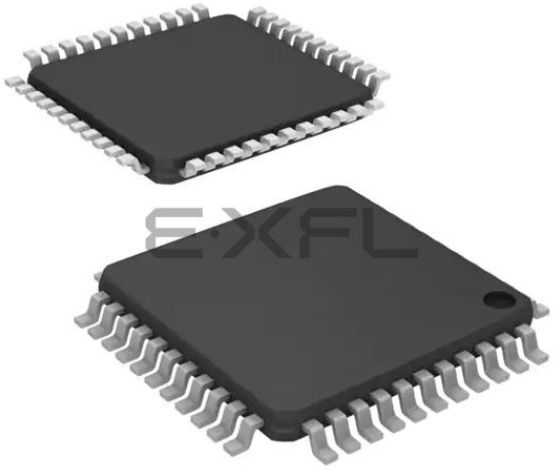


Welcome to E-XFL.COM

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	40MHz
Connectivity	CANbus, I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	96KB (48K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	3.25K x 8
Voltage - Supply (Vcc/Vdd)	4.2V ~ 5.5V
Data Converters	A/D 11x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f4685t-i-pt

MICROCHIP PIC18F2682/2685/4682/4685

28/40/44-Pin Enhanced Flash Microcontrollers with ECAN™ Technology, 10-Bit A/D and nanoWatt Technology

Power-Managed Modes:

- Run: CPU on, peripherals on
- Idle: CPU off, peripherals on
- Sleep: CPU off, peripherals off
- Idle mode currents down to 5.8 μA typical
- Sleep mode currents down to 0.1 μA typical
- Timer1 Oscillator: 1.1 μA, 32 kHz, 2V
- Watchdog Timer: 2.1 μA
- Two-Speed Oscillator Start-up

Flexible Oscillator Structure:

- Four Crystal modes, up to 40 MHz
- 4x Phase Lock Loop (PLL) – available for crystal and internal oscillators
- Two External RC modes, up to 4 MHz
- Two External Clock modes, up to 40 MHz
- Internal Oscillator Block:
  - 8 user-selectable frequencies, from 31 kHz to 8 MHz
  - Provides a complete range of clock speeds, from 31 kHz to 32 MHz when used with PLL
  - User-tunable to compensate for frequency drift
- Secondary Oscillator using Timer1 @ 32 kHz
- Fail-Safe Clock Monitor
  - Allows for safe shutdown if peripheral clock stops

Special Microcontroller Features:

- C compiler Optimized Architecture with optional Extended Instruction Set
- 100,000 Erase/Write Cycle Enhanced Flash Program Memory typical
- 1,000,000 Erase/Write Cycle Data EEPROM Memory typical
- Flash/Data EEPROM Retention: > 40 years
- Self-Programmable under Software Control
- Priority Levels for Interrupts
- 8 x 8 Single-Cycle Hardware Multiplier
- Extended Watchdog Timer (WDT):
  - Programmable period from 41 ms to 131s
- Single-Supply 5V In-Circuit Serial Programming™ (ICSP™) via two pins
- In-Circuit Debug (ICD) via two pins
- Wide operating voltage range: 2.0V to 5.5V

Peripheral Highlights:

- High-Current Sink/source 25 mA/25 mA
- Three External Interrupts
- One Capture/Compare/PWM (CCP1) module
- Enhanced Capture/Compare/PWM (ECCP1) module (40/44-pin devices only):
  - One, two or four PWM outputs
  - Selectable polarity
  - Programmable dead time
  - Auto-shutdown and auto-restart
- Master Synchronous Serial Port (MSSP) module supporting 3-Wire SPI (all 4 modes) and I^2C™ Master and Slave modes
- Enhanced Addressable USART module:
  - Supports RS-485, RS-232 and LIN 1.3
  - RS-232 operation using internal oscillator block (no external crystal required)
  - Auto-wake-up on Start bit
  - Auto-Baud Detect
- 10-Bit, up to 11-Channel Analog-to-Digital Converter module (A/D), up to 100 ksps:
  - Auto-acquisition capability
  - Conversion available during Sleep
- Dual Analog Comparators with Input Multiplexing

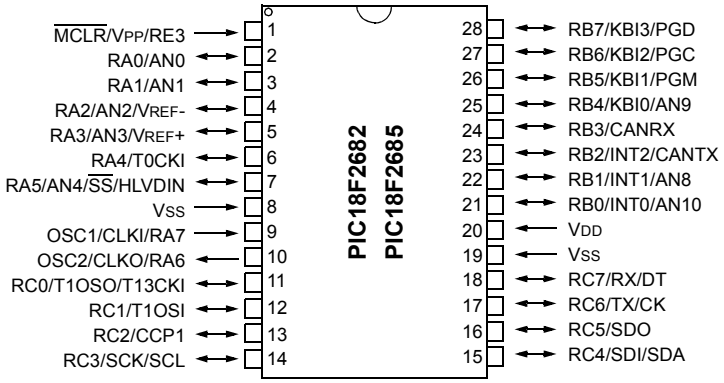
ECAN Module Features:

- Message bit rates up to 1 Mbps
- Conforms to CAN 2.0B ACTIVE Specification
- Fully Backward Compatible with PIC18XXX8 CAN modules
- Three Modes of Operation:
  - Legacy, Enhanced Legacy, FIFO
- Three Dedicated Transmit Buffers with Prioritization
- Two Dedicated Receive Buffers
- Six Programmable Receive/Transmit Buffers
- Three Full, 29-Bit Acceptance Masks
- 16 Full, 29-Bit Acceptance Filters w/Dynamic Association
- DeviceNet™ Data Byte Filter Support
- Automatic Remote Frame Handling
- Advanced Error Management Features

