Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit
Speed	20 MIPS
Connectivity	CANbus, I²C, SPI, UART/USART
Peripherals	AC'97, Brown-out Detect/Reset, I²S, LVD, POR, PWM, WDT
Number of I/O	68
Program Memory Size	144KB (48K x 24)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-TQFP
Supplier Device Package	80-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic30f6014-20i-pf

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

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

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

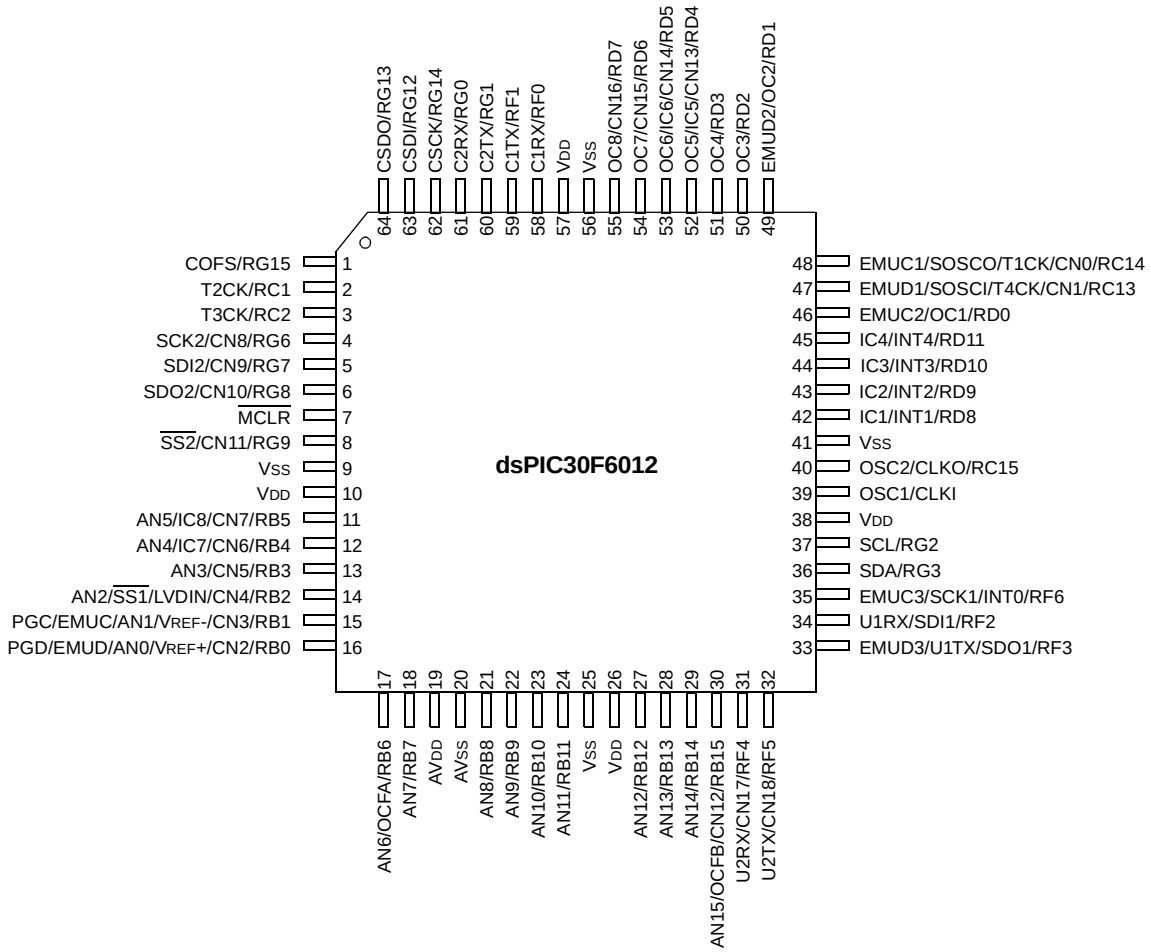
Trademarks

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona, Gresham, Oregon and Mountain View, California. The Company's quality system processes and procedures are for its PIC® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Pin Diagrams (Continued)

