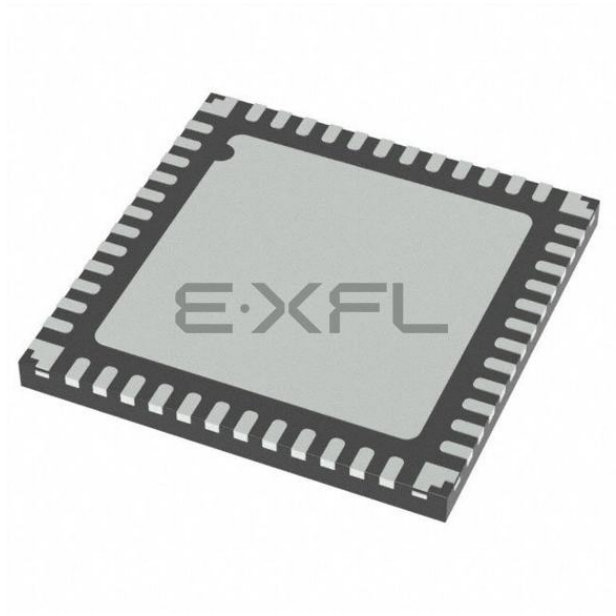


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	16-Bit
Speed	32MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, I²S, LVD, POR, PWM, WDT
Number of I/O	40
Program Memory Size	128KB (43K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 14x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-UFQFN Exposed Pad
Supplier Device Package	48-UQFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic24fj128ga705-i-m4

Peripheral Features

- High-Current Sink/Source 18 mA/18 mA on All I/O Pins
- Independent, Low-Power 32 kHz Timer Oscillator
- Timer1: 16-Bit Timer/Counter with External Crystal Oscillator; Timer1 can Provide an A/D Trigger
- Timer2,3: 16-Bit Timer/Counter, can Create 32-Bit Timer; Timer3 can Provide an A/D Trigger
- Three Input Capture modules, Each with a 16-Bit Timer
- Three Output Compare/PWM modules, Each with a 16-Bit Timer
- Four MCCP modules, Each with a Dedicated 16/32-Bit Timer:
  - One 6-output MCCP module
  - Three 2-output MCCP modules
- Three Variable Width, Synchronous Peripheral Interface (SPI) Ports on All Devices; 3 Operation modes:
  - 3-wire SPI (supports all 4 SPI modes)
  - 8 by 16-bit or 8 by 8-bit FIFO
  - I^2S mode
- Two I^2C Master and Slave w/Address Masking, and IPMI Support
- Two UART modules:
  - LIN/J2602 bus support (auto-wake-up, Auto-Baud Detect (ABD), Break character support)
  - RS-232 and RS-485 support
  - IrDA® mode (hardware encoder/decoder functions)
- Five External Interrupt Pins
- Parallel Master Port/Enhanced Parallel Slave Port (PMP/EPSP), 8-Bit Data with External Programmable Control (polarity and protocol)
- Enhanced CRC module
- Reference Clock Output with Programmable Divider
- Two Configurable Logic Cell (CLC) Blocks:
  - Two inputs and one output, all mappable to peripherals or I/O pins
  - AND/OR/XOR logic and D/JK flip-flop functions
- Peripheral Pin Select (PPS) with Independent I/O Mapping of Many Peripherals

