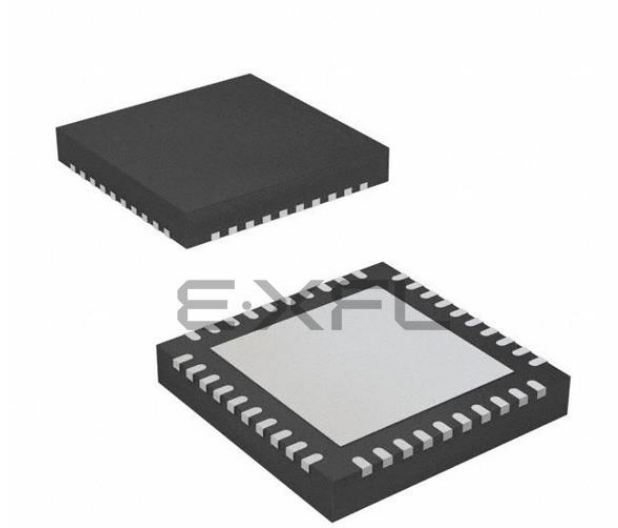


Welcome to E-XFL.COM

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	64MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, HLVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	32KB (16K x 16)
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 35x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	40-UFQFN Exposed Pad
Supplier Device Package	40-UQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f45k42-i-mv

3.0 PIC18 CPU

This family of devices contains a PIC18 8-bit CPU core based on the modified Harvard architecture. The PIC18 CPU supports:

- System Arbitration, which decides memory access allocation depending on user priorities
- Vectored Interrupt capability with automatic two level deep context saving
- 31-level deep hardware stack with overflow and underflow reset capabilities
- Support Direct, Indirect, and Relative Addressing modes
- 8x8 Hardware Multiplier

4.5.2 GENERAL PURPOSE REGISTER FILE

General Purpose RAM is available starting Bank 0 of data memory. GPRs are not initialized by a Power-on Reset and are unchanged on all other Resets.

4.5.3 SPECIAL FUNCTION REGISTERS

The Special Function Registers (SFRs) are registers used by the CPU and peripheral modules for controlling the desired operation of the device. These registers are implemented as static RAM. SFRs start at the top of data memory (3FFFh) and extend downward to occupy Bank 56 through 63 (3800h to 3FFFh). A list of these registers is given in Table 4-3 to Table 4-11. A bitwise summary of these registers can be found in **Section 42.0 "Register Summary"**.

4.5.4 ACCESS BANK

To streamline access for the most commonly used data memory locations, the data memory is configured with an Access Bank, which allows users to access a mapped block of memory without specifying a BSR. The Access Bank consists of the first 96 bytes of memory (00h-5Fh) in Bank 0 and the last 160 bytes of memory (60h-FFh) in Bank 63. The lower half is known as the "Access RAM" and is composed of GPRs. This upper half is also where some of the SFRs of the device are mapped. These two areas are mapped contiguously in the Access Bank and can be addressed linearly by an 8-bit address (Figure 4-4).

The Access Bank is used by core PIC18 instructions that include the Access RAM bit (the 'a' parameter in the instruction). When 'a' is equal to '1', the instruction uses the BSR and the 8-bit address included in the opcode for the data memory address. When 'a' is '0', however, the instruction uses the Access Bank address map; the current value of the BSR is ignored.

Using this "forced" addressing allows the instruction to operate on a data address in a single cycle, without updating the BSR first. For 8-bit addresses of 60h and above, this means that users can evaluate and operate on SFRs more efficiently. The Access RAM below 60h is a good place for data values that the user might need to access rapidly, such as immediate computational results or common program variables. Access RAM also allows for faster and more code efficient and switching of variables.

The mapping of the Access Bank is slightly different when the extended instruction set is enabled (XINST Configuration bit = 1). This is discussed in more detail in **Section 4.8.3 "Mapping the Access Bank in Indexed Literal Offset Mode"**.

