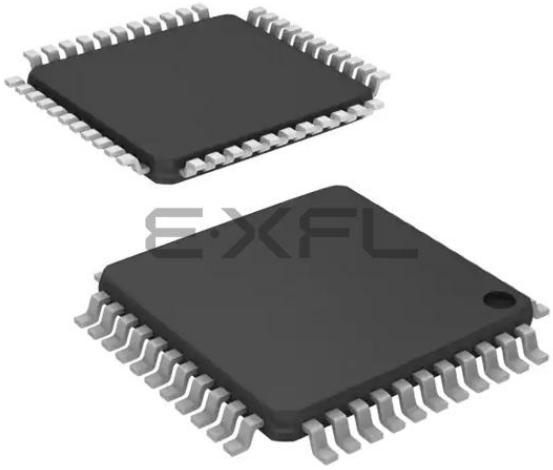
Microchip Technology - DSPIC33FJ128MC804-H/PT Datasheet



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, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, QEI, WDT
Number of I/O	35
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 9x10b/12b; D/A 6x16b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 150°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33fj128mc804-h-pt

FIGURE 3-2: dsPIC33FJ32MC302/304, dsPIC33FJ64MCX02/X04 AND dsPIC33FJ128MCX02/X04 PROGRAMMER'S MODEL

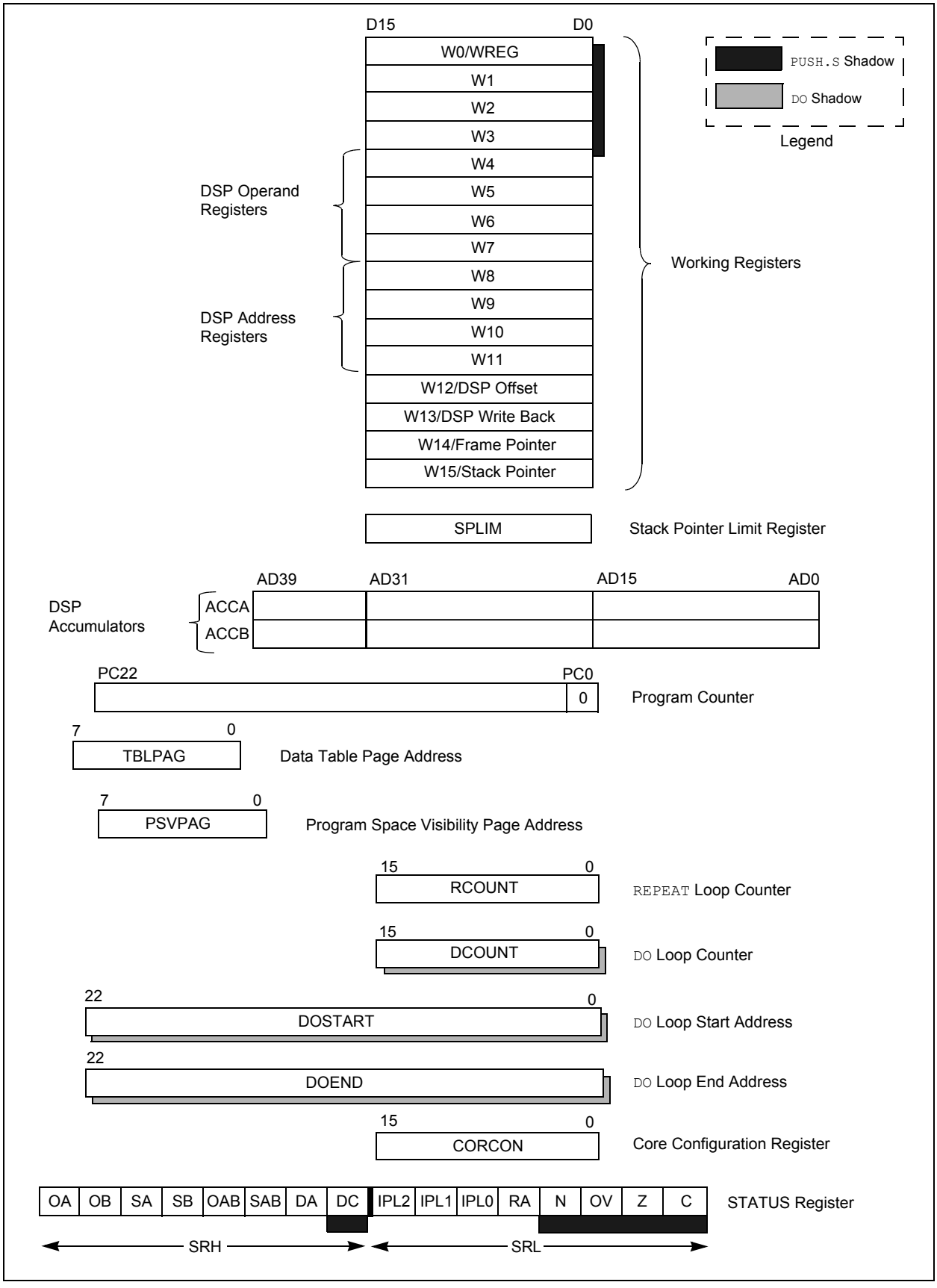


FIGURE 4-3: DATA MEMORY MAP FOR dsPIC33FJ32MC302/304 DEVICES WITH 4 KB RAM

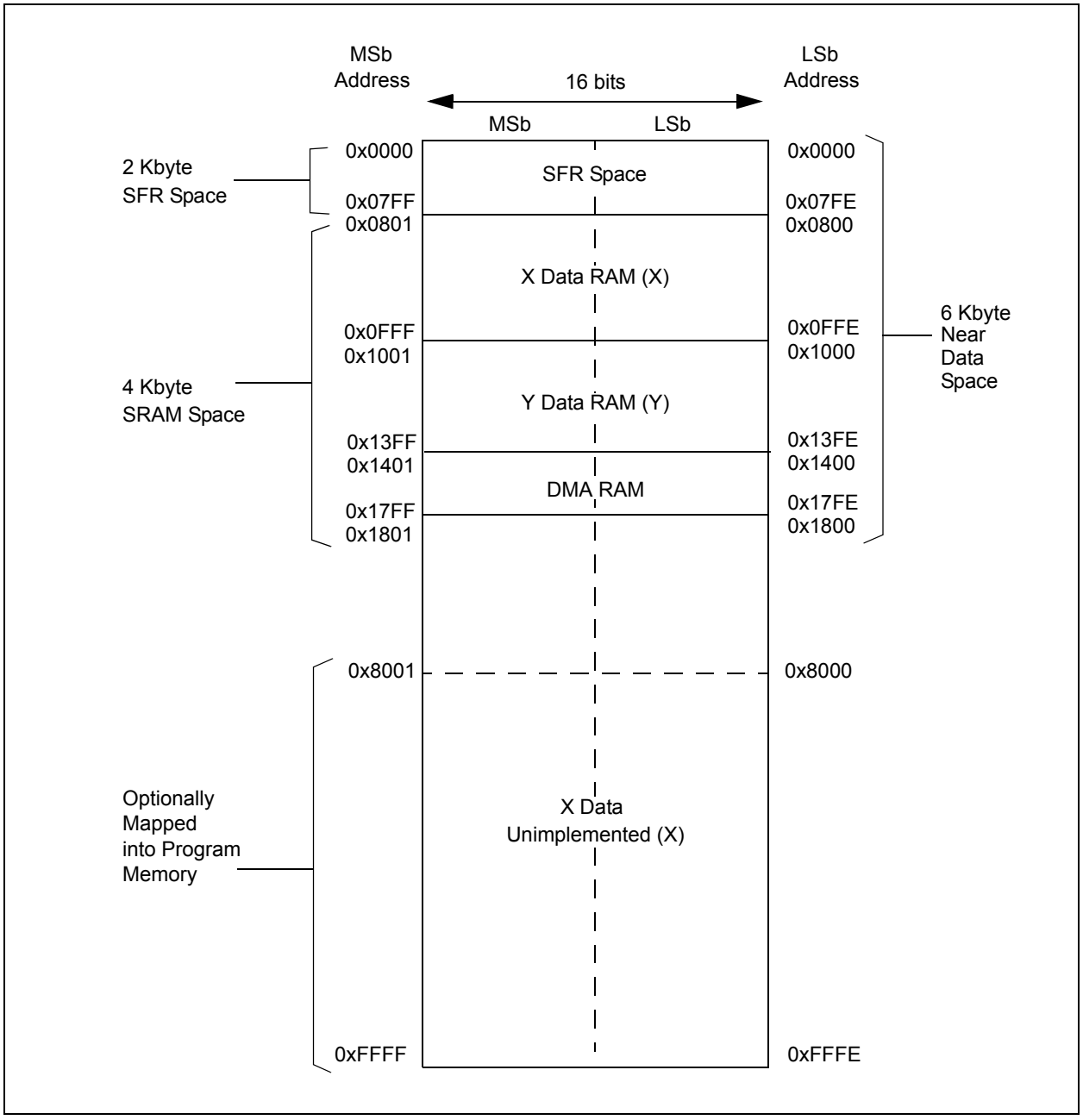


TABLE 4-24: PERIPHERAL PIN SELECT INPUT REGISTER MAP

File 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	All Resets
RPINR0	0680	—	—	—	INT1R<4:0>					—	—	—	—	—	—	—	—	1F00
RPINR1	0682	—	—	—	—	—	—	—	—	—	—	—	INT2R<4:0>					001F
RPINR3	0686	—	—	—	T3CKR<4:0>					—	—	—	T2CKR<4:0>					1F1F
RPINR4	0688	—	—	—	T5CKR<4:0>					—	—	—	T4CKR<4:0>					1F1F
RPINR7	068E	—	—	—	IC2R<4:0>					—	—	—	IC1R<4:0>					1F1F
RPINR10	0694	—	—	—	IC8R<4:0>					—	—	—	IC7R<4:0>					1F1F
RPINR11	0696	—	—	—	—	—	—	—	—	—	—	—	OCFAR<4:0>					001F
