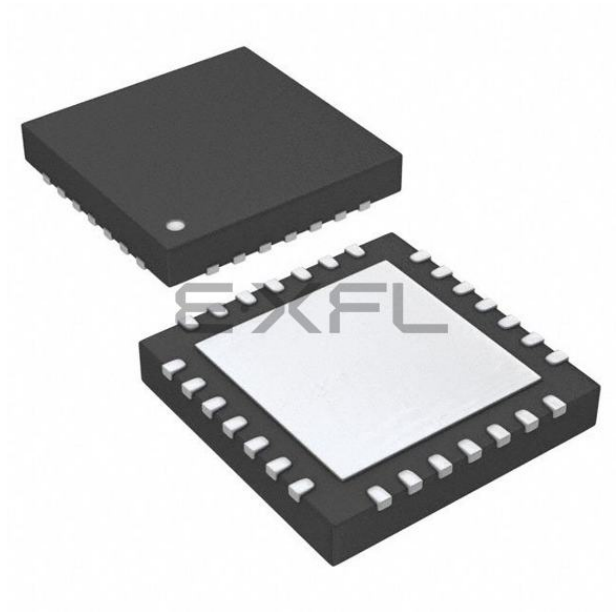


Welcome to E-XFL.COM

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	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	25
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 24x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f18856-i-ml

TABLE 1-2: PIC16F18856 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
OUT(2)	CWG3C	—	CMOS/OD	Complementary Waveform Generator 3 output C.
	CWG3D	—	CMOS/OD	Complementary Waveform Generator 3 output D.
	CLC1OUT	—	CMOS/OD	Configurable Logic Cell 1 output.
	CLC2OUT	—	CMOS/OD	Configurable Logic Cell 2 output.
	CLC3OUT	—	CMOS/OD	Configurable Logic Cell 3 output.
	CLC4OUT	—	CMOS/OD	Configurable Logic Cell 4 output.
	NCO1	—	CMOS/OD	Numerically Controller Oscillator output.
	CLKR	—	CMOS/OD	Clock Reference module output.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open-Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C = Schmitt Trigger input with I^2C
HV = High Voltage XTAL = Crystal levels

Note 1: This is a PPS remappable input signal. The input function may be moved from the default location shown to one of several other PORTx pins. Refer to Table 13-1 for details on which PORT pins may be used for this signal.

2: All output signals shown in this row are PPS remappable. These signals may be mapped to output onto one of several PORTx pin options as described in Table 13-3.

3: This is a bidirectional signal. For normal module operation, the firmware should map this signal to the same pin in both the PPS input and PPS output registers.

4: These pins are configured for I^2C logic levels. The SCLx/SDAx signals may be assigned to any of the RB1/RB2/RC3/RC4 pins. PPS assignments to the other pins (e.g., RA5) will operate, but input logic levels will be standard TTL/ST, as selected by the INLVL register, instead of the I^2C specific or SMBus input buffer thresholds.

TABLE 3-10: PIC16(L)F18856 MEMORY MAP, BANK 30

	Bank 30
F0Ch	—
F0Dh	—
F0Eh	—
F0Fh	—
F10h	RA0PPS
F11h	RA1PPS
F12h	RA2PPS
F13h	RA3PPS
F14h	RA4PPS
F15h	RA5PPS
F16h	RA6PPS
F17h	RA7PPS
F18h	RB0PPS
F19h	RB1PPS
F1Ah	RB2PPS
F1Bh	RB3PPS
F1Ch	RB4PPS
F1Dh	RB5PPS
F1Eh	RB6PPS
F1Fh	RB7PPS
F20h	RC0PPS
F21h	RC1PPS
F22h	RC2PPS
F23h	RC3PPS
F24h	RC4PPS
F25h	RC5PPS
F26h	RC6PPS
F27h	RC7PPS
F28h F37h	—
F38h	ANSELA
F39h	WPUA
F3Ah	ODCONA
F3Bh	SLRCONA
F3Ch	INLVLA
F3Dh	IOCAP
F3Eh	IOCAN
F3Fh	IOCAF

	Bank 30
F40h	CCDNA
F41h	CCDPA
F42h	—
F43h	ANSELB
F44h	WPUB
F45h	ODCONB
F46h	SLRCONB
F47h	INLVLB
F48h	IOCBP
F49h	IOCBN
F4Ah	IOCBF
F4Bh	CCDNB
F4Ch	CCDPB
F4Dh	—
F4Eh	ANSELC
F4Fh	WPUC
F50h	ODCONC
F51h	SLRCONC
F52h	INLVLC
F53h	IOCCP
F54h	IOCCN
F55h	IOCCF
F56h	CCDNC
F57h	CCDPC
F58h F64h	—
F65h	WPUE
F66h	—
F67h	—
F68h	INLVLE
F69h	IOCEP
F6Ah	IOCEN
F6Bh	IOCEF
F6Ch	—
F6Dh	—
F6Eh	—
F6Fh	—

Legend: = Unimplemented data memory locations, read as '0'.