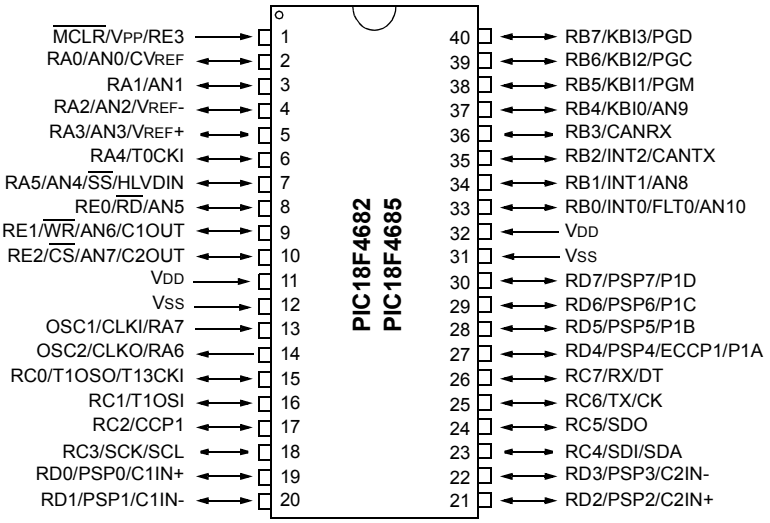
Device	Program Memory		Data Memory		I/O	10-Bit A/D (ch)	CCP1/ ECCP1 (PWM)	MSSP		EUSART	Comp.	Timers 8/16-bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)				SPI	Master I^2C™			
PIC18F2682	80K	40960	3328	1024	28	8	1/0	Y	Y	1	0	1/3
PIC18F2685	96K	49152	3328	1024	28	8	1/0	Y	Y	1	0	1/3
PIC18F4682	80K	40960	3328	1024	40/44	11	1/1	Y	Y	1	2	1/3
PIC18F4685	96K	49152	3328	1024	40/44	11	1/1	Y	Y	1	2	1/3

Pin Diagrams

28-Pin PDIP, SOIC



40-Pin PDIP



Note: Pinouts are subject to change.

2.7 Clock Sources and Oscillator Switching

Like previous PIC18 devices, the PIC18F2682/2685/4682/4685 family includes a feature that allows the device clock source to be switched from the main oscillator to an alternate low-frequency clock source. PIC18F2682/2685/4682/4685 devices offer two alternate clock sources. When an alternate clock source is enabled, the various power-managed operating modes are available.

Essentially, there are three clock sources for these devices:

- Primary oscillators
- Secondary oscillators
- Internal oscillator block

The **primary oscillators** include the External Crystal and Resonator modes, the External RC modes, the External Clock modes and the internal oscillator block. The particular mode is defined by the FOSC3:FOSC0 Configuration bits. The details of these modes are covered earlier in this chapter.

The **secondary oscillators** are those external sources not connected to the OSC1 or OSC2 pins. These sources may continue to operate even after the controller is placed in a power-managed mode.

PIC18F2682/2685/4682/4685 devices offer the Timer1 oscillator as a secondary oscillator. In all power-managed modes, this oscillator is often the time base for functions such as a Real-Time Clock.

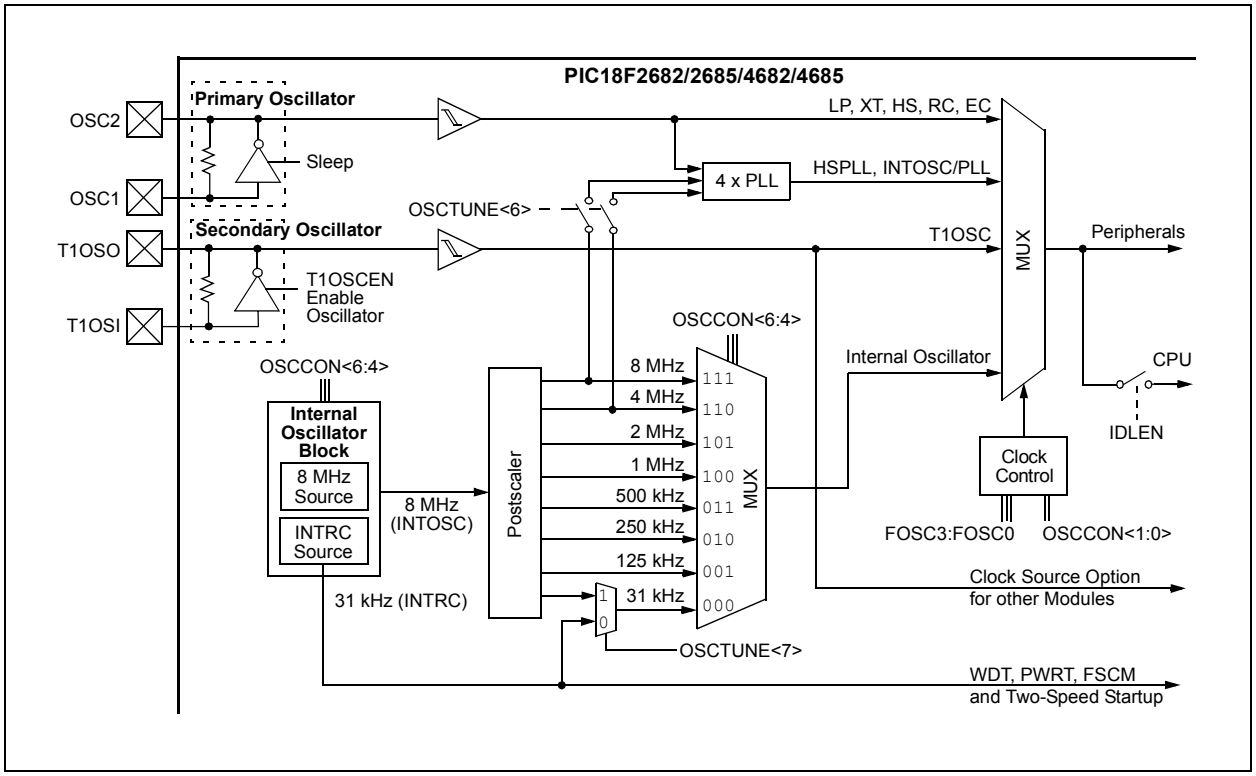
Most often, a 32.768 kHz watch crystal is connected between the RC0/T1OSO/T13CKI and RC1/T1OSI pins. Like the LP mode oscillator circuit, loading capacitors are also connected from each pin to ground.

The Timer1 oscillator is discussed in greater detail in **Section 12.3 "Timer1 Oscillator"**.

In addition to being a primary clock source, the **internal oscillator block** is available as a power-managed mode clock source. The INTRC source is also used as the clock source for several special features, such as the WDT and Fail-Safe Clock Monitor.

The clock sources for the PIC18F2682/2685/4682/4685 devices are shown in Figure 2-8. See **Section 24.0 "Special Features of the CPU"** for Configuration register details.