RPINR12	0698	—	—	—	—	—	—	—	—	—	—	—	FLTA1R<4:0>					001F
RPINR13	069A	—	—	—	—	—	—	—	—	—	—	—	FLTA2R<4:0>					001F
RPINR14	069C	—	—	—	QEB1R<4:0>					—	—	—	QEA1R<4:0>					1F1F
RPINR15	069E	—	—	—	—	—	—	—	—	—	—	—	INDX1R<4:0>					001F
RPINR16	06A0	—	—	—	QEB2R<4:0>					—	—	—	QEA2R<4:0>					1F1F
RPINR17	06A2	—	—	—	—	—	—	—	—	—	—	—	INDX2R<4:0>					001F
RPINR18	06A4	—	—	—	U1CTSR<4:0>					—	—	—	U1RXR<4:0>					1F1F
RPINR19	06A6	—	—	—	U2CTSR<4:0>					—	—	—	U2RXR<4:0>					1F1F
RPINR20	06A8	—	—	—	SCK1R<4:0>					—	—	—	SDI1R<4:0>					1F1F
RPINR21	06AA	—	—	—	—	—	—	—	—	—	—	—	SS1R<4:0>					001F
RPINR22	06AC	—	—	—	SCK2R<4:0>					—	—	—	SDI2R<4:0>					1F1F
RPINR23	06AE	—	—	—	—	—	—	—	—	—	—	—	SS2R<4:0>					001F
RPINR26(1)	06B4	—	—	—	—	—	—	—	—	—	—	—	C1RXR<4:0>					001F

Legend: x = unknown value on Reset, — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: This register is present in dsPIC33FJ128MC802/804 and dsPIC33FJ64MC802/804 devices only.

4.8.3 READING DATA FROM PROGRAM MEMORY USING PROGRAM SPACE VISIBILITY

The upper 32 Kbytes of data space may optionally be mapped into any 16K word page of the program space. This option provides transparent access to stored constant data from the data space without the need to use special instructions (such as TBLRDH).