64-Pin TQFP

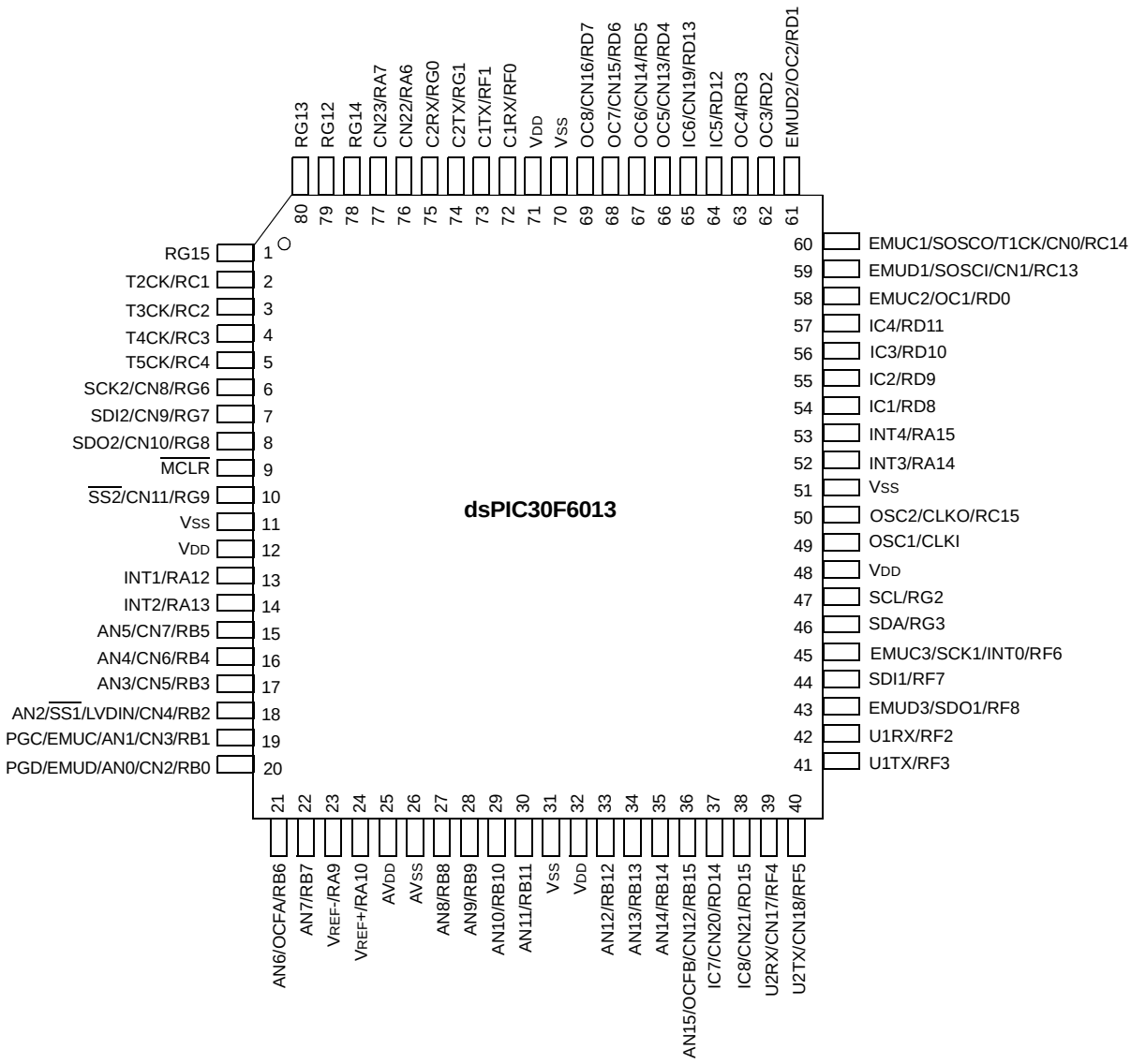


*dsPIC30F6012A recommended for new designs

Note: For descriptions of individual pins, see **Section 1.0 "Device Overview"**.

Pin Diagrams (Continued)

80-Pin TQFP



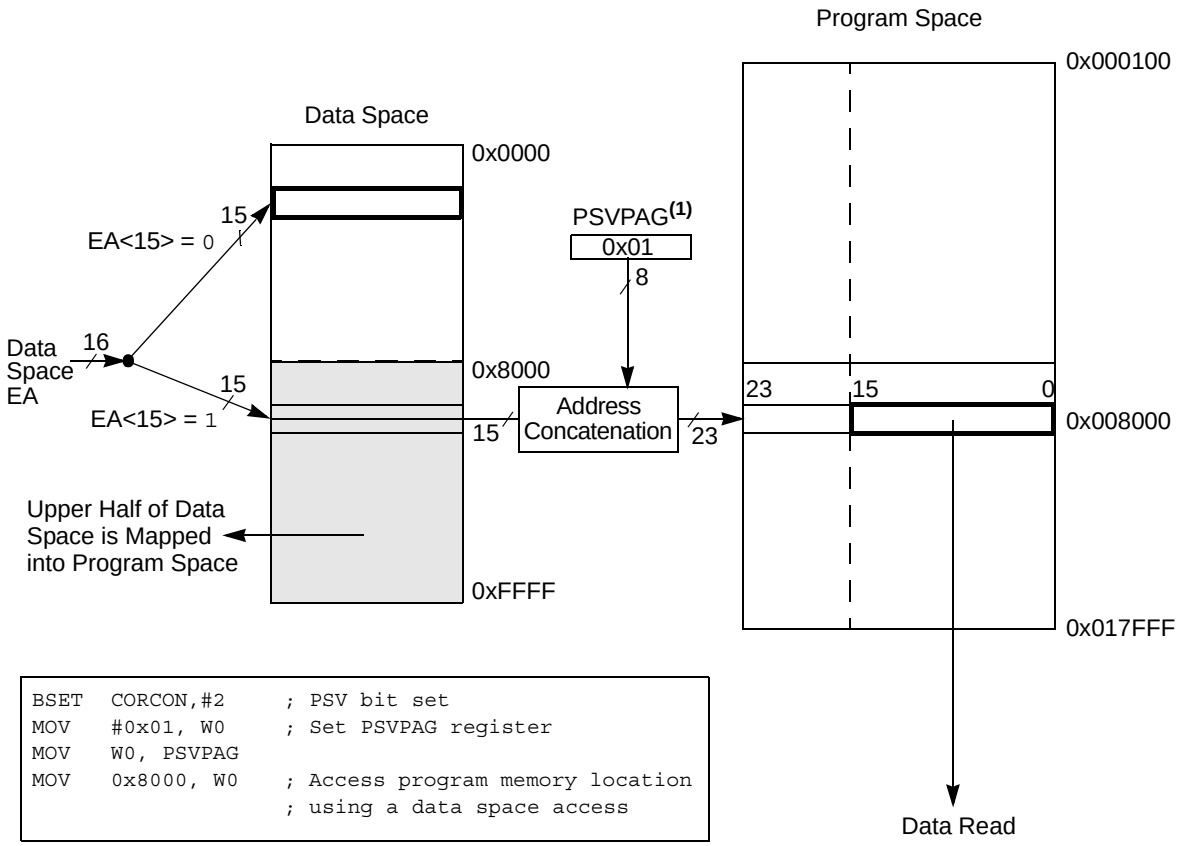
*dsPIC30F6013A recommended for new designs

Note: For descriptions of individual pins, see **Section 1.0 "Device Overview"**.

NOTES:

NOTES:

FIGURE 3-6: DATA SPACE WINDOW INTO PROGRAM SPACE OPERATION

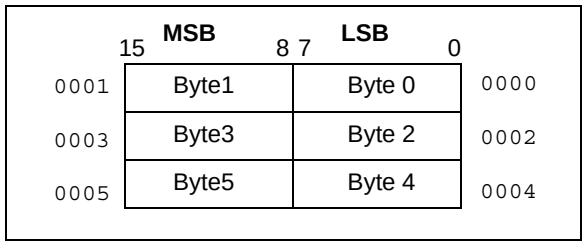



```
BSET   CORCON,#2     ; PSV bit set
MOV    #0x01, W0     ; Set PSVPAG register
MOV    W0, PSVPAG
MOV    0x8000, W0    ; Access program memory location
                     ; using a data space access
```

Note: PSVPAG is an 8-bit register, containing bits <22:15> of the program space address (i.e., it defines the page in program space to which the upper half of data space is being mapped).