Device	Memory		Pins	GPIO	DMA Channels	Peripherals													JTAG
	Program (bytes)	SRAM (bytes)				10/12-Bit A/D Channels	Comparators	CRC	MCCP 6-Output/2-Output	IC/OC PWM	16-Bit Timers	I^2C	Variable Width SPI	LIN-USART/IrDA®	CTMU Channels	EPMP (Address/Data Line)	CLC	RTCC	
PIC24FJ64GA705	64K	16K	48	40	6	14	3	Yes	1/3	3/3	3	2	3	2	13	10/8	2	Yes	Yes
PIC24FJ128GA705	128K	16K	48	40	6	14	3	Yes	1/3	3/3	3	2	3	2	13	10/8	2	Yes	Yes
PIC24FJ256GA705	256K	16K	48	40	6	14	3	Yes	1/3	3/3	3	2	3	2	13	10/8	2	Yes	Yes
PIC24FJ64GA704	64K	16K	44	36	6	14	3	Yes	1/3	3/3	3	2	3	2	13	10/8	2	Yes	Yes
PIC24FJ128GA704	128K	16K	44	36	6	14	3	Yes	1/3	3/3	3	2	3	2	13	10/8	2	Yes	Yes
PIC24FJ256GA704	256K	16K	44	36	6	14	3	Yes	1/3	3/3	3	2	3	2	13	10/8	2	Yes	Yes
PIC24FJ64GA702	64K	16K	28	22	6	10	3	Yes	1/3	3/3	3	2	3	2	12	No	2	Yes	Yes
PIC24FJ128GA702	128K	16K	28	22	6	10	3	Yes	1/3	3/3	3	2	3	2	12	No	2	Yes	Yes
PIC24FJ256GA702	256K	16K	28	22	6	10	3	Yes	1/3	3/3	3	2	3	2	12	No	2	Yes	Yes

1.0 DEVICE OVERVIEW

This document contains device-specific information for the following devices:

- PIC24FJ64GA705
- PIC24FJ128GA705
- PIC24FJ256GA705
- PIC24FJ64GA704
- PIC24FJ128GA704
- PIC24FJ256GA704
- PIC24FJ64GA702
- PIC24FJ128GA702
- PIC24FJ256GA702

The PIC24FJ256GA705 family introduces large Flash and SRAM memory in smaller package sizes. This is a 16-bit microcontroller family with a broad peripheral feature set and enhanced computational performance. This family also offers a new migration option for those high-performance applications which may be outgrowing their 8-bit platforms, but do not require the numerical processing power of a Digital Signal Processor (DSP).

Table 1-3 lists the functions of the various pins shown in the pinout diagrams.

1.1 Core Features

1.1.1 16-BIT ARCHITECTURE

Central to all PIC24F devices is the 16-bit modified Harvard architecture, first introduced with Microchip's dsPIC® Digital Signal Controllers (DSCs). The PIC24F CPU core offers a wide range of enhancements, such as:

- 16-bit data and 24-bit address paths with the ability to move information between data and memory spaces
- Linear addressing of up to 12 Mbytes (program space) and 32 Kbytes (data)
- A 16-element Working register array with built-in software stack support
- A 17 x 17 hardware multiplier with support for integer math
- Hardware support for 32 by 16-bit division
- An instruction set that supports multiple addressing modes and is optimized for high-level languages, such as 'C'
- Operational performance up to 16 MIPS

1.1.2 POWER-SAVING TECHNOLOGY

The PIC24FJ256GA705 family of devices includes Retention Sleep, a low-power mode with essential circuits being powered from a separate low-voltage regulator.

This new low-power mode also supports the continuous operation of the low-power, on-chip Real-Time Clock/Calendar (RTCC), making it possible for an application to keep time while the device is otherwise asleep.

Aside from this new feature, PIC24FJ256GA705 family devices also include all of the legacy power-saving features of previous PIC24F microcontrollers, such as:

- On-the-Fly Clock Switching, allowing the selection of a lower power clock during run time
- Doze Mode Operation, for maintaining peripheral clock speed while slowing the CPU clock
- Instruction-Based Power-Saving Modes, for quick invocation of the Idle and Sleep modes

1.1.3 OSCILLATOR OPTIONS AND FEATURES

All of the devices in the PIC24FJ256GA705 family offer six different oscillator options, allowing users a range of choices in developing application hardware. These include:

- Two Crystal modes
- External Clock (EC) mode
- A Phase-Locked Loop (PLL) frequency multiplier, which allows processor speeds up to 32 MHz
- An internal Fast RC Oscillator (FRC), a nominal 8 MHz output with multiple frequency divider options
- A separate internal Low-Power RC Oscillator (LPRC), 31 kHz nominal for low-power, timing-insensitive applications.

