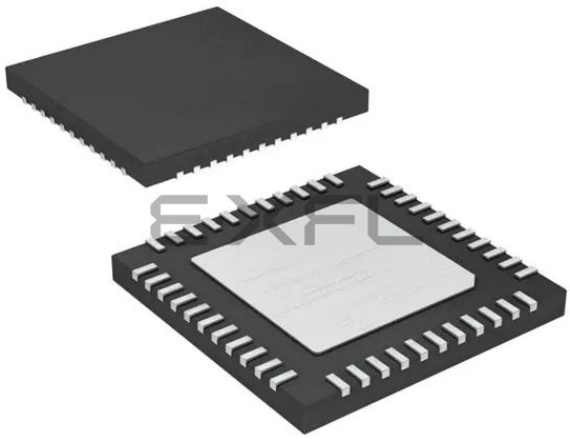


Welcome to E-XFL.COM

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	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 35x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf46k42-i-ml

6.1 Power-on Reset (POR)

The POR circuit holds the device in Reset until VDD has reached an acceptable level for minimum operation. Slow rising VDD, fast operating speeds or analog performance may require greater than minimum VDD. The PWRT, BOR or MCLR features can be used to extend the start-up period until all device operation conditions have been met.

6.2 Brown-out Reset (BOR)

The BOR circuit holds the device in Reset when VDD reaches a selectable minimum level. Between the POR and BOR, complete voltage range coverage for execution protection can be implemented.

The Brown-out Reset module has four operating modes controlled by the BOREN<1:0> bits in Configuration Words. The four operating modes are:

- BOR is always on
- BOR is off when in Sleep
- BOR is controlled by software
- BOR is always off

Refer to Table 6-1 for more information.

The Brown-out Reset voltage level is selectable by configuring the BORV<1:0> bits in Configuration Words.

A VDD noise rejection filter prevents the BOR from triggering on small events. If VDD falls below VBOR for a duration greater than parameter TBORDC, the device will reset. See Table 44-13 for more information.

6.2.1 BOR IS ALWAYS ON

When the BOREN bits of Configuration Words are programmed to '11', the BOR is always on. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is active during Sleep. The BOR does not delay wake-up from Sleep.

6.2.2 BOR IS OFF IN SLEEP

When the BOREN bits of Configuration Words are programmed to '10', the BOR is on, except in Sleep. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is not active during Sleep. The device wake-up will be delayed until the BOR is ready.

6.2.3 BOR CONTROLLED BY SOFTWARE

When the BOREN bits of Configuration Words are programmed to '01', the BOR is controlled by the SBOREN bit of the BORCON register. The device start-up is not delayed by the BOR ready condition or the VDD level.

BOR protection begins as soon as the BOR circuit is ready. The status of the BOR circuit is reflected in the BORRDY bit of the BORCON register.

BOR protection is unchanged by Sleep.

6.2.4 BOR AND BULK ERASE

BOR is forced ON during PFM Bulk Erase operations to make sure that a safe erase voltage is maintained for a successful erase cycle.

During Bulk Erase, the BOR is enabled at 2.45V for F and LF devices, even if it is configured to some other value. If VDD falls, the erase cycle will be aborted, but the device will not be reset.

9.3 Interrupt Priority

The final priority level for any pending source of interrupt is determined first by the user-assigned priority of that source in the IPRx register, then by the natural order priority within the IVT. The sections below detail the operation of Interrupt priorities.

9.3.1 USER (SOFTWARE) PRIORITY

User-assigned interrupt priority is enabled by setting the IPEN bit in the INTCON0 register (Register 9-1). Each peripheral interrupt source can be assigned a high or low priority level by the user. The user-assignable interrupt priority control bits for each interrupt are located in the IPRx registers (Registers 9-25 through 9-35).

The interrupts are serviced based on predefined interrupt priority scheme defined below.

1. Interrupts set by the user as high-priority interrupt have higher precedence of execution. High-priority interrupts will override a low-priority request when:
   a) A low priority interrupt has been requested or its request is already pending.
   b) A low- and high-priority interrupt are triggered concurrently, i.e., on the same instruction cycle**(1)**.
   c) A low-priority interrupt was requested and the corresponding Interrupt Service Routine is currently executing. In this case, the lower priority interrupt routine will complete executing after the high-priority interrupt has been serviced**(2)**.
2. Interrupts set by the user as a low priority have the lower priority of execution and are preempted by any high-priority interrupt.
3. Interrupts defined with the same software priority cannot preempt or interrupt each other. Concurrent pending interrupts with the same user priority are resolved using the natural order priority. (when MVECEN = ON) or in the order the interrupt flag bits are polled in the ISR (when MVECEN = OFF).

Note 1: When a high priority interrupt preempts a concurrent low priority interrupt, the GIEL bit may be cleared in the high priority Interrupt Service Routine. If the GIEL bit is cleared, the low priority interrupt will NOT be serviced even if it was originally requested. The corresponding interrupt flag needs to be cleared in user code.

2: When a high priority interrupt is requested while a low priority Interrupt Service Routine is executing, the GIEL bit may be cleared in the high priority Interrupt Service Routine. The pending low priority interrupt will resume even if the GIEL bit is cleared.

9.12 Register Definitions: Interrupt Control

REGISTER 9-1: INTCON0: INTERRUPT CONTROL REGISTER 0

R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1
GIE/GIEH	GIEL	IPEN	—	—	INT2EDG	INT1EDG	INT0EDG
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 **GIE/GIEH:** Global Interrupt Enable bits

If IPEN = 0:

GIE:

1 = Enables all unmasked interrupts
0 = Disables all interrupts

If IPEN = 1:

GIEH:

1 = Enables all unmasked high priority interrupts: bit also needs to be set for enabling low priority interrupts
0 = Disables all interrupts

bit 6 **GIEL:** Global Low Priority Interrupt Enable bit

If IPEN = 0:

Reserved, read as '0'

If IPEN = 1:

GIEL:

1 = Enables all unmasked low priority interrupts, GIEH also needs to be set for low priority interrupts
0 = Disables all low priority

bit 5 **IPEN:** Interrupt Priority Enable bit

1 = Enable priority levels on interrupts
0 = Disable priority levels on interrupts; all interrupts are treated as high priority interrupts

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

bit 2 **INT2EDG:** External Interrupt 2 Edge Select bit

1 = Interrupt on rising edge of INT2 pin
0 = Interrupt on falling edge of INT2 pin

bit 1 **INT1EDG:** External Interrupt 1 Edge Select bit

1 = Interrupt on rising edge of INT1 pin
0 = Interrupt on falling edge of INT1 pin

bit 0 **INT0EDG:** External Interrupt 0 Edge Select bit

1 = Interrupt on rising edge of INT0 pin
0 = Interrupt on falling edge of INT0 pin

REGISTER 9-32: IPR7: PERIPHERAL INTERRUPT PRIORITY REGISTER 7

U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	U-0	R/W-1/1	R/W-1/1
—	—	INT2IP	CLC2IP	CWG2IP	—	CCP2IP	TMR4IP
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 **INT2IP:** External Interrupt 2 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 4 **CLC2IP:** CLC2 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 3 **CWG2IP:** CWG2 Interrupt Priority bit
1 = High priority
0 = Low priority

