ECC FAULT INJECT ADDRESS COMPARE REGISTER HIGH

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCADDR<31:24>							
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCADDR<23:16>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **ECCADDR<31:16>:** ECC Fault Injection NVM Address Match Compare bits

REGISTER 3-130: C1TXQCONH: CAN TRANSMIT QUEUE CONTROL REGISTER HIGH

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PLSIZE2(1)	PLSIZE1(1)	PLSIZE0(1)	FSIZE4(1)	FSIZE3(1)	FSIZE2(1)	FSIZE1(1)	FSIZE0(1)
bit 15							bit 8

U-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TXAT1	TXAT0	TXPRI4	TXPRI3	TXPRI2	TXPRI1	TXPRI0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-13 **PLSIZE<2:0>:** Payload Size bits(1)
111 = 64 data bytes
110 = 48 data bytes
101 = 32 data bytes
100 = 24 data bytes
011 = 20 data bytes
010 = 16 data bytes
001 = 12 data bytes
000 = 8 data bytes

bit 12-8 **FSIZE<4:0>:** FIFO Size bits(1)
11111 = FIFO is 32 messages deep
...
00010 = FIFO is 3 messages deep
00001 = FIFO is 2 messages deep
00000 = FIFO is 1 message deep

bit 7 **Unimplemented:** Read as '0'

bit 6-5 **TXAT<1:0>:** Retransmission Attempts bits
This feature is enabled when RTXAT (C1CONH<0>) is set.
11 = Unlimited number of retransmission attempts
10 = Unlimited number of retransmission attempts
01 = Three retransmission attempts
00 = Disables retransmission attempts

bit 4-0 **TXPRI<4:0>:** Message Transmit Priority bits
11111 = Highest message priority
...
00000 = Lowest message priority

Note 1: These bits can only be modified in Configuration mode (OPMOD<2:0> = 100).

REGISTER 3-139: C1FIFOUAHx: CAN FIFO USER ADDRESS REGISTER x (x = 1 TO 7) HIGH(1)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
FIFOUA<31:24>							
bit 15							bit 8

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
FIFOUA<23:16>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **FIFOUA<31:16>:** FIFO User Address bits

TXEN = 1 (FIFO configured as a transmit buffer):
A read of this register will return the address where the next message is to be written (FIFO head).

TXEN = 0 (FIFO configured as a receive buffer):
A read of this register will return the address where the next message is to be read (FIFO tail).

Note 1: This register is not ensured to read correctly in Configuration mode and should only be accessed when the module is not in Configuration mode.

REGISTER 3-140: C1FIFOUALx: CAN FIFO USER ADDRESS REGISTER x (x = 1 TO 7) LOW(1)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
FIFOUA<15:8>							
bit 15							bit 8

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
FIFOUA<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **FIFOUA<15:0>:** FIFO User Address bits

TXEN = 1 (FIFO configured as a transmit buffer):
A read of this register will return the address where the next message is to be written (FIFO head).

TXEN = 0 (FIFO configured as a receive buffer):
A read of this register will return the address where the next message is to be read (FIFO tail).

Note 1: This register is not ensured to read correctly in Configuration mode and should only be accessed when the module is not in Configuration mode.

REGISTER 3-141: C1TEFUAH: CAN TRANSMIT EVENT FIFO USER ADDRESS REGISTER HIGH(1)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
TEFUA<31:24>							
bit 15							bit 8

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
TEFUA<23:16>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **TEFUA<31:16>:** Transmit Event FIFO User Address bits
A read of this register will return the address where the next event is to be read (FIFO tail).

Note 1: This register is not ensured to read correctly in Configuration mode and should only be accessed when the module is not in Configuration mode.

REGISTER 3-142: C1TEFUAL: CAN TRANSMIT EVENT FIFO USER ADDRESS REGISTER LOW(1)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
TEFUA<15:8>							
bit 15							bit 8

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
TEFUA<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-0 **TEFUA<15:0>:** Transmit Event FIFO User Address bits
A read of this register will return the address where the next event is to be read (FIFO tail).

Note 1: This register is not ensured to read correctly in Configuration mode and should only be accessed when the module is not in Configuration mode.