TABLE 3-13: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)

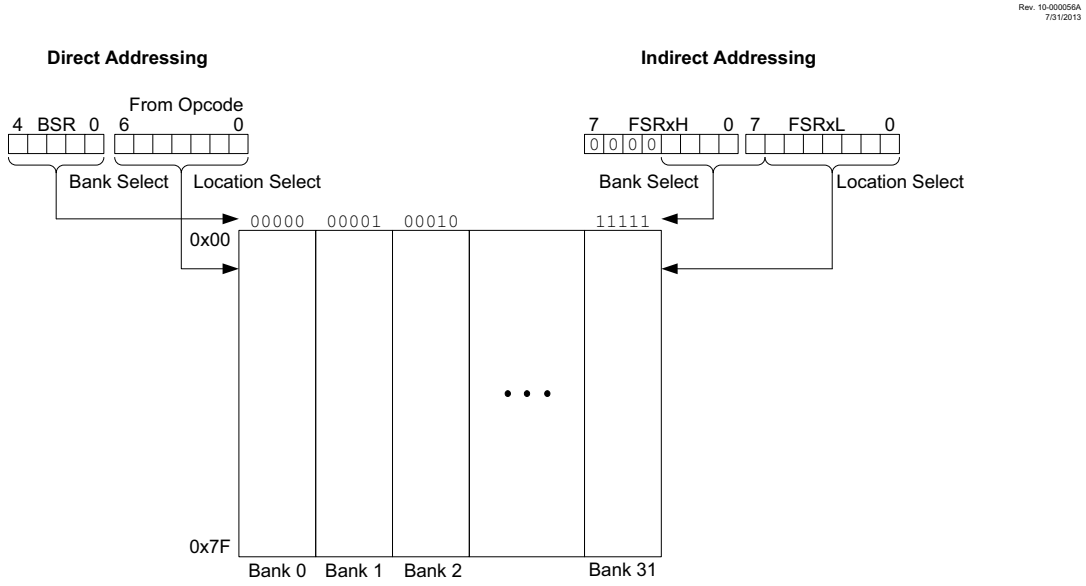
Address	Name	PIC16(L)F18856	PIC16(L)F18876	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
Bank 9													
CPU CORE REGISTERS; see Table 3-2 for specifics													
48Ch	SMT1TMRL			TMR<7:0>								0000 0000	0000 0000
48Dh	SMT1TMRH			TMR<15:8>								0000 0000	0000 0000
48Eh	SMT1TMRU			TMR<23:16>								0000 0000	0000 0000
48Fh	SMT1CPRL			CPR<7:0>								xxxx xxxx	uuuu uuuu
490h	SMT1CPRH			CPR<15:8>								xxxx xxxx	uuuu uuuu
491h	SMT1CPRU			CPR<23:16>								xxxx xxxx	uuuu uuuu
492h	SMT1CPWL			CPW<7:0>								xxxx xxxx	uuuu uuuu
493h	SMT1CPWH			CPW<15:8>								xxxx xxxx	uuuu uuuu
494h	SMT1CPWU			CPW<23:16>								xxxx xxxx	uuuu uuuu
495h	SMT1PRL			PR<7:0>								1111 1111	1111 1111
496h	SMT1PRH			PR<15:8>								1111 1111	1111 1111
497h	SMT1PRU			PR<23:16>								1111 1111	1111 1111
498h	SMT1CON0			EN	—	STP	WPOL	SPOL	CPOL	SMT1PS<1:0>		0-00 0000	0-00 0000
499h	SMT1CON1			SMT1GO	REPEAT	—	—	MODE<3:0>				00-- 0000	00-- 0000
49Ah	SMT1STAT			CPRUP	CPWUP	RST	—	—	TS	WS	AS	000- -000	000- -000
49Bh	SMT1CLK			—	—	—	—	—	CSEL<2:0>			---- -000	---- -000
49Ch	SMT1SIG			—	—	—	SSEL<4:0>					---0 0000	---0 0000
49Dh	SMT1WIN			—	—	—	WSEL<4:0>					---0 0000	---0 0000
49Eh	—	—		Unimplemented								—	—
49Fh	—	—		Unimplemented								—	—

Legend: x = unknown, u = unchanged, q =depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Register present on PIC16F18855/75 devices only.

2: Unimplemented, read as '1'.

FIGURE 3-9: TRADITIONAL DATA MEMORY MAP



11.7 Configuring the CRC

The following steps illustrate how to properly configure the CRC.

1. Determine if the automatic Program Memory scan will be used with the scanner or manual calculation through the SFR interface and perform the actions specified in **Section 11.4 "CRC Data Sources"**, depending on which decision was made.
2. If desired, seed a starting CRC value into the CRCACCH/L registers.
3. Program the CRCXORH/L registers with the desired generator polynomial.
4. Program the DLEN<3:0> bits of the CRCCON1 register with the length of the data word – 1 (refer to Example 11-1). This determines how many times the shifter will shift into the accumulator for each data word.
5. Program the PLEN<3:0> bits of the CRCCON1 register with the length of the polynomial – 2 (refer to Example 11-1).
6. Determine whether shifting in trailing zeros is desired and set the ACCM bit of CRCCON0 register appropriately.
7. Likewise, determine whether the MSb or LSb should be shifted first and write the SHIFTM bit of CRCCON0 register appropriately.
8. Write the CRCGO bit of the CRCCON0 register to begin the shifting process.