FIGURE 2-8: PIC18F2682/2685/4682/4685 CLOCK DIAGRAM



NOTES:

TABLE 4-4: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)

Register	Applicable Devices				Power-on Reset, Brown-out Reset	$\overline{\text{MCLR}}$ Resets, WDT Reset, RESET Instruction, Stack Resets	Wake-up via WDT or Interrupt
RXB1SIDH	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
RXB1CON	2682	2685	4682	4685	000- 0000	000- 0000	uuu- uuuu
TXB0D7	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0D6	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0D5	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0D4	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0D3	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0D2	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0D1	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0D0	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0DLC	2682	2685	4682	4685	-x-- xxxx	-u-- uuuu	-u-- uuuu
TXB0EIDL	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0EIDH	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0SIDL	2682	2685	4682	4685	xxx- x-xx	uuu- u-uu	uuu- u-uu
TXB0SIDH	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB0CON	2682	2685	4682	4685	0000 0-00	0000 0-00	uuuu u-uu
TXB1D7	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D6	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D5	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D4	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D3	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D2	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D1	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1D0	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1DLC	2682	2685	4682	4685	-x-- xxxx	-u-- uuuu	-u-- uuuu
TXB1EIDL	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1EIDH	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1SIDL	2682	2685	4682	4685	xxx- x-xx	uuu- u-uu	uuu- uu-u
TXB1SIDH	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXB1CON	2682	2685	4682	4685	0000 0-00	0000 0-00	uuuu u-uu
TXB2D7	2682	2685	4682	4685	xxxx xxxx	uuuu uuuu	0uuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition. Shaded cells indicate conditions do not apply for the designated device.

Note 1: One or more bits in the INTCONx or PIRx registers will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the PC is loaded with the interrupt vector (0008h or 0018h).

3: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the TOSU, TOSH and TOSL are updated with the current value of the PC. The STKPTR is modified to point to the next location in the hardware stack.

4: See Table 4-3 for Reset value for specific condition.

5: Bits 6 and 7 of PORTA, LATA and TRISA are enabled depending on the oscillator mode selected. When not enabled as PORTA pins, they are disabled and read '0'.

6: This register reads all '0's until the ECAN™ technology is set up in Mode 1 or Mode 2.

5.0 MEMORY ORGANIZATION

There are three types of memory in PIC18 Enhanced microcontroller devices:

- Program Memory
- Data RAM
- Data EEPROM

As Harvard architecture devices, the data and program memories use separate busses; this allows for concurrent access of the two memory spaces. The data EEPROM, for practical purposes, can be regarded as a peripheral device, since it is addressed and accessed through a set of control registers.

Additional detailed information on the operation of the Flash program memory is provided in **Section 6.0 "Flash Program Memory"**. Data EEPROM is discussed separately in **Section 7.0 "Data EEPROM Memory"**.

5.1 Program Memory Organization

PIC18 microcontrollers implement a 21-bit program counter, which is capable of addressing a 2-Mbyte program memory space. Accessing a location between the upper boundary of the physically implemented memory and the 2-Mbyte address will return all '0's (a NOP instruction).

The PIC18F2682 and PIC18F4682 each have 80 Kbytes of Flash memory and can store up to 40,960 single-word instructions. The PIC18F2685 and PIC18F4685 each have 96 Kbytes of Flash memory and can store up to 49,152 single-word instructions.

PIC18 devices have two interrupt vectors. The Reset vector address is at 0000h and the interrupt vector addresses are at 0008h and 0018h.

The program memory maps for PIC18F2682/4682 and PIC18F2685/4685 devices are shown in Figure 5-1.

FIGURE 5-1: PROGRAM MEMORY MAP AND STACK FOR PIC18F2682/2685/4682/4685 DEVICES

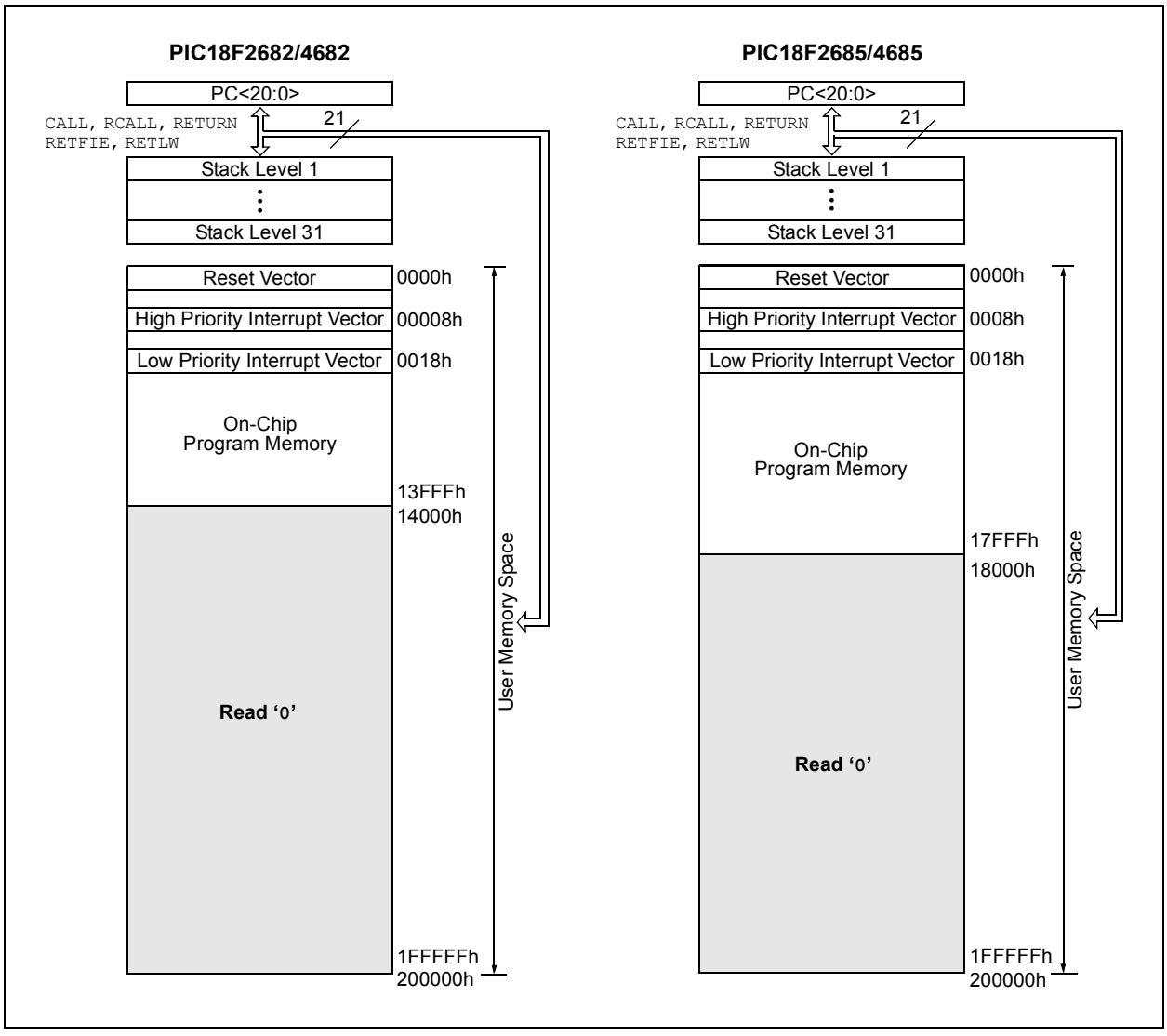


TABLE 5-1: SPECIAL FUNCTION REGISTER MAP FOR PIC18F2682/2685/4682/4685 DEVICES (CONTINUED)