TABLE 4-10: SPECIAL FUNCTION REGISTER MAP FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES BANK 57

39FFh	—	39DFh	OSCFRQ	39BFh	—	399Fh	—	397Fh	—	395Fh	WDTU	393Fh	—	391Fh	—
39FEh	—	39DEh	OSCTUNE	39BEh	—	399Eh	—	397Eh	—	395Eh	WDTH	393Eh	—	391Eh	—
39FDh	—	39DDh	OSCEN	39BDh	—	399Dh	—	397Dh	SCANTRIG	395Dh	WDTL	393Dh	—	391Dh	—
39FCh	—	39DCh	OSCSTAT	39BCh	—	399Ch	—	397Ch	SCANCON0	395Ch	WDTCON1	393Ch	—	391Ch	—
39FBh	—	39DBh	OSCCON3	39BBh	—	399Bh	—	397Bh	SCANHADRU	395Bh	WDTCON0	393Bh	—	391Bh	—
39FAh	—	39DAh	OSCCON2	39BAh	—	399Ah	PIE10	397Ah	SCANHADRH	395Ah	—	393Ah	—	391Ah	—
39F9h	—	39D9h	OSCCON1	39B9h	—	3999h	PIE9	3979h	SCANHADRL	3959h	—	3939h	—	3919h	—
39F8h	—	39D8h	CPUDOZE	39B8h	—	3998h	PIE8	3978h	SCANLADRU	3958h	—	3938h	—	3918h	—
39F7h	SCANPR	39D7h	—	39B7h	—	3997h	PIE7	3977h	SCANLADRH	3957h	—	3937h	—	3917h	—
39F6h	—	39D6h	—	39B6h	—	3996h	PIE6	3976h	SCANLADRL	3956h	—	3936h	—	3916h	—
39F5h	—	39D5h	—	39B5h	—	3995h	PIE5	3975h	—	3955h	—	3935h	—	3915h	—
39F4h	DMA2PR	39D4h	—	39B4h	—	3994h	PIE4	3974h	—	3954h	—	3934h	—	3914h	—
39F3h	DMA1PR	39D3h	—	39B3h	—	3993h	PIE3	3973h	—	3953h	—	3933h	—	3913h	—
39F2h	MAINPR	39D2h	—	39B2h	—	3992h	PIE2	3972h	—	3952h	—	3932h	—	3912h	—
39F1h	ISRPR	39D1h	VREGCON(1)	39B1h	—	3991h	PIE1	3971h	—	3951h	—	3931h	—	3911h	—
39F0h	—	39D0h	BORCON	39B0h	—	3990h	PIE0	3970h	—	3950h	—	3930h	—	3910h	—
39EFh	PRLOCK	39CFh	—	39AFh	—	398Fh	—	396Fh	—	394Fh	—	392Fh	—	390Fh	—
39EEh	—	39CEh	—	39AEh	—	398Eh	—	396Eh	—	394Eh	—	392Eh	—	390Eh	—
39EDh	—	39CDh	—	39ADh	—	398Dh	—	396Dh	—	394Dh	—	392Dh	—	390Dh	—
39ECh	—	39CCh	—	39ACh	—	398Ch	—	396Ch	—	394Ch	—	392Ch	—	390Ch	—
39EBh	—	39CBh	—	39ABh	—	398Bh	—	396Bh	—	394Bh	—	392Bh	—	390Bh	—
39EAh	—	39CAh	—	39AAh	PIR10	398Ah	IPR10	396Ah	—	394Ah	—	392Ah	—	390Ah	—
39E9h	—	39C9h	—	39A9h	PIR9	3989h	IPR9	3969h	CRCCON1	3949h	—	3929h	—	3909h	—
39E8h	—	39C8h	—	39A8h	PIR8	3988h	IPR8	3968h	CRCCON0	3948h	—	3928h	—	3908h	—
39E7h	—	39C7h	PMD7	39A7h	PIR7	3987h	IPR7	3967h	CRCXORH	3947h	—	3927h	—	3907h	—
39E6h	NVMCON2	39C6h	PMD6	39A6h	PIR6	3986h	IPR6	3966h	CRCXORL	3946h	—	3926h	—	3906h	—
39E5h	NVMCON1	39C5h	PMD5	39A5h	PIR5	3985h	IPR5	3965h	CRCSHIFTH	3945h	—	3925h	—	3905h	—
39E4h	—	39C4h	PMD4	39A4h	PIR4	3984h	IPR4	3964h	CRCSHIFTL	3944h	—	3924h	—	3904h	—
39E3h	NVMDAT	39C3h	PMD3	39A3h	PIR3	3983h	IPR3	3963h	CRCACCH	3943h	—	3923h	—	3903h	—
39E2h	—	39C2h	PMD2	39A2h	PIR2	3982h	IPR2	3962h	CRCACCL	3942h	—	3922h	—	3902h	—
39E1h	NVMADRH(4)	39C1h	PMD1	39A1h	PIR1	3981h	IPR1	3961h	CRCDATH	3941h	—	3921h	—	3901h	—
39E0h	NVMADRL	39C0h	PMD0	39A0h	PIR0	3980h	IPR0	3960h	CRCDATL	3940h	—	3920h	—	3900h	—

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

Note 1: Unimplemented in LF devices.

2: Unimplemented in PIC18(L)F26/27K42.

3: Unimplemented in PIC18(L)F26/27/45/46/47K42.

4: Unimplemented in PIC18(L)F45/55K42.

REGISTER 9-16: PIE2: PERIPHERAL INTERRUPT ENABLE REGISTER 2

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
I2C1RXIE	SPI1IE	SPI1TXIE	SPI1RXIE	DMA1AIE	DMA1ORIE	DMA1DCNTIE	DMA1SCNTIE
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 **I2C1RXIE:** I^2C1 Receive Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 6 **SPI1IE:** SPI1 Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 5 **SPI1TXIE:** SPI1 Transmit Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 4 **SPI1RXIE:** SPI1 Receive Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 3 **DMA1AIE:** DMA1 Abort Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 2 **DMA1ORIE:** DMA1 Overrun Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 1 **DMA1DCNTIE:** DMA1 Destination Count Interrupt Enable bit
1 = Enabled
0 = Disabled

bit 0 **DMA1SCNTIE:** DMA1 Source Count Interrupt Enable bit
1 = Enabled
0 = Disabled

REGISTER 11-5: WDTTMR: WDT TIMER REGISTER (READ-ONLY)

R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
WDTTMR<4:0>					STATE	PSCNT<17:16>	
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 7-3 **WDTTMR<4:0>:** Watchdog Window Value bits

WINDOW	WDT Window State		Open Percent
	Closed	Open	
111	N/A	00000-11111	100
110	00000-00011	00100-11111	87.5
101	00000-00111	01000-11111	75
100	00000-01011	01100-11111	62.5
011	00000-01111	10000-11111	50
010	00000-10011	10100-11111	37.5
001	00000-10111	11000-11111	25
000	00000-11011	11100-11111	12.5

bit 2 **STATE:** WDT Armed Status bit
1 = WDT is armed
0 = WDT is not armed

bit 1-0 **PSCNT<17:16>:** Prescale Select Upper Byte bits(1)

Note 1: The 18-bit WDT prescale value, PSCNT<17:0> includes the WDTPSL, WDTPSH and the lower bits of the WDTTMR registers. PSCNT<17:0> is intended for debug operations and should not be read during normal operation.