Program space access through the data space occurs if the Most Significant bit of the data space EA is '1' and program space visibility is enabled by setting the PSV bit in the Core Control register (CORCON<2>). The location of the program memory space to be mapped into the data space is determined by the Program Space Visibility Page register (PSVPAG). This 8-bit register defines any one of 256 possible pages of 16K words in program space. In effect, PSVPAG functions as the upper 8 bits of the program memory address, with the 15 bits of the EA functioning as the lower bits. By incrementing the PC by 2 for each program memory word, the lower 15 bits of data space addresses directly map to the lower 15 bits in the corresponding program space addresses.

Data reads to this area add a cycle to the instruction being executed, since two program memory fetches are required.

Although each data space address 0x8000 and higher maps directly into a corresponding program memory address (see Figure 4-11), only the lower 16 bits of the 24-bit program word are used to contain the data. The upper 8 bits of any program space location used as data should be programmed with '1111 1111' or '0000 0000' to force a NOP. This prevents possible issues should the area of code ever be accidentally executed.

Note:	PSV access is temporarily disabled during table reads/writes.

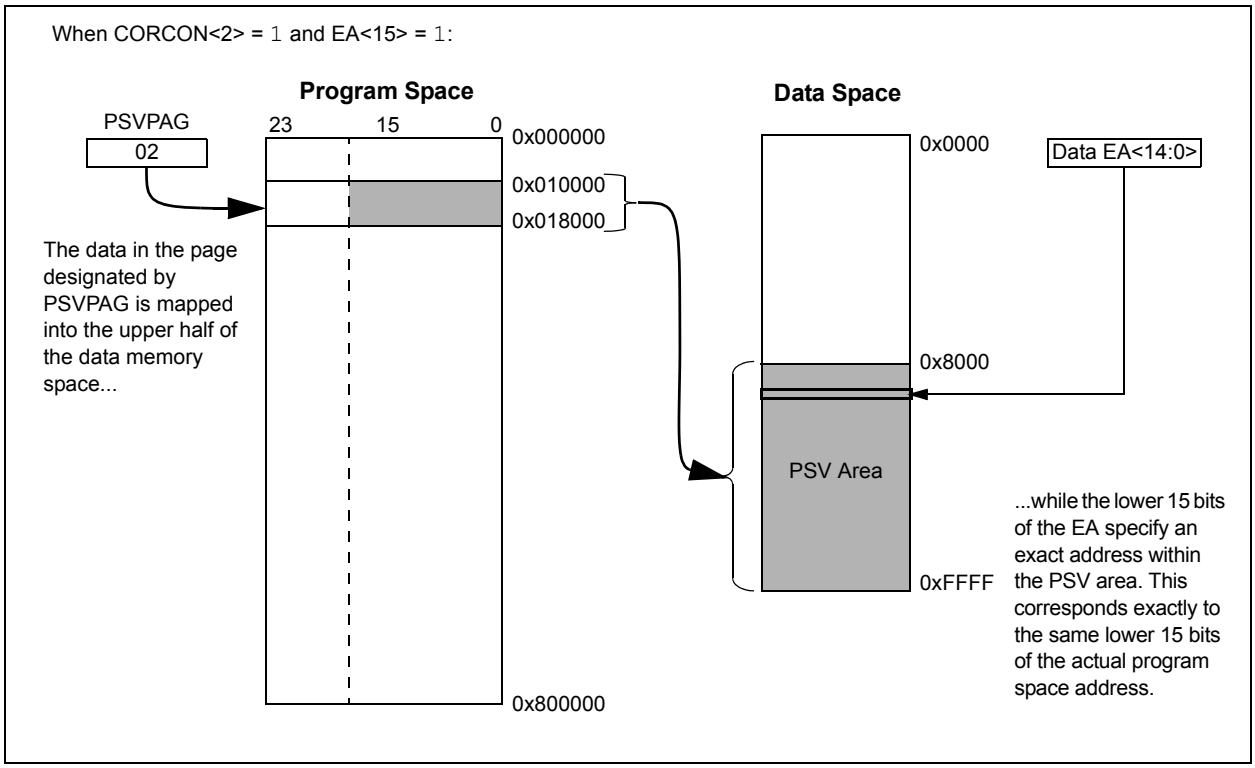
For operations that use PSV and are executed outside a REPEAT loop, the MOV and MOV.D instructions require one instruction cycle in addition to the specified execution time. All other instructions require two instruction cycles in addition to the specified execution time.

For operations that use PSV, and are executed inside a REPEAT loop, these instances require two instruction cycles in addition to the specified execution time of the instruction:

- Execution in the first iteration
- Execution in the last iteration
- Execution prior to exiting the loop due to an interrupt
- Execution upon re-entering the loop after an interrupt is serviced

Any other iteration of the REPEAT loop allows the instruction using PSV to access data, to execute in a single cycle.

FIGURE 4-11: PROGRAM SPACE VISIBILITY OPERATION



EXAMPLE 5-2: LOADING THE WRITE BUFFERS

```
; Set up NVMCON for row programming operations
        MOV     #0x4001, W0             ;
        MOV     W0, NVMCON              ; Initialize NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
        MOV     #0x0000, W0             ;
        MOV     W0, TBLPAG              ; Initialize PM Page Boundary SFR
        MOV     #0x6000, W0             ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
        MOV     #LOW_WORD_0, W2         ;
        MOV     #HIGH_BYTE_0, W3        ;
        TBLWTL  W2, [W0]                ; Write PM low word into program latch
        TBLWTH  W3, [W0++]              ; Write PM high byte into program latch
; 1st_program_word
        MOV     #LOW_WORD_1, W2         ;
        MOV     #HIGH_BYTE_1, W3        ;
        TBLWTL  W2, [W0]                ; Write PM low word into program latch
        TBLWTH  W3, [W0++]              ; Write PM high byte into program latch
;  2nd_program_word
        MOV     #LOW_WORD_2, W2         ;
        MOV     #HIGH_BYTE_2, W3        ;
        TBLWTL  W2, [W0]                ; Write PM low word into program latch
        TBLWTH  W3, [W0++]              ; Write PM high byte into program latch
        •
        •
        •
; 63rd_program_word
        MOV     #LOW_WORD_31, W2        ;
        MOV     #HIGH_BYTE_31, W3       ;
        TBLWTL  W2, [W0]                ; Write PM low word into program latch
        TBLWTH  W3, [W0++]              ; Write PM high byte into program latch
```

