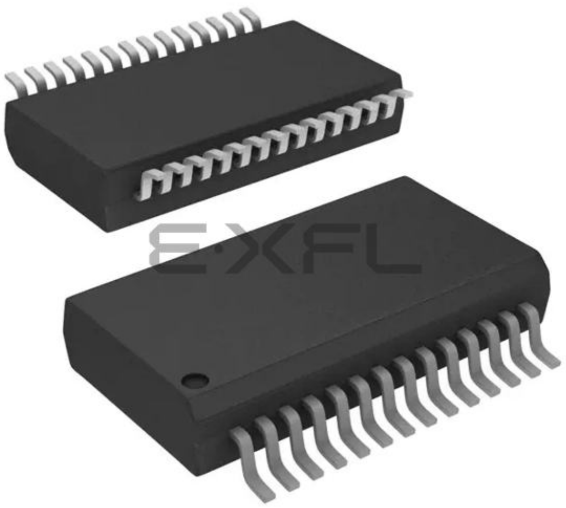




## What is "Embedded - Microcontrollers"?

"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 | 40 MIPs |
| Connectivity | I²C, IrDA, LINbus, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, POR, PWM, WDT |
| Number of I/O | 21 |
| Program Memory Size | 6KB (2K x 24) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 1K x 8 |
| Voltage - Supply (Vcc/Vdd) | 3V ~ 3.6V |
| Data Converters | A/D 6x10b; D/A 2x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 28-SSOP (0.209", 5.30mm Width) |
| Supplier Device Package | 28-SSOP |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/dspic33fj06gs202a-i-ss |

### TABLE 1-1: PINOUT I/O DESCRIPTIONS

| Pin Name | Pin Type | Buffer Type | PPS Capable | Description |
|---|---|---|---|---|
| AN0-AN7 | I | Analog | No | Analog input channels. |
| CLKI | I | ST/CMOS | No | External clock source input. Always associated with OSC1 pin function. |
| CLKO | O | — | No | Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. Optionally functions as CLKO in RC and EC modes. Always associated with OSC2 pin function. |
| OSC1 | I | ST/CMOS | No | Oscillator crystal input. ST buffer when configured in RC mode; CMOS otherwise. |
| OSC2 | I/O | — | No | Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. Optionally functions as CLKO in RC and EC modes. |
| CN0-CN15 | I | ST | No | Change notification inputs. Can be software programmed for internal weak pull-ups on all inputs. |
| IC1 | I | ST | Yes | Capture Input 1. |
| OCFA | I | ST | Yes | Compare Fault A input (for Compare Channel 1). |
| OC1 | O | — | Yes | Compare Output 1. |
| INT0 | I | ST | No | External Interrupt 0. |
| INT1 | I | ST | Yes | External Interrupt 1. |
| INT2 | I | ST | Yes | External Interrupt 2. |
| RA0-RA4 | I/O | ST | No | PORTA is a bidirectional I/O port. |
| RB0-RB15[1] | I/O | ST | No | PORTB is a bidirectional I/O port. |
| RP0-RP15[1] | I/O | ST | No | Remappable I/O pins. |
| T1CK | I | ST | Yes | Timer1 external clock input. |
| T2CK | I | ST | Yes | Timer2 external clock input. |
| $\overline{\text{U1CTS}}$ | I | ST | Yes | UART1 Clear-to-Send. |
| $\overline{\text{U1RTS}}$ | O | — | Yes | UART1 Ready-to-Send. |
| U1RX | I | ST | Yes | UART1 receive. |
| U1TX | O | — | Yes | UART1 transmit. |
| SCK1 | I/O | ST | Yes | Synchronous serial clock input/output for SPI1. |
| SDI1 | I | ST | Yes | SPI1 data in. |
| SDO1 | O | — | Yes | SPI1 data out. |
| $\overline{\text{SS1}}$ | I/O | ST | Yes | SPI1 slave synchronization or frame pulse I/O. |
| SCL1 | I/O | ST | No | Synchronous serial clock input/output for I2C1. |
| SDA1 | I/O | ST | No | Synchronous serial data input/output for I2C1. |
| TMS | I | TTL | No | JTAG Test mode select pin. |
| TCK | I | TTL | No | JTAG test clock input pin. |
| TDI | I | TTL | No | JTAG test data input pin. |
| TDO | O | — | No | JTAG test data output pin. |

**Legend:** CMOS = CMOS compatible input or output; Analog = Analog input; I = Input
ST = Schmitt Trigger input with CMOS levels; P = Power; O = Output
TTL = Transistor-Transistor Logic; PPS = Peripheral Pin Select; — = Does not apply

**Note 1:** Not all pins are available on all devices. Refer to the specific device in the **"Pin Diagrams"** section for availability.