9a. If manual SFR entry is used, monitor the FULL bit of CRCCON0 register. When FULL = 0, another word of data can be written to the CRCDATH/L registers, keeping in mind that CRCDATH should be written first if the data has >8 bits, as the shifter will begin upon the CRCDATL register being written.

9b. If the scanner is used, the scanner will automatically stuff words into the CRCDATH/L registers as needed, as long as the SCANGO bit is set.

10a. If using the Flash memory scanner, monitor the SCANIF (or the SCANGO bit) for the scanner to finish pushing information into the CRCDATA registers. After the scanner is completed, monitor the CRCIF (or the BUSY bit) to determine that the CRC has been completed and the check value can be read from the CRCACC registers. If both the interrupt flags are set (or both BUSY and SCANGO bits are cleared), the completed CRC calculation can be read from the CRCACCH/L registers.

10b. If manual entry is used, monitor the CRCIF (or BUSY bit) to determine when the CRCACC registers will hold the check value.

11.8 Program Memory Scan Configuration

If desired, the Program Memory Scan module may be used in conjunction with the CRC module to perform a CRC calculation over a range of program memory addresses. In order to set up the Scanner to work with the CRC you need to perform the following steps:

1. Set the EN bit to enable the module. This can be performed at any point preceding the setting of the SCANGO bit, but if it gets disabled, all internal states of the Scanner are reset (registers are unaffected).
2. Choose which memory access mode is to be used (see **Section 11.10 "Scanning Modes"**) and set the MODE bits of the SCANCON0 register appropriately.
3. Based on the memory access mode, set the INTM bits of the SCANCON0 register to the appropriate interrupt mode (see **Section 11.10.5 "Interrupt Interaction"**)
4. Set the SCANLADRL/H and SCANHADRL/H registers with the beginning and ending locations in memory that are to be scanned.
5. Begin the scan by setting the SCANGO bit in the SCANCON0 register. The scanner will wait (CRCGO must be set) for the signal from the CRC that it is ready for the first Flash memory location, then begin loading data into the CRC. It will continue to do so until it either hits the configured end address or an address that is unimplemented on the device, at which point the SCANGO bit will clear, Scanner functions will cease, and the SCANIF interrupt will be triggered. Alternately, the SCANGO bit can be cleared in software if desired.

11.9 Scanner Interrupt

The scanner will trigger an interrupt when the SCANGO bit transitions from '1' to '0'. The SCANIF interrupt flag of PIR7 is set when the last memory location is reached and the data is entered into the CRCDATA registers. The SCANIF bit can only be cleared in software. The SCAN interrupt enable is the SCANIE bit of the PIE7 register.

11.10 Scanning Modes

The memory scanner can scan in four modes: Burst, Peek, Concurrent, and Triggered. These modes are controlled by the MODE bits of the SCANCON0 register. The four modes are summarized in Table 11-1.

REGISTER 12-16: WPUB: WEAK PULL-UP PORTB REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0 **WPUB<7:0>**: Weak Pull-up Register bits(1)
1 = Pull-up enabled
0 = Pull-up disabled

Note 1: The weak pull-up device is automatically disabled if the pin is configured as an output.

REGISTER 12-17: ODCONB: PORTB OPEN-DRAIN CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ODCB7	ODCB6	ODCB5	ODCB4	ODCB3	ODCB2	ODCB1	ODCB0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0 **ODCB<7:0>:** PORTB Open-Drain Enable bits
For RB<7:0> pins, respectively
1 = Port pin operates as open-drain drive (sink current only)
0 = Port pin operates as standard push-pull drive (source and sink current)

REGISTER 13-2: RxyPPS: PIN Rxy OUTPUT SOURCE SELECTION REGISTER

U-0	U-0	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u
—	—	RxyPPS<5:0>					
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **RxyPPS<5:0>:** Pin Rxy Output Source Selection bits
See Table 13-2.

Note 1: TRIS control is overridden by the peripheral as required.

REGISTER 13-3: PPSLOCK: PPS LOCK REGISTER

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

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-1 **Unimplemented:** Read as ‘0’

bit 0 **PPSLOCKED:** PPS Locked bit
1= PPS is locked. PPS selections can not be changed.
0= PPS is not locked. PPS selections can be changed.

14.0 PERIPHERAL MODULE DISABLE

The PIC16F18855/75 provides the ability to disable selected modules, placing them into the lowest possible Power mode.

For legacy reasons, all modules are ON by default following any Reset.

14.1 Disabling a Module

Disabling a module has the following effects:

- All clock and control inputs to the module are suspended; there are no logic transitions, and the module will not function.
- The module is held in Reset.
- Any SFRs become “unimplemented”
  - Writing is disabled
  - Reading returns 00h
- Module outputs are disabled; I/O goes to the next module according to pin priority

14.2 Enabling a module

When the register bit is cleared, the module is re-enabled and will be in its Reset state; SFR data will reflect the POR Reset values.