All word accesses must be aligned to an even address. Misaligned word data fetches are not supported so care must be taken when mixing byte and word operations, or translating from 8-bit MCU code. Should a misaligned read or write be attempted, an address error trap will be generated. If the error occurred on a read, the instruction underway is completed, whereas if it occurred on a write, the instruction will be executed but the write will not occur. In either case, a trap will then be executed, allowing the system and/or user to examine the machine state prior to execution of the address fault.

FIGURE 3-10: DATA ALIGNMENT



All byte loads into any W register are loaded into the LSB. The MSB is not modified.

A sign-extend (SE) instruction is provided to allow users to translate 8-bit signed data to 16-bit signed values. Alternatively, for 16-bit unsigned data, users can clear the MSB of any W register by executing a zero-extend (ZE) instruction on the appropriate address.

Although most instructions are capable of operating on word or byte data sizes, it should be noted that some instructions, including the DSP instructions, operate only on words.

3.2.5 NEAR DATA SPACE

An 8-Kbyte ‘near’ data space is reserved in X address memory space between 0x0000 and 0x1FFF, which is directly addressable via a 13-bit absolute address field within all memory direct instructions. The remaining X address space and all of the Y address space is addressable indirectly. Additionally, the whole of X data space is addressable using MOV instructions, which support memory direct addressing with a 16-bit address field.

3.2.6 SOFTWARE STACK

The dsPIC DSC devices contain a software stack. W15 is used as the Stack Pointer.

The Stack Pointer always points to the first available free word and grows from lower addresses towards higher addresses. It pre-decrements for stack pops and post-increments for stack pushes as shown in Figure 3-11. Note that for a PC push during any CALL instruction, the MSB of the PC is zero-extended before the push, ensuring that the MSB is always clear.

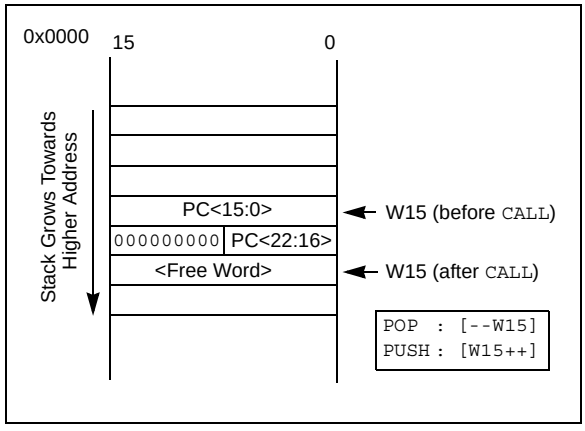
Note:	A PC push during exception processing will concatenate the SRL register to the MSB of the PC prior to the push.

There is a Stack Pointer Limit register (SPLIM) associated with the Stack Pointer. SPLIM is uninitialized at Reset. As is the case for the Stack Pointer, SPLIM<0> is forced to ‘0’ because all stack operations must be word aligned. Whenever an Effective Address (EA) is generated using W15 as a source or destination pointer, the address thus generated is compared with the value in SPLIM. If the contents of the Stack Pointer (W15) and the SPLIM register are equal and a push operation is performed, a stack error trap will not occur. The stack error trap will occur on a subsequent push operation. Thus, for example, if it is desirable to cause a stack error trap when the stack grows beyond address 0x2000 in RAM, initialize the SPLIM with the value, 0x1FFE.

Similarly, a Stack Pointer underflow (stack error) trap is generated when the Stack Pointer address is found to be less than 0x0800, thus preventing the stack from interfering with the Special Function Register (SFR) space.

A write to the SPLIM register should not be immediately followed by an indirect read operation using W15.

FIGURE 3-11: CALL STACK FRAME



5.0 INTERRUPTS

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the "*dsPIC30F Family Reference Manual*" (DS70046). For more information on the device instruction set and programming, refer to the "*dsPIC30F/33F Programmer's Reference Manual*" (DS70157).

The dsPIC30F Sensor and General Purpose Family has up to 41 interrupt sources and 4 processor exceptions (traps) which must be arbitrated based on a priority scheme.

The CPU is responsible for reading the Interrupt Vector Table (IVT) and transferring the address contained in the interrupt vector to the program counter. The interrupt vector is transferred from the program data bus into the program counter via a 24-bit wide multiplexer on the input of the program counter.

The Interrupt Vector Table (IVT) and Alternate Interrupt Vector Table (AIVT) are placed near the beginning of program memory (0x000004). The IVT and AIVT are shown in Table 5-1.

The interrupt controller is responsible for preprocessing the interrupts and processor exceptions prior to them being presented to the processor core. The peripheral interrupts and traps are enabled, prioritized and controlled using centralized Special Function Registers:

- IFS0<15:0>, IFS1<15:0>, IFS2<15:0>
  All interrupt request flags are maintained in these three registers. The flags are set by their respective peripherals or external signals, and they are cleared via software.
- IEC0<15:0>, IEC1<15:0>, IEC2<15:0>
  All interrupt enable control bits are maintained in these three registers. These control bits are used to individually enable interrupts from the peripherals or external signals.
- IPC0<15:0>... IPC10<7:0>
  The user assignable priority level associated with each of these 41 interrupts is held centrally in these twelve registers.
- IPL<3:0>
  The current CPU priority level is explicitly stored in the IPL bits. IPL<3> is present in the CORCON register, whereas IPL<2:0> are present in the STATUS register (SR) in the processor core.
- INTCON1<15:0>, INTCON2<15:0>
  Global interrupt control functions are derived from these two registers. INTCON1 contains the control and status flags for the processor exceptions. The INTCON2 register controls the external interrupt request signal behavior and the use of the alternate vector table.

Note: Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

All interrupt sources can be user assigned to one of 7 priority levels, 1 through 7, via the IPCx registers. Each interrupt source is associated with an interrupt vector, as shown in Table 5-1. Levels 7 and 1 represent the highest and lowest maskable priorities, respectively.

Note: Assigning a priority level of '0' to an interrupt source is equivalent to disabling that interrupt.

If the NSTDIS bit (INTCON1<15>) is set, nesting of interrupts is prevented. Thus, if an interrupt is currently being serviced, processing of a new interrupt is prevented even if the new interrupt is of higher priority than the one currently being serviced.

Note: The IPL bits become read only whenever the NSTDIS bit has been set to '1'.

Certain interrupts have specialized control bits for features like edge or level triggered interrupts, interrupt-on-change, etc. Control of these features remains within the peripheral module which generates the interrupt.

The `DISI` instruction can be used to disable the processing of interrupts of priorities 6 and lower for a certain number of instructions, during which the DISI bit (INTCON2<14>) remains set.