**2:** This pin is available on dsPIC33FJ09GS302 devices only.

## 3.4 CPU Control Registers

### REGISTER 3-1: SR: CPU STATUS REGISTER

| R-0 | R-0 | R/C-0 | R/C-0 | R-0 | R/C-0 | R -0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| OA | OB | SA(1) | SB(1) | OAB | SAB(1,4) | DA | DC |
| bit 15 | | | | | | | bit 8 |

| R/W-0(3) | R/W-0(3) | R/W-0(3) | R-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| IPL<2:0>(2) | | | RA | N | OV | Z | C |
| bit 7 | | | | | | | bit 0 |

| Legend: | | |
|---|---|---|
| C = Clearable bit | R = Readable bit | U = Unimplemented bit, read as ‘0’ |
| S = Settable bit | W = Writable bit | -n = Value at POR |
| ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |

bit 15 **OA:** Accumulator A Overflow Status bit
1 = Accumulator A overflowed
0 = Accumulator A has not overflowed

bit 14 **OB:** Accumulator B Overflow Status bit
1 = Accumulator B overflowed
0 = Accumulator B has not overflowed

bit 13 **SA:** Accumulator A Saturation ‘Sticky’ Status bit(1)
1 = Accumulator A is saturated or has been saturated at some time
0 = Accumulator A is not saturated

bit 12 **SB:** Accumulator B Saturation ‘Sticky’ Status bit(1)
1 = Accumulator B is saturated or has been saturated at some time
0 = Accumulator B is not saturated

bit 11 **OAB:** OA || OB Combined Accumulator Overflow Status bit
1 = Accumulators A or B have overflowed
0 = Neither Accumulators A or B have overflowed

bit 10 **SAB:** SA || SB Combined Accumulator ‘Sticky’ Status bit(1,4)
1 = Accumulators A or B are saturated or have been saturated at some time in the past
0 = Neither Accumulator A or B are saturated

bit 9 **DA:** `DO` Loop Active bit
1 = `DO` loop in progress
0 = `DO` loop not in progress

bit 8 **DC:** MCU ALU Half Carry/$\overline{\text{Borrow}}$ bit
1 = A carry-out from the 4th low-order bit (for byte-sized data) or 8th low-order bit (for word-sized data) of the result occurred
0 = No carry-out from the 4th low-order bit (for byte-sized data) or 8th low-order bit (for word-sized data) of the result occurred

**Note 1:** This bit can be read or cleared (not set).
**2:** The IPL<2:0> bits are concatenated with the IPL3 bit (CORCON<3>) to form the CPU Interrupt Priority Level (IPL). The value in parentheses indicates the IPL if IPL3 = 1. User interrupts are disabled when IPL3 = 1.
**3:** The IPL<2:0> Status bits are read-only when NSTDIS = 1 (INTCON1<15>).
**4:** Clearing this bit will clear SA and SB.

**REGISTER 3-1: SR: CPU STATUS REGISTER (CONTINUED)**

bit 7-5 **IPL<2:0>:** CPU Interrupt Priority Level Status bits[2,3]

`111` = CPU Interrupt Priority Level is 7 (15), user interrupts disabled
`110` = CPU Interrupt Priority Level is 6 (14)
`101` = CPU Interrupt Priority Level is 5 (13)
`100` = CPU Interrupt Priority Level is 4 (12)
`011` = CPU Interrupt Priority Level is 3 (11)
`010` = CPU Interrupt Priority Level is 2 (10)
`001` = CPU Interrupt Priority Level is 1 (9)
`000` = CPU Interrupt Priority Level is 0 (8)

bit 4 **RA:** `REPEAT` Loop Active bit

`1` = `REPEAT` loop in progress
`0` = `REPEAT` loop not in progress

bit 3 **N:** MCU ALU Negative bit

`1` = Result was negative
`0` = Result was non-negative (zero or positive)

bit 2 **OV:** MCU ALU Overflow bit

This bit is used for signed arithmetic (2's complement). It indicates an overflow of a magnitude that causes the sign bit to change state.
`1` = Overflow occurred for signed arithmetic (in this arithmetic operation)
`0` = No overflow occurred

bit 1 **Z:** MCU ALU Zero bit

`1` = An operation that affects the Z bit has set it at some time in the past
`0` = The most recent operation that affects the Z bit has cleared it (i.e., a non-zero result)

bit 0 **C:** MCU ALU Carry/$\overline{\text{Borrow}}$ bit

`1` = A carry-out from the Most Significant bit of the result occurred
`0` = No carry-out from the Most Significant bit of the result occurred

**Note 1:** This bit can be read or cleared (not set).

**2:** The IPL<2:0> bits are concatenated with the IPL3 bit (CORCON<3>) to form the CPU Interrupt Priority Level (IPL). The value in parentheses indicates the IPL if IPL3 = `1`. User interrupts are disabled when IPL3 = `1`.

**3:** The IPL<2:0> Status bits are read-only when NSTDIS = `1` (INTCON1<15>).

**4:** Clearing this bit will clear SA and SB.

## 4.5 Modulo Addressing

Modulo Addressing mode is a method used to provide an automated means to support circular data buffers using hardware. The objective is to remove the need for software to perform data address boundary checks when executing tightly looped code, as is typical in many DSP algorithms.

Modulo Addressing can operate in either data or program space (since the data pointer mechanism is essentially the same for both). One circular buffer can be supported in each of the X (which also provides the pointers into program space) and Y data spaces. Modulo Addressing can operate on any W register pointer. However, it is not advisable to use W14 or W15 for Modulo Addressing since these two registers are used as the Stack Frame Pointer and Stack Pointer, respectively.