Depending on the module, it may take up to one full instruction cycle for the module to become active. There should be no interaction with the module (e.g., writing to registers) for at least one instruction after it has been re-enabled.

14.3 Disabling a Module

When a module is disabled, any and all associated input selection registers (ISMs) are also disabled.

14.4 System Clock Disable

Setting SYSCMD (PMD0, Register 14-1) disables the system clock (FOSC) distribution network to the peripherals. Not all peripherals make use of SYSCLK, so not all peripherals are affected. Refer to the specific peripheral description to see if it will be affected by this bit.

16.0 FIXED VOLTAGE REFERENCE (FVR)

The Fixed Voltage Reference, or FVR, is a stable voltage reference, independent of VDD, with 1.024V, 2.048V or 4.096V selectable output levels. The output of the FVR can be configured to supply a reference voltage to the following:

- ADC input channel
- ADC positive reference
- Comparator positive input
- Digital-to-Analog Converter (DAC)

The FVR can be enabled by setting the FVREN bit of the FVRCON register.

Note:	Fixed Voltage Reference output cannot exceed VDD.

16.1 Independent Gain Amplifiers

The output of the FVR, which is connected to the ADC, comparators, and DAC, is routed through two independent programmable gain amplifiers. Each amplifier can be programmed for a gain of 1x, 2x or 4x, to produce the three possible voltage levels.

The ADFVR<1:0> bits of the FVRCON register are used to enable and configure the gain amplifier settings for the reference supplied to the ADC module. Reference **Section 23.0 "Analog-to-Digital Converter With Computation (ADC2) Module"** for additional information.

The CDAFVR<1:0> bits of the FVRCON register are used to enable and configure the gain amplifier settings for the reference supplied to the DAC and comparator module. Reference **Section 25.0 "5-Bit Digital-to-Analog Converter (DAC1) Module"** and **Section 18.0 "Comparator Module"** for additional information.

16.2 FVR Stabilization Period

When the Fixed Voltage Reference module is enabled, it requires time for the reference and amplifier circuits to stabilize. Once the circuits stabilize and are ready for use, the FVRRDY bit of the FVRCON register will be set.