The internal oscillator block also provides a stable reference source for the Fail-Safe Clock Monitor (FSCM). This option constantly monitors the main clock source against a reference signal provided by the internal oscillator and enables the controller to switch to the internal oscillator, allowing for continued low-speed operation or a safe application shutdown.

1.1.4 EASY MIGRATION

Regardless of the memory size, all devices share the same rich set of peripherals, allowing for a smooth migration path as applications grow and evolve. The consistent pinout scheme used throughout the entire family also aids in migrating from one device to the next larger device.

The PIC24F family is pin-compatible with devices in the dsPIC33 family, and shares some compatibility with the pinout schema for PIC18 and dsPIC30. This extends the ability of applications to grow from the relatively simple, to the powerful and complex, yet still selecting a Microchip device.

REGISTER 6-1: NVMCON: FLASH MEMORY CONTROL REGISTER

R/S-0, HC(1)	R/W-0(1)	R-0, HSC(1)	R/W-0	r-0	r-0	U-0	U-0
WR	WREN	WRERR	NVMSIDL	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-0(1)	R/W-0(1)	R/W-0(1)	R/W-0(1)
—	—	—	—	NVMOP3(2)	NVMOP2(2)	NVMOP1(2)	NVMOP0(2)
bit 7							bit 0

Legend:	S = Settable bit	HC = Hardware Clearable bit	r = Reserved bit
R = Readable bit	W = Writable bit	'0' = Bit is cleared	x = Bit is unknown
-n = Value at POR	'1' = Bit is set	U = Unimplemented bit, read as '0'	
HSC = Hardware Settable/Clearable bit			

bit 15 **WR:** Write Control bit(1)
- 1 = Initiates a Flash memory program or erase operation; the operation is self-timed and the bit is cleared by hardware once the operation is complete
- 0 = Program or erase operation is complete and inactive

bit 14 **WREN:** Write Enable bit(1)
- 1 = Enables Flash program/erase operations
- 0 = Inhibits Flash program/erase operations

bit 13 **WRERR:** Write Sequence Error Flag bit(1)
- 1 = An improper program or erase sequence attempt, or termination has occurred (bit is set automatically on any set attempt of the WR bit)
- 0 = The program or erase operation completed normally

bit 12 **NVMSIDL:** NVM Stop in Idle bit
- 1 = Removes power from the program memory when device enters Idle mode
- 0 = Powers program memory in Standby mode when the device enters Idle mode

bit 11-10 **Reserved:** Maintain as '0'

bit 9-4 **Unimplemented:** Read as '0'

bit 3-0 **NVMOP<3:0>:** NVM Operation Select bits(1,2)
- 1110 = Chip erases user memory (does not erase Device ID, customer OTP or executive memory)
- 0100 = Unused
- 0011 = Erases a page of program or executive memory
- 0010 = Row programming operation
- 0001 = Double-word programming operation

Note 1: These bits can only be reset on a Power-on Reset.

2: All other combinations of NVMOP<3:0> are unimplemented.

EXAMPLE 6-1: ERASING A PROGRAM MEMORY BLOCK ('C' LANGUAGE CODE)

```
// C example using MPLAB XC16
    unsigned   long progAddr = 0xXXXXXX;         // Address of row to write
    unsigned   int  offset;
//Set up pointer to the first memory location to be written
    NVMADRU = progAddr>>16;                      // Initialize PM Page Boundary SFR
    NVMADR  = progAddr & 0xFFFF;                 // Initialize lower word of address
    NVMCON  = 0x4003;                            // Initialize NVMCON
    asm("DISI #5");                              // Block all interrupts with priority <7
                                                 // for next 5 instructions
    __builtin_write_NVM();                       // check function to perform unlock
                                                 // sequence and set WR
```

TABLE 6-2: CODE MEMORY PROGRAMMING EXAMPLE: ROW WRITES

Step 1: Set the NVMCON register to program 128 instruction words.
`MOV #0x4002, W0` `MOV W0, NVMCON`
Step 2: Initialize the TBLPAG register for writing to the latches.
`MOV #0xFA, W12` `MOV W12, TBLPAG`
Step 3: Load W0:W5 with the next 4 instruction words to program.