In general, any particular circular buffer can be configured to operate in only one direction as there are certain restrictions on the buffer start address (for incrementing buffers), or end address (for decrementing buffers), based upon the direction of the buffer.

The only exception to the usage restrictions is for buffers that have a power-of-two length. As these buffers satisfy the start and end address criteria, they can operate in a bidirectional mode (that is, address boundary checks are performed on both the lower and upper address boundaries).

### 4.5.1 START AND END ADDRESS

The Modulo Addressing scheme requires that a starting and ending address be specified and loaded into the 16-bit Modulo Buffer Address registers: XMODSRT, XMODEND, YMODSRT and YMODEND (see Table 4-1).

| **Note:** | Y space Modulo Addressing EA calculations assume word-sized data (LSb of every EA is always clear). |
|---|---|

The length of a circular buffer is not directly specified. It is determined by the difference between the corresponding start and end addresses. The maximum possible length of the circular buffer is 32K words (64 Kbytes).

### 4.5.2 W ADDRESS REGISTER SELECTION

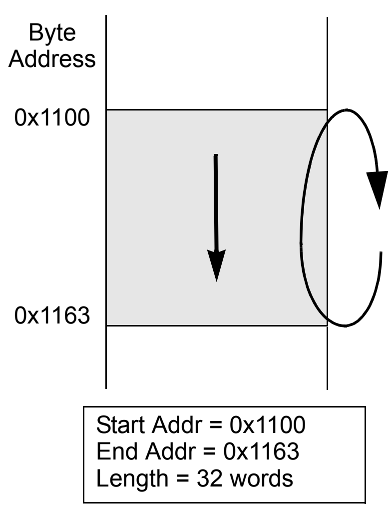
The Modulo and Bit-Reversed Addressing Control register, MODCON<15:0>, contains enable flags as well as a W register field to specify the W Address registers. The XWM and YWM fields select the registers that will operate with Modulo Addressing:

- If XWM = 15, X RAGU and X WAGU Modulo Addressing is disabled
- If YWM = 15, Y AGU Modulo Addressing is disabled

The X Address Space Pointer W register (XWM), to which Modulo Addressing is to be applied, is stored in MODCON<3:0> (see Table 4-1). Modulo Addressing is enabled for X data space when XWM is set to any value other than '15' and the XMODEN bit is set at MODCON<15>.

The Y Address Space Pointer W register (YWM) to which Modulo Addressing is to be applied is stored in MODCON<7:4>. Modulo Addressing is enabled for Y data space when YWM is set to any value other than '15' and the YMODEN bit is set at MODCON<14>.

**FIGURE 4-6: MODULO ADDRESSING OPERATION EXAMPLE**




```
MOV     #0x1100, W0
MOV     W0, XMODSRT    ;set modulo start address
MOV     #0x1163, W0
MOV     W0, MODEND     ;set modulo end address
MOV     #0x8001, W0
MOV     W0, MODCON     ;enable W1, X AGU for modulo

MOV     #0x0000, W0    ;W0 holds buffer fill value

MOV     #0x1110, W1    ;point W1 to buffer

DO      AGAIN, #0x31   ;fill the 50 buffer locations
MOV     W0, [W1++]     ;fill the next location
AGAIN:  INC W0, W0     ;increment the fill value
```

## 7.3 Interrupt Control and Status Registers

The following registers are implemented for the interrupt controller:

- INTCON1
- INTCON2
- IFSx
- IECx
- IPCx
- INTTREG

### 7.3.1 INTCON1 AND INTCON2

Global interrupt control functions are controlled from INTCON1 and INTCON2. INTCON1 contains the Interrupt Nesting Disable (NSTDIS) bit as well as the control and status flags for the processor trap sources. The INTCON2 register controls the external interrupt request signal behavior and the use of the Alternate Interrupt Vector Table.

### 7.3.2 IFSx

The IFSx registers maintain all of the interrupt request flags. Each source of interrupt has a status bit, which is set by the respective peripherals or external signal and is cleared via software.

### 7.3.3 IECx

The IECx registers maintain all of the interrupt enable bits. These control bits are used to individually enable interrupts from the peripherals or external signals.

### 7.3.4 IPCx

The IPCx registers are used to set the Interrupt Priority Level (IPL) for each source of interrupt. Each user interrupt source can be assigned to one of eight priority levels.

### 7.3.5 INTTREG

The INTTREG register contains the associated interrupt vector number and the new CPU Interrupt Priority Level, which are latched into the Vector Number (VECNUM<6:0>) and Interrupt Level (ILR<3:0>) bit fields in the INTTREG register. The new Interrupt Priority Level is the priority of the pending interrupt.

The interrupt sources are assigned to the IFSx, IECx and IPCx registers in the same sequence that they are listed in Table 7-1. For example, the INT0 (External Interrupt 0) is shown as having Vector Number 8 and a natural order priority of 0. Thus, the INT0IF bit is found in IFS0<0>, the INT0IE bit is found in IEC0<0> and the INT0IP bits are found in the first position of IPC0 (IPC0<2:0>).

### 7.3.6 STATUS/CONTROL REGISTERS

Although they are not specifically part of the interrupt control hardware, two of the CPU Control registers contain bits that control interrupt functionality.