When an interrupt is serviced, the PC is loaded with the address stored in the vector location in program memory that corresponds to the interrupt. There are 63 different vectors within the IVT (refer to Table 5-1). These vectors are contained in locations 0x000004 through 0x0000FE of program memory (refer to Table 5-1). These locations contain 24-bit addresses and in order to preserve robustness, an address error trap will take place should the PC attempt to fetch any of these words during normal execution. This prevents execution of random data as a result of accidentally decrementing a PC into vector space, accidentally mapping a data space address into vector space, or the PC rolling over to 0x000000 after reaching the end of implemented program memory space. Execution of a `GOTO` instruction to this vector space will also generate an address error trap.

13.0 OUTPUT COMPARE MODULE

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the "*dsPIC30F Family Reference Manual*" (DS70046).

This section describes the output compare module and associated operational modes. The features provided by this module are useful in applications requiring operational modes, such as:

- Generation of Variable Width Output Pulses
- Power Factor Correction

Figure 13-1 depicts a block diagram of the output compare module.

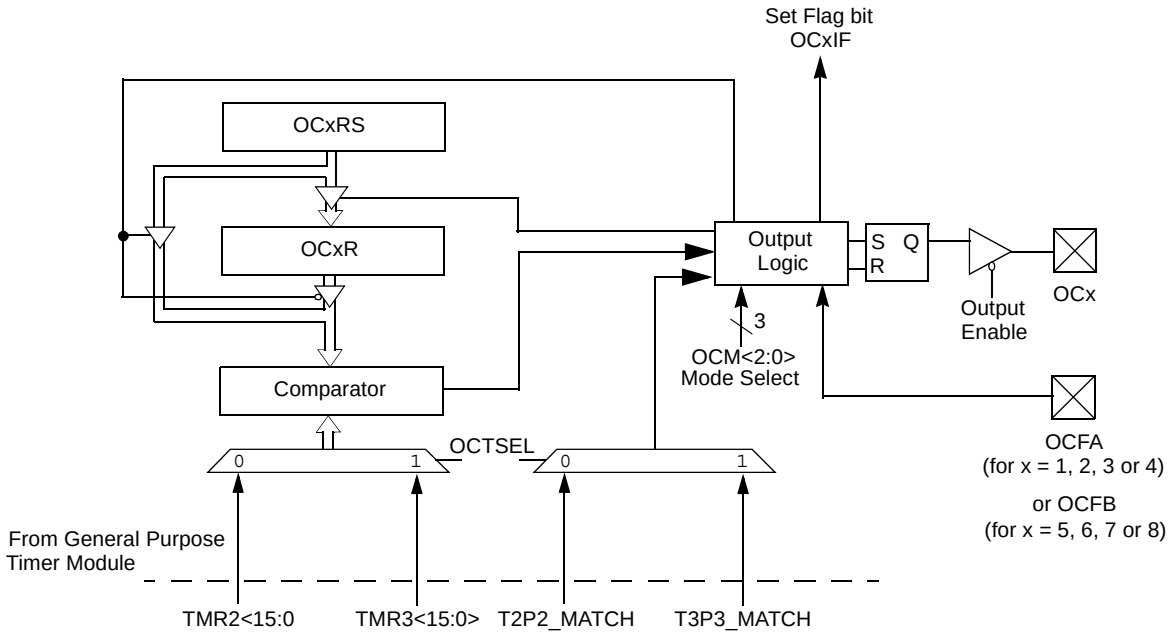
The key operational features of the output compare module include:

- Timer2 and Timer3 Selection mode
- Simple Output Compare Match mode
- Dual Output Compare Match mode
- Simple PWM mode
- Output Compare During Sleep and Idle modes
- Interrupt on Output Compare/PWM Event

These operating modes are determined by setting the appropriate bits in the 16-bit OCxCON SFR (where x = 1,2,3,...,N). The dsPIC DSC devices contain up to 8 compare channels (i.e., the maximum value of N is 8).

OCxRS and OCxR in Figure 13-1 represent the Dual Compare registers. In the Dual Compare mode, the OCxR register is used for the first compare and OCxRS is used for the second compare.

FIGURE 13-1: OUTPUT COMPARE MODE BLOCK DIAGRAM



Note: Where 'x' is shown, reference is made to the registers associated with the respective output compare channels 1 through N.

15.12.3 BAUD RATE GENERATOR

In I^2C Master mode, the reload value for the BRG is located in the I2CBRG register. When the BRG is loaded with this value, the BRG counts down to '0' and stops until another reload has taken place. If clock arbitration is taking place, for instance, the BRG is reloaded when the SCL pin is sampled high.

As per the I^2C standard, FSCK may be 100 kHz or 400 kHz. However, the user can specify any baud rate up to 1 MHz. I2CBRG values of '0' or '1' are illegal.

EQUATION 15-1: SERIAL CLOCK RATE

$$I2CBRG = \left(\frac{FCY}{FSCK} - \frac{FCY}{1{,}111{,}111} \right) - 1$$

15.12.4 CLOCK ARBITRATION

Clock arbitration occurs when the master de-asserts the SCL pin (SCL allowed to float high) during any receive, transmit, or Restart/Stop condition. When the SCL pin is allowed to float high, the Baud Rate Generator is suspended from counting until the SCL pin is actually sampled high. When the SCL pin is sampled high, the Baud Rate Generator is reloaded with the contents of I2CBRG and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count in the event that the clock is held low by an external device.

15.12.5 MULTI-MASTER COMMUNICATION, BUS COLLISION, AND BUS ARBITRATION

Multi-master operation support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA by letting SDA float high while another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin = 0, then a bus collision has taken place. The master will set the MI2CIF pulse and reset the master portion of the I^2C port to its Idle state.

If a transmit was in progress when the bus collision occurred, the transmission is halted, the TBF flag is cleared, the SDA and SCL lines are de-asserted and a value can now be written to I2CTRN. When the user services the I^2C master event Interrupt Service Routine, if the I^2C bus is free (i.e., the P bit is set), the user can resume communication by asserting a Start condition.

If a Start, Restart, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are de-asserted, and the respective control bits in the I2CCON register are cleared to '0'. When the user services the bus collision Interrupt Service Routine, and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

The master will continue to monitor the SDA and SCL pins, and if a Stop condition occurs, the MI2CIF bit will be set.

A write to the I2CTRN will start the transmission of data at the first data bit regardless of where the transmitter left off when bus collision occurred.