REGISTER 13-5: NVMDAT: DATA EEPROM MEMORY DATA

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
DAT<7:0>							
bit 7							bit 0

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

bit 7-0 **DAT<7:0>:** The value of the data memory word returned from NVMADR after a Read command, or the data written by a Write command.

TABLE 13-4: SUMMARY OF REGISTERS ASSOCIATED WITH NONVOLATILE MEMORY CONTROL

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
NVMCON1	REG<1:0>		—	FREE	WRERR	WREN	WR	RD	210
NVMCON2	Unlock Pattern								211
NVMADRL	NVMADR<7:0>								211
NVMADRH (1)	—	—	—	—	—	—	NVMADR<9:8>		211
NVMDAT	NVMDAT<7:0>								212

Legend: — = unimplemented, read as '0'. Shaded bits are not used during EEPROM access.

*Page provides register information.

Note 1: The NVMADRH register is not implemented on PIC18(L)F45/55K42.

REGISTER 14-11: SCANCON0: SCANNER ACCESS CONTROL REGISTER 0

R/W-0/0	R/W-0/0	R/W/HC-0/0	U-0	U-0	R/W-0/0	R/W-0/0	R-0/0
EN	TRIGEN	SGO	—	—	MREG	BURSTMD	BUSY
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	HC = Bit is cleared by hardware

bit 7 **EN:** Scanner Enable bit(1)
1 = Scanner is enabled
0 = Scanner is disabled

bit 6 **TRIGEN:** Scanner Trigger Enable bit(2)
1 = Scanner trigger is enabled
0 = Scanner trigger is disabled
Refer Table 14-1.

bit 5 **SGO:** Scanner GO bit(3, 4)
1 = When the CRC is ready, the Memory region set by the MREG bit will be accessed and data is passed to the CRC peripheral.
0 = Scanner operations will not occur

bit 4-3 **Unimplemented:** Read as '0'

bit 2 **MREG:** Scanner Memory Region Select bit(2)
1 = Scanner address points to Data EEPROM
0 = Scanner address points to Program Flash Memory

bit 1 **BURSTMD:** Scanner Burst Mode bit
1 = Memory access request to the CPU Arbiter is always true
0 = Memory access request to the CPU Arbiter is dependent on the CRC request and Trigger
Refer Table 14-1.

bit 0 **BUSY:** Scanner Busy Indicator bit
1 = Scanner cycle is in process
0 = Scanner cycle is compete (or never started)

Note 1: Setting EN = 1 (SCANCON0 register) does not affect any other register content.
2: Scanner trigger selection can be set using the SCANTRIG register.
3: This bit can be cleared in software. It is cleared in hardware when LADR>HADR (and a data cycle is not occurring) or when CRCGO = 0 (CRCCON0 register).
4: CRCEN and CRCGO bits (CRCCON0 register) must be set before setting the SGO bit.