- The CPU STATUS Register, SR, contains the IPL<2:0> bits (SR<7:5>). These bits indicate the current CPU Interrupt Priority Level. The user can change the current CPU priority level by writing to the IPL bits.
- The CORCON register contains the IPL3 bit, which together with IPL<2:0>, indicates the current CPU priority level. IPL3 is a read-only bit so that trap events cannot be masked by the user software.

All Interrupt registers are described in Register 7-1 through Register 7-35.

**REGISTER 7-11: IFS7: INTERRUPT FLAG STATUS REGISTER 7**

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | — | — |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | U-0 | R/W-0 | U-0 | U-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | ADCP6IF | — | — | ADCP3IF(1) | ADCP2IF(2) |
| 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 15-5 **Unimplemented:** Read as '0'

bit 4 **ADCP6IF:** ADC Pair 6 Conversion Done Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

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

bit 1 **ADCP3IF:** ADC Pair 3 Conversion Done Interrupt Flag Status bit(1)
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 0 **ADCP2IF:** ADC Pair 2 Conversion Done Interrupt Flag Status bit(2)
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

**Note 1:** This bit is not implemented in dsPIC33FJ06GS102A/202A devices.
**2:** This bit is not implemented in dsPIC33FJ06GS001/101A devices.

**REGISTER 7-13: IEC1: INTERRUPT ENABLE CONTROL REGISTER 1**

| U-0 | U-0 | R/W-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | — | INT2IE | — | — | — | — | — |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | INT1IE | CNIE | AC1IE(1) | MI2C1IE | SI2C1IE |
| 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 15-14 **Unimplemented:** Read as ‘0’

bit 13 **INT2IE:** External Interrupt 2 Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 12-5 **Unimplemented:** Read as ‘0’

bit 4 **INT1IE:** External Interrupt 1 Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 3 **CNIE:** Input Change Notification Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 2 **AC1IE:** Analog Comparator 1 Interrupt Enable bit(1)
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 1 **MI2C1IE:** I2C1 Master Events Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 0 **SI2C1IE:** I2C1 Slave Events Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

**Note 1:** This bit is not implemented in dsPIC33FJ06GS101A/102A devices.

**REGISTER 7-18: IEC7: INTERRUPT ENABLE CONTROL REGISTER 7**

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | — | — |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | U-0 | R/W-0 | U-0 | U-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | ADCP6IE | — | — | ADCP3IE(1) | ADCP2IE(2) |
| 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 15-5 **Unimplemented:** Read as ‘0’

bit 4 **ADCP6IE:** ADC Pair 6 Conversion Done Interrupt Enable bit
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 3-2 **Unimplemented:** Read as ‘0’

bit 1 **ADCP3IE:** ADC Pair 3 Conversion Done Interrupt Enable bit(1)
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

bit 0 **ADCP2IE:** ADC Pair 2 Conversion Done Interrupt Enable bit(2)
1 = Interrupt request is enabled
0 = Interrupt request is not enabled

**Note 1:** This bit is not implemented in dsPIC33FJ06GS102A/202A devices.
**2:** This bit is not implemented in dsPIC33FJ06GS001/101A devices.

**REGISTER 7-19: IPC0: INTERRUPT PRIORITY CONTROL REGISTER 0**

| U-0 | R/W-1 | R/W-0 | R/W-0 | U-0 | R/W-1 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | T1IP<2:0> | | | — | OC1IP<2:0>(1) | | |
| bit 15 | | | | | | | bit 8 |

| U-0 | R/W-1 | R/W-0 | R/W-0 | U-0 | R/W-1 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | IC1IP<2:0>(2) | | | — | INT0IP<2:0> | | |
| 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 **Unimplemented:** Read as ‘0’

bit 14-12 **T1IP<2:0>:** Timer1 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **OC1IP<2:0>:** Output Compare Channel 1 Interrupt Priority bits(1)
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 7 **Unimplemented:** Read as ‘0’

bit 6-4 **IC1IP<2:0>:** Input Capture Channel 1 Interrupt Priority bits(2)
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

bit 3 **Unimplemented:** Read as ‘0’

bit 2-0 **INT0IP<2:0>:** External Interrupt 0 Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

**Note 1:** These bits are not implemented in the dsPIC33FJ06GS001 device.
**2:** These bits are not implemented in dsPIC33FJ06GS001/101A/102A devices.

**REGISTER 10-4: RPINR3: PERIPHERAL PIN SELECT INPUT REGISTER 3**

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | — | — |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| — | — | T2CKR<5:0> | | | | | |
| 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 15-6 **Unimplemented:** Read as '0'

bit 5-0 **T2CKR<5:0>:** Assign Timer2 External Clock (T2CK) to the Corresponding RPn Pin bits

111111 = Input tied to VSS
100011 = Input tied to RP35
100010 = Input tied to RP34
100001 = Input tied to RP33
100000 = Input tied to RP32
•
•
•
00000 = Input tied to RP0

**REGISTER 10-8: RPINR20: PERIPHERAL PIN SELECT INPUT REGISTER 20**

| U-0 | U-0 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| — | — | SCK1R<5:0>(1) | | | | | |
| bit 15 | | | | | | | bit 8 |

| U-0 | U-0 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| — | — | SDI1R<5:0>(1) | | | | | |
| 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 15-14 **Unimplemented:** Read as ‘0’

bit 13-8 **SCK1R<5:0>:** Assign SPI1 Clock Input (SCK1) to the Corresponding RPn Pin bits(1)
111111 = Input tied to Vss
100011 = Input tied to RP35
100010 = Input tied to RP34
100001 = Input tied to RP33
100000 = Input tied to RP32
•
•
•
00000 = Input tied to RP0