In a multi-master environment, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I^2C bus can be taken when the P bit is set in the I2CSTAT register, or the bus is Idle and the S and P bits are cleared.

15.13 I^2C Module Operation During CPU Sleep and Idle Modes

15.13.1 I^2C OPERATION DURING CPU SLEEP MODE

When the device enters Sleep mode, all clock sources to the module are shutdown and stay at logic '0'. If Sleep occurs in the middle of a transmission and the state machine is partially into a transmission as the clocks stop, then the transmission is aborted. Similarly, if Sleep occurs in the middle of a reception, then the reception is aborted.

15.13.2 I^2C OPERATION DURING CPU IDLE MODE

For the I^2C, the I2CSIDL bit selects if the module will stop on Idle or continue on Idle. If I2CSIDL = 0, the module will continue operation on assertion of the Idle mode. If I2CSIDL = 1, the module will stop on Idle.

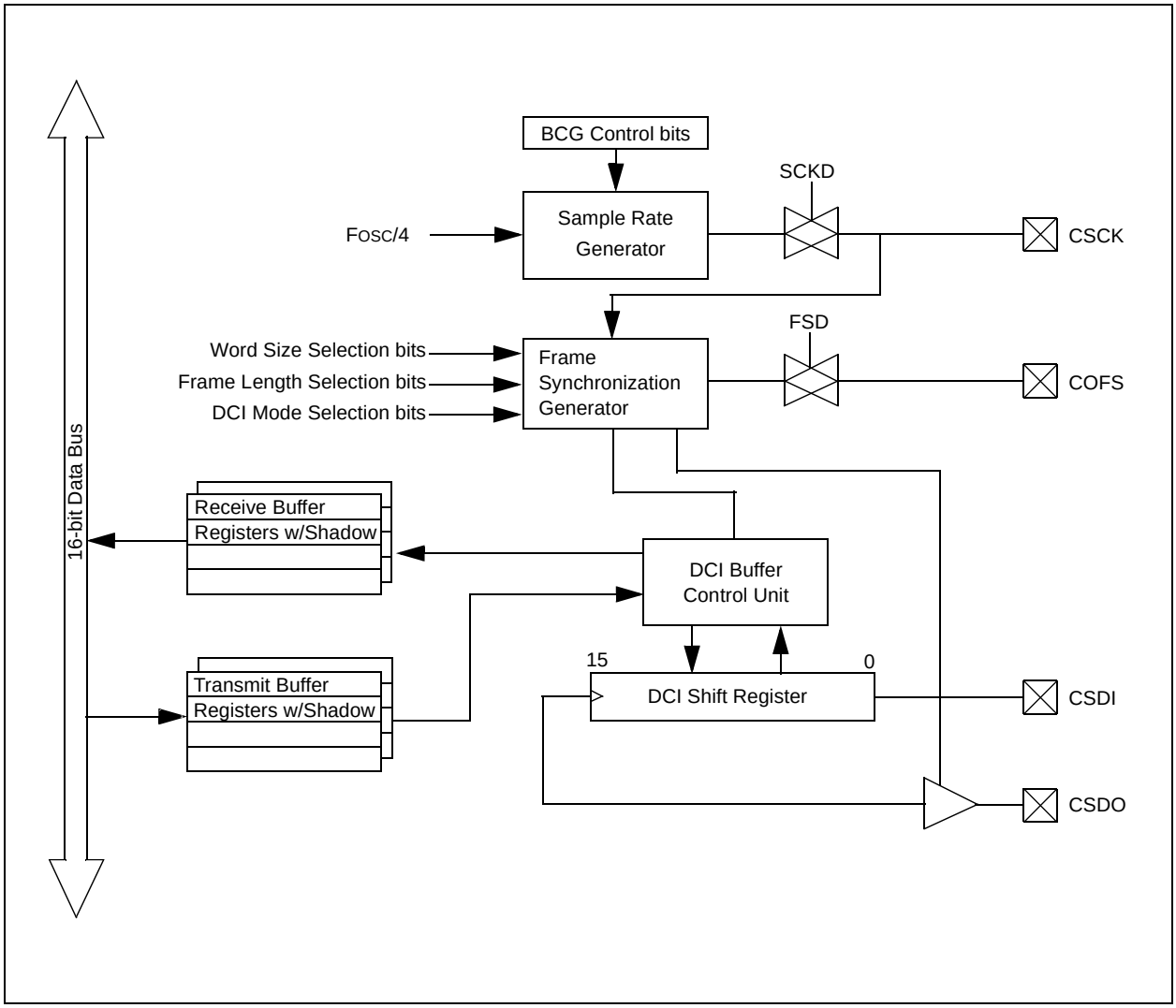
TABLE 17-1: CAN1 REGISTER MAP

SFR Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset State
C1RXF0SID	0300	—	—	—	Receive Acceptance Filter 0 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF0EIDH	0302	—	—	—	—	Receive Acceptance Filter 0 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF0EIDL	0304	Receive Acceptance Filter 0 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF1SID	0308	—	—	—	Receive Acceptance Filter 1 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF1EIDH	030A	—	—	—	—	Receive Acceptance Filter 1 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF1EIDL	030C	Receive Acceptance Filter 1 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF2SID	0310	—	—	—	Receive Acceptance Filter 2 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF2EIDH	0312	—	—	—	—	Receive Acceptance Filter 2 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF2EIDL	0314	Receive Acceptance Filter 2 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF3SID	0318	—	—	—	Receive Acceptance Filter 3 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF3EIDH	031A	—	—	—	—	Receive Acceptance Filter 3 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF3EIDL	031C	Receive Acceptance Filter 3 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF4SID	0320	—	—	—	Receive Acceptance Filter 4 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF4EIDH	0322	—	—	—	—	Receive Acceptance Filter 4 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF4EIDL	0324	Receive Acceptance Filter 4 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF5SID	0328	—	—	—	Receive Acceptance Filter 5 Standard Identifier <10:0>											—	EXIDE	000u uuuu uuuu uu0u
C1RXF5EIDH	032A	—	—	—	—	Receive Acceptance Filter 5 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXF5EIDL	032C	Receive Acceptance Filter 5 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXM0SID	0330	—	—	—	Receive Acceptance Mask 0 Standard Identifier <10:0>											—	MIDE	000u uuuu uuuu uu0u
C1RXM0EIDH	0332	—	—	—	—	Receive Acceptance Mask 0 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXM0EIDL	0334	Receive Acceptance Mask 0 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXM1SID	0338	—	—	—	Receive Acceptance Mask 1 Standard Identifier <10:0>											—	MIDE	000u uuuu uuuu uu0u
C1RXM1EIDH	033A	—	—	—	—	Receive Acceptance Mask 1 Extended Identifier <17:6>												0000 uuuu uuuu uuuu
C1RXM1EIDL	033C	Receive Acceptance Mask 1 Extended Identifier <5:0>						—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1TX2SID	0340	Transmit Buffer 2 Standard Identifier <10:6>					—	—	—	Transmit Buffer 2 Standard Identifier <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu
C1TX2EID	0342	Transmit Buffer 2 Extended Identifier <17:14>				—	—	—	—	Transmit Buffer 2 Extended Identifier <13:6>								uuuu 0000 uuuu uuuu
C1TX2DLC	0344	Transmit Buffer 2 Extended Identifier <5:0>						TXRTR	TXRB1	TXRB0	DLC<3:0>				—	—	—	uuuu uuuu uuuu u000
C1TX2B1	0346	Transmit Buffer 2 Byte 1								Transmit Buffer 2 Byte 0								uuuu uuuu uuuu uuuu
C1TX2B2	0348	Transmit Buffer 2 Byte 3								Transmit Buffer 2 Byte 2								uuuu uuuu uuuu uuuu
C1TX2B3	034A	Transmit Buffer 2 Byte 5								Transmit Buffer 2 Byte 4								uuuu uuuu uuuu uuuu
C1TX2B4	034C	Transmit Buffer 2 Byte 7								Transmit Buffer 2 Byte 6								uuuu uuuu uuuu uuuu
C1TX2CON	034E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>		0000 0000 0000 0000
C1TX1SID	0350	Transmit Buffer 1 Standard Identifier <10:6>					—	—	—	Transmit Buffer 1 Standard Identifier <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu

Legend: u = uninitialized bit

Note: Refer to "*dsPIC30F Family Reference Manual*" (DS70046) for descriptions of register bit fields.

FIGURE 18-1: DCI MODULE BLOCK DIAGRAM



18.3.8 SAMPLE CLOCK EDGE CONTROL BIT