FIGURE 16-1: VOLTAGE REFERENCE BLOCK DIAGRAM

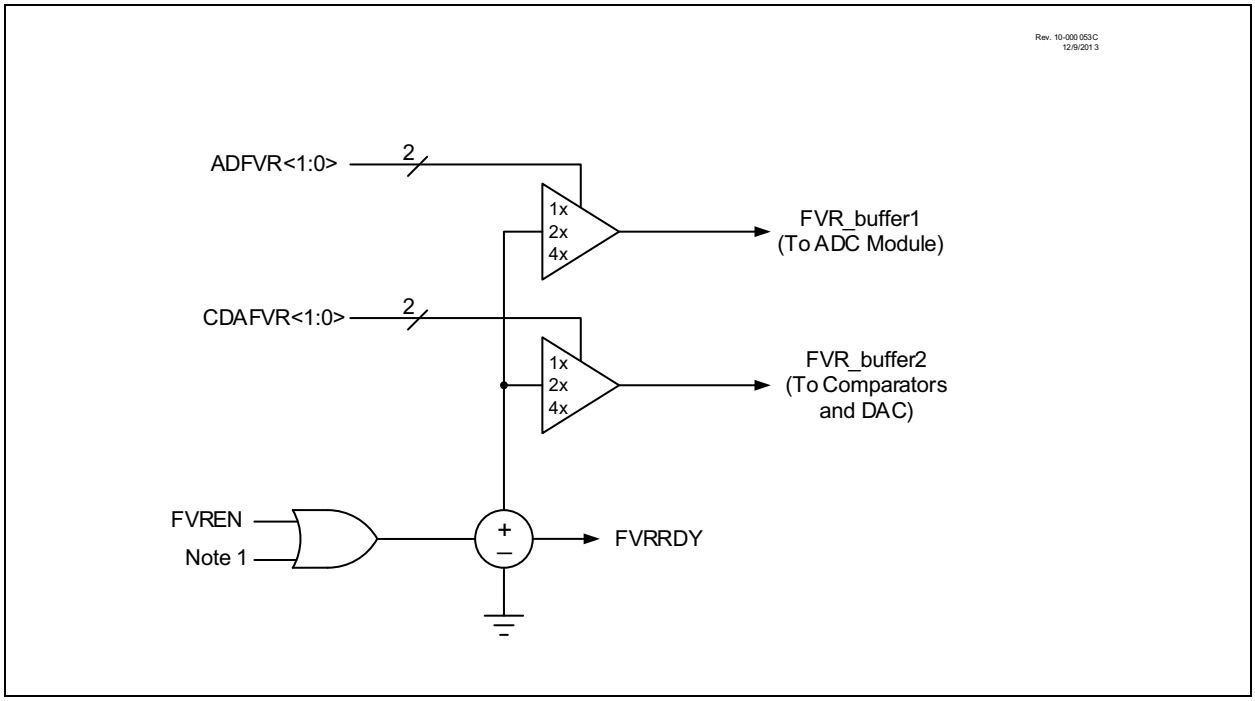


TABLE 26-1: SUMMARY OF REGISTERS ASSOCIATED WITH DATA SIGNAL MODULATOR MODE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	ANSA7	ANSA6	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	205
ANSELC	ANSC7	ANSC6	ANSC5	ANSC4	ANSC3	ANSC2	ANSC1	ANSC0	221
INLVLA	INLVLA7	INLVLA6	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	207
INLVLC	INLVLC7	INLVLC6	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	222
MDCON0	MDEN	—	MDOUT	MDOPOL	—	—	—	MDBIT	397
MDCON1	—	—	MDCHPOL	MDCHSYNC	—	—	MDCLPOL	MDCLSYNC	398
MDSRC	—	—	—	MDMS<4:0>					399
MDCARH	—	—	—	—	MDCHS<3:0>				400
MDCARL	—	—	—	—	MDCLS<3:0>				401
MDCARLPPS	—	—	—	MDCARLPPS<4:0>					249
MDCARHPPS	—	—	—	MDCARHPPS<4:0>					249
MDSRCPPS	—	—	—	MDSRCPPS<4:0>					249
RxyPPS	—	—	—	RxyPPS<4:0>					250
SLRCONA	SLRA7	SLRA6	SLRA5	SLRA4	SLRA3	SLRA2	SLRA1	SLRA0	207
SLRCONC	SLRC7	SLRC6	SLRC5	SLRC4	SLRC3	SLRC2	SLRC1	SLRC0	222
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	204
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	220

Legend: — = unimplemented, read as '0'. Shaded cells are not used in the Data Signal Modulator mode.

REGISTER 30-4: CCPRxH REGISTER: CCPx REGISTER HIGH BYTE

R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x	R/W-x/x
CCPRx<15:8>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Reset
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0 CCPxMODE = Capture mode
CCPRxH<7:0>: Captured value of TMR1H
CCPxMODE = Compare mode
CCPRxH<7:0>: MS Byte compared to TMR1H
CCPxMODE = PWM modes when CCPxFMT = 0:
CCPRxH<7:2>: Not used
CCPRxH<1:0>: Pulse-width Most Significant two bits
CCPxMODE = PWM modes when CCPxFMT = 1:
CCPRxH<7:0>: Pulse-width Most Significant eight bits

REGISTER 30-5: CCPTMRS0: CCP TIMERS CONTROL 0 REGISTER

R/W-0/0	R/W-1/1	R/W-0/0	R/W-1/1	R/W-0/0	R/W-1/1	R/W-0/0	R/W-1/1
C4TSEL<1:0>		C3TSEL<1:0>		C2TSEL<1:0>		C1TSEL<1:0>	
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Reset
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6 **C4TSEL<1:0>:** CCP4 Timer Selection
11 = CCP4 based on TMR5 (Capture/Compare) or TMR6 (PWM)
10 = CCP4 based on TMR3 (Capture/Compare) or TMR4 (PWM)
01 = CCP4 based on TMR1 (Capture/Compare) or TMR2 (PWM)
00 = Reserved

bit 5-4 **C3TSEL<1:0>:** CCP4 Timer Selection
11 = CCP3 based on TMR5 (Capture/Compare) or TMR6 (PWM)
10 = CCP3 based on TMR3 (Capture/Compare) or TMR4 (PWM)
01 = CCP3 based on TMR1 (Capture/Compare) or TMR2 (PWM)
00 = Reserved

bit 3-2 **C2TSEL<1:0>:** CCP4 Timer Selection
11 = CCP2 based on TMR5 (Capture/Compare) or TMR6 (PWM)
10 = CCP2 based on TMR3 (Capture/Compare) or TMR4 (PWM)
01 = CCP2 based on TMR1 (Capture/Compare) or TMR2 (PWM)
00 = Reserved

bit 1-0 **C1TSEL<1:0>:** CCP4 Timer Selection
11 = CCP1 based on TMR5 (Capture/Compare) or TMR6 (PWM)
10 = CCP1 based on TMR3 (Capture/Compare) or TMR4 (PWM)
01 = CCP1 based on TMR1 (Capture/Compare) or TMR2 (PWM)
00 = Reserved

31.6.10 SLEEP OPERATION

While in Sleep mode, the I^2C slave module can receive addresses or data and when an address match or complete byte transfer occurs, wake the processor from Sleep (if the MSSP interrupt is enabled).

31.6.11 EFFECTS OF A RESET

A Reset disables the MSSP module and terminates the current transfer.

31.6.12 MULTI-MASTER MODE

In Multi-Master mode, the interrupt generation on the detection of the Start and Stop conditions allows the determination of when the bus is free. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSP module is disabled. Control of the I^2C bus may be taken when the P bit of the SSPxSTAT register is set, or the bus is Idle, with both the S and P bits clear. When the bus is busy, enabling the SSP interrupt will generate the interrupt when the Stop condition occurs.

In multi-master operation, the SDA line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed by hardware with the result placed in the BCL1IF bit.