Address	Name	Address	Name	Address	Name	Address	Name
EFFh	—	EDFh	—	EBFh	—	E9Fh	—
EFEh	—	EDEh	—	EBEh	—	E9Eh	—
EFDh	—	EDDh	—	EBDh	—	E9Dh	—
EFCh	—	EDCh	—	EBCh	—	E9Ch	—
EFBh	—	EDBh	—	EBBh	—	E9Bh	—
EFAh	—	EDAh	—	EBAh	—	E9Ah	—
EF9h	—	ED9h	—	EB9h	—	E99h	—
EF8h	—	ED8h	—	EB8h	—	E98h	—
EF7h	—	ED7h	—	EB7h	—	E97h	—
EF6h	—	ED6h	—	EB6h	—	E96h	—
EF5h	—	ED5h	—	EB5h	—	E95h	—
EF4h	—	ED4h	—	EB4h	—	E94h	—
EF3h	—	ED3h	—	EB3h	—	E93h	—
EF2h	—	ED2h	—	EB2h	—	E92h	—
EF1h	—	ED1h	—	EB1h	—	E91h	—
EF0h	—	ED0h	—	EB0h	—	E90h	—
EEFh	—	ECFh	—	EAFh	—	E8Fh	—
EEEh	—	ECEh	—	EAEh	—	E8Eh	—
EEDh	—	ECDh	—	EADh	—	E8Dh	—
EECh	—	ECCh	—	EACh	—	E8Ch	—
EEBh	—	ECBh	—	EABh	—	E8Bh	—
EEAh	—	ECAh	—	EAAh	—	E8Ah	—
EE9h	—	EC9h	—	EA9h	—	E89h	—
EE8h	—	EC8h	—	EA8h	—	E88h	—
EE7h	—	EC7h	—	EA7h	—	E87h	—
EE6h	—	EC6h	—	EA6h	—	E86h	—
EE5h	—	EC5h	—	EA5h	—	E85h	—
EE4h	—	EC4h	—	EA4h	—	E84h	—
EE3h	—	EC3h	—	EA3h	—	E83h	—
EE2h	—	EC2h	—	EA2h	—	E82h	—
EE1h	—	EC1h	—	EA1h	—	E81h	—
EE0h	—	EC0h	—	EA0h	—	E80h	—

Note 1: Registers available only on PIC18F4X8X devices; otherwise, the registers read as '0'.
2: When any TX_ENn bit in RX_TX_SELn is set, then the corresponding bit in this register has transmit properties.
3: This is not a physical register.

TABLE 5-2: REGISTER FILE SUMMARY (PIC18F2682/2685/4682/4685) (CONTINUED)

File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Details on page:
TXB1CON	TXBIF	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI1	TXPRI0	0000 0-00	56, 284
TXB2D7	TXB2D77	TXB2D76	TXB2D75	TXB2D74	TXB2D73	TXB2D72	TXB2D71	TXB2D70	xxxx xxxx	56, 286
TXB2D6	TXB2D67	TXB2D66	TXB2D65	TXB2D64	TXB2D63	TXB2D62	TXB2D61	TXB2D60	xxxx xxxx	57, 286
TXB2D5	TXB2D57	TXB2D56	TXB2D55	TXB2D54	TXB2D53	TXB2D52	TXB2D51	TXB2D50	xxxx xxxx	57, 286
TXB2D4	TXB2D47	TXB2D46	TXB2D45	TXB2D44	TXB2D43	TXB2D42	TXB2D41	TXB2D40	xxxx xxxx	57, 286
TXB2D3	TXB2D37	TXB2D36	TXB2D35	TXB2D34	TXB2D33	TXB2D32	TXB2D31	TXB2D30	xxxx xxxx	57, 286
TXB2D2	TXB2D27	TXB2D26	TXB2D25	TXB2D24	TXB2D23	TXB2D22	TXB2D21	TXB2D20	xxxx xxxx	57, 286
TXB2D1	TXB2D17	TXB2D16	TXB2D15	TXB2D14	TXB2D13	TXB2D12	TXB2D11	TXB2D10	xxxx xxxx	57, 286
TXB2D0	TXB2D07	TXB2D06	TXB2D05	TXB2D04	TXB2D03	TXB2D02	TXB2D01	TXB2D00	xxxx xxxx	57, 286
TXB2DLC	—	TXRTR	—	—	DLC3	DLC2	DLC1	DLC0	-x-- xxxx	57, 287
TXB2EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	57, 286
TXB2EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	57, 285
TXB2SIDL	SID2	SID1	SID0	—	EXIDE	—	EID17	EID16	xxxx x-xx	57, 285
TXB2SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxx- x-xx	57, 285
TXB2CON	TXBIF	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI1	TXPRI0	0000 0-00	57, 284
RXM1EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	57, 307
RXM1EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	57, 307
RXM1SIDL	SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16	xxx- x-xx	57, 307
RXM1SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	57, 306
RXM0EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	57, 307
RXM0EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	57, 307
RXM0SIDL	SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16	xxx- x-xx	57, 307
RXM0SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	57, 306
RXF5EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	57, 306
RXF5EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	57, 306
RXF5SIDL	SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16	xxx- x-xx	57, 305
RXF5SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	57, 306
RXF4EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	57, 306
RXF4EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	57, 306
RXF4SIDL	SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16	xxx- x-xx	57, 305
RXF4SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	57, 306
RXF3EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	57, 306
RXF3EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	57, 306
RXF3SIDL	SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16	xxx- x-xx	58, 305
RXF3SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	58, 306
RXF2EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	58, 306
RXF2EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	58, 306
RXF2SIDL	SID2	SID1	SID0	—	EXIDEN	—	EID17	EID16	xxx- x-xx	58, 305
RXF2SIDH	SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3	xxxx xxxx	58, 306
RXF1EIDL	EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	xxxx xxxx	58, 306
RXF1EIDH	EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	xxxx xxxx	58, 306

Legend: x = unknown, u = unchanged, - = unimplemented, q = value depends on condition. Shaded cells are unimplemented, read as '0'.