FIGURE 25-3: TIMER MODE TIMING DIAGRAM

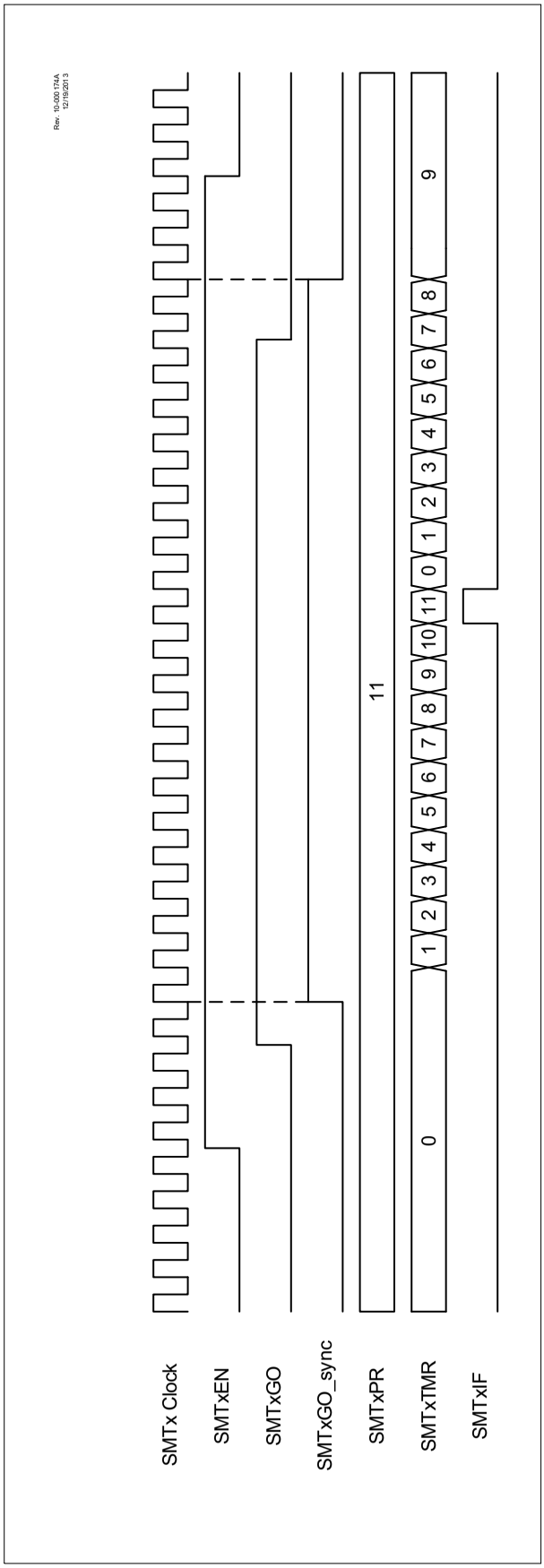
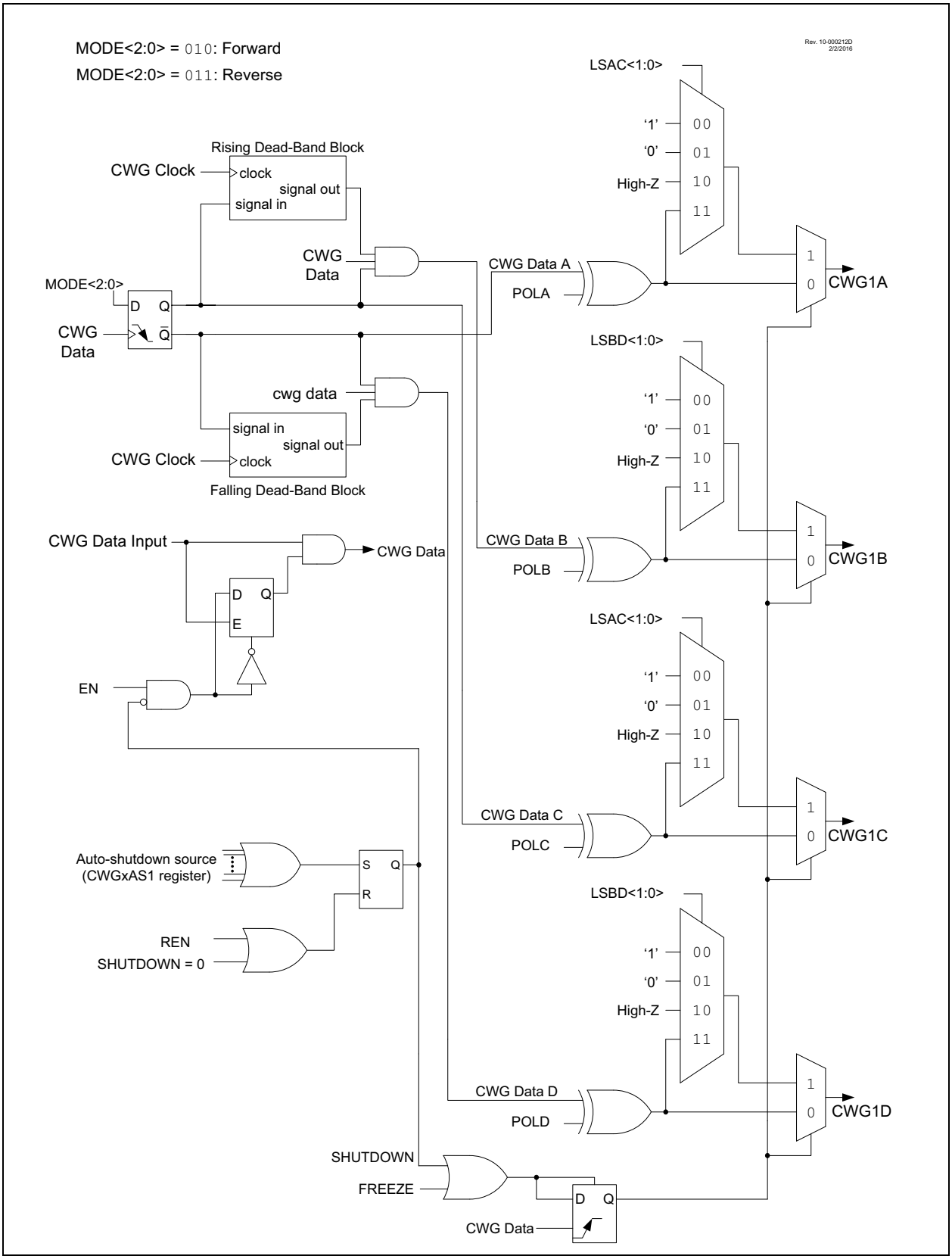


FIGURE 26-6: SIMPLIFIED CWG BLOCK DIAGRAM (FORWARD AND REVERSE FULL-BRIDGE MODES)



27.0 CONFIGURABLE LOGIC CELL (CLC)

The Configurable Logic Cell (CLCx) module provides programmable logic that operates outside the speed limitations of software execution. The logic cell takes up to 32 input signals and, through the use of configurable gates, reduces the 32 inputs to four logic lines that drive one of eight selectable single-output logic functions.

Input sources are a combination of the following:

- I/O pins
- Internal clocks
- Peripherals
- Register bits

The output can be directed internally to peripherals and to an output pin.

There are four CLC modules available on this device - CLC1, CLC2, CLC3 and CLC4.