The states where arbitration can be lost are:

- Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

31.6.13 MULTI -MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA, by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin is '0', then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCL1IF and reset the I^2C port to its Idle state (Figure 31-32).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are deasserted and the SSPxBUF can be written to. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

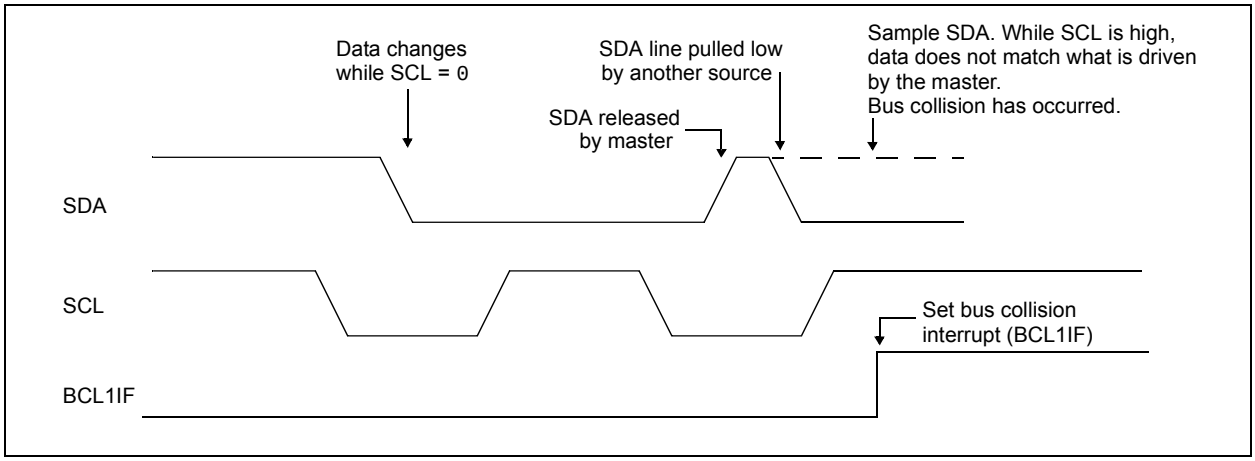
If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the SSPxCON2 register are cleared. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

The master will continue to monitor the SDA and SCL pins. If a Stop condition occurs, the SSPxIF bit will be set.

A write to the SSPxBUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when the bus collision occurred.

In Multi-Master mode, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I^2C bus can be taken when the P bit is set in the SSPxSTAT register, or the bus is Idle and the S and P bits are cleared.

FIGURE 31-32: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



32.6.4 HIGH AND LOW MEASURE MODE

This mode measures the high and low pulse time of the SMTSIGx relative to the SMT clock. It begins incrementing the SMTxTMR on a rising edge on the SMTSIGx input, then updates the SMTxCPW register with the value and resets the SMTxTMR on a falling edge, starting to increment again. Upon observing another rising edge, it updates the SMTxCPR register with its current value and once again resets the SMTxTMR value and begins incrementing again. See Figure 32-8 and Figure 32-9.

32.6.11 WINDOWED COUNTER MODE

This mode counts pulses on the SMTx_signal input, within a window dictated by the SMTxWIN input. It begins counting upon seeing a rising edge of the SMTxWIN input, updates the SMTxCPW register on a falling edge of the SMTxWIN input, and updates the SMTxCPR register on each rising edge of the SMTxWIN input beyond the first. See Figure 32-21 and Figure 32-22.

REGISTER 32-3: SMTxSTAT: SMT STATUS REGISTER

R/W/HC-0/0	R/W/HC-0/0	R/W/HC-0/0	U-0	U-0	R-0/0	R-0/0	R-0/0
CPRUP	CPWUP	RST	—	—	TS	WS	AS
bit 7							bit 0

Legend:		
HC = Bit is cleared by hardware		HS = Bit is set by hardware
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	q = Value depends on condition

bit 7 **CPRUP:** SMT Manual Period Buffer Update bit
1 = Request update to SMTxPRx registers
0 = SMTxPRx registers update is complete

bit 6 **CPWUP:** SMT Manual Pulse Width Buffer Update bit
1 = Request update to SMTxCPW registers
0 = SMTxCPW registers update is complete

bit 5 **RST:** SMT Manual Timer Reset bit
1 = Request Reset to SMTxTMR registers
0 = SMTxTMR registers update is complete

bit 4-3 **Unimplemented**: Read as ‘0’

bit 2 **TS:** SMT GO Value Status bit
1 = SMT timer is incrementing
0 = SMT timer is not incrementing

bit 1 **WS:** SMTxWIN Value Status bit
1 = SMT window is open
0 = SMT window is closed

bit 0 **AS:** SMT_signal Value Status bit
1 = SMT acquisition is in progress
0 = SMT acquisition is not in progress

REGISTER 32-5: SMTxWIN: SMT1 WINDOW INPUT SELECT REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	WSEL<4:0>				
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7-5 **Unimplemented**: Read as '0'

bit 4-0 **WSEL<4:0>:** SMTx Window Selection bits
11111 = Reserved
•
•
•
11000 = Reserved
10111 = LC4_out
10110 = LC3_out
10101 = LC2_out
10100 = LC1_out
10011 = ZCD1_output
10010 = C2OUT_sync
10001 = C1OUT_sync
10000 = PWM7_out
01111 = PWM6_out
01110 = CCP5_out
01101 = CCP4_out
01100 = CCP3_out
01011 = CCP2_out
01010 = CCP1_out
01001 = SMT2_match(1)
01000 = SMT1_match(1)
00111 = TMR6_postscaled
00110 = TMR4_postscaled
00101 = TMR2_postscaled
00100 = TMR0_overflow
00011 = SOSC
00010 = MFINTOSC/16
00001 = LFINTOSC
00000 = SMTxWINPPS

Note 1: The SMT_match corresponding to the SMT selected becomes reserved.