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **SDI1R<5:0>:** Assign SPI1 Data Input (SDI1) to the Corresponding RPn Pin bits(1)
111111 = Input tied to Vss
100011 = Input tied to RP35
100010 = Input tied to RP34
100001 = Input tied to RP33
100000 = Input tied to RP32
•
•
•
00000 = Input tied to RP0

**Note 1:** These bits are not implemented in the dsPIC33FJ06GS001 device.

**REGISTER 17-1: I2C1CON: I2C1 CONTROL REGISTER (CONTINUED)**

bit 8 **SMEN:** SMBus Input Levels bit
1 = Enables I/O pin thresholds compliant with SMBus specification
0 = Disables SMBus input thresholds

bit 7 **GCEN:** General Call Enable bit (when operating as $I^2C$ slave)
1 = Enables interrupt when a general call address is received in the I2C1RSR (module is enabled for reception)
0 = General call address is disabled

bit 6 **STREN:** SCL1 Clock Stretch Enable bit (when operating as $I^2C$ slave)
Used in conjunction with SCLREL bit.
1 = Enables software or receives clock stretching
0 = Disables software or receives clock stretching

bit 5 **ACKDT:** Acknowledge Data bit (when operating as $I^2C$ master, applicable during master receive)
Value that is transmitted when the software initiates an Acknowledge sequence.
1 = Sends NACK during Acknowledge
0 = Sends ACK during Acknowledge

bit 4 **ACKEN:** Acknowledge Sequence Enable bit
(when operating as $I^2C$ master, applicable during master receive)
1 = Initiates Acknowledge sequence on SDA1 and SCL1 pins and transmits ACKDT data bit. Hardware is clear at end of master Acknowledge sequence.
0 = Acknowledge sequence is not in progress

bit 3 **RCEN:** Receive Enable bit (when operating as $I^2C$ master)
1 = Enables Receive mode for $I^2C$. Hardware is clear at end of eighth bit of master receive data byte.
0 = Receive sequence is not in progress

bit 2 **PEN:** Stop Condition Enable bit (when operating as $I^2C$ master)
1 = Initiates Stop condition on SDA1 and SCL1 pins. Hardware is clear at end of master Stop sequence.
0 = Stop condition is not in progress

bit 1 **RSEN:** Repeated Start Condition Enable bit (when operating as $I^2C$ master)
1 = Initiates Repeated Start condition on SDA1 and SCL1 pins. Hardware is clear at end of master Repeated Start sequence.
0 = Repeated Start condition is not in progress

bit 0 **SEN:** Start Condition Enable bit (when operating as $I^2C$ master)
1 = Initiates Start condition on SDA1 and SCL1 pins. Hardware is clear at end of master Start sequence.
0 = Start condition is not in progress

**REGISTER 17-3: I2C1MSK: I2C1 SLAVE MODE ADDRESS MASK REGISTER**

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | AMSK<9:8> | |
| 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 |
|---|---|---|---|---|---|---|---|
| AMSK<7:0> | | | | | | | |
| 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 15-10 | **Unimplemented:** Read as ‘0’ |
| bit 9-0 | **AMSK<9:0>:** Mask for Address bit x Select bits<br>1 = Enables masking for bit x of incoming message address; bit match not required in this position<br>0 = Disables masking for bit x; bit match required in this position |