REGISTER 3-163: ADCON5L: ADC CONTROL REGISTER 5 LOW

HSC/R-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
SHRRDY	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
SHRPWR	—	—	—	—	—	—	—
bit 7							bit 0

Legend:	U = Unimplemented bit, read as '0'		
R = Readable bit	W = Writable bit	HSC = Hardware Settable/Clearable bit	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **SHRRDY:** Shared ADC Core Ready Flag bit
1 = ADC core is powered and ready for operation
0 = ADC core is not ready for operation

bit 14-8 **Unimplemented:** Read as '0'

bit 7 **SHRPWR:** Shared ADC Core Power Enable bit
1 = ADC core is powered
0 = ADC core is off

bit 6-0 **Unimplemented:** Read as '0'

REGISTER 4-89: ADCON4L: ADC CONTROL REGISTER 4 LOW

U-0	U-0	U-0	U-0	U-0	U-0	r-0	r-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	SAMC1EN	SAMC0EN
bit 7							bit 0

Legend:	r = Reserved bit		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-10 **Unimplemented:** Read as '0'

bit 9-8 **Reserved:** Must be written as '0'

bit 7-2 **Unimplemented:** Read as '0'

bit 1 **SAMC1EN:** Dedicated ADC Core 1 Conversion Delay Enable bit

1 = After trigger, the conversion will be delayed and the ADC core will continue sampling during the time specified by the SAMC<9:0> bits in the ADCORE1L register

0 = After trigger, the sampling will be stopped immediately and the conversion will be started on the next core clock cycle

bit 0 **SAMC0EN:** Dedicated ADC Core 0 Conversion Delay Enable bit

1 = After trigger, the conversion will be delayed and the ADC core will continue sampling during the time specified by the SAMC<9:0> bits in the ADCORE0L register

0 = After trigger, the sampling will be stopped immediately and the conversion will be started on the next core clock cycle

REGISTER 4-94: ADCOREXH: DEDICATED ADC CORE x CONTROL REGISTER HIGH (x = 0 TO 1)

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	EISEL2	EISEL1	EISEL0	RES1	RES2
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ADCS6	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-13 **Unimplemented:** Read as '0'

bit 12-10 **EISEL<2:0>:** ADC Core x Early Interrupt Time Selection bits
111 = Early interrupt is set and an interrupt is generated 8 TADCORE clocks prior to when the data is ready
110 = Early interrupt is set and an interrupt is generated 7 TADCORE clocks prior to when the data is ready
101 = Early interrupt is set and an interrupt is generated 6 TADCORE clocks prior to when the data is ready
100 = Early interrupt is set and an interrupt is generated 5 TADCORE clocks prior to when the data is ready
011 = Early interrupt is set and an interrupt is generated 4 TADCORE clocks prior to when the data is ready
010 = Early interrupt is set and an interrupt is generated 3 TADCORE clocks prior to when the data is ready
001 = Early interrupt is set and an interrupt is generated 2 TADCORE clocks prior to when the data is ready
000 = Early interrupt is set and an interrupt is generated 1 TADCORE clock prior to when the data is ready

bit 9-8 **RES<1:0>:** ADC Core x Resolution Selection bits
11 = 12-bit resolution
10 = 10-bit resolution
01 = 8-bit resolution(1)
00 = 6-bit resolution(1)

bit 7 **Unimplemented:** Read as '0'

bit 6-0 **ADCS<6:0>:** ADC Core x Input Clock Divider bits
These bits determine the number of Source Clock Periods (TCORESRC) for one Core Clock Period (TADCORE).
1111111 = 254 Source Clock Periods
...
0000011 = 6 Source Clock Periods
0000010 = 4 Source Clock Periods
0000001 = 2 Source Clock Periods
0000000 = 2 Source Clock Periods

Note 1: For the 6-bit ADC core resolution (RES<1:0> = 00), the EISEL<2:0> bits settings, from '100' to '111', are not valid and should not be used. For the 8-bit ADC core resolution (RES<1:0> = 01), the EISEL<2:0> bits settings, '110' and '111', are not valid and should not be used.

5.3 Slave Processor Control

The MSI contains three control bits related to Slave processor control within the MSI1CON register.

5.3.1 SLAVE ENABLE (SLVEN) CONTROL

The SLVEN (MSI1CON<15>) control bit provides a means for the Master processor to enable or disable the Slave processor.

The Slave is disabled when SLVEN (MSI1CON<15>) = 0. In this state:

- The Slave is held in the Reset state
- The Master has access to the Slave PRAM (to load it out of a device Reset)
- The Slave Reset status bit, SLVRST (MSI1STAT<15>) = 1

The Slave is enabled when SLVEN (MSI1CON<15>) = 1. In this state:

- The Slave Reset is released and it will start to execute code in whatever mode it is configured to operate in
- The Master processor will no longer have access to the Slave PRAM
- The Slave Reset status bit, SLVRST (MSI1STAT<15>) = 0

Note: The SLVRST (MSI1STAT<15>) status bit indicates when the Slave is in Reset. The associated interrupt only occurs when the Slave enters the Reset state after having previously not been in Reset. That is, no interrupt can be generated until the Slave is first enabled.

The SLVEN bit may only be modified after satisfying the hardware write interlock. The SLVEN bit is protected from unexpected writes through a software unlocking sequence that is based on the MSI1KEY register. Given the critical nature of the MSI control interface, the MSI macro unlock mechanism is independent from that of the Flash controller for added robustness.