Note 1: Bit 21 of the PC is only available in Test mode and Serial Programming modes.

2: The SBOREN bit is only available when CONFIG2L<1:0> = 01; otherwise, it is disabled and reads as '0'. See **Section 4.4 "Brown-out Reset (BOR)"**.

3: These registers and/or bits are not implemented on PIC18F2682/2685 devices and are read as '0'. Reset values are shown for PIC18F4682/4685 devices; individual unimplemented bits should be interpreted as '—'.

4: The PLLEN bit is only available in specific oscillator configurations; otherwise, it is disabled and reads as '0'. See **Section 2.6.4 "PLL in INTOSC Modes"**.

5: The RE3 bit is only available when Master Clear Reset is disabled (CONFIG3H<7> = 0); otherwise, RE3 reads as '0'. This bit is read-only.

6: RA6/RA7 and their associated latch and direction bits are individually configured as port pins based on various primary oscillator modes. When disabled, these bits read as '0'.

7: CAN bits have multiple functions depending on the selected mode of the CAN module.

8: This register reads all '0's until the ECAN™ technology is set up in Mode 1 or Mode 2.

9: These registers and/or bits are available on PIC18F4682/4685 devices only.

TABLE 10-3: PORTB I/O SUMMARY

Pin Name	Function	I/O	TRIS	Buffer	Description
RB0/INT0/FLT0/AN10	RB0	OUT	0	DIG	LATB<0> data output.
		IN	1	TTL	PORTB<0> data input. Weak pull-up available only in this mode.
	INT0	IN	1	ST	External interrupt 0 input.
	FLT0(1)	IN	1	ST	Enhanced PWM Fault input.
	AN10	IN	1	ANA	A/D input channel 10. Enabled on POR, this analog input overrides the digital input (read as clear – low level).
RB1/INT1/AN8	RB1	OUT	0	DIG	LATB<1> data output.
		IN	1	TTL	PORTB<1> data input. Weak pull-up available only in this mode.
	INT1	IN	1	ST	External interrupt 1 input.
	AN8	IN	1	ANA	A/D input channel 8. Enabled on POR, this analog input overrides the digital input (read as clear – low level).
RB2/INT2/CANTX	RB2	OUT	x	DIG	LATB<2> data output.
		IN	1	TTL	PORTB<2> data input. Weak pull-up available only in this mode.
	INT2	IN	1	ST	External interrupt 2 input.
	CANTX	OUT	1	DIG	CAN transmit signal output. The CAN interface overrides the TRIS<2> control when enabled.
RB3/CANRX	RB3	OUT	0	DIG	LATB<3> data output.
		IN	1	TTL	PORTB<3> data input. Weak pull-up available only in this mode.
	CANRX	IN	1	ST	CAN receive signal input. Pin must be configured as a digital input by setting TRISB<3>.
RB4/KBI0/AN9	RB4	OUT	0	DIG	LATB<4> data output.
		IN	1	TTL	PORTB<4> data input. Weak pull-up available only in this mode.
	KBI0	IN	1	TTL	Interrupt-on-pin change.
	AN9	IN	1	ANA	A/D input channel 9. Enabled on POR, this analog input overrides the digital input (read as clear – low level).
RB5/KBI1/PGM	RB5	OUT	0	DIG	LATB<5> data output.
		IN	1	TTL	PORTB<5> data input. Weak pull-up available only in this mode.
	KBI1	IN	1	TTL	Interrupt-on-pin change.
	PGM	IN	x	ST	Low-Voltage Programming mode entry (ICSP™). Enabling this function overrides digital output.
RB6/KBI2/PGC	RB6	OUT	0	DIG	LATB<6> data output.
		IN	1	TTL	PORTB<6> data input. Weak pull-up available only in this mode.
	KBI2	IN	1	TTL	Interrupt-on-pin change.
	PGC	IN	x	ST	Low-Voltage Programming mode entry (ICSP) clock input.
RB7/KBI3/PGD	RB7	OUT	0	DIG	LATB<7> data output.
		IN	1	TTL	PORTB<7> data input. Weak pull-up available only in this mode.
	KBI3	IN	1	TTL	Interrupt-on-pin change.
	PGD	OUT	x	DIG	Low-Voltage Programming mode entry (ICSP) clock output.
		IN	x	ST	Low-Voltage Programming mode entry (ICSP) clock input.

Legend: OUT = Output; IN = Input; ANA = Analog Signal; DIG = Digital Output; ST = Schmitt Buffer Input; TTL – TTL Buffer Input

Note 1: This bit is unimplemented on PIC18F2682/2685 devices.

11.3 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module. The prescaler is not directly readable or writable; its value is set by the PSA and T0PS2:T0PS0 bits (T0CON<3:0>) which determine the prescaler assignment and prescale ratio.

Clearing the PSA bit assigns the prescaler to the Timer0 module. When it is assigned, prescale values from 1:2 through 1:256 in power-of-2 increments are selectable.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., `CLRF TMR0`, `MOVWF TMR0`, `BSF TMR0`, etc.) clear the prescaler count.

Note:	Writing to TMR0 when the prescaler is assigned to Timer0 will clear the prescaler count but will not change the prescaler assignment.

11.3.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control and can be changed “on-the-fly” during program execution.

11.4 Timer0 Interrupt

The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h in 8-bit mode, or from FFFFh to 0000h in 16-bit mode. This overflow sets the TMR0IF flag bit. The interrupt can be masked by clearing the TMR0IE bit (INTCON<5>). Before re-enabling the interrupt, the TMR0IF bit must be cleared in software by the Interrupt Service Routine.

Since Timer0 is shut down in Sleep mode, the TMR0 interrupt cannot awaken the processor from Sleep.

TABLE 11-1: REGISTERS ASSOCIATED WITH TIMER0

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
TMR0L	Timer0 Register Low Byte								52
TMR0H	Timer0 Register High Byte								52
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	51
T0CON	TMR0ON	T08BIT	T0CS	T0SE	PSA	T0PS2	T0PS1	T0PS0	52
TRISA	TRISA7(1)	TRISA6(1)	PORTA Data Direction Register						54

Legend: `x` = unknown, `u` = unchanged, — = unimplemented locations, read as ‘0’. Shaded cells are not used by Timer0.

Note 1: RA7:RA6 and their associated latch and data direction bits are enabled as I/O pins based on oscillator configuration; otherwise, they are read as ‘0’.