**FIGURE 19-3: ADC BLOCK DIAGRAM FOR dsPIC33FJ06GS102A DEVICE**

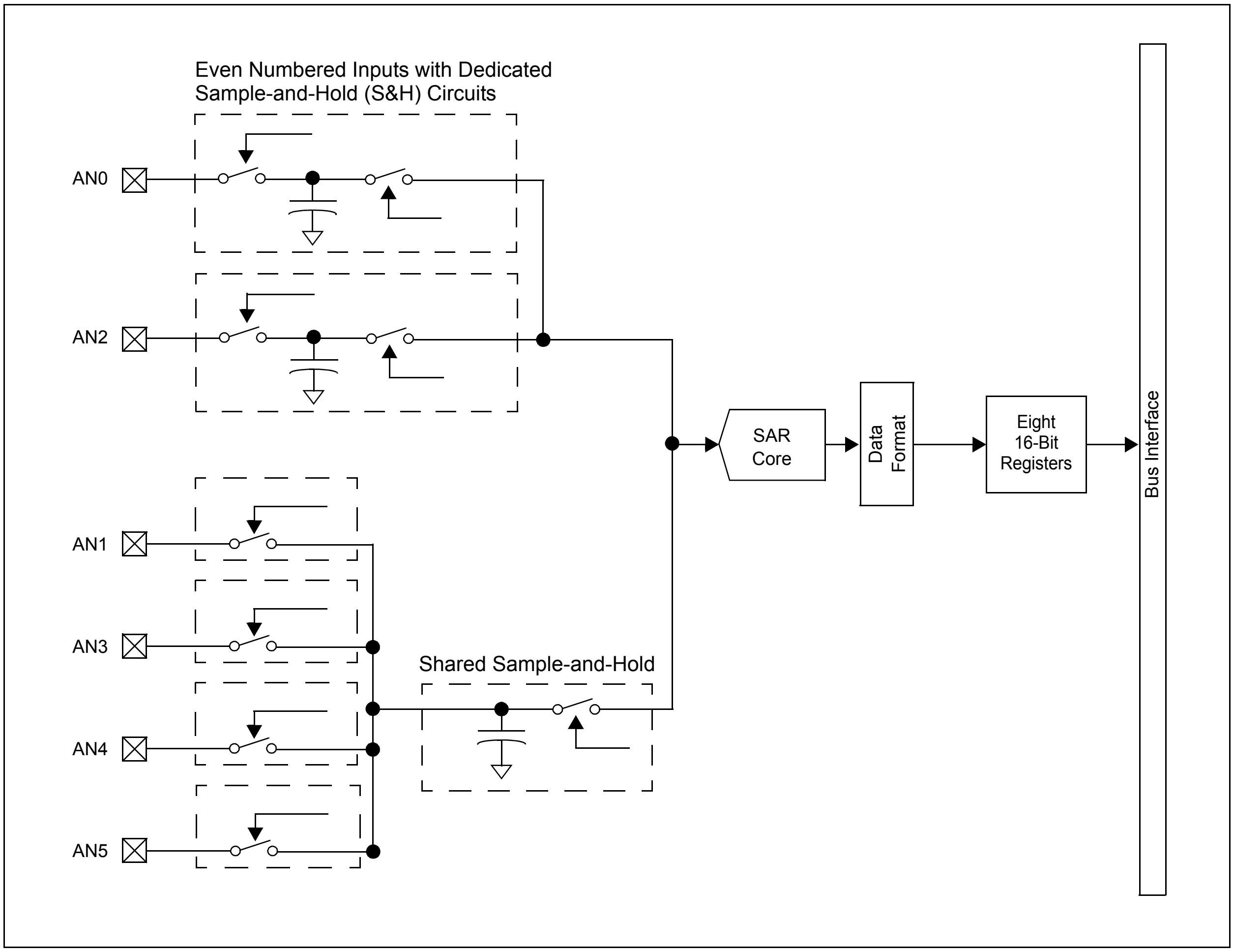
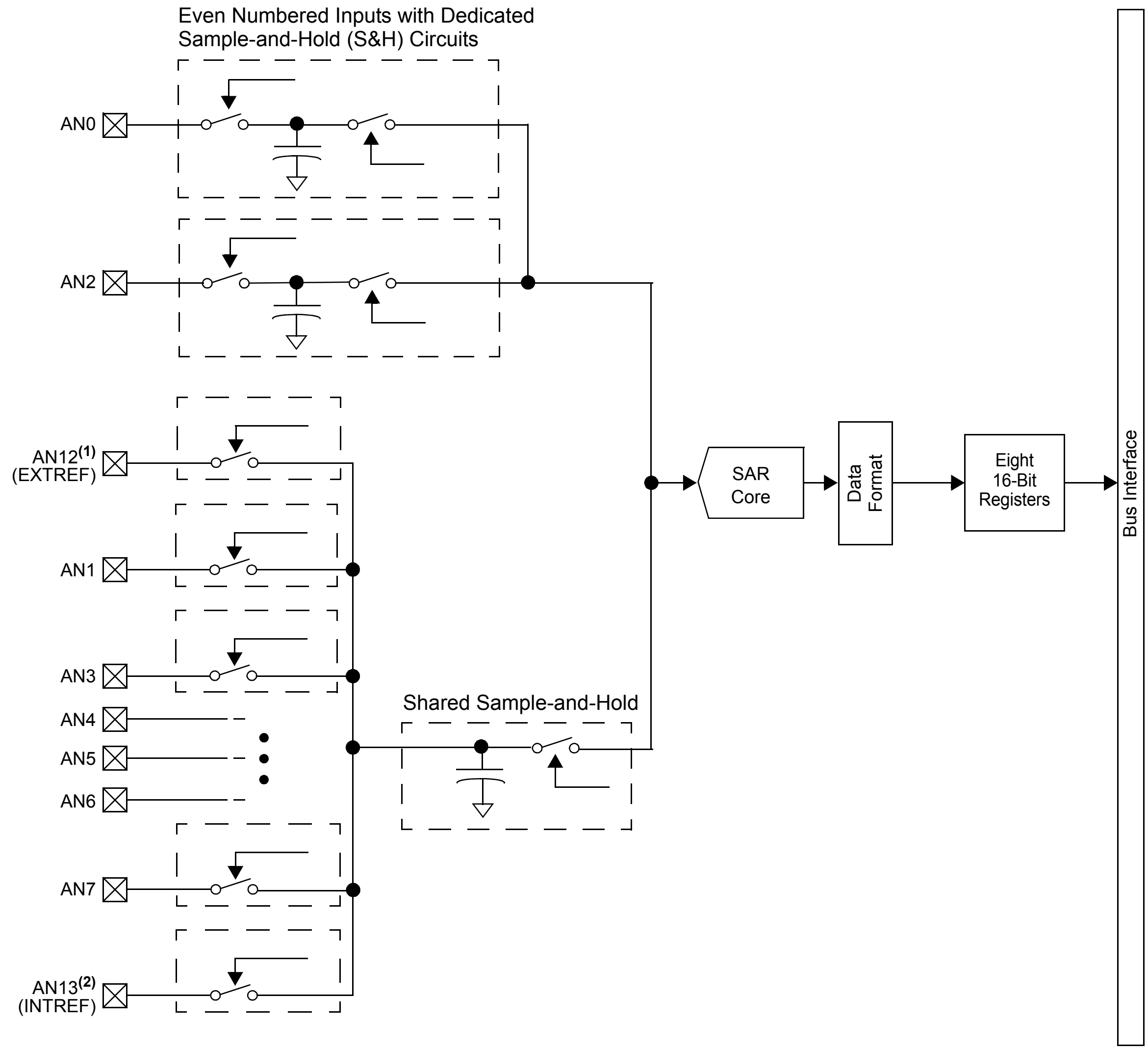