Completing a predefined data write sequence to the MSI1KEY register will open a window. The SLVEN bit should be written on the first instruction that follows the unlock sequence. No other bits within the MSI1CON register are affected by the interlock. The MSI1KEY register is not a physical register. A read of the MSI1KEY register will read all '0's.

When the SLVEN bit lock is enabled (i.e., the bits are locked and cannot be modified), the instruction sequence shown in Example 5-1 must be executed to open the lock. The unlock sequence is a prerequisite to both setting and clearing the target control bit.

Note: It is recommended to enable SRSTIE (MSI1CON<7>) = 1 prior to enabling the SLVEN bit. This will make the design robust and will update the Master with the Reset state of the Slave.

EXAMPLE 5-1: MSI ENABLE OPERATION

```
//Unlock Key to allow MSI Enable control
MOV.b   #0x55, W0
MOV.b   WREG, MSI1KEY
MOV.b   #0xAA, W0
MOV.b   WREG, MSI1KEY
// Enable MSI
BSET    MSI1CON, SLVEN
```

EXAMPLE 5-2: MSI ENABLE OPERATION IN C CODE

```
#include <libpic30.h>
_start_slave();
```

5.4 Slave Reset Coupling Control

In all operating modes, the user may couple or decouple the Master Run-Time Resets to the Slave Reset by using the Master Slave Reset Enable (S1MSRE) fuse. The Resets are effectively coupled by directing the selected Reset source to the SLVEN bit Reset.

In all operating modes, the user may also choose whether the SLVEN bit is reset or not in the event of a Slave Run-Time Reset by using the Slave Reset Enable (S1SSRE) fuse.

A user may choose to reset SLVEN in the event of a Slave Reset because that event could be an indicator of a problem with Slave execution. The Slave would be placed in Reset and the Master alerted (via the Slave Reset event interrupt, need to make SRSTIE (MSI1CON<7> = 1) to attempt to rectify the problem. The Master must re-enable the Slave by setting the SLVEN bit again.

Alternatively, the user may choose to not halt the Slave in the event of a Slave Reset, and just allow it to restart execution after a Reset and continue operation as soon as possible. The Slave Reset event interrupt would still occur, but could be ignored by the Master.

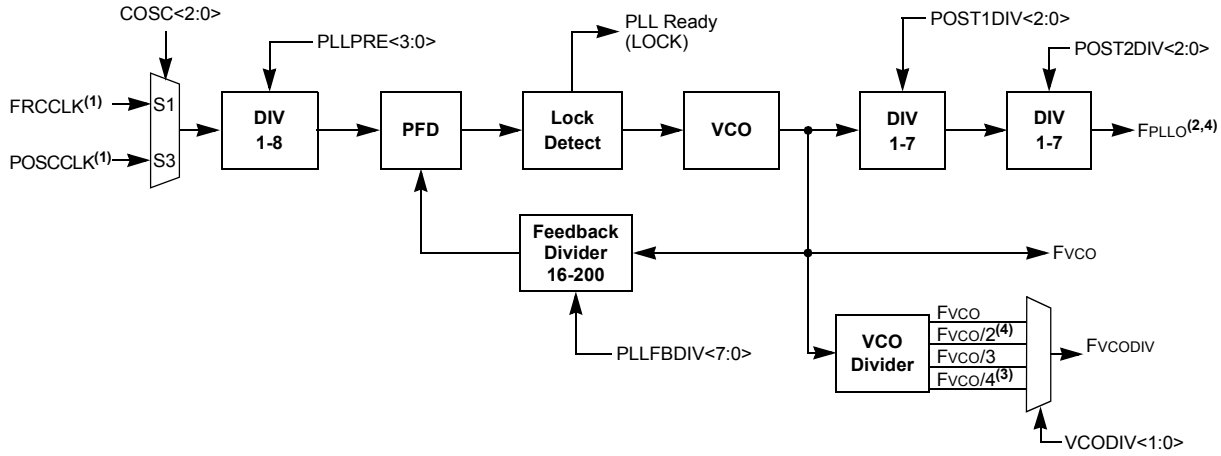
The Primary Oscillator and internal FRC Oscillator sources can optionally use an on-chip PLL to obtain higher operating speeds. There are two independent instantiations of PLL for the Master and Slave clock subsystems. Figure 6-4 illustrates a block diagram of the Master/Slave core PLL module.

For PLL operation, the following requirements must be met at all times without exception:

- The PLL Input Frequency (FPLLI) must be in the range of 8 MHz to 64 MHz
- The PFD Input Frequency (FPFD) must be in the range of 8 MHz to (FVCO/16) MHz

The VCO Output Frequency (FVCO) must be in the range of 400 MHz to 1600 MHz

FIGURE 6-4: MASTER/SLAVE CORE PLL AND VCO DETAIL



Note 1: From Master and Slave core shared oscillator source.
2: Clock option for PWM.
3: Clock option for ADC.
4: Clock option for DAC.