15.4 PWM Mode

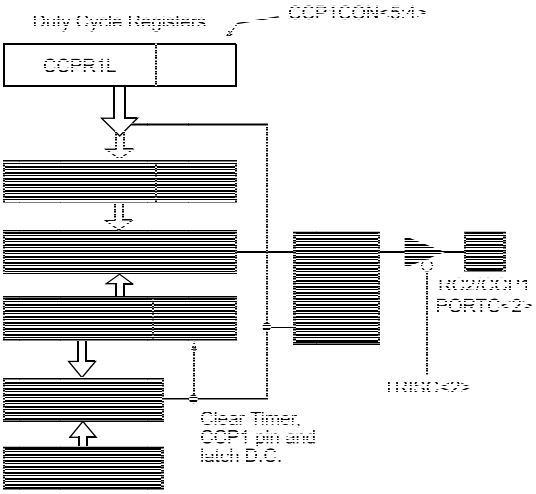
In Pulse-Width Modulation (PWM) mode, the CCP1 pin produces up to a 10-bit resolution PWM output. Since the CCP1 pin is multiplexed with a PORTB or PORTC data latch, the appropriate TRIS bit must be cleared to make the CCP1 pin an output.

Note:	Clearing the CCP1CON register will force the RC2 output latch to the default low level. This is not the PORTC I/O data latch.

Figure 15-3 shows a simplified block diagram of the CCP1 module in PWM mode.

For a step-by-step procedure on how to set up the CCP1 module for PWM operation, see **Section 15.4.4 "Setup for PWM Operation"**.

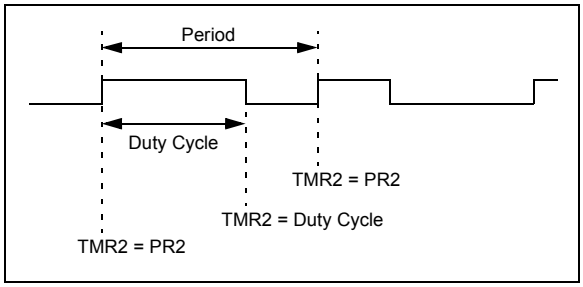
FIGURE 15-3: SIMPLIFIED PWM BLOCK DIAGRAM



Note 1: The 8-bit TMR2 value is concatenated with the 2-bit internal Q clock, or 2 bits of the prescaler, to create the 10-bit time base.

A PWM output (Figure 15-4) has a time base (period) and a time that the output stays high (duty cycle). The frequency of the PWM is the inverse of the period (1/period).

FIGURE 15-4: PWM OUTPUT



15.4.1 PWM PERIOD

The PWM period is specified by writing to the PR2 (PR4) register. The PWM period can be calculated using the following formula.

EQUATION 15-1:

$$\text{PWM Period} = [(PR2) + 1] \cdot 4 \cdot T_{OSC} \cdot (\text{TMR2 Prescale Value})$$

PWM frequency is defined as 1/[PWM period].

When TMR1 (TMR3) is equal to PR2 (PR4), the following three events occur on the next increment cycle:

- TMR2 is cleared
- The CCP1 pin is set (exception: if PWM duty cycle = 0%, the CCP1 pin will not be set)
- The PWM duty cycle is latched from CCPR1L into CCPR1H

Note:	The Timer2 postscaler (see **Section 13.0 "Timer2 Module"**) is not used in the determination of the PWM frequency. The postscaler could be used to have a servo update rate at a different frequency than the PWM output.

15.4.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the CCPR1L register and to the CCP1CON<5:4> bits. Up to 10-bit resolution is available. The CCPR1L contains the eight MSbs and the CCP1CON<5:4> contains the two LSbs. This 10-bit value is represented by CCPR1L:CCP1CON<5:4>. The following equation is used to calculate the PWM duty cycle in time.

EQUATION 15-2:

$$\text{PWM Duty Cycle} = (\text{CCPR1L:CCP1CON<5:4>}) \cdot T_{OSC} \cdot (\text{TMR2 Prescale Value})$$

CCPR1L and CCP1CON<5:4> can be written to at any time, but the duty cycle value is not latched into CCPR1H until after a match between PR2 and TMR2 occurs (i.e., the period is complete). In PWM mode, CCPR1H is a read-only register.

REGISTER 19-3: ADCON2: A/D CONTROL REGISTER 2

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	—	ACQT2	ACQT1	ACQT0	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 7 **ADFM:** A/D Result Format Select bit
1 = Right justified
0 = Left justified

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

bit 5-3 **ACQT2:ACQT0:** A/D Acquisition Time Select bits
111 = 20 TAD
110 = 16 TAD
101 = 12 TAD
100 = 8 TAD
011 = 6 TAD
010 = 4 TAD
001 = 2 TAD
000 = 0 TAD(1)

bit 2-0 **ADCS2:ADCS0:** A/D Conversion Clock Select bits
111 = FRC (clock derived from A/D RC oscillator)(1)
110 = FOSC/64
101 = FOSC/16
100 = FOSC/4
011 = FRC (clock derived from A/D RC oscillator)(1)
010 = FOSC/32
001 = FOSC/8
000 = FOSC/2

Note 1: If the A/D FRC clock source is selected, a delay of one TCY (instruction cycle) is added before the A/D clock starts. This allows the SLEEP instruction to be executed before starting a conversion.

REGISTER 23-24: BnSIDH: TX/RX BUFFER n STANDARD IDENTIFIER REGISTERS, HIGH BYTE IN RECEIVE MODE [0 ≤ n ≤ 5, TXnEN (BSEL0<n>) = 0](1)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
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 7-0 **SID10:SID3:** Standard Identifier bits (if EXIDE (BnSIDL<3>) = 0)
Extended Identifier bits EID28:EID21 (if EXIDE = 1).

Note 1: These registers are available in Mode 1 and 2 only.

REGISTER 23-25: BnSIDH: TX/RX BUFFER n STANDARD IDENTIFIER REGISTERS, HIGH BYTE IN TRANSMIT MODE [0 ≤ n ≤ 5, TXnEN (BSEL0<n>) = 1](1)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
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 7-0 **SID10:SID3:** Standard Identifier bits (if EXIDE (BnSIDL<3>) = 0)
Extended Identifier bits EID28:EID21 (if EXIDE = 1).

Note 1: These registers are available in Mode 1 and 2 only.

BCF **Bit Clear f**

Syntax: BCF f, b {,a}

Operands: $0 \le f \le 255$
$0 \le b \le 7$
$a \in [0,1]$

Operation: $0 \rightarrow f<b>$

Status Affected: None

Encoding:

1001	bbba	ffff	ffff

Description: Bit 'b' in register 'f' is cleared.