The sample clock edge (CSCKE) control bit determines the sampling edge for the CSCK signal. If the CSCK bit is cleared (default), data will be sampled on the falling edge of the CSCK signal. The AC-Link protocols and most Multi-Channel formats require that data be sampled on the falling edge of the CSCK signal. If the CSCK bit is set, data will be sampled on the rising edge of CSCK. The I^2S protocol requires that data be sampled on the rising edge of the CSCK signal.

18.3.9 DATA JUSTIFICATION CONTROL BIT

In most applications, the data transfer begins one CSCK cycle after the COFS signal is sampled active. This is the default configuration of the DCI module. An alternate data alignment can be selected by setting the DJST control bit in the DCICON1 SFR. When DJST = 1, data transfers will begin during the same CSCK cycle when the COFS signal is sampled active.

18.3.10 TRANSMIT SLOT ENABLE BITS

The TSCON SFR has control bits that are used to enable up to 16 time slots for transmission. These control bits are the TSE<15:0> bits. The size of each time slot is determined by the WS<3:0> word size selection bits and can vary up to 16 bits.

If a transmit time slot is enabled via one of the TSE bits (TSEx = 1), the contents of the current transmit shadow buffer location will be loaded into the CSDO Shift register and the DCI buffer control unit is incremented to point to the next location.

During an unused transmit time slot, the CSDO pin will drive '0's or will be tri-stated during all disabled time slots depending on the state of the CSDOM bit in the DCICON1 SFR.

The data frame size in bits is determined by the chosen data word size and the number of data word elements in the frame. If the chosen frame size has less than 16 elements, the additional slot enable bits will have no effect.

Each transmit data word is written to the 16-bit transmit buffer as left justified data. If the selected word size is less than 16 bits, then the LSbs of the transmit buffer memory will have no effect on the transmitted data. The user should write '0's to the unused LSbs of each transmit buffer location.

18.3.11 RECEIVE SLOT ENABLE BITS

The RSCON SFR contains control bits that are used to enable up to 16 time slots for reception. These control bits are the RSE<15:0> bits. The size of each receive time slot is determined by the WS<3:0> word size selection bits and can vary from 1 to 16 bits.

If a receive time slot is enabled via one of the RSE bits (RSEx = 1), the shift register contents will be written to the current DCI receive shadow buffer location and the buffer control unit will be incremented to point to the next buffer location.

Data is not packed in the receive memory buffer locations if the selected word size is less than 16 bits. Each received slot data word is stored in a separate 16-bit buffer location. Data is always stored in a left justified format in the receive memory buffer.

18.3.12 SLOT ENABLE BITS OPERATION WITH FRAME SYNC

The TSE and RSE control bits operate in concert with the DCI frame sync generator. In the Master mode, a COFS signal is generated whenever the frame sync generator is reset. In the Slave mode, the frame sync generator is reset whenever a COFS pulse is received.

The TSE and RSE control bits allow up to 16 consecutive time slots to be enabled for transmit or receive. After the last enabled time slot has been transmitted/received, the DCI will stop buffering data until the next occurring COFS pulse.

18.3.13 SYNCHRONOUS DATA TRANSFERS

The DCI buffer control unit will be incremented by one word location whenever a given time slot has been enabled for transmission or reception. In most cases, data input and output transfers will be synchronized, which means that a data sample is received for a given channel at the same time a data sample is transmitted. Therefore, the transmit and receive buffers will be filled with equal amounts of data when a DCI interrupt is generated.

In some cases, the amount of data transmitted and received during a data frame may not be equal. As an example, assume a two-word data frame is used. Furthermore, assume that data is only received during slot #0 but is transmitted during slot #0 and slot #1. In this case, the buffer control unit counter would be incremented twice during a data frame but only one receive register location would be filled with data.

The 20-bit mode treats each 256-bit AC-Link frame as sixteen, 16-bit time slots. In the 20-bit AC-Link mode, the module operates as if COFSG<3:0> = 1111 and WS<3:0> = 1111. The data alignment for 20-bit data slots is ignored. For example, an entire AC-Link data frame can be transmitted and received in a packed fashion by setting all bits in the TSCON and RSCON SFRs. Since the total available buffer length is 64 bits, it would take 4 consecutive interrupts to transfer the AC-Link frame. The application software must keep track of the current AC-Link frame segment.