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

bit 1 **CCP2IP:** CRC Interrupt Priority bit
1 = High priority
0 = Low priority

bit 0 **TMR4IP:** TMR4 Interrupt Priority bit
1 = High priority
0 = Low priority

REGISTER 9-33: IPR8: PERIPHERAL INTERRUPT PRIORITY REGISTER 8

R/W-1/1	R/W-1/1	U-0	U-0	U-0	U-0	U-0	U-0
TMR5GIP	TMR5IP	—	—	—	—	—	—
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 **TMR5GIP:** TMR5 Gate Interrupt Priority bit
1 = High priority
0 = Low priority

bit 6 **TMR5IP:** TMR5 Interrupt Priority bit
1 = High priority
0 = Low priority

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

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 15-2: DMAxCON1: DMAx CONTROL REGISTER1

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
DMODE<1:0>		DSTP	SMR<1:0>		SMODE<1:0>		SSTP
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

bit 7-6 **DMODE<1:0>:** Destination Address Mode Selection bits
- 11 = Reserved, Do not use
- 10 = DMAxDPTR<15:0> is decremented after each transfer completion
- 01 = DMAxDPTR<15:0> is incremented after each transfer completion
- 00 = DMAxDPTR<15:0> remains unchanged after each transfer completion

bit 5 **DSTP:** Destination Counter Reload Stop bit
- 1 = SIRQEN bit is cleared when Destination Counter reloads
- 0 = SIRQEN bit is not cleared when Destination Counter reloads

bit 4-3 **SMR[1:0]:** Source Memory Region Select bits
- 1x = DMAxSSA<21:0> points to Data EEPROM
- 01 = DMAxSSA<21:0> points to Program Flash Memory
- 00 = DMAxSSA<21:0> points to SFR/GPR Data Space

bit 2-1 **SMODE[1:0]:** Source Address Mode Selection bits
- 11 = Reserved, Do not use
- 10 = DMAxSPTR<21:0> is decremented after each transfer completion
- 01 = DMAxSPTR<21:0> is incremented after each transfer completion
- 00 = DMAxSPTR<21:0> remains unchanged after each transfer completion

bit 0 **SSTP:** Source Counter Reload Stop bit
- 1 = SIRQEN bit is cleared when Source Counter reloads
- 0 = SIRQEN bit is not cleared when Source Counter reloads

TABLE 15-3: SUMMARY OF REGISTERS ASSOCIATED WITH DMA

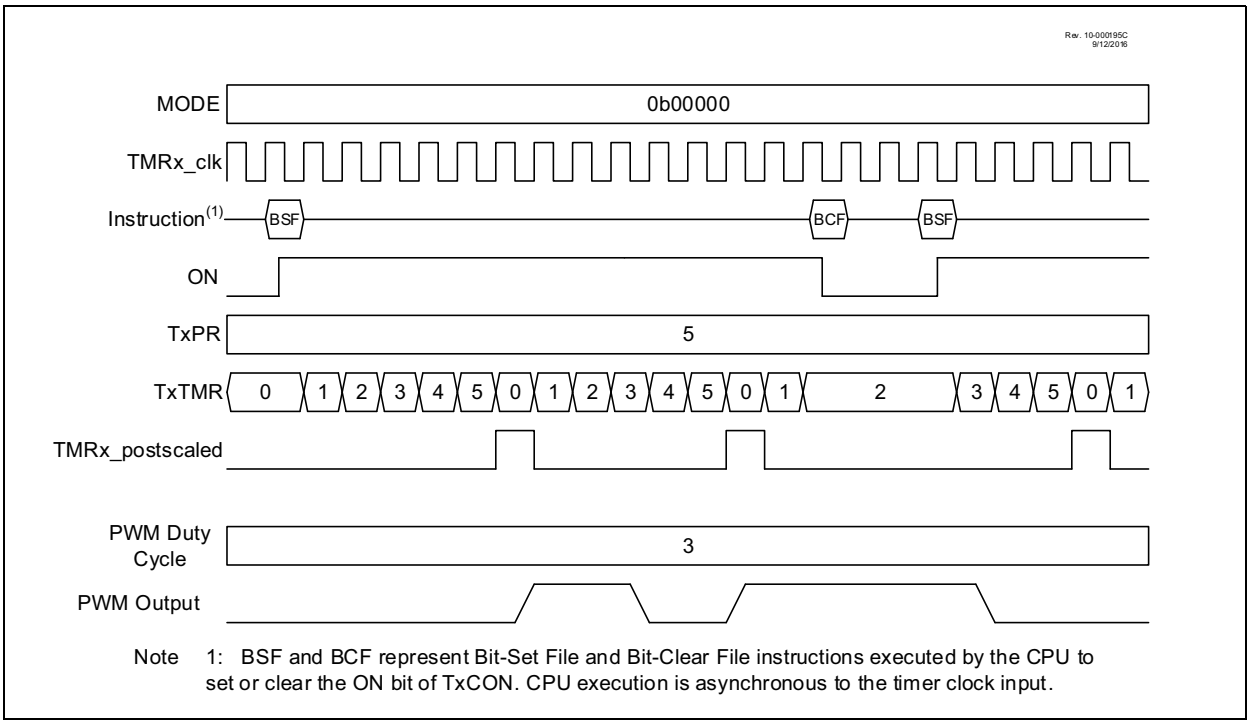
Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
DMAxCON0	EN	SIRQEN	DGO	—	—	AIRQEN	—	XIP	248
DMAxCON1	DMODE<1:0>		DSTP	SMR<1:0>		SMODE<1:0>		SSTP	249
DMAxBUF	DBUF7	DBUF6	DBUF5	DBUF4	DBUF3	DBUF2	DBUF1	DBUF0	250
DMAxSSAL	SSA<7:0>								250
DMAxSSAH	SSA<15:8>								250
DMAxSSAU	—	—	SSA<21:16>						251
DMAxSPTRL	SPTR<7:0>								251
DMAxSPTRH	SPTR<15:8>								251
DMAxSPTRU	—	—	SPTR<21:16>						252
DMAxSSZL	SSZ<7:0>								252
DMAxSSZH	—	—	—	—	SSZ<11:8>				252
DMAxSCNTL	SCNT<7:0>								253
DMAxSCNTH	—	—	—	—	SCNT<11:8>				253
DMAxDSAL	DSA<7:0>								253
DMAxDSAH	DSA<15:8>								254
DMAxDPTRL	DPTR<7:0>								254
DMAxDPTRH	DPTR<15:8>								254
DMAxDSZL	DSZ<7:0>								255
DMAxDSZH	—	—	—	—	DSZ<11:8>				255
DMAxDCNTL	DCNT<7:0>								255
DMAxDCNTH	—	—	—	—	DCNT<11:8>				256
DMAxSIRQ	—	SIRQ<6:0>							256
DMAxAIRQ	—	AIRQ<6:0>							256

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by DMA.