**FIGURE 19-5: ADC BLOCK DIAGRAM FOR dsPIC33FJ09GS302 DEVICE**




**Note 1:** To measure the voltage at AN12 (EXTREF), an analog comparator must be enabled and EXTREF must be selected as the comparator reference.

**2:** AN13 (INTREF) is an internal analog input and is not available on a pin.

### REGISTER 19-4: ADPCFG: ADC PORT CONFIGURATION REGISTER

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | — | — |
| bit 15 | | | | | | | bit 8 |

| R/W-0 | R/W-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| PCFG7(1) | PCFG6(1) | — | — | PCFG3 | PCFG2 | PCFG1 | PCFG0 |
| 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 15-8 **Unimplemented:** Read as ‘0’

bit 7-6 **PCFG<7:6>:** Analog-to-Digital Port Configuration Control bits(1)

1 = Port pin is in Digital mode; port read input is enabled; Analog-to-Digital input multiplexer is connected to $AV_{SS}$

0 = Port pin is in Analog mode; port read input is disabled; Analog-to-Digital samples pin voltage

bit 5-4 **Unimplemented:** Read as ‘0’

bit 3-0 **PCFG<3:0>:** Analog-to-Digital Port Configuration Control bits

1 = Port pin is in Digital mode; port read input is enabled; Analog-to-Digital input multiplexer is connected to $AV_{SS}$

0 = Port pin is in Analog mode; port read input is disabled; Analog-to-Digital samples pin voltage