EXAMPLE 5-3: INITIATING A PROGRAMMING SEQUENCE

```
        DISI    #5                      ; Block all interrupts with priority < 7
                                        ; for next 5 instructions
        MOV     #0x55, W0
        MOV     W0, NVMKEY              ; Write the 55 key
        MOV     #0xAA, W1               ;
        MOV     W1, NVMKEY              ; Write the AA key
        BSET    NVMCON, #WR             ; Start the erase sequence
        NOP                             ; Insert two NOPs after the
        NOP                             ; erase command is asserted
```

REGISTER 7-5: IFS0: INTERRUPT FLAG STATUS REGISTER 0 (CONTINUED)

bit 2 **OC1IF:** Output Compare Channel 1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 1 **IC1IF:** Input Capture Channel 1 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 0 **INT0IF:** External Interrupt 0 Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 7-10: IEC0: INTERRUPT ENABLE CONTROL REGISTER 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	INT0IE
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 15 **Unimplemented:** Read as '0'

bit 14 **DMA1IE:** DMA Channel 1 Data Transfer Complete Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 13 **AD1IE:** ADC1 Conversion Complete Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 12 **U1TXIE:** UART1 Transmitter Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 11 **U1RXIE:** UART1 Receiver Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 10 **SPI1IE:** SPI1 Event Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 9 **SPI1EIE:** SPI1 Error Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 8 **T3IE:** Timer3 Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 7 **T2IE:** Timer2 Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 6 **OC2IE:** Output Compare Channel 2 Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 5 **IC2IE:** Input Capture Channel 2 Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 4 **DMA0IE:** DMA Channel 0 Data Transfer Complete Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

bit 3 **T1IE:** Timer1 Interrupt Enable bit
1 = Interrupt request enabled
0 = Interrupt request not enabled

REGISTER 7-32: INTTREG: INTERRUPT CONTROL AND STATUS REGISTER

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	ILR<3:0>			
bit 15							bit 8

U-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
—	VECNUM<6:0>						
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 15-12 **Unimplemented:** Read as '0'

bit 11-8 **ILR<3:0>:** New CPU Interrupt Priority Level bits
1111 = CPU Interrupt Priority Level is 15
•
•
•
0001 = CPU Interrupt Priority Level is 1
0000 = CPU Interrupt Priority Level is 0

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

bit 6-0 **VECNUM<6:0>:** Vector Number of Pending Interrupt bits
0111111 = Interrupt Vector pending is number 135
•
•
•
0000001 = Interrupt Vector pending is number 9
0000000 = Interrupt Vector pending is number 8

8.0 DIRECT MEMORY ACCESS (DMA)

Note 1: This data sheet summarizes the features of the dsPIC33FJ32MC302/304, dsPIC33FJ64MCX02/X04 and dsPIC33FJ128MCX02/X04 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 38. "Direct Memory Access (DMA) (Part III)"** (DS70215) of the "*dsPIC33F/PIC24H Family Reference Manual*", which is available from the Microchip web site (www.microchip.com).

2: Some registers and associated bits described in this section may not be available on all devices. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

Direct Memory Access (DMA) is a very efficient mechanism of copying data between peripheral SFRs (e.g., UART Receive register, Input Capture 1 buffer), and buffers or variables stored in RAM, with minimal CPU intervention. The DMA controller can automatically copy entire blocks of data without requiring the user software to read or write the peripheral Special Function Registers (SFRs) every time a peripheral interrupt occurs. The DMA controller uses a dedicated bus for data transfers and therefore, does not steal cycles from the code execution flow of the CPU. To exploit the DMA capability, the corresponding user buffers or variables must be located in DMA RAM.

The dsPIC33FJ32MC302/304, dsPIC33FJ64MCX02/X04 and dsPIC33FJ128MCX02/X04 peripherals that can utilize DMA are listed in Table 8-1.

TABLE 8-1: DMA CHANNEL TO PERIPHERAL ASSOCIATIONS

Peripheral to DMA Association	DMAxREQ Register IRQSEL<6:0> Bits	DMAxPAD Register Values to Read From Peripheral	DMAxPAD Register Values to Write to Peripheral
INT0 – External Interrupt 0	0000000	—	—
IC1 – Input Capture 1	0000001	0x0140 (IC1BUF)	—
OC1 – Output Compare 1 Data	0000010	—	0x0182 (OC1R)
OC1 – Output Compare 1 Secondary Data	0000010	—	0x0180 (OC1RS)
IC2 – Input Capture 2	0000101	0x0144 (IC2BUF)	—
OC2 – Output Compare 2 Data	0000110	—	0x0188 (OC2R)
OC2 – Output Compare 2 Secondary Data	0000110	—	0x0186 (OC2RS)
TMR2 – Timer2	0000111	—	—
TMR3 – Timer3	0001000	—	—
SPI1 – Transfer Done	0001010	0x0248 (SPI1BUF)	0x0248 (SPI1BUF)
UART1RX – UART1 Receiver	0001011	0x0226 (U1RXREG)	—
UART1TX – UART1 Transmitter	0001100	—	0x0224 (U1TXREG)
ADC1 – ADC1 Convert Done	0001101	0x0300 (ADC1BUF0)	—
UART2RX – UART2 Receiver	0011110	0x0236 (U2RXREG)	—
UART2TX – UART2 Transmitter	0011111	—	0x0234 (U2TXREG)
SPI2 – Transfer Done	0100001	0x0268 (SPI2BUF)	0x0268 (SPI2BUF)
ECAN1 – RX Data Ready	0100010	0x0440 (C1RXD)	—
PMP - Master Data Transfer	0101101	0x0608 (PMDIN1)	0x0608 (PMDIN1)
ECAN1 – TX Data Request	1000110	—	0x0442 (C1TXD)
DAC1 - Right Data Output	1001110	—	0x3F6 (DAC1RDAT)
DAC2 - Left Data Output	1001111	—	0x03F8 (DAC1LDAT)