22.5.1 SOFTWARE GATE MODE

The timer increments with each clock input when ON = 1 and does not increment when ON = 0. When the T2TMR count equals the T2PR period count the timer resets on the next clock and continues counting from 0. Operation with the ON bit software controlled is illustrated in Figure 22-4. With T2PR = 5, the counter advances until T2TMR = 5, and goes to zero with the next clock.

FIGURE 22-4: SOFTWARE GATE MODE TIMING DIAGRAM



Note 1: BSF and BCF represent Bit-Set File and Bit-Clear File instructions executed by the CPU to set or clear the ON bit of TxCON. CPU execution is asynchronous to the timer clock input.

REGISTER 26-6: CWGxAS0: CWG AUTO-SHUTDOWN CONTROL REGISTER 0

R/W/HS/HC-0/0	R/W-0/0	R/W-0/0	R/W-1/1	R/W-0/0	R/W-1/1	U-0	U-0
SHUTDOWN	REN	LSBD<1:0>		LSAC<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 Resets
'1' = Bit is set	'0' = Bit is cleared	HS/HC = Bit is set/cleared by hardware
q = Value depends on condition		

bit 7 **SHUTDOWN:** Auto-Shutdown Event Status bit(1,2)
- 1 = An auto-shutdown state is in effect
- 0 = No auto-shutdown event has occurred

bit 6 **REN:** Auto-Restart Enable bit
- 1 = Auto-restart is enabled
- 0 = Auto-restart is disabled

bit 5-4 **LSBD<1:0>:** CWGxB and CWGxD Auto-Shutdown State Control bits
- 11 = A logic '1' is placed on CWGxB/D when an auto-shutdown event occurs.
- 10 = A logic '0' is placed on CWGxB/D when an auto-shutdown event occurs.
- 01 = Pin is tri-stated on CWGxB/D when an auto-shutdown event occurs.
- 00 = The inactive state of the pin, including polarity, is placed on CWGxB/D after the required dead-band interval when an auto-shutdown event occurs.

bit 3-2 **LSAC<1:0>:** CWGxA and CWGxC Auto-Shutdown State Control bits
- 11 = A logic '1' is placed on CWGxA/C when an auto-shutdown event occurs.
- 10 = A logic '0' is placed on CWGxA/C when an auto-shutdown event occurs.
- 01 = Pin is tri-stated on CWGxA/C when an auto-shutdown event occurs.
- 00 = The inactive state of the pin, including polarity, is placed on CWGxA/C after the required dead-band interval when an auto-shutdown event occurs.

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

Note 1: This bit may be written while EN = 0 (Register 26-1), to place the outputs into the shutdown configuration.

2: The outputs will remain in auto-shutdown state until the next rising edge of the CWG data input after this bit is cleared.

FIGURE 30-2: On Off Keying (OOK) Synchronization

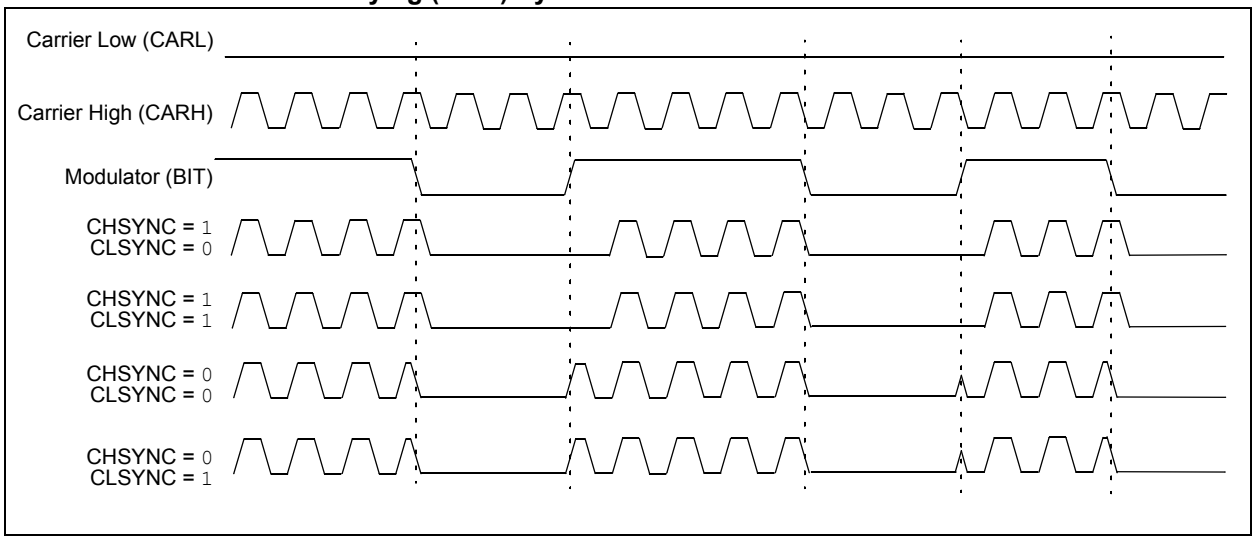


FIGURE 30-3: No Synchronization (CHSYNC = 0, CLSYNC = 0)

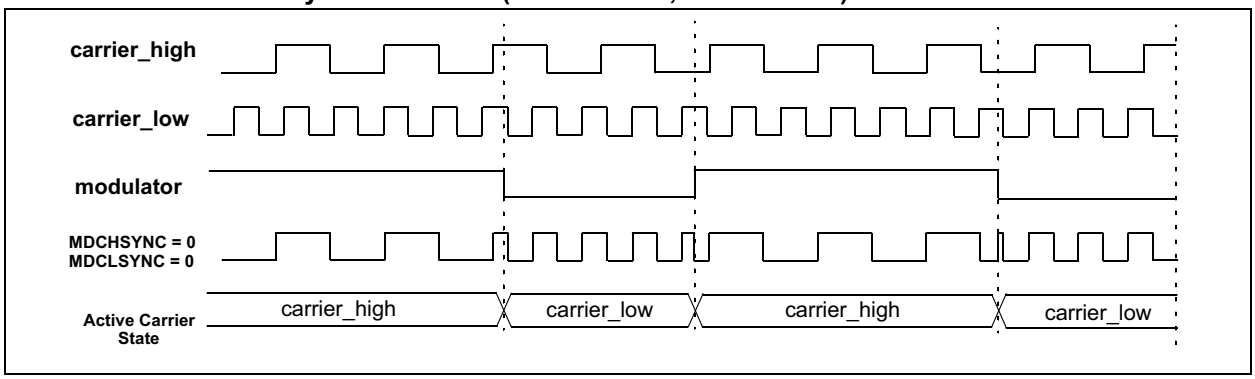
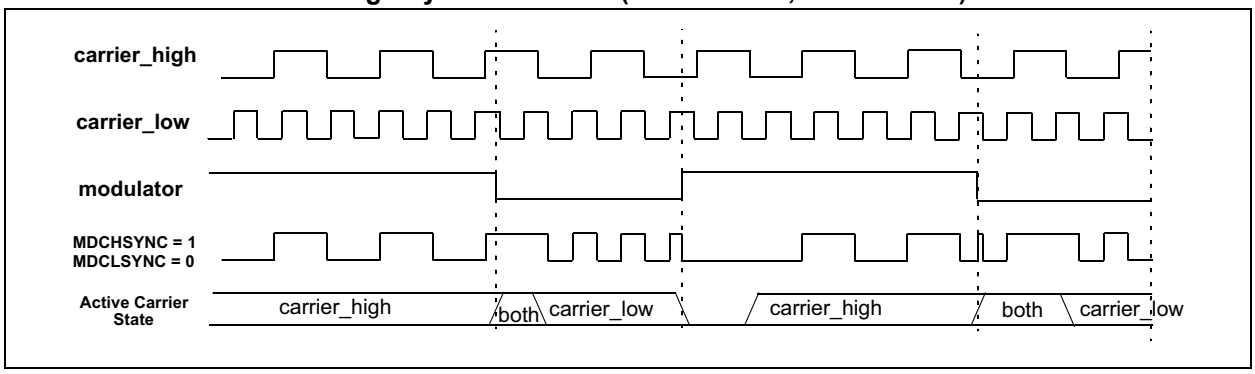