REGISTER 6-15: PLLDIV: PLL OUTPUT DIVIDER REGISTER (SLAVE)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	VCODIV<1:0>	
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-1
—	POST1DIV<2:0>(1,2)			—	POST2DIV<2:0>(1,2)		
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-10 **Unimplemented:** Read as '0'

bit 9-8 **VCODIV<1:0>:** PLL VCO Output Divider Select bits

11 = FVCO
10 = FVCO/2
01 = FVCO/3
00 = FVCO/4

bit 7 **Unimplemented:** Read as '0'

bit 6-4 **POST1DIV<2:0>:** PLL Output Divider #1 Ratio bits(1,2)

POST1DIV<2:0> can have a valid value, from 1 to 7 (POST1DIVx value should be greater than or equal to the POST2DIVx value). The POST1DIVx divider is designed to operate at higher clock rates than the POST2DIVx divider.

bit 3 **Unimplemented:** Read as '0'

bit 2-0 **POST2DIV<2:0>:** PLL Output Divider #2 Ratio bits(1,2)

POST2DIV<2:0> can have a valid value, from 1 to 7 (POST2DIVx value should be less than or equal to the POST1DIVx value). The POST1DIVx divider is designed to operate at higher clock rates than the POST2DIVx divider.

Note 1: The POST1DIVx and POST2DIVx divider values must not be changed while the PLL is operating.

2: The default values for POST1DIVx and POST2DIVx are 4 and 1, respectively, yielding a 150 MHz system source clock.

REGISTER 6-18: APLLDIV: APLL OUTPUT DIVIDER REGISTER (SLAVE)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	AVCODIV<1:0>	
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-1
—	APOST1DIV<2:0>(1,2)			—	APOST2DIV<2:0>(1,2)		
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-10 **Unimplemented:** Read as '0'

bit 9-8 **AVCODIV<1:0>:** APLL VCO Output Divider Select bits
11 = AFVCO
10 = AFVCO/2
01 = AFVCO/3
00 = AFVCO/4

bit 7 **Unimplemented:** Read as '0'

bit 6-4 **APOST1DIV<2:0>:** APLL Output Divider #1 Ratio bits(1,2)
APOST1DIV<2:0> can have a valid value, from 1 to 7 (APOST1DIVx value should be greater than or equal to the APOST2DIVx value). The APOST1DIVx divider is designed to operate at higher clock rates than the APOST2DIVx divider.

bit 3 **Unimplemented:** Read as '0'

bit 2-0 **APOST2DIV<2:0>:** APLL Output Divider #2 Ratio bits(1,2)
APOST2DIV<2:0> can have a valid value, from 1 to 7 (APOST2DIVx value should be less than or equal to the APOST1DIVx value). The APOST1DIVx divider is designed to operate at higher clock rates than the APOST2DIVx divider.

Note 1: The APOST1DIVx and APOST2DIVx divider values must not be changed while the PLL is operating.
2: The default values for APOST1DIVx and APOST2DIVx are 4 and 1, respectively, yielding a 150 MHz system source clock.

REGISTER 8-3: DMAINTn: DMA CHANNEL n INTERRUPT REGISTER

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DBUFWF(1)	CHSEL6	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
HIGHIF(1,2)	LOWIF(1,2)	DONEIF(1)	HALFIF(1)	OVRUNIF(1)	—	—	HALFEN
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **DBUFWF:** DMA Buffered Data Write Flag bit(1)
- 1 = The content of the DMA buffer has not been written to the location specified in DMADSTn or DMASRCn in Null Write mode
- 0 = The content of the DMA buffer has been written to the location specified in DMADSTn or DMASRCn in Null Write mode

bit 14-8 **CHSEL<6:0>:** DMA Channel Trigger Selection bits

See Table 8-2 for a complete list.

bit 7 **HIGHIF:** DMA High Address Limit Interrupt Flag bit(1,2)
- 1 = The DMA channel has attempted to access an address higher than DMAH or the upper limit of the data RAM space
- 0 = The DMA channel has not invoked the high address limit interrupt

bit 6 **LOWIF:** DMA Low Address Limit Interrupt Flag bit(1,2)
- 1 = The DMA channel has attempted to access the DMA SFR address lower than DMAL, but above the SFR range (07FFh)
- 0 = The DMA channel has not invoked the low address limit interrupt

bit 5 **DONEIF:** DMA Complete Operation Interrupt Flag bit(1)

If CHEN = 1:
- 1 = The previous DMA session has ended with completion
- 0 = The current DMA session has not yet completed