**Note 1:** This bit is not implemented in dsPIC33FJ06GS102A/202A devices.

**2:** This bit is not implemented in dsPIC33FJ06GS001/101A devices.

**TABLE 23-2: INSTRUCTION SET OVERVIEW (CONTINUED)**

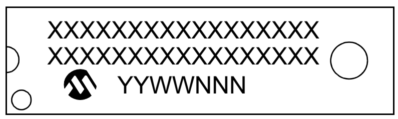
| Base Instr # | Assembly Mnemonic | Assembly Syntax | Description | # of Words | # of Cycles | Status Flags Affected |
|---|---|---|---|---|---|---|
| 29 | DIV | DIV.S Wm,Wn | Signed 16/16-bit Integer Divide | 1 | 18 | N,Z,C,OV |
| | | DIV.SD Wm,Wn | Signed 32/16-bit Integer Divide | 1 | 18 | N,Z,C,OV |
| | | DIV.U Wm,Wn | Unsigned 16/16-bit Integer Divide | 1 | 18 | N,Z,C,OV |
| | | DIV.UD Wm,Wn | Unsigned 32/16-bit Integer Divide | 1 | 18 | N,Z,C,OV |
| 30 | DIVF | DIVF Wm,Wn | Signed 16/16-bit Fractional Divide | 1 | 18 | N,Z,C,OV |
| 31 | DO | DO #lit14,Expr | Do code to PC + Expr, lit14 + 1 times | 2 | 2 | None |
| | | DO Wn,Expr | Do code to PC + Expr, (Wn) + 1 times | 2 | 2 | None |
| 32 | ED | ED Wm*Wm,Acc,Wx,Wy,Wxd | Euclidean Distance (no accumulate) | 1 | 1 | OA,OB,OAB, SA,SB,SAB |
| 33 | EDAC | EDAC Wm*Wm,Acc,Wx,Wy,Wxd | Euclidean Distance | 1 | 1 | OA,OB,OAB, SA,SB,SAB |
| 34 | EXCH | EXCH Wns,Wnd | Swap Wns with Wnd | 1 | 1 | None |
| 35 | FBCL | FBCL Ws,Wnd | Find Bit Change from Left (MSb) Side | 1 | 1 | C |
| 36 | FF1L | FF1L Ws,Wnd | Find First One from Left (MSb) Side | 1 | 1 | C |
| 37 | FF1R | FF1R Ws,Wnd | Find First One from Right (LSb) Side | 1 | 1 | C |
| 38 | GOTO | GOTO Expr | Go to Address | 2 | 2 | None |
| | | GOTO Wn | Go to Indirect | 1 | 2 | None |
| 39 | INC | INC f | f = f + 1 | 1 | 1 | C,DC,N,OV,Z |
| | | INC f,WREG | WREG = f + 1 | 1 | 1 | C,DC,N,OV,Z |
| | | INC Ws,Wd | Wd = Ws + 1 | 1 | 1 | C,DC,N,OV,Z |
| 40 | INC2 | INC2 f | f = f + 2 | 1 | 1 | C,DC,N,OV,Z |
| | | INC2 f,WREG | WREG = f + 2 | 1 | 1 | C,DC,N,OV,Z |
| | | INC2 Ws,Wd | Wd = Ws + 2 | 1 | 1 | C,DC,N,OV,Z |
| 41 | IOR | IOR f | f = f .IOR. WREG | 1 | 1 | N,Z |
| | | IOR f,WREG | WREG = f .IOR. WREG | 1 | 1 | N,Z |
| | | IOR #lit10,Wn | Wd = lit10 .IOR. Wd | 1 | 1 | N,Z |
| | | IOR Wb,Ws,Wd | Wd = Wb .IOR. Ws | 1 | 1 | N,Z |
| | | IOR Wb,#lit5,Wd | Wd = Wb .IOR. lit5 | 1 | 1 | N,Z |
| 42 | LAC | LAC Wso,#Slit4,Acc | Load Accumulator | 1 | 1 | OA,OB,OAB, SA,SB,SAB |
| 43 | LNK | LNK #lit14 | Link Frame Pointer | 1 | 1 | None |
| 44 | LSR | LSR f | f = Logical Right Shift f | 1 | 1 | C,N,OV,Z |
| | | LSR f,WREG | WREG = Logical Right Shift f | 1 | 1 | C,N,OV,Z |
| | | LSR Ws,Wd | Wd = Logical Right Shift Ws | 1 | 1 | C,N,OV,Z |
| | | LSR Wb,Wns,Wnd | Wnd = Logical Right Shift Wb by Wns | 1 | 1 | N,Z |
| | | LSR Wb,#lit5,Wnd | Wnd = Logical Right Shift Wb by lit5 | 1 | 1 | N,Z |
| 45 | MAC | MAC Wm*Wn,Acc,Wx,Wxd,Wy,Wyd , AWB | Multiply and Accumulate | 1 | 1 | OA,OB,OAB, SA,SB,SAB |
| | | MAC Wm*Wm,Acc,Wx,Wxd,Wy,Wyd | Square and Accumulate | 1 | 1 | OA,OB,OAB, SA,SB,SAB |
| 46 | MOV | MOV f,Wn | Move f to Wn | 1 | 1 | None |
| | | MOV f | Move f to f | 1 | 1 | N,Z |
| | | MOV f,WREG | Move f to WREG | 1 | 1 | None |
| | | MOV #lit16,Wn | Move 16-bit Literal to Wn | 1 | 1 | None |
| | | MOV.b #lit8,Wn | Move 8-bit Literal to Wn | 1 | 1 | None |
| | | MOV Wn,f | Move Wn to f | 1 | 1 | None |
| | | MOV Wso,Wdo | Move Ws to Wd | 1 | 1 | None |
| | | MOV WREG,f | Move WREG to f | 1 | 1 | None |
| | | MOV.D Wns,Wd | Move Double from W(ns):W(ns + 1) to Wd | 1 | 2 | None |
| | | MOV.D Ws,Wnd | Move Double from Ws to W(nd + 1):W(nd) | 1 | 2 | None |
| 47 | MOVSAC | MOVSAC Acc,Wx,Wxd,Wy,Wyd,AWB | Prefetch and Store Accumulator | 1 | 1 | None |

**NOTES:**

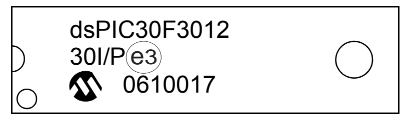
# 27.0 PACKAGING INFORMATION

## 27.1 Package Marking Information

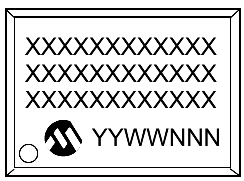
18-Lead PDIP



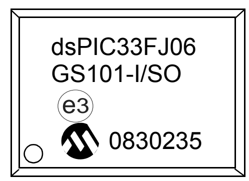

Example



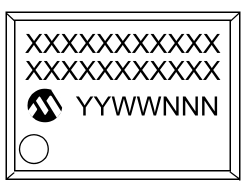

18-Lead SOIC (.300”)



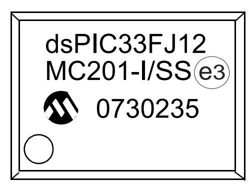

Example




20-Lead SSOP




Example




| | | |
|---|---|---|
| **Legend:** | XX...X | Customer-specific information |
| | Y | Year code (last digit of calendar year) |
| | YY | Year code (last 2 digits of calendar year) |
| | WW | Week code (week of January 1 is week ‘01’) |
| | NNN | Alphanumeric traceability code |
| | (e3) | Pb-free JEDEC designator for Matte Tin (Sn) |
| | * | This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package. |

**Note:** If the full Microchip part number cannot be marked on one line, it is carried over to the next line, thus limiting the number of available characters for customer-specific information.

# INDEX