FIGURE 30-4: Carrier High Synchronization (CHSYNC = 1, CLSYNC = 0)



31.2.2 UART ASYNCHRONOUS RECEIVER

The Asynchronous mode is typically used in RS-232 systems. The receiver block diagram is shown in Figure 31-2. The data is received on the RX pin and drives the data recovery block. The data recovery block is actually a high-speed shifter operating at 4 or 16 times the baud rate, whereas the serial Receive Shift Register (RSR) operates at the bit rate. When all bits of the character have been shifted in, they are immediately transferred to a two character First-In-First-Out (FIFO) memory. The FIFO buffering allows reception of two complete characters and the start of a third character before software must start servicing the UART receiver. The FIFO registers and RSR are not directly accessible by software. Access to the received data is via the UxRXB register.

31.2.2.1 Enabling the Receiver

The UART receiver is enabled for asynchronous operation by configuring the following control bits:

- RXEN = 1
- MODE<3:0> = 0h through 3h
- UxBRGH:L = desired baud rate
- RXPPS = code for desired input pin
- Input pin ANSEL bit = 0
- ON = 1

All other UART control bits are assumed to be in their default state.

Setting the RXEN bit in the UxCON0 register enables the receiver circuitry of the UART. Setting the MODE<3:0> bits in the UxCON0 register configures the UART for the desired asynchronous mode. Setting the ON bit in the UxCON1 register enables the UART. The TRIS bit corresponding to the selected RX I/O pin must be set to configure the pin as an input.

Note:	If the RX function is on an analog pin, the corresponding ANSEL bit must be cleared for the receiver to function.

31.2.2.2 Receiving Data

Data is recovered from the bit stream by timing to the center of the bits and sampling the input level. In High-Speed mode, there are four BRG clocks per bit and only one sample is taken per bit. In Normal-Speed mode, there are 16 BRG clocks per bit and three samples are taken per bit.

The receiver data recovery circuit initiates character reception on the falling edge of the Start bit. The Start bit, is always a '0'. The Start bit is qualified in the middle of the bit. In Normal-Speed mode only, the Start bit is also qualified at the leading edge of the bit. The following paragraphs describe the majority detect sampling of Normal-Speed mode.

The falling edge starts the baud rate generator (BRG) clock. The input is sampled at the first and second BRG clocks.

If both samples are high then the falling edge is deemed a glitch and the UART returns to the Start bit detection state without generating an error.

If either sample is low, the data recovery circuit continues counting BRG clocks and takes samples at clock counts 7, 8, and 9. When less than two samples are low, the Start bit is deemed invalid and the data recovery circuit aborts character reception, without generating an error, and resumes looking for the falling edge of the Start bit.

When two or more samples are low, the Start bit is deemed valid and the data recovery continues. After a valid Start bit is detected, the BRG clock counter continues and resets at count 16. This is the beginning of the first data bit.

The data recovery circuit counts BRG clocks from the beginning of the bit and takes samples at clocks 7, 8, and 9. The bit value is determined from the majority of the samples. The resulting '0' or '1' is shifted into the RSR.The BRG clock counter continues and resets at count 16. This sequence repeats until all data bits have been sampled and shifted into the RSR.

After all data bits have been shifted in, the first Stop bit is sampled. Stop bits are always a '1'. If the bit sampling determines that a '0' is in the Stop bit position, the framing error is set for this character. Otherwise, the framing error is cleared for this character. See **Section 31.2.2.4 "Receive Framing Error"** for more information on framing errors.

31.2.2.3 Receive Interrupts

Immediately after all data bits and the Stop bit have been received, the character in the RSR is transferred to the UART receive FIFO. The UxRXIF interrupt flag in the respective PIR register is set at this time, provided it is not being suppressed.

The UxRXIF is suppressed by any of the following:

- FERIF if FERIE is set
- PERIF if PERIE is set

This suspends DMA transfer of data until software processes the error and reads UxRXB to advance the FIFO beyond the error.

UxRXIF interrupts are enabled by setting all of the following bits:

- UxRXIE, Interrupt Enable bit in the PIE register
- GIE, Global Interrupt Enable bits in the INTCON0 register

The UxRXIF interrupt flag bit will be set when not suppressed and there is an unread character in the FIFO, regardless of the state of interrupt enable bits. Reading the UxRXB register will transfer the top character out of the FIFO and reduce the FIFO contents by one. The UxRXIF interrupt flag bit is read-only, it cannot be set or cleared by software.