18.7 I²S Mode Operation

The DCI module is configured for I²S mode by writing a value of '01' to the COFSM<1:0> control bits in the DCICON1 SFR. When operating in the I²S mode, the DCI module will generate frame synchronization signals with a 50% duty cycle. Each edge of the frame synchronization signal marks the boundary of a new data word transfer.

The user must also select the frame length and data word size using the COFSG and WS control bits in the DCICON2 SFR.

18.7.1 I²S FRAME AND DATA WORD LENGTH SELECTION

The WS and COFSG control bits are set to produce the period for one half of an I²S data frame. That is, the frame length is the total number of CSCK cycles required for a left or a right data word transfer.

The BLEN bits must be set for the desired buffer length. Setting BLEN<1:0> = 01 will produce a CPU interrupt, once per I²S frame.

18.7.2 I²S DATA JUSTIFICATION

As per the I²S specification, a data word transfer will, by default, begin one CSCK cycle after a transition of the WS signal. A 'MS bit left justified' option can be selected using the DJST control bit in the DCICON2 SFR.

If DJST = 1, the I²S data transfers will be MSb left justified. The MSb of the data word will be presented on the CSDO pin during the same CSCK cycle as the rising or falling edge of the COFS signal. The CSDO pin is tri-stated after the data word has been sent.

19.9 Module Power-down Modes

The module has 2 internal Power modes.

When the ADON bit is '1', the module is in Active mode; it is fully powered and functional.

When ADON is '0', the module is in Off mode. The digital and analog portions of the circuit are disabled for maximum current savings.

In order to return to the Active mode from Off mode, the user must wait for the ADC circuitry to stabilize.

19.10 ADC Operation During CPU Sleep and Idle Modes

19.10.1 ADC OPERATION DURING CPU SLEEP MODE

When the device enters Sleep mode, all clock sources to the module are shutdown and stay at logic '0'.

If Sleep occurs in the middle of a conversion, the conversion is aborted. The converter will not continue with a partially completed conversion on exit from Sleep mode.

Register contents are not affected by the device entering or leaving Sleep mode.

The ADC module can operate during Sleep mode if the ADC clock source is set to RC (ADRC = 1). When the RC clock source is selected, the A/D module waits one instruction cycle before starting the conversion. This allows the SLEEP instruction to be executed which eliminates all digital switching noise from the conversion. When the conversion is complete, the CONV bit will be cleared and the result loaded into the ADCBUF register.

If the A/D interrupt is enabled, the device will wake-up from Sleep. If the A/D interrupt is not enabled, the ADC module will then be turned off, although the ADON bit will remain set.

19.10.2 A/D OPERATION DURING CPU IDLE MODE

The ADSIDL bit selects if the module will stop on Idle or continue on Idle. If ADSIDL = 0, the module will continue operation on assertion of Idle mode. If ADSIDL = 1, the module will stop on Idle.

19.11 Effects of a Reset

A device Reset forces all registers to their Reset state. This forces the ADC module to be turned off, and any conversion and sampling sequence is aborted. The values that are in the ADCBUF registers are not modified. The A/D Result register will contain unknown data after a Power-on Reset.

19.12 Output Formats

The ADC result is 12 bits wide. The data buffer RAM is also 12 bits wide. The 12-bit data can be read in one of four different formats. The FORM<1:0> bits select the format. Each of the output formats translates to a 16-bit result on the data bus.

FIGURE 19-5: ADC OUTPUT DATA FORMATS

RAM Contents:					d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
Read to Bus:																
Signed Fractional	$\overline{d11}$	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0
Fractional	d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00	0	0	0	0
Signed Integer	$\overline{d11}$	$\overline{d11}$	$\overline{d11}$	$\overline{d11}$	$\overline{d11}$	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00
Integer	0	0	0	0	d11	d10	d09	d08	d07	d06	d05	d04	d03	d02	d01	d00

TABLE 23-5: DC CHARACTERISTICS: OPERATING CURRENT (I_{DD})

DC CHARACTERISTICS			Standard Operating Conditions: 2.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Parameter No.	**Typical(1)**	**Max**	**Units**	**Conditions**		
Operating Current (I_{DD})(2)						
DC31a	6.8	10	mA	25°C	3.3V	0.128 MIPS LPRC (512 kHz)
DC31b	6.3	10	mA	85°C		
DC31c	6.1	10	mA	125°C		
DC31e	16	22	mA	25°C	5V	
DC31f	15	22	mA	85°C		
DC31g	15	22	mA	125°C		
DC30a	13	19	mA	25°C	3.3V	(1.8 MIPS) FRC (7.37 MHz)
DC30b	13	19	mA	85°C		
DC30c	13	19	mA	125°C		
DC30e	27	39	mA	25°C	5V	
DC30f	26	39	mA	85°C		
DC30g	25	39	mA	125°C		
DC23a	27	41	mA	25°C	3.3V	4 MIPS
DC23b	27	41	mA	85°C		
DC23c	27	41	mA	125°C		
DC23e	41	60	mA	25°C	5V	
DC23f	40	60	mA	85°C		
DC23g	40	60	mA	125°C		
DC24a	46	71	mA	25°C	3.3V	10 MIPS
DC24b	46	71	mA	85°C		
DC24c	47	71	mA	125°C		
DC24e	79	120	mA	25°C	5V	
DC24f	78	120	mA	85°C		
DC24g	78	120	mA	125°C		
DC27a	83	120	mA	25°C	3.3V	20 MIPS
DC27b	83	120	mA	85°C		
DC27d	138	190	mA	25°C	5V	
DC27e	137	190	mA	85°C		
DC27f	136	190	mA	125°C		
DC29a	194	255	mA	25°C	5V	30 MIPS
DC29b	192	255	mA	85°C		

Note 1: Data in "Typical" column is at 5V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature also have an impact on the current consumption. The test conditions for all I_{DD} measurements are as follows: OSC1 driven with external square wave from rail to rail. All I/O pins are configured as Inputs and pulled to V_{DD}. $\overline{MCLR}$ = V_{DD}, WDT, FSCM, LVD and BOR are disabled. CPU, SRAM, Program Memory and Data Memory are operational. No peripheral modules are operating.

TABLE 23-17: INTERNAL CLOCK TIMING EXAMPLES

Note 1: Assumption: Oscillator Postscaler is divide by 1.
2: Instruction Execution Cycle Time: TCY = 1 / MIPS.
3: Instruction Execution Frequency: MIPS = (FOSC * PLLx)/4 [since there are 4 Q clocks per instruction cycle].