18.3 SPI Control Registers

REGISTER 18-1: SPIxSTAT: SPIx STATUS AND CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
SPIEN	—	SPISIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/C-0	U-0	U-0	U-0	U-0	R-0	R-0
—	SPIROV	—	—	—	—	SPITBF	SPIRBF
bit 7							bit 0

Legend:	C = Clearable bit		
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 15 **SPIEN:** SPIx Enable bit
1 = Enables module and configures SCKx, SDOx, SDIx and $\overline{SSx}$ as serial port pins
0 = Disables module

bit 14 **Unimplemented:** Read as '0'

bit 13 **SPISIDL:** Stop in Idle Mode bit
1 = Discontinue module operation when device enters Idle mode
0 = Continue module operation in Idle mode

bit 12-7 **Unimplemented:** Read as '0'

bit 6 **SPIROV:** Receive Overflow Flag bit
1 = A new byte/word is completely received and discarded. The user software has not read the previous data in the SPIxBUF register
0 = No overflow has occurred

bit 5-2 **Unimplemented:** Read as '0'

bit 1 **SPITBF:** SPIx Transmit Buffer Full Status bit
1 = Transmit not yet started, SPIxTXB is full
0 = Transmit started, SPIxTXB is empty
Automatically set in hardware when CPU writes SPIxBUF location, loading SPIxTXB
Automatically cleared in hardware when SPIx module transfers data from SPIxTXB to SPIxSR

bit 0 **SPIRBF:** SPIx Receive Buffer Full Status bit
1 = Receive complete, SPIxRXB is full
0 = Receive is not complete, SPIxRXB is empty
Automatically set in hardware when SPIx transfers data from SPIxSR to SPIxRXB
Automatically cleared in hardware when core reads SPIxBUF location, reading SPIxRXB

REGISTER 19-1: I2CxCON: I2Cx CONTROL REGISTER

R/W-0	U-0	R/W-0	R/W-1, HC	R/W-0	R/W-0	R/W-0	R/W-0
I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC	R/W-0, HC
GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

Legend:	U = Unimplemented bit, read as '0'		
R = Readable bit	W = Writable bit	HS = Set in hardware	HC = Cleared in Hardware
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **I2CEN:** I2Cx Enable bit
1 = Enables the I2Cx module and configures the SDAx and SCLx pins as serial port pins
0 = Disables the I2Cx module. All I^2C™ pins are controlled by port functions

bit 14 **Unimplemented:** Read as '0'

bit 13 **I2CSIDL:** Stop in Idle Mode bit
1 = Discontinue module operation when device enters an Idle mode
0 = Continue module operation in Idle mode

bit 12 **SCLREL:** SCLx Release Control bit (when operating as I^2C slave)
1 = Release SCLx clock
0 = Hold SCLx clock low (clock stretch)
If STREN = 1:
Bit is R/W (i.e., software can write '0' to initiate stretch and write '1' to release clock). Hardware clear at beginning of slave transmission. Hardware clear at end of slave reception.
If STREN = 0:
Bit is R/S (i.e., software can only write '1' to release clock). Hardware clear at beginning of slave transmission.

bit 11 **IPMIEN:** Intelligent Peripheral Management Interface (IPMI) Enable bit
1 = IPMI mode is enabled; all addresses Acknowledged
0 = IPMI mode disabled

bit 10 **A10M:** 10-bit Slave Address bit
1 = I2CxADD is a 10-bit slave address
0 = I2CxADD is a 7-bit slave address

bit 9 **DISSLW:** Disable Slew Rate Control bit
1 = Slew rate control disabled
0 = Slew rate control enabled

bit 8 **SMEN:** SMbus Input Levels bit
1 = Enable I/O pin thresholds compliant with SMbus specification
0 = Disable SMbus input thresholds

bit 7 **GCEN:** General Call Enable bit (when operating as I^2C slave)
1 = Enable interrupt when a general call address is received in the I2CxRSR (module is enabled for reception)
0 = General call address disabled

bit 6 **STREN:** SCLx Clock Stretch Enable bit (when operating as I^2C slave)
Used in conjunction with SCLREL bit.
1 = Enable software or receive clock stretching
0 = Disable software or receive clock stretching

21.3 Modes of Operation

The ECAN module can operate in one of several operation modes selected by the user. These modes include:

- Initialization mode
- Disable mode
- Normal Operation mode
- Listen Only mode
- Listen All Messages mode
- Loopback mode

Modes are requested by setting the REQOP<2:0> bits (CiCTRL1<10:8>). Entry into a mode is Acknowledged by monitoring the OPMODE<2:0> bits (CiCTRL1<7:5>). The module does not change the mode and the OPMODE bits until a change in mode is acceptable, generally during bus Idle time, which is defined as at least 11 consecutive recessive bits.

21.3.1 INITIALIZATION MODE

In the Initialization mode, the module does not transmit or receive. The error counters are cleared and the interrupt flags remain unchanged. The user application has access to Configuration registers that are access restricted in other modes. The module protects the user from accidentally violating the CAN protocol through programming errors. All registers which control the configuration of the module can not be modified while the module is on-line. The ECAN module is not allowed to enter the Configuration mode while a transmission is taking place. The Configuration mode serves as a lock to protect the following registers:

- All Module Control registers
- Baud Rate and Interrupt Configuration registers
- Bus Timing registers
- Identifier Acceptance Filter registers
- Identifier Acceptance Mask registers

21.3.2 DISABLE MODE

In Disable mode, the module does not transmit or receive. The module has the ability to set the WAKIF bit due to bus activity, however, any pending interrupts remains and the error counters retains their value.