FIGURE 31-7: MANCHESTER TIMING

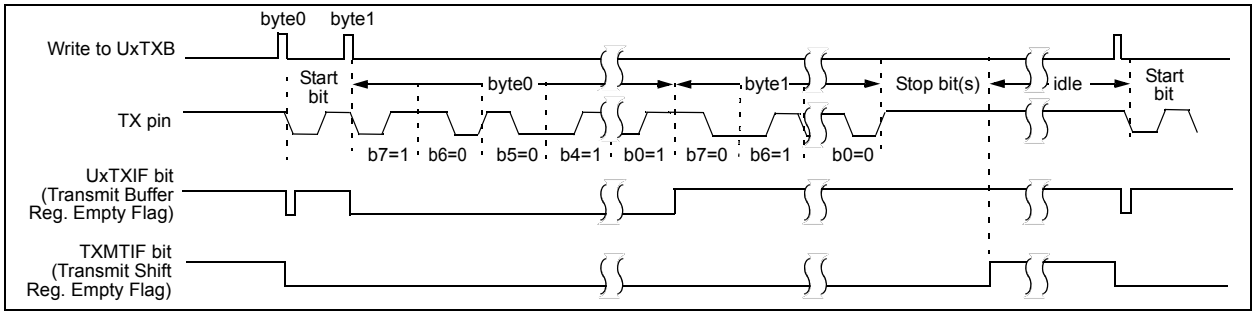


FIGURE 31-8: DALI FRAME TIMING

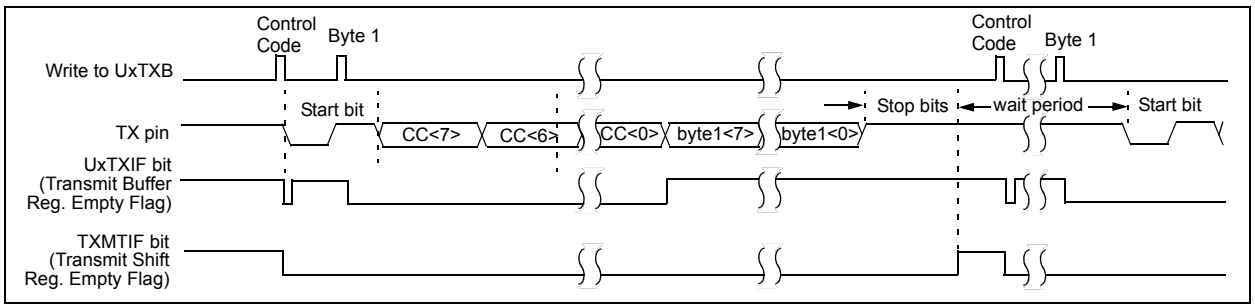
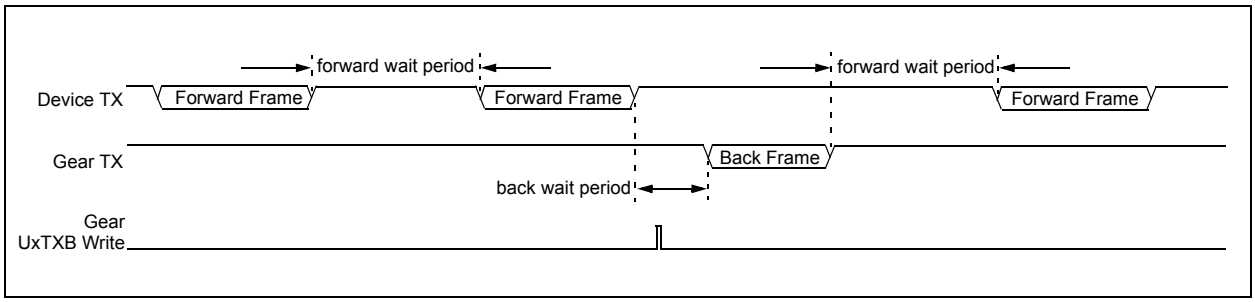


FIGURE 31-9: DALI FORWARD/BACK FRAME TIMING



31.7 General Purpose Manchester (UART1 only)

General purpose Manchester is a subset of the DALI mode. When the UxP1L register is cleared, there is no minimum wait time between frames. This allows full and half-duplex operation because writes to the UxTXB are not held waiting for a receive operation to complete.

General purpose Manchester operation maintains all other aspects of DALI mode such as:

- Single-pulse Start bit
- Most Significant bit first
- No stop periods between back-to-back bytes

General purpose Manchester mode is configured with the following settings:

- MODE<3:0> = 1000
- TXEN = 1
- RXEN = 1
- UxP1 = 0h
- UxBRGH:L = desired baud rate
- TXPOL and RXPOL = desired Idle state
- STP = desired number of stop periods
- RxyPPS = TX pin selection code
- TX pin TRIS control = 0
- RXPPS = RX pin selection code
- RX pin TRIS control = 1
- Input pin ANSEL bit = 0
- ON = 1

The Manchester bit stream timing is shown in Figure 31-7.

31.17.1 AUTO-BAUD DETECT

The UART module supports automatic detection and calibration of the baud rate in the 8-bit asynchronous and LIN modes. However, setting ABDEN to start auto-baud detection is neither necessary, nor possible in LIN mode because that mode supports auto-baud detection automatically at the beginning of every data packet. Enabling auto-baud detect with the ABDEN bit applies to the asynchronous modes only.

Note:	In DALI Mode, ABDEN is ignored. The baud rate needs to be manually set to 1200 using the BRG registers.

When Auto-Baud Detect (ABD) is active, the clock to the BRG is reversed. Rather than the BRG clocking the incoming RX signal, the RX signal is timing the BRG. The Baud Rate Generator is used to time the period of a received 55h (ASCII “U”), which is the Sync character for the LIN bus. The unique feature of this character is that it has five falling edges, including the Start bit edge, five rising edges including the Stop bit edge.

In 8-bit Asynchronous mode, setting the ABDEN bit in the UxCON0 register enables the auto-baud calibration sequence. The first falling edge of the RX input after ABDEN is set will start the auto-baud calibration sequence. While the ABD sequence takes place, the UART state machine is held in idle. On the first falling edge of the receive line, the UxBRG begins counting up using the BRG counter clock as shown in Figure 31-12. The fifth falling edge will occur on the RX pin at the beginning of the bit 7 period. At that time, an accumulated value totaling the proper BRG period is left in the UxBRGH, UxBRGL register pair, the ABDEN bit is automatically cleared and the ABDIF interrupt flag is set. ABDIF must be cleared by software.

RXIDL indicates that the sync input is active. RXIDL will go low on the first falling edge and go high on the fifth rising edge.

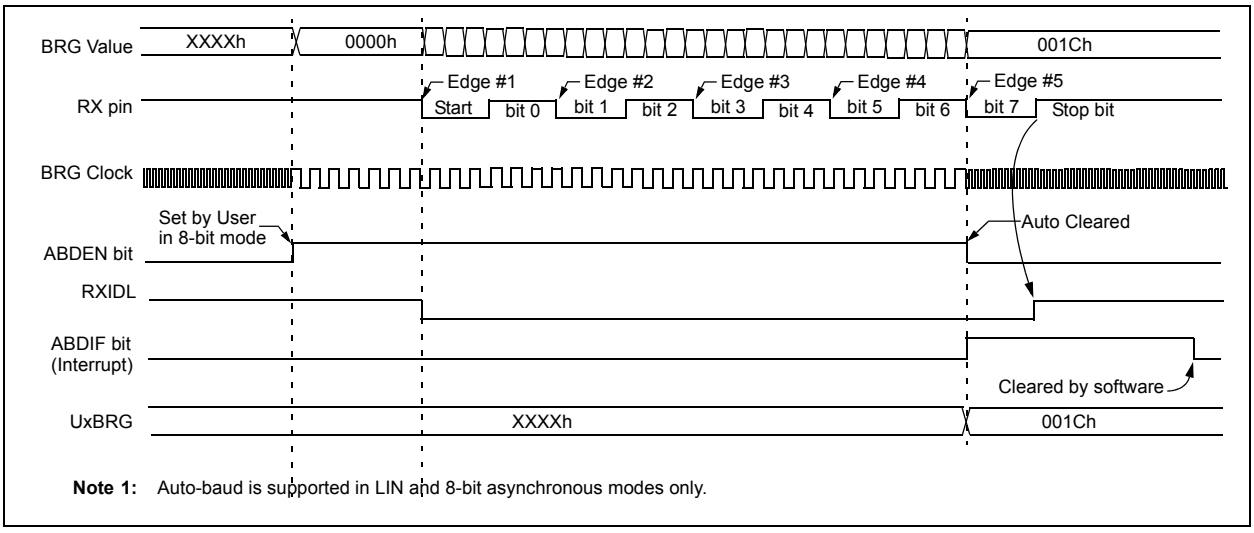
The BRG auto-baud clock is determined by the BRGS bit as shown in Table 31-2. During ABD, the internal BRG register is used as a 16-bit counter. However, the UxBRGH and UxBRGL registers retain the previous BRG value until the auto-baud process is successfully completed. While calibrating the baud rate period, the internal BRG register is clocked at 1/8th the BRG base clock rate. The resulting byte measurement is the average bit time when clocked at full speed and is transferred to the UxBRGH and UxBRGL registers when complete.