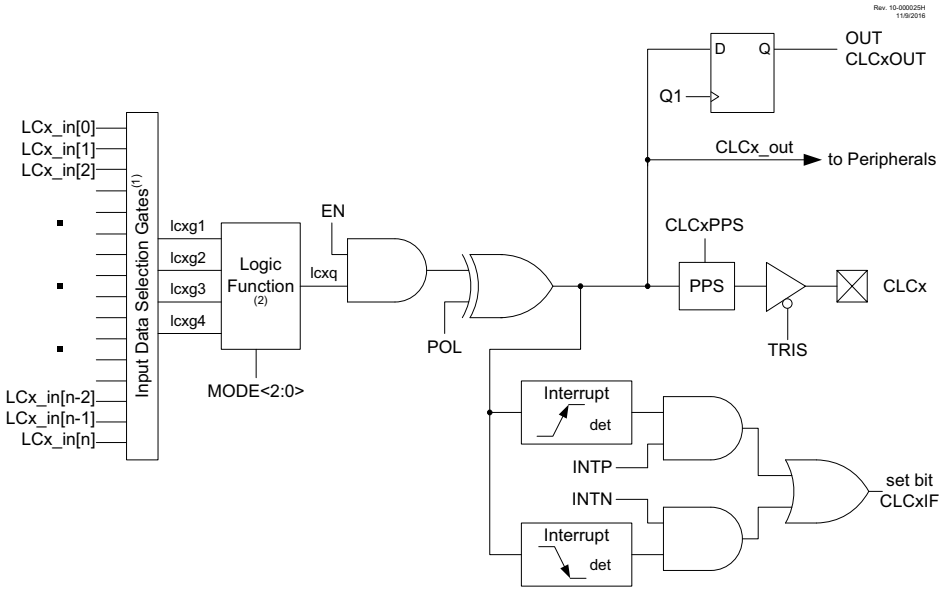
Note:	The CLC1, CLC2, CLC3 and CLC4 are four separate module instances of the same CLC module design. Throughout this section, the lower case 'x' in register names is a generic reference to the CLC number (which should be substituted with 1, 2, 3, or 4 during code development). For example, the control register is generically described in this chapter as CLCxCON, but the actual device registers are CLC1CON, CLC2CON, CLC3CON and CLC4CON.

Refer to Figure 27-1 for a simplified diagram showing signal flow through the CLCx.

Possible configurations include:

- Combinatorial Logic
  - AND
  - NAND
  - AND-OR
  - AND-OR-INVERT
  - OR-XOR
  - OR-XNOR
- Latches
  - S-R
  - Clocked D with Set and Reset
  - Transparent D with Set and Reset

FIGURE 27-1: CLCx SIMPLIFIED BLOCK DIAGRAM



Note 1: See Figure 27-2: Input Data Selection and Gating

2: See Figure 27-3: Programmable Logic Functions.

27.1 CLCx Setup

Programming the CLCx module is performed by configuring the four stages in the logic signal flow. The four stages are:

- Data selection
- Data gating
- Logic function selection
- Output polarity

Each stage is setup at run time by writing to the corresponding CLCx Special Function Registers. This has the added advantage of permitting logic reconfiguration on-the-fly during program execution.

27.1.1 DATA SELECTION

There are 32 signals available as inputs to the configurable logic. Four 32-input multiplexers are used to select the inputs to pass on to the next stage.

Data selection is through four multiplexers as indicated on the left side of Figure 27-2. Data inputs in the figure are identified by a generic numbered input name.

Table 27-1 correlates the generic input name to the actual signal for each CLC module. The column labeled 'DyS<4:0> Value' indicates the MUX selection code for the selected data input. DyS is an abbreviation for the MUX select input codes: D1S<4:0> through D4S<4:0>.

Data inputs are selected with CLCxSEL0 through CLCxSEL3 registers (Register 27-3 through Register 27-6).

Note: Data selections are undefined at power-up.

29.10 Register Definitions: ZCD Control

REGISTER 29-1: ZCDCON: ZERO-CROSS DETECT CONTROL REGISTER

R/W-0/0	U-0	R-x	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
SEN	—	OUT	POL	—	—	INTP	INTN
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 **SEN:** Zero-Cross Detect Software Enable bit
This bit is ignored when ZCDSEN configuration bit is set.
1= Zero-cross detect is enabled.
0= Zero-cross detect is disabled. ZCD pin operates according to PPS and TRIS controls.

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

bit 5 **OUT:** Zero-Cross Detect Data Output bit
ZCDPOL bit = 0:
1 = ZCD pin is sinking current
0 = ZCD pin is sourcing current
ZCDPOL bit = 1:
1 = ZCD pin is sourcing current
0 = ZCD pin is sinking current

bit 4 **POL:** Zero-Cross Detect Polarity bit
1 = ZCD logic output is inverted
0 = ZCD logic output is not inverted

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **INTP:** Zero-Cross Detect Positive-Going Edge Interrupt Enable bit
1 = ZCDIF bit is set on low-to-high ZCD_output transition
0 = ZCDIF bit is unaffected by low-to-high ZCD_output transition

bit 0 **INTN:** Zero-Cross Detect Negative-Going Edge Interrupt Enable bit
1 = ZCDIF bit is set on high-to-low ZCD_output transition
0 = ZCDIF bit is unaffected by high-to-low ZCD_output transition

TABLE 29-1: SUMMARY OF REGISTERS ASSOCIATED WITH THE ZCD MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
ZCDCON	SEN	—	OUT	POL	—	—	INTP	INTN	462

Legend: — = unimplemented, read as '0'. Shaded cells are unused by the ZCD module.

The Master process is started by writing the PID to the UxP1L register when UxP2 is '0' and the UART is idle. The UxTXIF will not be set in this case. Only the six Least Significant bits of UxP1L are used in the PID transmission.

The two Most Significant bits of the transmitted PID are PID parity bits. PID<6> is the exclusive-or of PID bits 0,1,2,and 4. PID<7> is the inverse of the exclusive-or of PID bits 1,3,4,and 5.

The UART calculates and inserts these bits in the serial stream.

Writing UxP1L automatically clears the UxTXCHK and UxRXCHK registers and generates the Break, delimiter bit, Sync character (55h), and PID transmission portion of the transaction. The data portion of the transaction that follows, if there is one, is a Slave process. See **Section 31.5.2 "LIN Slave Mode"** for more details of that process. The Master receives it's own PID when RXEN is set. Software performs the Slave process corresponding to the PID that was sent and received. Attempting to write UxP1L before an active master process is complete will not succeed. Instead, the TXWRE bit will be set.