If CHEN = 0:
- 1 = The previous DMA session has ended with completion
- 0 = The previous DMA session has ended without completion

bit 4 **HALFIF:** DMA 50% Watermark Level Interrupt Flag bit(1)
- 1 = DMACNTn has reached the halfway point to 0000h
- 0 = DMACNTn has not reached the halfway point

bit 3 **OVRUNIF:** DMA Channel Overrun Flag bit(1)
- 1 = The DMA channel is triggered while it is still completing the operation based on the previous trigger
- 0 = The overrun condition has not occurred

bit 2-1 **Unimplemented:** Read as '0'

bit 0 **HALFEN:** Halfway Completion Watermark bit
- 1 = Interrupts are invoked when DMACNTn has reached its halfway point and at completion
- 0 = An interrupt is invoked only at the completion of the transfer

Note 1: Setting these flags in software does not generate an interrupt.

2: Testing for address limit violations (DMASRCn or DMADSTn is either greater than DMAH or less than DMAL) is NOT done before the actual access.

REGISTER 9-14: PGxSTAT: PWM GENERATOR x STATUS REGISTER (CONTINUED)

bit 5 **CAP:** Capture Status bit(1)
1 = PWM Generator time base value has been captured in PGxCAP
0 = No capture has occurred

bit 4 **UPDATE:** PWM Data Register Update Status/Control bit
1 = PWM Data register update is pending – user Data registers are not writable
0 = No PWM Data register update is pending

bit 3 **UPDREQ:** PWM Data Register Update Request bit
User software writes a '1' to this bit location to request a PWM Data register update. The bit location always reads as '0'. The UPDATE status bit will indicate '1' when an update is pending.

bit 2 **STEER:** Output Steering Status bit (Push-Pull Output mode only)
1 = PWM Generator is in 2nd cycle of Push-Pull mode
0 = PWM Generator is in 1st cycle of Push-Pull mode

bit 1 **CAHALF:** Half Cycle Status bit (Center-Aligned modes only)
1 = PWM Generator is in 2nd half of time base cycle
0 = PWM Generator is in 1st half of time base cycle

bit 0 **TRIG:** PWM Trigger Status bit
1 = PWM Generator is triggered and PWM cycle is in progress
0 = No PWM cycle is in progress

Note 1: User software may write a '1' to CAP as a request to initiate a software capture. The CAP status bit will be set when the capture event has occurred. No further captures will occur until CAP is cleared by software.

REGISTER 9-17: PGxyPCIL: PWM GENERATOR xy PCI REGISTER LOW (x = PWM GENERATOR #; y = F, CL, FF OR S) (CONTINUED)

bit 4-0 **PSS<4:0>:** PCI Source Selection bits

For Master:

11111 = Master CLC1
11110 = Slave Comparator 3 output
11101 = Slave Comparator 2 output
11100 = Slave Comparator 1 output
11011 = Master Comparator 1 output
11010 = Slave PWM Event F
11001 = Slave PWM Event E
11000 = Slave PWM Event D
10111 = Slave PWM Event C
10110 = Device pin, PCI<22>
10101 = Device pin, PCI<21>
10100 = Device pin, PCI<20>
10011 = Device pin, PCI<19>
10010 = Master RPn input, Master PCI18R
10001 = Master RPn input, Master PCI17R
10000 = Master RPn input, Master PCI16R
01111 = Master RPn input, Master PCI15R
01110 = Master RPn input, Master PCI14R
01101 = Master RPn input, Master PCI13R
01100 = Master RPn input, Master PCI12R
01011 = Master RPn input, Master PCI11R
01010 = Master RPn input, Master PCI10R
01001 = Master RPn input, Master PCI9R
01000 = Master RPn input, Master PCI8R
00111 = Reserved
00110 = Reserved
00101 = Reserved
00100 = Reserved
00011 = Internally connected to Combo Trigger B
00010 = Internally connected to Combo Trigger A
00001 = Internally connected to the output of PWMPCI<2:0> MUX
00000 = Tied to '0'

REGISTER 9-25: PGxDCA: PWM GENERATOR x DUTY CYCLE ADJUSTMENT REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PGxDCA<7:0>							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-8 **Unimplemented:** Read as ‘0’

bit 7-0 **PGxDCA<7:0>:** PWM Generator x Duty Cycle Adjustment Value bits

Depending on the state of the selected PCI source, the PGxDCA value will be added to the value in the PGxDC register to create the effective duty cycle. When the PCI source is active, PGxDCA is added. When the PCI source is inactive, no adjustment is made. Duty cycle adjustment is disabled when PGxDCA<7:0> = 0. The PCI source is selected using the DTCMPSEL bit.