Note 1:	If the WUE bit is set with the ABDEN bit, auto-baud detection will occur on the byte following the Break character (see **Section 31.17.3 “Auto-Wake-up on Break”**).
2:	It is up to the user to determine that the incoming character baud rate is within the range of the selected BRG clock source. Some combinations of oscillator frequency and UART baud rates are not possible.

TABLE 31-2: BRG COUNTER CLOCK RATES

BRGS	BRG Base Clock	BRG ABD Clock
1	FOSC/4	FOSC/32
0	FOSC/16	FOSC/128

FIGURE 31-12: AUTOMATIC BAUD RATE CALIBRATION



Note 1: Auto-baud is supported in LIN and 8-bit asynchronous modes only.

REGISTER 32-4: SPIxTCNTH: SPI TRANSFER COUNTER MSB REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	—	TCNT10	TCNT9	TCNT8
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'

bit 7-3 **Unimplemented**: Read as '0'

bit 2-0 TCNT<10:8>:

BMODE = 0

Bits 13-11 of the Transfer Counter, counting the total number of bits to transfer

BMODE = 1

Bits 10-8 of the Transfer Counter, counting the total number of bytes to transfer

Note: This register should not be written to while a transfer is in progress (BUSY bit of SPIxCON2 is set).

REGISTER 32-5: SPIxTWIDTH: SPI TRANSFER WIDTH REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	—	TWIDTH2	TWIDTH1	TWIDTH0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'

bit 7-3 **Unimplemented**: Read as '0'

bit 2-0 TWIDTH<2:0>:

BMODE = 0

Bits 2-0 of the Transfer Counter, counting the total number of bits to transfer

BMODE = 1

Size (in bits) of each transfer counted by the transfer counter

111 = 7 bits

110 = 6 bits

101 = 5 bits

100 = 4 bits

011 = 3 bits

010 = 2 bits

001 = 1 bit

000 = 8 bits

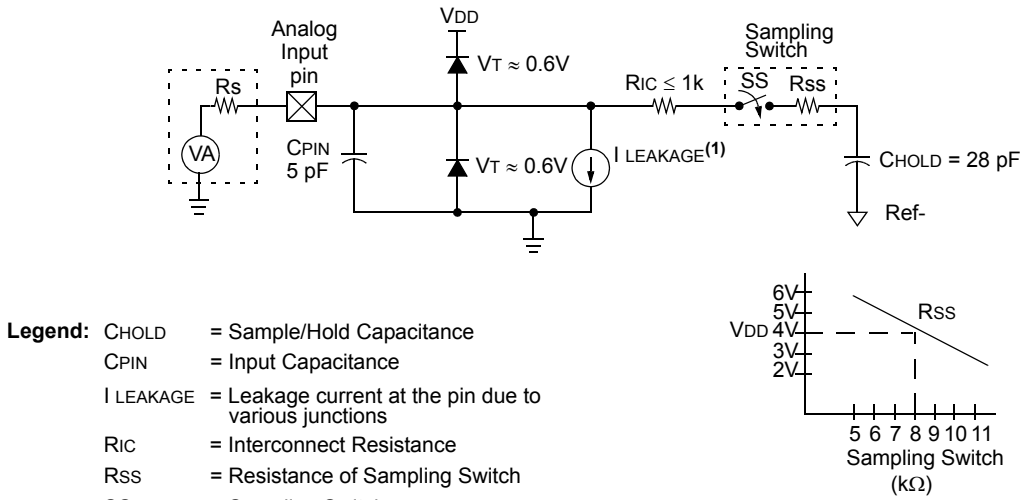
Note: This register should not be written to while a transfer is in progress (BUSY bit of SPIxCON2 is set).

33.4.3.5 Slave Transmission (10-bit Addressing Mode)

This section describes the sequence of events for the I^2C module configured as an I^2C slave in 10-bit Addressing mode and is transmitting data. Figure 33-12 is used as a visual reference for this description.

1. Master asserts Start condition (can also be a restart) on the bus. Start condition Interrupt Flag (SCIF) in I2CxPIR register is set. If Start condition interrupt is enabled (SCIE bit is set), generic interrupt I2CxIF is set.
2. Master transmits high address byte with R/W = 0.
3. The received high address is compared with the values in I2CxADR1 and I2CxADR3 registers.
4. If high address matches; R/W is copied to R/W bit, D/A bit is cleared, high address data is copied to I2CxADB1. If the address does not match; module becomes idle.
5. If Address hold interrupt is enabled (ADRIE = 1), CSTR is set. I2CxIF is set.
6. Slave software can read high address from I2CxADB1 and set/clear ACKDT before releasing SCL.
7. ACKDT value is copied out to SDA for ACK pulse. SCL line is released by clearing CSTR.
8. Master sends ninth SCL pulse for ACK.
9. Slave can force a NACK at this point due to previous error not being cleared. E.g. Receive buffer overflow or transmit buffer underflow errors. In these cases the Slave hardware forces a NACK and the module becomes idle.
10. Master transmits low address data byte.
11. If the low address matches; SMA is set, ADRIF is set, R/W is copied to R/W bit, D/A bit is cleared, low address data is copied to I2CxADB0, and ACTDT is copied to SDA. If the address does not match; module becomes idle.
12. If address hold interrupt is enabled, the CSTR bit is set as mentioned in step 6. Slave software can read low address byte from I2CxADB0 register and change ACKDT value before releasing SCL.
13. Master sends 9th SCL pulse for ACK.
14. If the Acknowledge interrupt and hold is enabled (ACKTIE = 1), CSTR is set, I2CxIF is set.
15. Slave software can read address from I2CxADB0 and I2CxADB1 registers and change the value of ACKDT before releasing SCL by clearing CSTR.
16. Master asserts Restart condition (cannot be Start) on the bus. Restart Condition Interrupt Flag (RSCIF) is set. If the Restart Condition Interrupt is enabled, generic interrupt I2CxIF is set.
17. Master transmits high address byte with R/W = 1.
18. If SMA = 1, and if high address matches; R/W is copied to R/W bit, D/A bit is cleared, high address data is copied to I2CxADB1, and ACTDT is output to SDA. If the address does not match or SMA = 0; module become idle.
19. If ADRIE = 1, CSTR is set. I2CIF is set. Slave software can read address from I2CxADB0/1 and set/clear ACKDT. The ACKDT value is copied out to SDA. SCL is released by clearing CSTR bit.
20. If TXBE = 1 and I2CCNT!= 0 (I2CTXIF = 1), CSTR is set. Slave software must load data into I2CxTXB to release SCL.
21. Master sends SCL pulse for ACK. If I2CCNT = 0, CNTIF is set.
22. If NACK; NACKIF is set, slave goes idle.
23. If ACKTIE = 1, CSTR is set, I2CIF is set. Slave software can read address from I2CxADB0/1 before releasing SCL by clearing CSTR.
24. Master sends eight SCL pulses to clock out data.
25. Go to step 20.