`MOV #<LSW0>, W0`
`MOV #<MSB1:MSB0>, W1`
`MOV #<LSW1>, W2`
`MOV #<LSW2>, W3`
`MOV #<MSB3:MSB2>, W4`
`MOV #<LSW3>, W5`

Step 4: Set the Read Pointer (W6) and load the (next set of) write latches.

`CLR W6`
`CLR W7`
`TBLWTL [W6++], [W7]`
`TBLWTH.B [W6++], [W7++]`
`TBLWTH.B [W6++], [++W7]`
`TBLWTL [W6++], [W7++]`
`TBLWTL [W6++], [W7]`
`TBLWTH.B [W6++], [W7++]`
`TBLWTH.B [W6++], [++W7]`
`TBLWTL [W6++], [W7++]`

Step 5: Repeat Steps 4 and 5, for a total of 32 times, to load the write latches with 128 instructions.
Step 6: Set the NVMADRU/NVMADR register pair to point to the correct address.

`MOV #DestinationAddress<15:0>, W3`
`MOV #DestinationAddress<23:16>, W4`
`MOV W3, NVMADR`
`MOV W4, NVMADRU`

Step 7: Execute the WR bit unlock sequence and initiate the write cycle.

`MOV #0x55, W0`
`MOV W0, NVMKEY`
`MOV #0xAA, W0`
`MOV W0, NVMKEY`
`BSET NVMCON, #WR`
`NOP`
`NOP`
`NOP`

7.1 Special Function Register Reset States

Most of the Special Function Registers (SFRs) associated with the PIC24F CPU and peripherals are reset to a particular value at a device Reset. The SFRs are grouped by their peripheral or CPU function and their Reset values are specified in each section of this manual.

The Reset value for each SFR does not depend on the type of Reset, with the exception of four registers. The Reset value for the Reset Control register, RCON, will depend on the type of device Reset. The Reset value for the Oscillator Control register, OSCCON, will depend on the type of Reset and the programmed values of the FNOSC<2:0> bits in the FOSCSEL Flash Configuration Word (see Table 7-2). The RCFGCAL and NVMCON registers are only affected by a POR.

7.2 Device Reset Times

The Reset times for various types of device Reset are summarized in Table 7-3. Note that the Master Reset Signal, $\overline{\text{SYSRST}}$, is released after the POR delay time expires.

The time at which the device actually begins to execute code will also depend on the system oscillator delays, which include the Oscillator Start-up Timer (OST) and the PLL lock time. The OST and PLL lock times occur in parallel with the applicable $\overline{\text{SYSRST}}$ delay times.

The Fail-Safe Clock Monitor (FSCM) delay determines the time at which the FSCM begins to monitor the system clock source after the $\overline{\text{SYSRST}}$ signal is released.

7.3 Brown-out Reset (BOR)

PIC24FJ256GA705 family devices implement a BOR circuit that provides the user with several configuration and power-saving options. The BOR is controlled by the BOREN<1:0> (FPOR<1:0>) Configuration bits.

When BOR is enabled, any drop of V_{DD} below the BOR threshold results in a device BOR. Threshold levels are described in **Section 32.1 "DC Characteristics"**.

7.4 Clock Source Selection at Reset

If clock switching is enabled, the system clock source at device Reset is chosen, as shown in Table 7-2. If clock switching is disabled, the system clock source is always selected according to the Oscillator Configuration bits. For more information, refer to the *"dsPIC33/PIC24 Family Reference Manual*", **"Oscillator"** (DS39700).