REGISTER 9-26: PGxPER: PWM GENERATOR x PERIOD REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PGxPER<15:8>(1)							
bit 15							bit 8

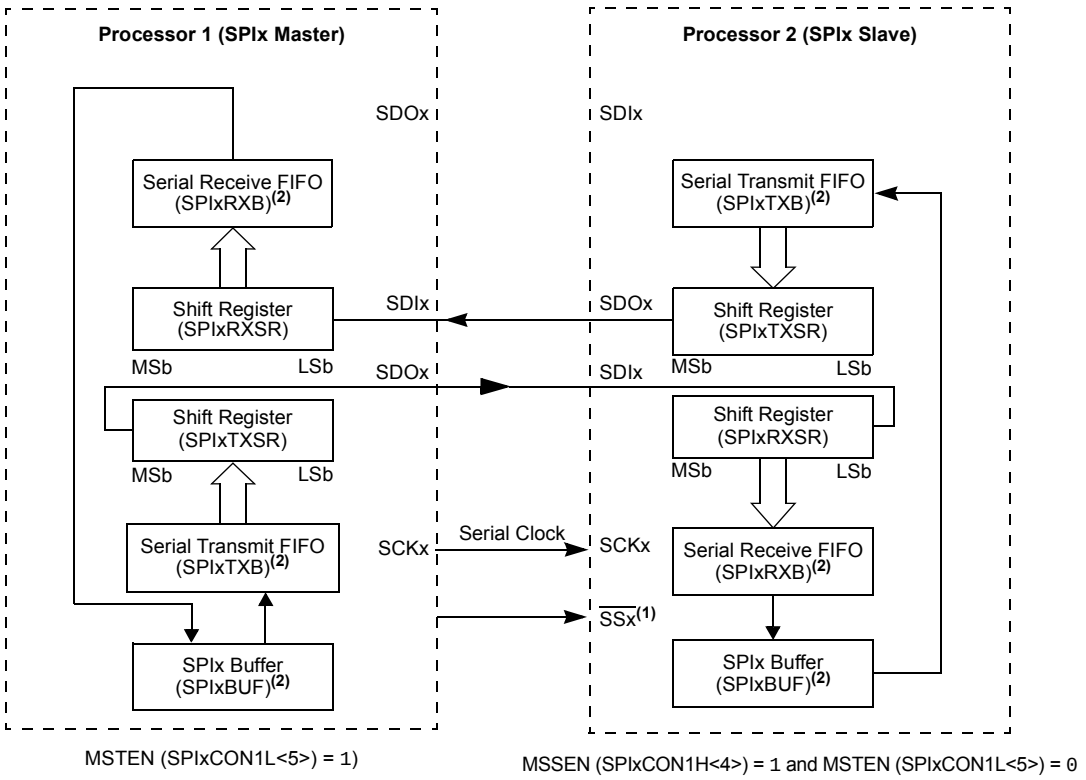
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PGxPER<7:0>(1)							
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 15-0 **PGxPER<15:0>:** PWM Generator x Period Register bits(1)

Note 1: Period values less than ‘0x0010’ should not be selected.

FIGURE 14-4: SPIx MASTER/SLAVE CONNECTION (ENHANCED BUFFER MODES)



Note 1: Using the $\overline{SSx}$ pin in Slave mode of operation is optional.

2: User must write transmit data to read the received data from SPIxBUF. The SPIxTXB and SPIxRXB registers are memory-mapped to SPIxBUF.