If the REQOP<2:0> bits (CiCTRL1<10:8>) = 001, the module enters the Module Disable mode. If the module is active, the module waits for 11 recessive bits on the CAN bus, detect that condition as an Idle bus, then accept the module disable command. When the OPMODE<2:0> bits (CiCTRL1<7:5>) = 001, that indicates whether the module successfully went into Module Disable mode. The I/O pins reverts to normal I/O function when the module is in the Module Disable mode.

The module can be programmed to apply a low-pass filter function to the CiRX input line while the module or the CPU is in Sleep mode. The WAKFIL bit (CiCFG2<14>) enables or disables the filter.

Note:	Typically, if the ECAN module is allowed to transmit in a particular mode of operation and a transmission is requested immediately after the ECAN module has been placed in that mode of operation, the module waits for 11 consecutive recessive bits on the bus before starting transmission. If the user switches to Disable mode within this 11-bit period, then this transmission is aborted and the corresponding TXABT bit is set and TXREQ bit is cleared.

21.3.3 NORMAL OPERATION MODE

Normal Operation mode is selected when the REQOP<2:0> = 000. In this mode, the module is activated and the I/O pins assumes the CAN bus functions. The module transmits and receive CAN bus messages via the CiTX and CiRX pins.

21.3.4 LISTEN ONLY MODE

If the Listen Only mode is activated, the module on the CAN bus is passive. The transmitter buffers revert to the port I/O function. The receive pins remain inputs. For the receiver, no error flags or Acknowledge signals are sent. The error counters are deactivated in this state. The Listen Only mode can be used for detecting the baud rate on the CAN bus. To use this, it is necessary that there are at least two further nodes that communicate with each other.

21.3.5 LISTEN ALL MESSAGES MODE

The module can be set to ignore all errors and receive any message. The Listen All Messages mode is activated by setting the REQOP<2:0> = 111. In this mode, the data which is in the message assembly buffer, until the time an error occurred, is copied in the receive buffer and can be read via the CPU interface.

21.3.6 LOOPBACK MODE

If the Loopback mode is activated, the module connects the internal transmit signal to the internal receive signal at the module boundary. The transmit and receive pins revert to their port I/O function.

TABLE 28-5: CODE FLASH SECURITY SEGMENT SIZES FOR 128 KB DEVICES

CONFIG BITS	BSS<2:0> = x11 0K	BSS<2:0> = x10 1K	BSS<2:0> = x01 4K	BSS<2:0> = x00 8K
SSS<2:0> = x11 0K	VS = 256 IW GS = 43776 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh	VS = 256 IW BS = 768 IW GS = 43008 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh	VS = 256 IW BS = 3840 IW GS = 39936 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh	VS = 256 IW BS = 7936 IW GS = 35840 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh
SSS<2:0> = x10 4K	VS = 256 IW SS = 3840 IW GS = 39936 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00ABFEh 0157FEh	VS = 256 IW BS = 768 IW SS = 3072 IW GS = 39936 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00ABFEh 0157FEh	VS = 256 IW BS = 3840 IW GS = 39936 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00ABFEh 0157FEh	VS = 256 IW BS = 7936 IW GS = 35840 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00ABFEh 0157FEh
SSS<2:0> = x01 8K	VS = 256 IW SS = 7936 IW GS = 35840 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh	VS = 256 IW BS = 768 IW SS = 7168 IW GS = 35840 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh	VS = 256 IW BS = 3840 IW SS = 4096 IW GS = 35840 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh	VS = 256 IW BS = 7936 IW GS = 35840 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh
SSS<2:0> = x00 16K	VS = 256 IW SS = 16128 IW GS = 27648 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh	VS = 256 IW BS = 768 IW SS = 15360 IW GS = 27648 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh	VS = 256 IW BS = 3840 IW SS = 12288 IW GS = 27648 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh	VS = 256 IW BS = 7936 IW SS = 8192 IW GS = 27648 IW 000000h 0001FEh 000200h 0007FEh 000800h 001FFEh 002000h 003FFEh 004000h 007FFEh 008000h 00FFFEh 010000h 0157FEh

TABLE 31-4: DC TEMPERATURE AND VOLTAGE SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ + 85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
Operating Voltage							
DC10	**Supply Voltage**						
	VDD		3.0	—	3.6	V	Industrial and Extended
DC12	VDR	**RAM Data Retention Voltage(2)**	1.8	—	—	V	—
DC16	VPOR	**VDD Start Voltage** to ensure internal Power-on Reset signal	—	—	VSS	V	—
DC17	SVDD	**VDD Rise Rate** to ensure internal Power-on Reset signal	0.03	—	—	V/ms	0-3.0V in 0.1s

Note 1: Data in "Typ" column is at 3.3V, 25°C unless otherwise stated.

2: This is the limit to which VDD may be lowered without losing RAM data.

TABLE 31-8: DC CHARACTERISTICS: DOZE CURRENT (I_{DOZE})

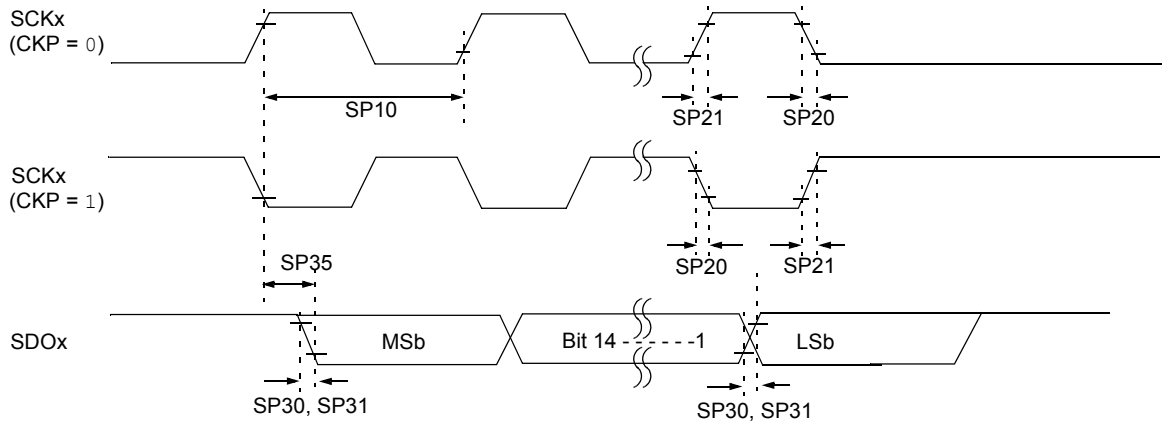
DC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤+ 85°C for Industrial -40°C ≤ TA ≤+125°C for Extended				
Parameter No.	**Typical(1)**	**Max**	**Doze Ratio**	**Units**	**Conditions**		
DC73a	20	50	1:2	mA	-40°C	3.3V	40 MIPS
DC73f	17	30	1:64	mA			
DC73g	17	30	1:128	mA			
DC70a	20	50	1:2	mA	+25°C	3.3V	40 MIPS
DC70f	17	30	1:64	mA			
DC70g	17	30	1:128	mA			
DC71a	20	50	1:2	mA	+85°C	3.3V	40 MIPS
DC71f	17	30	1:64	mA			
DC71g	17	30	1:128	mA			
DC72a	21	50	1:2	mA	+125°C	3.3V	40 MIPS
DC72f	18	30	1:64	mA			
DC72g	18	30	1:128	mA			