If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default).

If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See **Section 25.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write register 'f'

Example: `BCF FLAG_REG, 7, 0`

Before Instruction
FLAG_REG = C7h
After Instruction
FLAG_REG = 47h

BN **Branch if Negative**

Syntax: BN n

Operands: $-128 \le n \le 127$

Operation: if Negative bit is '1'
$(PC) + 2 + 2n \rightarrow PC$

Status Affected: None

Encoding:

1110	0110	nnnn	nnnn

Description: If the Negative bit is '1', then the program will branch.

The 2's complement number '2n' is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

Example:
```
HERE    BN   Jump
```

Before Instruction
PC = address (HERE)
After Instruction
If Negative = 1;
PC = address (Jump)
If Negative = 0;
PC = address (HERE + 2)

BNC **Branch if Not Carry**

Syntax: BNC n

Operands: -128 ≤ n ≤ 127

Operation: if Carry bit is '0'
(PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1110	0011	nnnn	nnnn

Description: If the Carry bit is '0', then the program will branch.
The 2's complement number '2n' is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

Example: HERE BNC Jump

Before Instruction
PC = address (HERE)
After Instruction
If Carry = 0;
PC = address (Jump)
If Carry = 1;
PC = address (HERE + 2)

BNN **Branch if Not Negative**

Syntax: BNN n

Operands: -128 ≤ n ≤ 127

Operation: if Negative bit is '0'
(PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1110	0111	nnnn	nnnn

Description: If the Negative bit is '0', then the program will branch.
The 2's complement number '2n' is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

Example: HERE BNN Jump

Before Instruction
PC = address (HERE)
After Instruction
If Negative = 0;
PC = address (Jump)
If Negative = 1;
PC = address (HERE + 2)

RCALL **Relative Call**

Syntax: RCALL n

Operands: -1024 ≤ n ≤ 1023

Operation: (PC) + 2 → TOS,
(PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1101	1nnn	nnnn	nnnn

Description: Subroutine call with a jump up to 1K from the current location. First, return address (PC + 2) is pushed onto the stack. Then, add the 2's complement number '2n' to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is a two-cycle instruction.

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'n' Push PC to stack	Process Data	Write to PC
No operation	No operation	No operation	No operation

Example: `HERE RCALL Jump`

Before Instruction
PC = Address `(HERE)`
After Instruction
PC = Address `(Jump)`
TOS = Address `(HERE + 2)`

RESET **Reset**

Syntax: RESET

Operands: None

Operation: Reset all registers and flags that are affected by a $\overline{\text{MCLR}}$ Reset.

Status Affected: All

Encoding:

0000	0000	1111	1111

Description: This instruction provides a way to execute a $\overline{\text{MCLR}}$ Reset in software.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Start Reset	No operation	No operation

Example: `RESET`

After Instruction
Registers = Reset Value
Flags* = Reset Value

25.2 Extended Instruction Set

In addition to the standard 75 instructions of the PIC18 instruction set, PIC18F2682/2685/4682/4685 devices also provide an optional extension to the core CPU functionality. The added features include eight additional instructions that augment indirect and indexed addressing operations and the implementation of Indexed Literal Offset Addressing mode for many of the standard PIC18 instructions.

The additional features are disabled by default. To enable them, users must set the XINST Configuration bit.

The instructions in the extended set can all be classified as literal operations, which either manipulate the File Select Registers or use them for indexed addressing. Two of the instructions, `ADDFSR` and `SUBFSR`, each have an additional special instantiation for using FSR2. These versions (`ADDULNK` and `SUBULNK`) allow for automatic return after execution.

The extended instructions are specifically implemented to optimize re-entrant program code (that is, code that is recursive or that uses a software stack) written in high-level languages, particularly C. Among other things, they allow users working in high-level languages to perform certain operations on data structures more efficiently. These include:

- Dynamic allocation and deallocation of software stack space when entering and leaving subroutines
- Function Pointer invocation
- Software Stack Pointer manipulation
- Manipulation of variables located in a software stack

A summary of the instructions in the extended instruction set is provided in Table 25-3. Detailed descriptions are provided in **Section 25.2.2 "Extended Instruction Set"**. The opcode field descriptions in Table 25-1 apply to both the standard and extended PIC18 instruction sets.

Note:	The instruction set extension and the Indexed Literal Offset Addressing mode were designed for optimizing applications written in C; the user may likely never use these instructions directly in assembler. The syntax for these commands is provided as a reference for users who may be reviewing code that has been generated by a compiler.

25.2.1 EXTENDED INSTRUCTION SYNTAX

Most of the extended instructions use indexed arguments, using one of the File Select Registers and some offset to specify a source or destination register. When an argument for an instruction serves as part of indexed addressing, it is enclosed in square brackets ("[ ]"). This is done to indicate that the argument is used as an index or offset. MPASM™ Assembler will flag an error if it determines that an index or offset value is not bracketed.

When the extended instruction set is enabled, brackets are also used to indicate index arguments in bit-oriented and byte-oriented instructions. This is in addition to other changes in their syntax. For more details, see **Section 25.2.3.1 "Extended Instruction Syntax with Standard PIC18 Commands"**.

Note:	In the past, square brackets have been used to denote optional arguments in the PIC18 and earlier instruction sets. In this text and going forward, optional arguments are denoted by braces ("{ }").

TABLE 25-3: EXTENSIONS TO THE PIC18 INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	16-Bit Instruction Word				Status Affected
				MSb			LSb	
ADDFSR	f, k	Add literal to FSR	1	1110	1000	ffkk	kkkk	None
ADDULNK	k	Add literal to FSR2 and return	2	1110	1000	11kk	kkkk	None
CALLW		Call subroutine using WREG	2	0000	0000	0001	0100	None
MOVSF	z_s, f_d	Move z_s (source) to 1st word	2	1110	1011	0zzz	zzzz	None
		f_d (destination) 2nd word		1111	ffff	ffff	ffff	
MOVSS	z_s, z_d	Move z_s (source) to 1st word	2	1110	1011	1zzz	zzzz	None
		z_d (destination) 2nd word		1111	xxxx	xzzz	zzzz	
PUSHL	k	Store literal at FSR2, decrement FSR2	1	1110	1010	kkkk	kkkk	None
SUBFSR	f, k	Subtract literal from FSR	1	1110	1001	ffkk	kkkk	None
SUBULNK	k	Subtract literal from FSR2 and return	2	1110	1001	11kk	kkkk	None