33.5 EUSART Operation During Sleep

The EUSART will remain active during Sleep only in the Synchronous Slave mode. All other modes require the system clock and therefore cannot generate the necessary signals to run the Transmit or Receive Shift registers during Sleep.

Synchronous Slave mode uses an externally generated clock to run the Transmit and Receive Shift registers.

33.5.1 SYNCHRONOUS RECEIVE DURING SLEEP

To receive during Sleep, all the following conditions must be met before entering Sleep mode:

- RC1STA and TX1STA Control registers must be configured for Synchronous Slave Reception (see **Section 33.4.2.4 "Synchronous Slave Reception Set-up:"**).
- If interrupts are desired, set the RCIE bit of the PIE3 register and the GIE and PEIE bits of the INTCON register.
- The RCIF interrupt flag must be cleared by reading RCREG to unload any pending characters in the receive buffer.

Upon entering Sleep mode, the device will be ready to accept data and clocks on the RX/DT and TX/CK pins, respectively. When the data word has been completely clocked in by the external device, the RCIF interrupt flag bit of the PIR3 register will be set. Thereby, waking the processor from Sleep.

Upon waking from Sleep, the instruction following the SLEEP instruction will be executed. If the Global Interrupt Enable (GIE) bit of the INTCON register is also set, then the Interrupt Service Routine at address 004h will be called.

33.5.2 SYNCHRONOUS TRANSMIT DURING SLEEP

To transmit during Sleep, all the following conditions must be met before entering Sleep mode:

- The RC1STA and TX1STA Control registers must be configured for synchronous slave transmission (see **Section 33.4.2.2 "Synchronous Slave Transmission Set-up:"**).
- The TXIF interrupt flag must be cleared by writing the output data to the TXREG, thereby filling the TSR and transmit buffer.
- If interrupts are desired, set the TXIE bit of the PIE3 register and the PEIE bit of the INTCON register.
- Interrupt enable bits TXIE of the PIE3 register and PEIE of the INTCON register must set.

Upon entering Sleep mode, the device will be ready to accept clocks on TX/CK pin and transmit data on the RX/DT pin. When the data word in the TSR has been completely clocked out by the external device, the pending byte in the TXREG will transfer to the TSR and the TXIF flag will be set. Thereby, waking the processor from Sleep. At this point, the TXREG is available to accept another character for transmission, which will clear the TXIF flag.

Upon waking from Sleep, the instruction following the SLEEP instruction will be executed. If the Global Interrupt Enable (GIE) bit is also set then the Interrupt Service Routine at address 0004h will be called.

Note: Unless otherwise noted, VIN = 5V, FOSC = 300 kHz, CIN = 0.1 µF, TA = 25°C.