FIGURE 14-5: SPIx MASTER, FRAME MASTER CONNECTION DIAGRAM

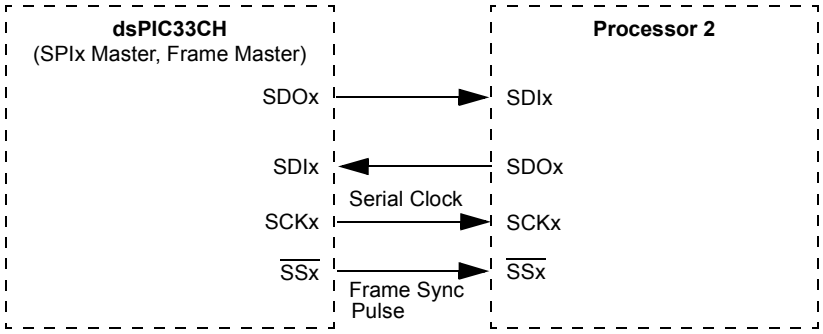


FIGURE 16-1: SENTx MODULE BLOCK DIAGRAM

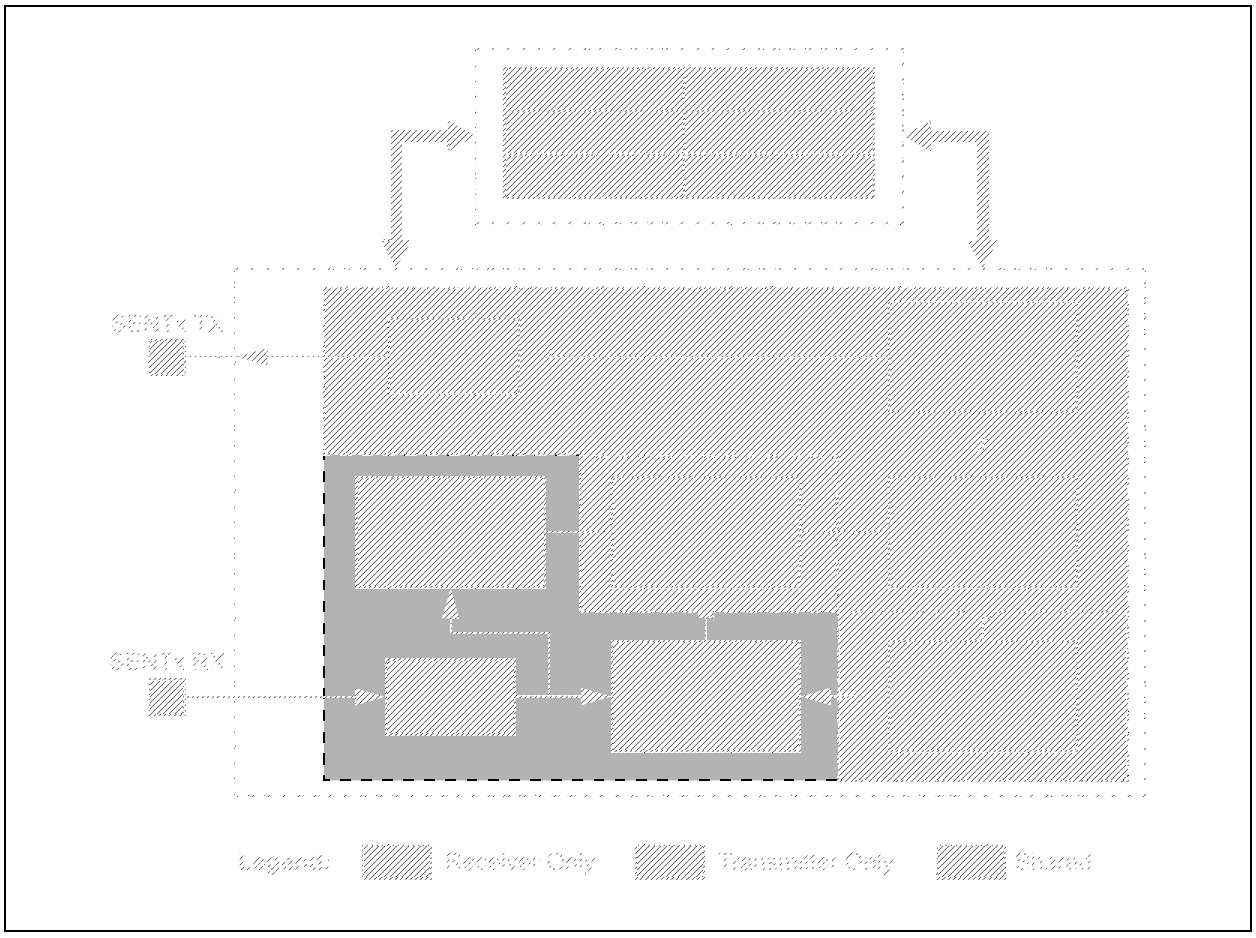
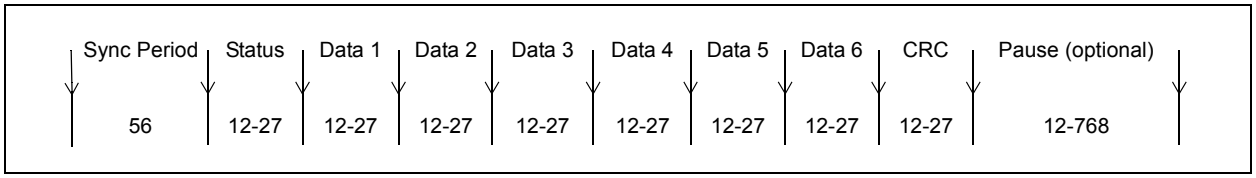
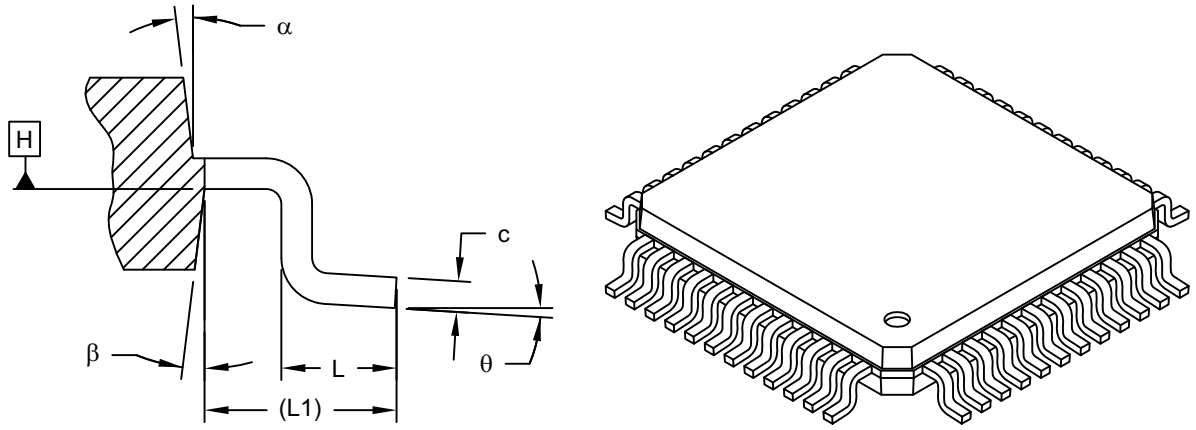