27.2 DC Characteristics: Power-Down and Supply Current PIC18F2682/2685/4682/4685 (Industrial) PIC18LF2682/2685/4682/4685 (Industrial) (Continued)

PIC18LF2682/2685/4682/4685 (Industrial)		**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial					
PIC18F2682/2685/4682/4685 (Industrial, Extended)		**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	**Device**	**Typ**	**Max**	**Units**	**Conditions**		
	Supply Current (IDD)(2,3)						
	PIC18LF268X/468X	65	220	µA	-40°C	VDD = 2.0V	FOSC = 1 MHz (**PRI_IDLE** mode, EC oscillator)
		65	220	µA	+25°C		
		70	220	µA	+85°C		
	PIC18LF268X/468X	120	330	µA	-40°C	VDD = 3.0V	
		120	330	µA	+25°C		
		130	330	µA	+85°C		
	All devices	300	600	µA	-40°C	VDD = 5.0V	
		240	600	µA	+25°C		
		300	600	µA	+85°C		
	Extended devices only	320	600	µA	+125°C		
	PIC18LF268X/468X	260	760	µA	-40°C	VDD = 2.0V	FOSC = 4 MHz (**PRI_IDLE** mode, EC oscillator)
		255	760	µA	+25°C		
		270	760	µA	+85°C		
	PIC18LF268X/468X	420	1.4	µA	-40°C	VDD = 3.0V	
		430	1.4	µA	+25°C		
		450	1.4	µA	+85°C		
	All devices	0.9	2.2	mA	-40°C	VDD = 5.0V	
		0.9	2.2	mA	+25°C		
		0.9	2.2	mA	+85°C		
	Extended devices only	1	3	mA	+125°C		
	Extended devices only	2.8	7	mA	+125°C	VDD = 4.2V	FOSC = 25 MHz (**PRI_IDLE** mode, EC oscillator)
		4.3	11	mA	+125°C	VDD = 5.0V	
	All devices	6	18	mA	-40°C	VDD = 4.2 V	FOSC = 40 MHz (**PRI_IDLE** mode, EC oscillator)
		6.2	18	mA	+25°C		
		6.6	18	mA	+85°C		
	All devices	8.1	22	mA	-40°C	VDD = 5.0V	
		9.1	22	mA	+25°C		
		8.3	22	mA	+85°C		

Legend: Shading of rows is to assist in readability of the table.

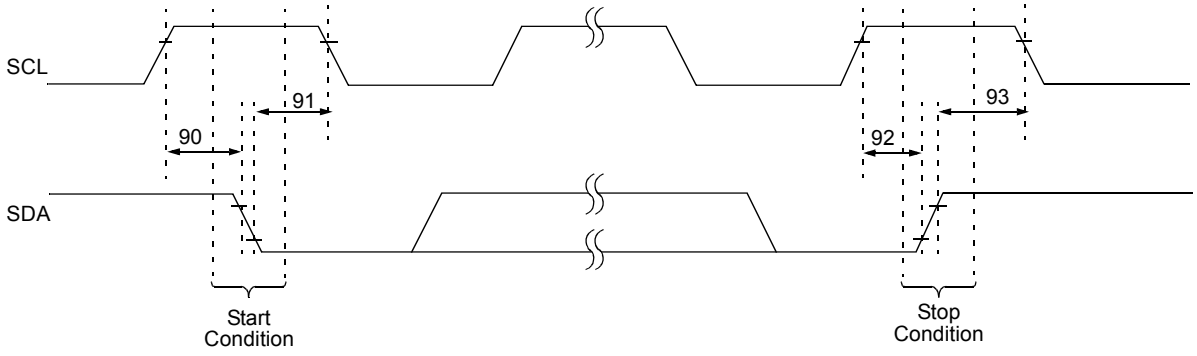
Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD or VSS and all features that add delta current disabled (such as WDT, Timer1 Oscillator, BOR, etc.).

2: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.
The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD;
MCLR = VDD; WDT enabled/disabled as specified.

3: For RC oscillator configurations, current through REXT is not included. The current through the resistor can be estimated by the formula Ir = VDD/2REXT (mA) with REXT in kΩ.

4: Standard low-cost 32 kHz crystals have an operating temperature range of -10°C to +70°C. Extended temperature crystals are available at a much higher cost.

FIGURE 27-18: MASTER SSP I²C™ BUS START/STOP BITS TIMING WAVEFORMS



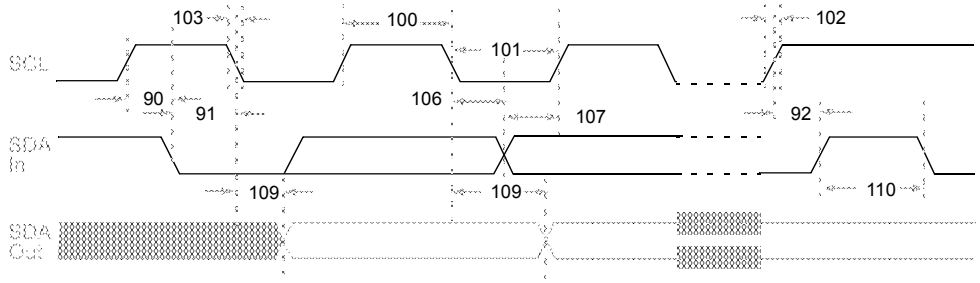
Note: Refer to Figure 27-4 for load conditions.

TABLE 27-20: MASTER SSP I²C™ BUS START/STOP BITS REQUIREMENTS

Param. No.	Symbol	Characteristic		Min	Max	Units	Conditions
90	TSU:STA	Start condition Setup Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	Only relevant for Repeated Start condition
			400 kHz mode	2(TOSC)(BRG + 1)	—		
			1 MHz mode(1)	2(TOSC)(BRG + 1)	—		
91	THD:STA	Start Condition Hold Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	After this period, the first clock pulse is generated
			400 kHz mode	2(TOSC)(BRG + 1)	—		
			1 MHz mode(1)	2(TOSC)(BRG + 1)	—		
92	TSU:STO	Stop Condition Setup Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	
			400 kHz mode	2(TOSC)(BRG + 1)	—		
			1 MHz mode(1)	2(TOSC)(BRG + 1)	—		
93	THD:STO	Stop Condition Hold Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	
			400 kHz mode	2(TOSC)(BRG + 1)	—		
			1 MHz mode(1)	2(TOSC)(BRG + 1)	—		

Note 1: Maximum pin capacitance = 10 pF for all I²C pins.

FIGURE 27-19: MASTER SSP I²C™ BUS DATA TIMING



Note: Refer to Figure 27-4 for load conditions.

F

G

H

I