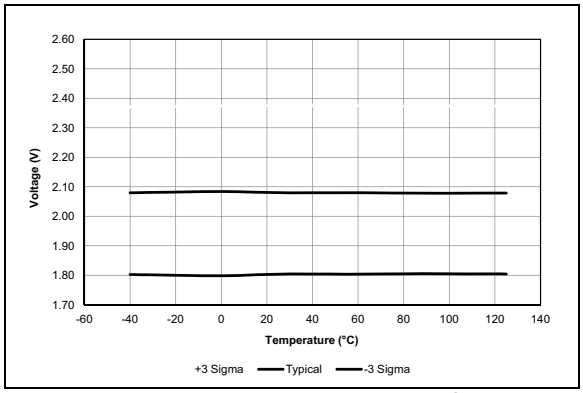


FIGURE 38-19: *LPBOR Reset Voltage, PIC16LF18856/76 Only.*

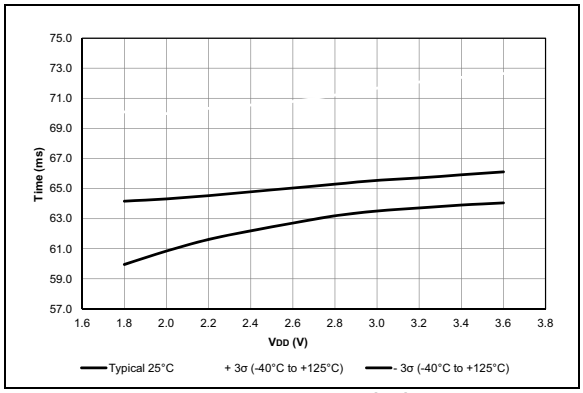


FIGURE 38-22: *PWRT Period, PIC16LF18856/76 Only.*

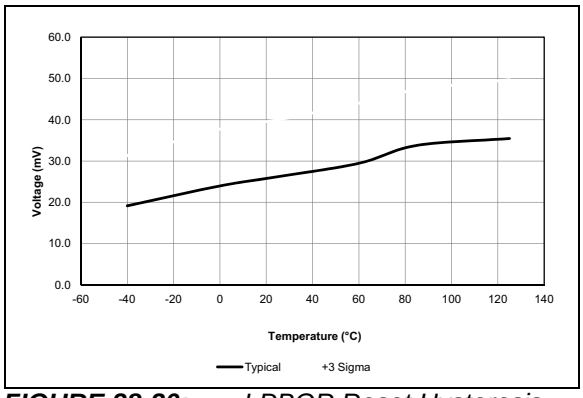


FIGURE 38-20: *LPBOR Reset Hysteresis, PIC16LF18856/76 Only.*

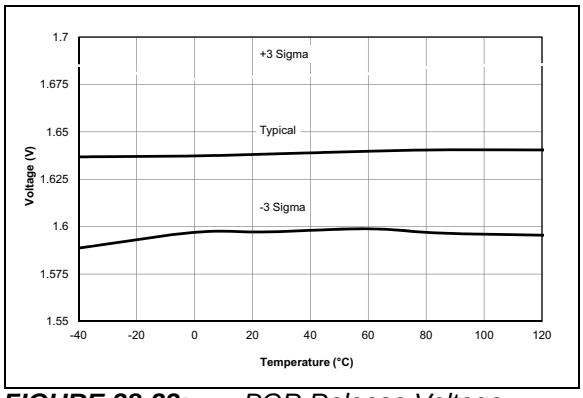


FIGURE 38-23: *POR Release Voltage.*

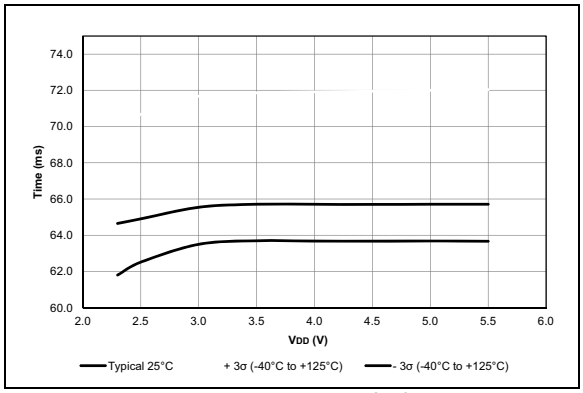


FIGURE 38-21: *PWRT Period, PIC16F18856/76 Only.*

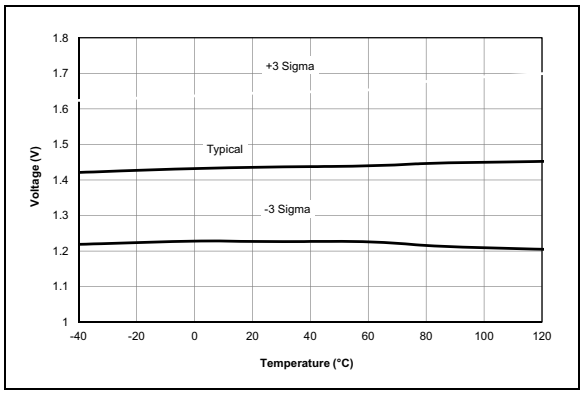


FIGURE 38-24: *POR Rearm Voltage, VREGPM1 = 0, PIC16F18856/76 Only.*

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A (01/2016)

Initial release of the document.

Revision B (04/2017)

Removed Preliminary Status - Added Char Graphs; Updated Figures 6-1, 23-2, 27-1, 28-1, 29-2, 29-3, 29-8, 29-9, 29-10, 29-11, 29-12, 29-13, 32-14, 32-15, 32-18, and 37-10; Registers 4-1, 4-3, 6-3, 8-2, 9-2, 12-2, 12-4, 12-6, 12-12, 12-14, 12-16, 12-32, 12-33, 12-34, 12-35, 12-36, 12-37, 12-43, 12-45, 12-49, 20-9, 23-1, 23-3, 23-4, 27-2, 28-1, 28-3, 29-1, 31-4, 31-5, 31-6, 34-1, and 34-2; Sections 9.1, 10.4.3, 21.5, 23.1.1, 23.1.4, 23.4.4, 23.5.2, 23.5.3, 29.1, 29.2, 31.6, 32.1.1, 32.6.9, 34.2, and 34.4; Tables 3-5, 3-6, 3-7, 10-2, 20-2, 23-1, 31-3, 36-4, 37-3, 37-5, 37-11 and 37-13.

Added Figure 37-11. Added Section 6.2.2.4 MFINTOSC, 21.5.1 Correction by AC Coupling. Added Section 28.4: Timer1 16-Bit Read/Write Mode.

Updated Instruction Sets MOVWF and NOP.

Removed Figure 37-11.

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Trademarks

ISBN: 978-1-5224-1628-9