TABLE 23-18: AC CHARACTERISTICS: INTERNAL RC ACCURACY(2)

AC CHARACTERISTICS		Standard Operating Conditions: 2.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended					
Param No.	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**	
	Internal FRC Jitter @ FRC Freq. = 7.37 MHz(1)						
OS62	FRC	—	±0.04	±0.16	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
		—	±0.07	±0.23	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5-5.5V
	Internal FRC Accuracy @ FRC Freq. = 7.37 MHz(1)						
OS63	FRC	—	—	±1.50	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0-5.5V
	Internal FRC Drift @ FRC Freq. = 7.37 MHz(1)						
OS64		-0.7	—	0.5	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
		-0.7	—	0.7	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0-3.6V
		-0.7	—	0.5	%	-40°C ≤ TA ≤ +85°C	VDD = 4.5-5.5V
		-0.7	—	0.7	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5-5.5V

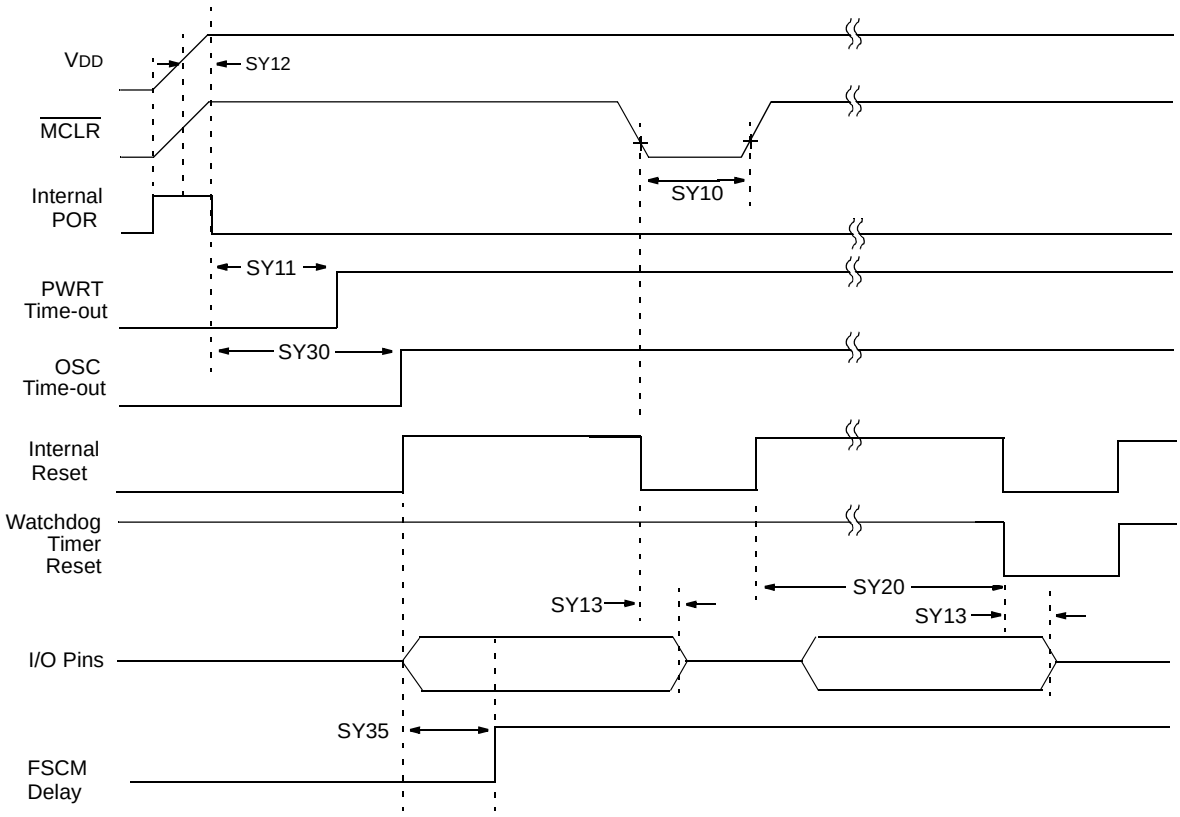
Note 1: Frequency calibrated at 7.372 MHz ±2%, 25°C and 5V. TUN <3:0> bits can be used to compensate for temperature drift.
2: Overall FRC variation can be calculated by adding the absolute values of jitter, accuracy and drift percentages.

TABLE 23-19: INTERNAL RC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 2.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
	LPRC @ Freq. = 512 kHz(1)					
OS65		-35	—	+35	%	—

Note 1: Change of LPRC frequency as VDD changes.

FIGURE 23-6: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING CHARACTERISTICS



Note: Refer to Figure 23-3 for load conditions.

TABLE 23-21: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER AND BROWN-OUT RESET TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic(1)**	**Min**	**Typ(2)**	**Max**	**Units**	**Conditions**
SY10	TmcL	$\overline{\text{MCLR}}$ Pulse Width (low)	2	—	—	µs	-40°C to +85°C
SY11	TPWRT	Power-up Timer Period	3 12 50	4 16 64	6 22 90	ms	-40°C to +85°C User programmable
SY12	TPOR	Power On Reset Delay	3	10	30	µs	-40°C to +85°C
SY13	TIOZ	I/O High-impedance from $\overline{\text{MCLR}}$ Low or Watchdog Timer Reset	—	0.8	1.0	µs	
SY20	TWDT1	Watchdog Timer Time-out Period (No Prescaler)	1.4	2.1	2.8	ms	VDD = 3.3V, -40°C to +85°C
	TWDT2		1.4	2.1	2.8	ms	VDD = 5.0V, -40°C to +85°C

Note 1: These parameters are characterized but not tested in manufacturing.
2: Data in "Typ" column is at 5V, 25°C unless otherwise stated.
3: Refer to Figure 23-2 and Table 23-11 for BOR.

NOTES:

READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

To: Technical Publications Manager Total Pages Sent _____

RE: Reader Response

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

Telephone: (_____) _____ - _____ FAX: (_____) _____ - _____

Application (optional):

Would you like a reply? ___Y ___N

Device: dsPIC30F6011/6012/6013/6014 Literature Number: DS70117F

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this document easy to follow? If not, why?

4. What additions to the document do you think would enhance the structure and subject?

5. What deletions from the document could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?