Note 1: Data in the “Typical” column is at 3.3V, 25°C unless otherwise stated.

TABLE 31-32: SPIx MAXIMUM DATA/CLOCK RATE SUMMARY

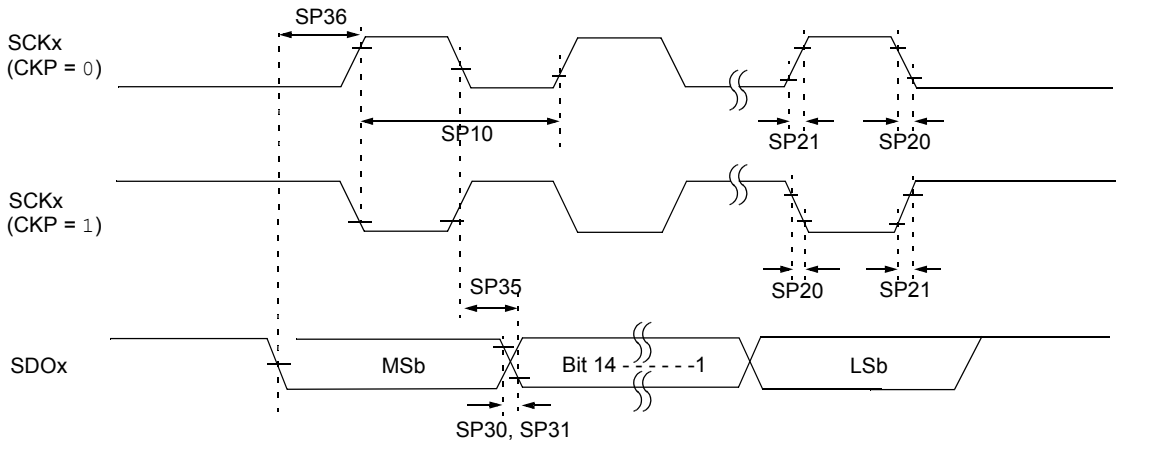
AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended			
Maximum Data Rate	**Master Transmit Only (Half-Duplex)**	**Master Transmit/Receive (Full-Duplex)**	**Slave Transmit/Receive (Full-Duplex)**	**CKE**	**CKP**	**SMP**
15 Mhz	Table 31-33	—	—	0,1	0,1	0,1
9 Mhz	—	Table 31-34	—	1	0,1	1
9 Mhz	—	Table 31-35	—	0	0,1	1
15 Mhz	—	—	Table 31-36	1	0	0
11 Mhz	—	—	Table 31-37	1	1	0
15 Mhz	—	—	Table 31-38	0	1	0
11 Mhz	—	—	Table 31-39	0	0	0

FIGURE 31-14: SPIx MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY CKE = 0) TIMING CHARACTERISTICS



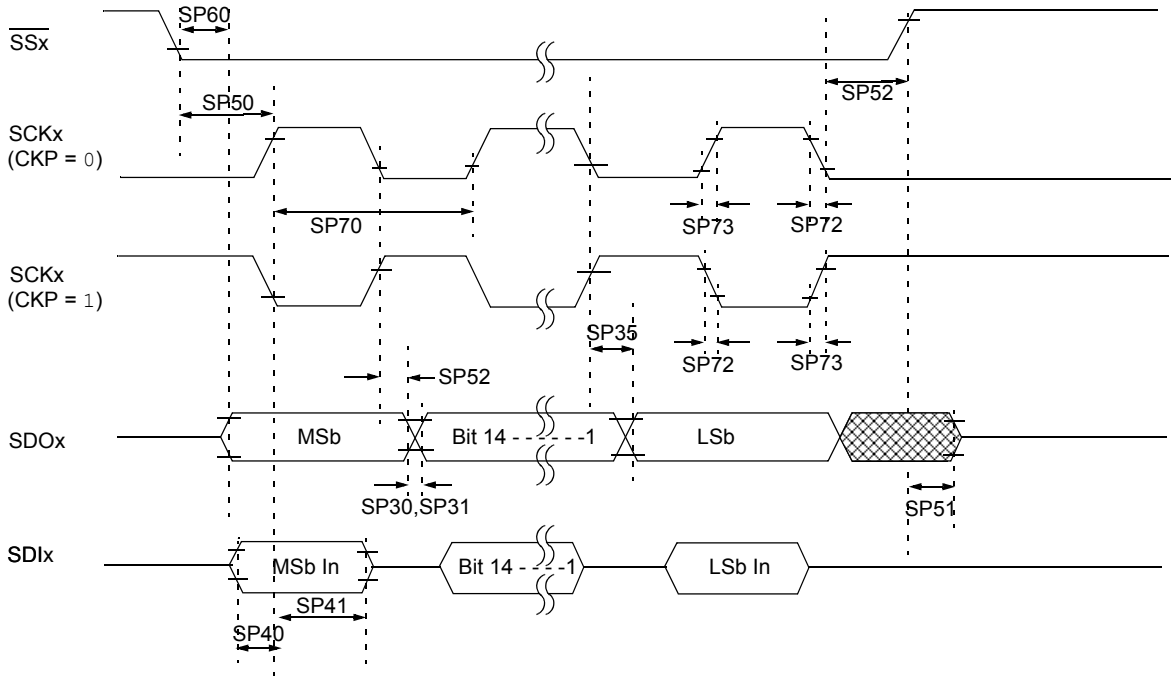
Note: Refer to Figure 31-1 for load conditions.