FIGURE 36-4: ANALOG INPUT MODEL

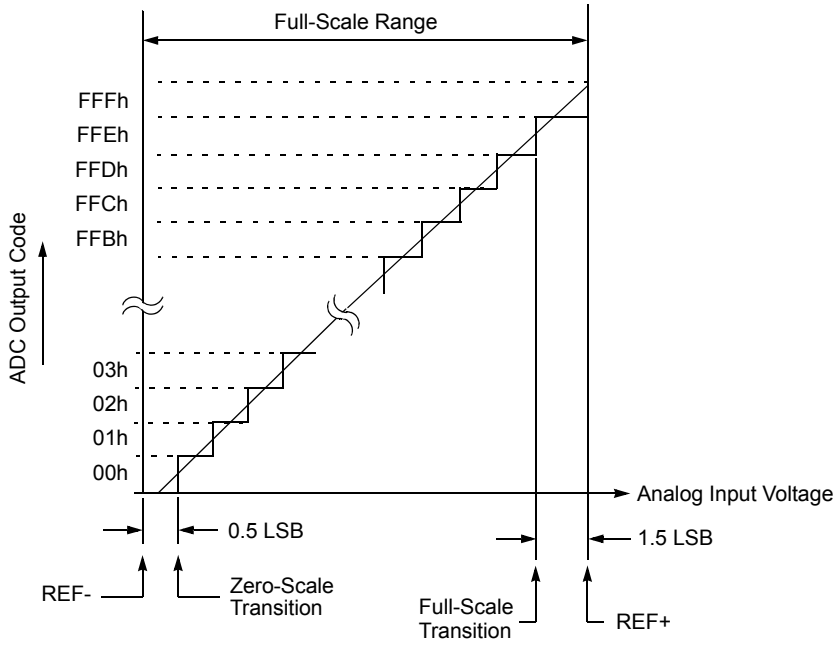


Legend:

CHOLD	= Sample/Hold Capacitance
CPIN	= Input Capacitance
I LEAKAGE	= Leakage current at the pin due to various junctions
RIC	= Interconnect Resistance
RSS	= Resistance of Sampling Switch
SS	= Sampling Switch
VT	= Threshold Voltage

Note 1: Refer to Table 44-6 (parameter D340 and D341).

FIGURE 36-5: ADC TRANSFER FUNCTION



36.7 Register Definitions: ADC Control

REGISTER 36-1: ADCON0: ADC CONTROL REGISTER 0

R/W-0/0	R/W-0/0	U-0	R/W-0/0	U-0	R/W-0/0	U-0	R/W/HC-0
ON	CONT	—	CS	—	FM	—	GO
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 **ON:** ADC Enable bit
1 = ADC is enabled
0 = ADC is disabled

bit 6 **CONT:** ADC Continuous Operation Enable bit
1 = GO is retriggered upon completion of each conversion trigger until ADTIF is set (if ADSOI is set) or until GO is cleared (regardless of the value of ADSOI)
0 = ADC is cleared upon completion of each conversion trigger

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

bit 4 **CS:** ADC Clock Selection bit
1 = Clock supplied from FRC dedicated oscillator
0 = Clock supplied by FOSC, divided according to ADCLK register

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

bit 2 **FM:** ADC results Format/alignment Selection
1 = ADRES and PREV data are right-justified
0 = ADRES and PREV data are left-justified, zero-filled

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

bit 0 **GO:** ADC Conversion Status bit(1)
1 = ADC conversion cycle in progress. Setting this bit starts an ADC conversion cycle. The bit is cleared by hardware as determined by the CONT bit
0 = ADC conversion completed/not in progress

Note 1: This bit requires ON bit to be set.

2: If cleared by software while a conversion is in progress, the results of the conversion up to this point will be transfered to ADRES and the state machine will be reset, but the ADIF interrupt flag bit will not be set; filter and threshold operations will not be performed.

37.0 5-BIT DIGITAL-TO-ANALOG CONVERTER (DAC) MODULE

The Digital-to-Analog Converter supplies a variable voltage reference, ratiometric with the input source, with 32 selectable output levels.

The positive input source (VSOURCE+) of the DAC can be connected to:

- FVR Buffer
- External VREF+ pin
- VDD supply voltage

The negative input source (VSOURCE-) of the DAC can be connected to:

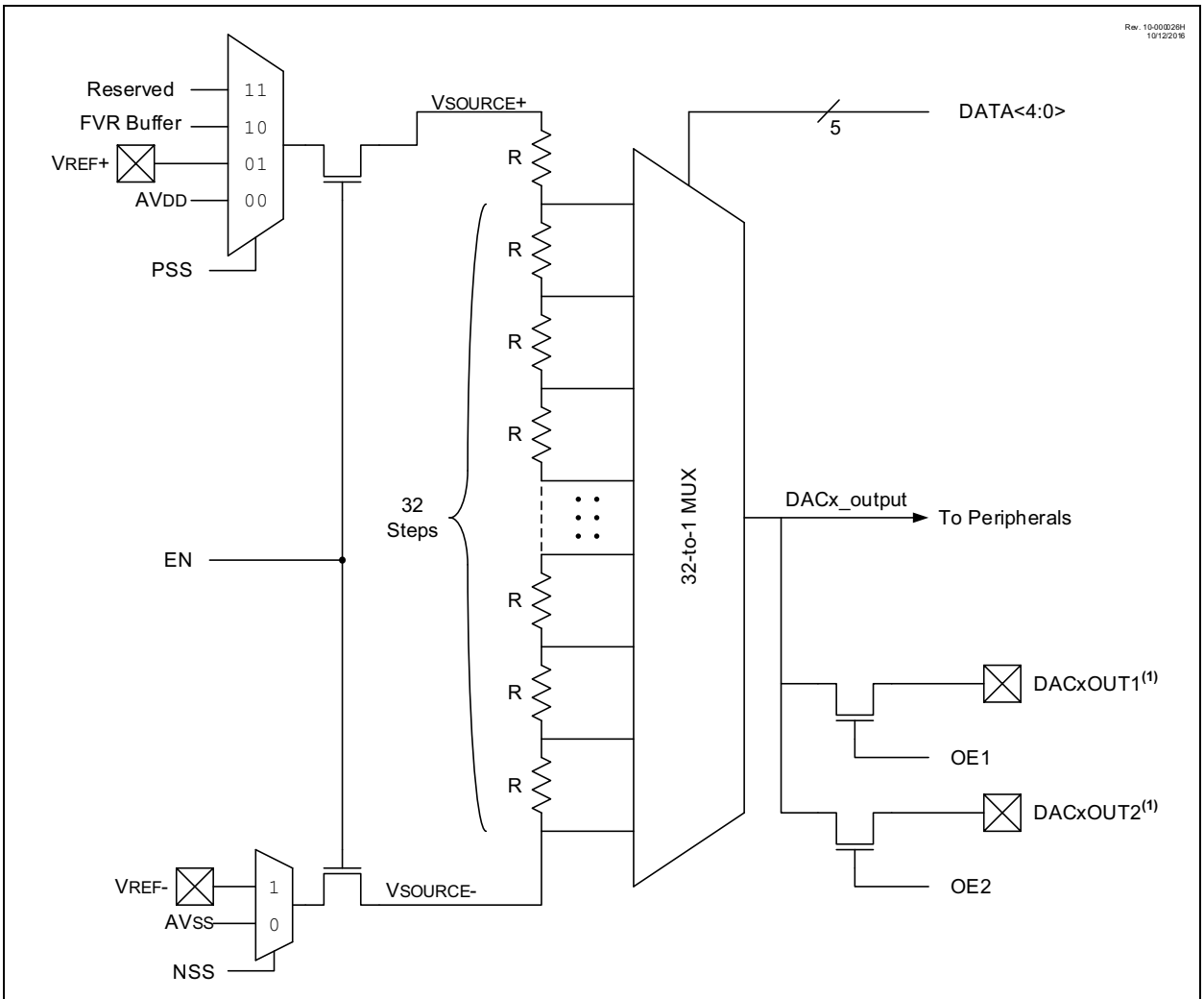
- External VREF- pin
- Vss

The output of the DAC (DAC1_output) can be selected as a reference voltage to the following:

- Comparator positive input
- ADC input channel
- DAC1OUT1 pin
- DAC1OUT2 pin

The Digital-to-Analog Converter (DAC) can be enabled by setting the EN bit of the DAC1CON0 register.

FIGURE 37-1: DIGITAL-TO-ANALOG CONVERTER BLOCK DIAGRAM



Note 1: The unbuffered DACx_output is provided on the DACxOUT pin(s).

38.13 Register Definitions: Comparator Control

Long bit name prefixes for the Comparators are shown in Table 38-2. Refer to **Section 1.3.2.2 "Long Bit Names"** for more information.

TABLE 38-2:

Peripheral	Bit Name Prefix
C1	C1
C2	C2

REGISTER 38-1: CMxCON0: COMPARATOR x CONTROL REGISTER 0

R/W-0/0	R-0/0	U-0	R/W-0/0	U-0	U-1	R/W-0/0	R/W-0/0
EN	OUT	—	POL	—	—	HYS	SYNC
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 **EN:** Comparator Enable bit
- 1 = Comparator is enabled
- 0 = Comparator is disabled and consumes no active power

bit 6 **OUT:** Comparator Output bit

If POL = 0 (noninverted polarity):
- 1 = CxVP > CxVN
- 0 = CxVP < CxVN

If POL = 1 (inverted polarity):
- 1 = CxVP < CxVN
- 0 = CxVP > CxVN

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

bit 4 **POL:** Comparator Output Polarity Select bit
- 1 = Comparator output is inverted
- 0 = Comparator output is not inverted

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

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

bit 1 **HYS:** Comparator Hysteresis Enable bit
- 1 = Comparator hysteresis enabled
- 0 = Comparator hysteresis disabled

bit 0 **SYNC:** Comparator Output Synchronous Mode bit
- 1 = Comparator output to Timer1/3/5 and I/O pin is synchronous to changes on Timer1 clock source.
- 0 = Comparator output to Timer1/3/5 and I/O pin is asynchronous
  Output updated on the falling edge of Timer1/3/5 clock source.

TABLE 41-2: INSTRUCTION SET (CONTINUED)

Mnemonic, Operands		Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
				MSb			LSb		
LITERAL INSTRUCTIONS									
ADDLW	k	Add literal and WREG	1	0000	1111	kkkk	kkkk	C, DC, Z, OV, N	
ANDLW	k	AND literal with WREG	1	0000	1011	kkkk	kkkk	Z, N	
IORLW	k	Inclusive OR literal with WREG	1	0000	1001	kkkk	kkkk	Z, N	
LFSR	f_n, k	Load FSR(f_n) with a 14-bit literal (k)	2	1110	1110	00ff	kkkk	None	
				1111	00kk	kkkk	kkkk		
ADDFSR	f_n, k	Add FSR(f_n) with (k)	1	1110	1000	ffkk	kkkk	None	
SUBFSR	f_n, k	Subtract (k) from FSR(f_n)	1	1110	1001	ffkk	kkkk	None	
MOVLB	k	Move literal to BSR<5:0>	1	0000	0001	00kk	kkkk	None	
MOVLW	k	Move literal to WREG	1	0000	1110	kkkk	kkkk	None	
MULLW	k	Multiply literal with WREG	1	0000	1101	kkkk	kkkk	None	
RETLW	k	Return with literal in WREG	2	0000	1100	kkkk	kkkk	None	
SUBLW	k	Subtract WREG from literal	1	0000	1000	kkkk	kkkk	C, DC, Z, OV, N	
XORLW	k	Exclusive OR literal with WREG	1	0000	1010	kkkk	kkkk	Z, N	
DATA MEMORY – PROGRAM MEMORY INSTRUCTIONS									
TBLRD*		Table Read	2 - 5	0000	0000	0000	1000	None	
TBLRD*+		Table Read with post-increment		0000	0000	0000	1001	None	
TBLRD*-		Table Read with post-decrement		0000	0000	0000	1010	None	
TBLRD+*		Table Read with pre-increment		0000	0000	0000	1011	None	
TBLWT*		Table Write	2 - 5	0000	0000	0000	1100	None	
TBLWT*+		Table Write with post-increment		0000	0000	0000	1101	None	
TBLWT*-		Table Write with post-decrement		0000	0000	0000	1110	None	
TBLWT+*		Table Write with pre-increment		0000	0000	0000	1111	None	

Note 1: If Program Counter (PC) is modified or a conditional test is true, the instruction requires an additional cycle. The extra cycle is executed as a NOP.

2: Some instructions are multi word instructions. The second/third words of these instructions will be decoded as a NOP, unless the first word of the instruction retrieves the information embedded in these 16-bits. This ensures that all program memory locations have a valid instruction.

3: f_s and f_d do not cover the full memory range. 2 MSBs of bank selection are forced to ‘b00 to limit the range of these instructions to lower 4k addressing space.