FIGURE 16-2: SENTx PROTOCOL DATA FRAMES



48-Lead Thin Quad Flatpack (PT) - 7x7x1.0 mm Body [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



SECTION A-A

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Leads	N		48	
Lead Pitch	e		0.50 BSC	
Overall Height	A	-	-	1.20
Standoff	A1	0.05	-	0.15
Molded Package Thickness	A2	0.95	1.00	1.05
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	ϕ	0°	3.5°	7°
Overall Width	E		9.00 BSC	
Overall Length	D		9.00 BSC	
Molded Package Width	E1		7.00 BSC	
Molded Package Length	D1		7.00 BSC	
Lead Thickness	c	0.09	-	0.16
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

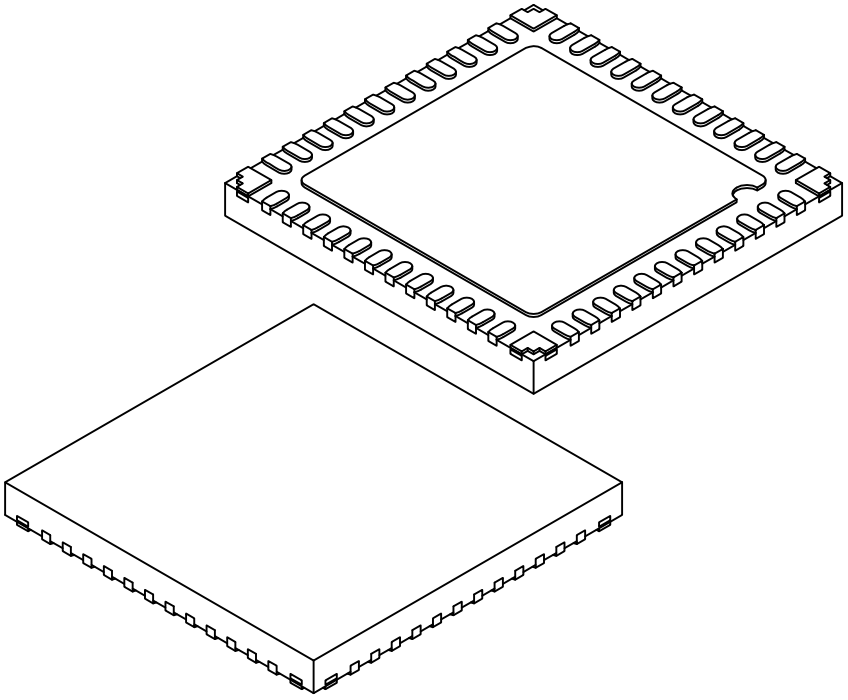
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.
5. Datums A-B and D to be determined at center line between leads where leads exit plastic body at datum plane H

Microchip Technology Drawing C04-300-PT Rev A Sheet 2 of 2

48-Lead Ultra Thin Plastic Quad Flat, No Lead Package (M4) - 6x6 mm Body [UQFN] With Corner Anchors and 4.6x4.6 mm Exposed Pad

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Terminals	N		48	
Pitch	e		0.40 BSC	
Overall Height	A	0.50	0.55	0.60
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3		0.15 REF	
Overall Length	D		6.00 BSC	
Exposed Pad Length	D2	4.50	4.60	4.70
Overall Width	E		6.00 BSC	
Exposed Pad Width	E2	4.50	4.60	4.70
Terminal Width	b	0.15	0.20	0.25
Corner Anchor Pad	b1		0.45 REF	
Corner Anchor Pad, Metal-free Zone	b2		0.23 REF	
Terminal Length	L	0.35	0.40	0.45
Terminal-to-Exposed-Pad	K		0.30 REF	

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-442A-M4 Sheet 2 of 2