31.5.2 LIN SLAVE MODE

LIN Slave mode is configured by the following settings:

- MODE<3:0> = 1011
- TXEN = 1
- RXEN = 1
- UxP2 = Number of data bytes to transmit
- UxP3 = Number of data bytes to receive
- UxBRGH:L = Value to achieve default baud rate
- TXPOL = 0 (for high Idle state)
- STP = desired Stop bits selection
- C0EN = desired checksum mode
- RxyPPS = TX pin selection code
- TX pin TRIS control = 0
- ON = 1

The Slave process starts upon detecting a Break on the RX pin. The Break clears the UxTXCHK, UxRXCHK, UxP2, and UxP3 registers. At the end of the Break, the auto-baud circuity is activated and the baud rate is automatically set using the Sync character following the Break. The character following the Sync character is received as the PID code and is saved in the receive FIFO. The UART computes the two PID parity bits from the six Least Significant bits of the PID. If either parity bit does not match the corresponding bit of the received PID code, the PERIF flag is set and saved at the same FIFO location as the PID code. The UxRXIF bit is set indicating that the PID is available.

Software retrieves the PID by reading the UxRXB register and determines the Slave process to execute from that. The checksum method, number of data bytes, and whether to send or receive data, is defined by software according to the PID code.

31.5.2.1 LIN Slave Receiver

When the Slave process is a receiver, the software performs the following tasks:

- UxP3 register is written with a value equal to the number of data bytes to receive.
- C0EN bit is set or cleared to select the appropriate checksum. This must be completed before the Start bit of the checksum byte is received.
- Each byte of the process response is read from UxRXB when UxRXIF is set.

The UART updates the checksum on each received byte. When the last data byte is received, the computed checksum total is stored in the UxRXCHK register. The next received byte is saved in the receive FIFO and added with the value in UxRXCHK. The result of this addition is not accessible. However, if the result is not all '1's, the CERIF bit in the UxERRIR is set. The CERIF flag persists until cleared by software. Software needs to read UxRXB to remove the checksum byte from the FIFO, but the byte can be discarded if not needed for any other purpose.

After the checksum is received, the UART ignores all activity on the RX pin until a Break starts the next transaction.

31.5.2.2 LIN Slave Transmitter

When the Slave process is a transmitter, software performs the following tasks in the order shown:

- UxP2 register is written with a value equal to the number of bytes to transmit. This will enable TXIF flag which is disabled when UxP2 is '0'.
- C0EN bit is set or cleared to select the appropriate checksum
- Inter-byte delay is performed
- Each byte of the process response is written to UxTXB when UxTXIF is set

The UART accumulates the checksum as each byte is written to UxTXB. After the last byte is written, the UART stores the calculated checksum in the UxTXCHK register and transmits the inverted result as the last byte in the response.

The TXIF flag is disabled when UxP2 bytes have been written. Any writes to UxTXB that exceed the UxP2 count will be ignored and set the TXWRE flag in the UxFIFO register.

When TXMTIF goes true, indicating the transmit shift register has completed sending the last byte in the frame, the TX output is held in Idle state for the number of half-bit periods selected by the STP bits in the UxCON2 register.

After the last Stop bit, the TX output is held in Idle state for an additional wait time determined by the half-bit period count in the UxP1 register. For example, a 2450 µs delay (~6 half-bit times) requires a value of 6 in UxP1L.

Any writes to the UxTXB register that occur after TXMTIF goes true, but before the UxP1 wait time expires, are held and then transmitted immediately following the wait time. If a backward frame is received during the wait time, any bytes that may have been written to UxTXB will be transmitted after completion of the backward frame reception plus the UxP1 wait time.

The wait timer is reset by the backward frame and starts over immediately following the reception of the Stop bits of the backward frame. Data pending in the transmit shift register will be sent when the wait time elapses.

To replace or delete any pending forward frame data, the TXBE bit needs to be set to flush the shift register and transmit buffer. A new control byte can then be written to the UxTXB register. The control byte will be held in the buffer and sent at the beginning of the next forward frame following the UxP1 wait time.

In Control Device mode, PERIF is set when a forward frame is received. This helps the software to determine whether the received byte is part of a forward frame from a Control Device (either from the Control Device under consideration or from another Control Device on the bus) or a backward frame from a Control Gear.

31.6.2 CONTROL GEAR

The Control Gear mode is configured with the following settings:

- MODE = `0b1001`
- TXEN = `1`
- RXEN = `1`
- UxP1 = Back Frames are held for transmission this number of half-bit periods after the completion of a Forward Frame.
- UxP2 = Forward/Back Frame threshold delimiter. Idle periods more than this number of half-bit periods are detected as Forward Frames.
- UxBRGH:L = Value to achieve 1200 baud rate
- TXPOL = appropriate polarity for interface circuit
- RXPOL = same as TXPOL
- STP = `0b10` for two Stop bits
- RxyPPS = TX pin output code
- TX pin TRIS control = `0`
- RXPPS = RX pin selection code
- RX pin TRIS control = `1`
- Input pin ANSEL bit = `0`
- ON = `1`

The UART starts listening for a forward frame when the Control Gear mode is entered. Only the frames that follow an Idle period longer than UxP2 half-bit periods are detected as forward frames. Backward frames from other Control Gear are ignored. Only forward frames will be stored in UxRXB. This is necessary because a backward frame can be sent only as a response to a forward frame.

The forward frame is received one byte at a time in the receive FIFO and retrieved by reading the UxRXB register. The end of the forward frame starts a timer to delay the backward frame response by wait time equal to the number of half-bit periods stored in UxP1.

The data received in the forward frame is processed by the application software. If the application decides to send a backward frame in response to the forward frame, the value of the backward frame is written to UxTXB. This value is held for transmission in the transmit shift register until the wait time expires and is then transmitted.