FIGURE 31-15: SPIx MASTER MODE (HALF-DUPLEX, TRANSMIT ONLY CKE = 1) TIMING CHARACTERISTICS



Note: Refer to Figure 31-1 for load conditions.

FIGURE 31-19: SPIx SLAVE MODE (FULL-DUPLEX, CKE = 1, CKP = 1, SMP = 0) TIMING CHARACTERISTICS



Note: Refer to Figure 31-1 for load conditions.

TABLE 31-44: ADC MODULE SPECIFICATIONS (12-BIT MODE)

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
ADC Accuracy (12-bit Mode) – Measurements with external VREF+/VREF-							
AD20a	Nr	Resolution(1)	12 data bits			bits	
AD21a	INL	Integral Nonlinearity	-2	—	+2	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD22a	DNL	Differential Nonlinearity	>-1	—	<1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD23a	GERR	Gain Error	—	3.4	10	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD24a	EOFF	Offset Error	—	0.9	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD25a	—	Monotonicity	—	—	—	—	Guaranteed
ADC Accuracy (12-bit Mode) – Measurements with internal VREF+/VREF-							
AD20a	Nr	Resolution(1)	12 data bits			bits	
AD21a	INL	Integral Nonlinearity	-2	—	+2	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD22a	DNL	Differential Nonlinearity	>-1	—	<1	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD23a	GERR	Gain Error	2	10.5	20	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD24a	EOFF	Offset Error	2	3.8	10	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD25a	—	Monotonicity	—	—	—	—	Guaranteed
Dynamic Performance (12-bit Mode)							
AD30a	THD	Total Harmonic Distortion	—	—	-75	dB	—
AD31a	SINAD	Signal to Noise and Distortion	68.5	69.5	—	dB	—
AD32a	SFDR	Spurious Free Dynamic Range	80	—	—	dB	—
AD33a	FNYQ	Input Signal Bandwidth	—	—	250	kHz	—
AD34a	ENOB	Effective Number of Bits	11.09	11.3	—	bits	—

Note 1: Injection currents > |0| can affect the ADC results by approximately 4 to 6 counts (i.e., VIH source > (VDD + 0.3V) or VIL source < (VSS – 0.3V).

TABLE 32-12: SPIx MODULE SLAVE MODE (CKE = 0) TIMING REQUIREMENTS

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+150°C for High Temperature					
Param No.	Symbol	Characteristic(1)	Min	Typ	Max	Units	Conditions
HSP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	—	35	ns	—
HSP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	25	—	—	ns	—
HSP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	25	—	—	ns	—
HSP51	TssH2doZ	$\overline{SSx}$ ↑ to SDOx Output High-Impedance	15	—	55	ns	See **Note 2**

Note 1: These parameters are characterized but not tested in manufacturing.

2: Assumes 50 pF load on all SPIx pins.

TABLE 32-13: SPIx MODULE SLAVE MODE (CKE = 1) TIMING REQUIREMENTS

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+150°C for High Temperature					
Param No.	Symbol	Characteristic(1)	Min	Typ	Max	Units	Conditions
HSP35	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge	—	—	35	ns	—
HSP40	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge	25	—	—	ns	—
HSP41	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge	25	—	—	ns	—
HSP51	TssH2doZ	$\overline{SSx}$ ↑ to SDOx Output High-Impedance	15	—	55	ns	See **Note 2**
HSP60	TssL2doV	SDOx Data Output Valid after $\overline{SSx}$ Edge	—	—	55	ns	—

Note 1: These parameters are characterized but not tested in manufacturing.

2: Assumes 50 pF load on all SPIx pins.

FIGURE 32-13: TYPICAL FRC FREQUENCY @ V_{DD} = 3.3V

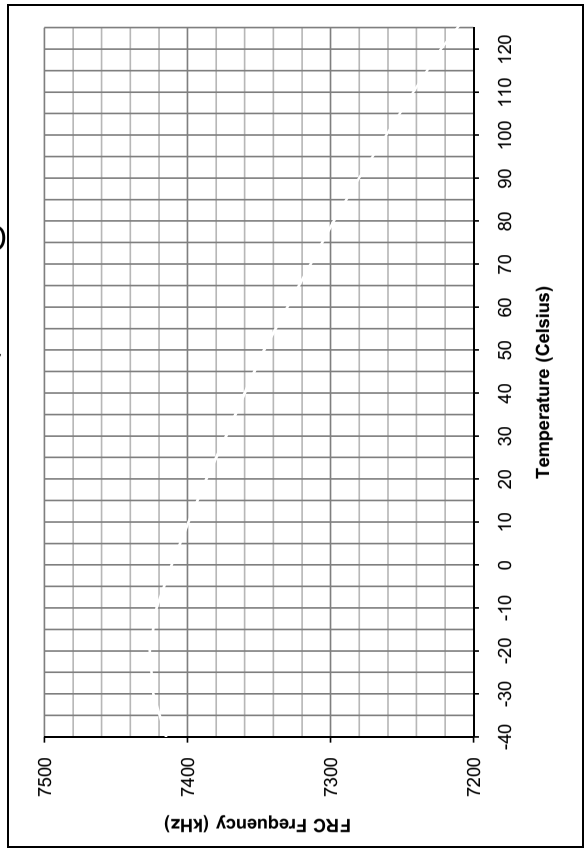


FIGURE 32-14: TYPICAL LPRC FREQUENCY @ V_{DD} = 3.3V