If the backward frame data is written to UxTXB after the wait time has expired, it is held in the UxTXB register until the end of the wait time following the next forward frame. The TXMTIF bit is false when the backward frame data is held in the transmit shift register. Receiving a UxRXIF interrupt before the TXMTIF goes true indicates that the backward frame write was too late and another forward frame was received before sending the backward frame. The pending backward frame has to be flushed by setting the TXBE bit, to prevent it from being sent after the next Forward Frame.

TABLE 36-2: COMPUTATION MODES

		Bit Clear Conditions	Value after Trigger completion		Threshold Operations			Value at ADTIF interrupt		
Mode	ADMD	ACC and CNT	ACC	CNT	Retrigger	Threshold Test	Interrupt	ADAOV	FLTR	CNT
Basic	0	ADACLR = 1	Unchanged	Unchanged	No	Every Sample	If threshold=true	N/A	N/A	count
Accumulate	1	ADACLR = 1	S + ACC or (S2-S1) + ACC	If (CNT=0xFF): CNT, otherwise: CNT+1	No	Every Sample	If threshold=true	ACC Overflow	ACC/2^{ADCRS}	count
Average	2	ADACLR = 1 or CNT>=RPT at GO or retrigger	S + ACC or (S2-S1) + ACC	If (CNT=0xFF): CNT, otherwise: CNT+1	No	If CNT>=RPT	If threshold=true	ACC Overflow	ACC/2^{ADCRS}	count
Burst Average	3	ADACLR = 1 or GO set or retrigger	Each repetition: same as Average End with sum of all samples	Each repetition: same as Average End with CNT=RPT	Repeat while CNT<RPT	If CNT>=RPT	If threshold=true	ACC Overflow	ACC/2^{ADCRS}	RPT
Low-pass Filter	4	ADACLR = 1	S+ACC-ACC/2^{ADCRS} or (S2-S1)+ACC-ACC/2^{ADCRS}	Count up, stop counting when CNT = 0xFF	No	If CNT>=RPT	If threshold=true	ACC Overflow	Filtered Value	count

Note: S1 and S2 are abbreviations for Sample 1 and Sample 2, respectively. When ADDSEN = 0, S1 = ADRES; When ADDSEN = 1, S1 = PREV and S2 = ADRES.

REGISTER 36-2: ADCON1: ADC CONTROL REGISTER 1

R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0	R/W-0/0
PPOL	IPEN	GPOL	—	—	—	—	DSEN
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 **PPOL:** Precharge Polarity bit
If PRE>0x00:

PPOL	Action During 1st Precharge Stage	
	External (selected analog I/O pin)	Internal (AD sampling capacitor)
1	Connected to V_{DD}	C_{HOLD} connected to V_{SS}
0	Connected to V_{SS}	C_{HOLD} connected to V_{DD}

Otherwise:
The bit is ignored

bit 6 **IPEN:** A/D Inverted Precharge Enable bit
If DSEN = 1
1 = The precharge and guard signals in the second conversion cycle are the opposite polarity of the first cycle
0 = Both Conversion cycles use the precharge and guards specified by ADPPOL and ADGPOL
Otherwise:
The bit is ignored

bit 5 **GPOL:** Guard Ring Polarity Selection bit
1 = ADC guard Ring outputs start as digital high during Precharge stage
0 = ADC guard Ring outputs start as digital low during Precharge stage

bit 4-1 **Unimplemented:** Read as '0'

bit 0 **DSEN:** Double-sample enable bit
1 = Two conversions are performed on each trigger. Data from the first conversion appears in PREV
0 = One conversion is performed for each trigger

ANDWF **AND W with f**

Syntax: ANDWF f {,d {,a}}

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

Operation: (W) .AND. (f) → dest

Status Affected: N, Z

Encoding:

0001	01da	ffff	ffff

Description: The contents of W are AND'ed with register 'f'. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f' (default).
If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank.
If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 95 (5Fh). See **Section 41.2.3 "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 to destination

Example: `ANDWF REG, 0, 0`

Before Instruction
W = 17h
REG = C2h
After Instruction
W = 02h
REG = C2h

BC **Branch if Carry**

Syntax: BC n

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

Operation: if CARRY bit is '1'
(PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1110	0010	nnnn	nnnn

Description: If the CARRY 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 2-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    BC   5
```
Before Instruction
PC = address (HERE)
After Instruction
If CARRY = 1;
PC = address (HERE + 12)
If CARRY = 0;
PC = address (HERE + 2)

TABLE 42-1: REGISTER FILE SUMMARY FOR PIC18(L)F26/27/45/46/47/55/56/57K42 DEVICES

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
3BC8h - 3AEBh	—	Unimplemented								
3AEAh	U2CTSPPS	—	—	—	U2CTSPPS					277
3AE8h	U2RXPPS	—	—	—	U2RXPPS					277
3AE7h	U1CTSPPS	—	—	—	U1CTSPPS					277
3AE5h	U1RXPPS	—	—	—	U1RXPPS					277
3AE4h	I2C2SDAPPS	—	—	—	I2C2SDAPPS					277
3AE3h	I2C2SCLPPS	—	—	—	I2C2SCLPPS					277
3AE2h	I2C1SDAPPS	—	—	—	I2C1SDAPPS					277
3AE1h	I2C1SCLPPS	—	—	—	I2C1SCLPPS					277
3AE0h	SPI1SSPPS	—	—	—	SPI1SSPPS					277
3ADFh	SPI1SDIPPS	—	—	—	SPI1SDIPPS					277
3ADEh	SPI1SCKPPS	—	—	—	SPI1SCKPPS					277
3ADDh	ADACTPPS	—	—	—	ADACTPPS					277
3ADCh	CLCIN3PPS	—	—	—	CLCIN3PPS					277
3ADBh	CLCIN2PPS	—	—	—	CLCIN2PPS					277
3ADAh	CLCIN1PPS	—	—	—	CLCIN1PPS					277
3AD9h	CLCIN0PPS	—	—	—	CLCIN0PPS					277
3AD8h	MD1SRCPPS	—	—	—	MD1SRCPPS					277
3AD7h	MD1CARHPPS	—	—	—	MD1CARHPPS					277
3AD6h	MD1CARLPPS	—	—	—	MD1CARLPPS					277
3AD5h	CWG3INPPS	—	—	—	CWG3INPPS					277
3AD4h	CWG2INPPS	—	—	—	CWG2INPPS					277
3AD3h	CWG1INPPS	—	—	—	CWG1INPPS					277
3AD2h	SMT1SIGPPS	—	—	—	SMT1SIGPPS					277
3AD1h	SMT1WINPPS	—	—	—	SMT1WINPPS					277
3AD0h	CCP4PPS	—	—	—	CCP4PPS					277
3ACFh	CCP3PPS	—	—	—	CCP3PPS					277
3ACEh	CCP2PPS	—	—	—	CCP2PPS					277
3ACDh	CCP1PPS	—	—	—	CCP1PPS					277
3ACCh	T6INPPS	—	—	—	T6INPPS					277
3ACBh	T4INPPS	—	—	—	T4INPPS					277
3ACAh	T2INPPS	—	—	—	T2INPPS					277
3AC9h	T5GPPS	—	—	—	T5GPPS					277
3AC8h	T5CLKIPPS	—	—	—	T5CLKIPPS					277
3AC7h	T3GPPS	—	—	—	T3GPPS					277
3AC6h	T3CLKIPPS	—	—	—	T3CLKIPPS					277
3AC5h	T1GPPS	—	—	—	T1GPPS					277
3AC4h	T1CLKIPPS	—	—	—	T1CLKIPPS					277
3AC3h	T0CLKIPPS	—	—	—	T0CLKIPPS					277
3AC2h	INT2PPS	—	—	—	INT2PPS					277
3AC1h	INT1PPS	—	—	—	INT1PPS					277
3AC0h	INT0PPS	—	—	—	INT0PPS					277
3ABFh	PPSLOCK	—	—	—	—	—	—	—	PPSLOCKED	283
3ABEh	—	Reserved, maintain as '0'								
3ABDh - 3A9Ah	—	Unimplemented								
3A99h	—	Reserved, maintain as '0'								

Legend: x = unknown, u = unchanged, — = unimplemented, q = value depends on condition

Note 1: Unimplemented in LF devices.
2: Unimplemented in PIC18(L)F26/27K42.
3: Unimplemented on PIC18(L)F26/27/45/46/47K42 devices.
4: Unimplemented in PIC18(L)F45/55K42.

TABLE 44-6: I/O PORTS

Standard Operating Conditions (unless otherwise stated)							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
	VIL	**Input Low Voltage**					
		I/O PORT:					
D300		with TTL buffer	—	—	0.8	V	4.5V ≤ VDD ≤ 5.5V
D301			—	—	0.15 VDD	V	1.8V ≤ VDD < 4.5V
D302		with Schmitt Trigger buffer	—	—	0.2 VDD	V	2.0V ≤ VDD ≤ 5.5V
D303		with I2C levels	—	—	0.3 VDD	V	
D304		with SMBus 2.0	—	—	0.8	V	2.7V ≤ VDD ≤ 5.5V
D305		with SMBus 3.0	—	—	0.8	V	1.8V ≤ VDD ≤ 5.5V
D306		$\overline{MCLR}$	—	—	0.2 VDD	V	
	VIH	**Input High Voltage**					
		I/O PORT:					
D320		with TTL buffer	2.0	—	—	V	4.5V ≤ VDD ≤ 5.5V
D321			0.25 VDD + 0.8	—	—	V	1.8V ≤ VDD < 4.5V
D322		with Schmitt Trigger buffer	0.8 VDD	—	—	V	2.0V ≤ VDD ≤ 5.5V
D323		with I2C levels	0.7 VDD	—	—	V	
D324		with SMBus 2.0	2.1	—	—	V	2.7V ≤ VDD ≤ 5.5V
D325		with SMBus 3.0	1.35	—	—	V	1.8V ≤ VDD ≤ 5.5V
D326		$\overline{MCLR}$	0.7 VDD	—	—	V	
	IIL	**Input Leakage Current(1)**					
D340		I/O Ports	—	± 5	± 125	nA	VSS ≤ VPIN ≤ VDD, Pin at high-impedance, 85°C
D341			—	± 5	± 1000	nA	VSS ≤ VPIN ≤ VDD, Pin at high-impedance, 125°C
D342		$\overline{MCLR}$(2)	—	± 50	± 200	nA	VSS ≤ VPIN ≤ VDD, Pin at high-impedance, 85°C
	IPUR	**Weak Pull-up Current**					
D350			25	120	200	µA	VDD = 3.0V, VPIN = VSS
	VOL	**Output Low Voltage**					
D360		I/O ports	—	—	0.6	V	IOL = 10.0mA, VDD = 3.0V
	VOH	**Output High Voltage**					
D370		I/O ports	VDD - 0.7	—	—	V	IOH = 6.0 mA, VDD = 3.0V
D380	CIO	**All I/O pins**	—	5	50	pF	

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Negative current is defined as current sourced by the pin.

2: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7289-7561

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820