TABLE 7-2: OSCILLATOR SELECTION vs. TYPE OF RESET (CLOCK SWITCHING ENABLED)

Reset Type	Clock Source Determinant
POR	FNOSC<2:0> Configuration bits (FOSCSEL<2:0>)
BOR	
$\overline{\text{MCLR}}$	COSC<2:0> Control bits (OSCCON<14:12>)
WDTO	
SWR	

TABLE 8-2: INTERRUPT VECTOR DETAILS

Interrupt Source	IRQ #	IVT Address	Interrupt Bit Location		
			Flag	Enable	Priority
Highest Natural Order Priority					
INT0 – External Interrupt 0	0	000014h	IFS0<0>	IEC0<0>	INT0Interrupt
IC1 – Input Capture 1	1	000016h	IFS0<1>	IEC0<1>	IC1Interrupt
OC1 – Output Compare 1	2	000018h	IFS0<2>	IEC0<2>	OC1Interrupt
T1 – Timer1	3	00001Ah	IFS0<3>	IEC0<3>	T1Interrupt
DMA0 – Direct Memory Access 0	4	00001Ch	IFS0<4>	IEC0<4>	DMA0Interrupt
IC2 – Input Capture 2	5	00001Eh	IFS0<5>	IEC0<5>	IC2Interrupt
OC2 – Output Compare 2	6	000020h	IFS0<6>	IEC0<6>	OC2Interrupt
T2 – Timer2	7	000022h	IFS0<7>	IEC0<7>	T2Interrupt
T3 – Timer3	8	000024h	IFS0<8>	IEC0<8>	T3Interrupt
SPI1 – SPI1 General	9	000026h	IFS0<9>	IEC0<9>	SPI1Interrupt
SPI1TX – SPI1 Transfer Done	10	000028h	IFS0<10>	IEC0<10>	SPI1TXInterrupt
U1RX – UART1 Receiver	11	00002Ah	IFS0<11>	IEC0<11>	U1RXInterrupt
U1TX – UART1 Transmitter	12	00002Ch	IFS0<12>	IEC0<12>	U1TXInterrupt
ADC1 – A/D Converter 1	13	00002Eh	IFS0<13>	IEC0<13>	ADC1Interrupt
DMA1 – Direct Memory Access 1	14	000030h	IFS0<14>	IEC0<14>	DMA1Interrupt
NVM – NVM Program/Erase Complete	15	000032h	IFS0<15>	IEC0<15>	NVMInterrupt
SI2C1 – I2C1 Slave Events	16	000034h	IFS1<0>	IEC1<0>	SI2C1Interrupt
MI2C1 – I2C1 Master Events	17	000036h	IFS1<1>	IEC1<1>	MI2C1Interrupt
Comp – Comparator	18	000038h	IFS1<2>	IEC1<2>	CompInterrupt
IOC – Interrupt-on-Change Interrupt	19	00003Ah	IFS1<3>	IEC1<3>	IOCInterrupt
INT1 – External Interrupt 1	20	00003Ch	IFS1<4>	IEC1<4>	INT1Interrupt
—	21	—	—	—	—
—	22	—	—	—	—
—	23	—	—	—	—
DMA2 – Direct Memory Access 2	24	000044h	IFS1<8>	IEC1<8>	DMA2Interrupt
OC3 – Output Compare 3	25	000046h	IFS1<9>	IEC1<9>	OC3Interrupt
—	26	—	—	—	—
—	27	—	—	—	—
—	28	—	—	—	—
INT2 – External Interrupt 2	29	00004Eh	IFS1<13>	IEC1<13>	INT2Interrupt
U2RX – UART2 Receiver	30	000050h	IFS1<14>	IEC1<14>	U2RXInterrupt
U2TX – UART2 Transmitter	31	000052h	IFS1<15>	IEC1<15>	U2TXInterrupt
SPI2 – SPI2 General	32	000054h	IFS2<0>	IEC2<0>	SPI2Interrupt
SPI2TX – SPI2 Transfer Done	33	000056h	IFS2<1>	IEC2<1>	SPI2TXInterrupt
—	34	—	—	—	—
—	35	—	—	—	—
DMA3 – Direct Memory Access 3	36	00005Ch	IFS2<4>	IEC2<4>	DMA3Interrupt
IC3 – Input Capture 3	37	00005Eh	IFS2<5>	IEC2<5>	IC3Interrupt
—	38	—	—	—	—
—	39	—	—	—	—
—	40	—	—	—	—
CCT3 – Capture/Compare Timer3	43	00006Ah	IFS2<11>	IEC2<11>	CCT3Interrupt

REGISTER 8-5: INTCON4: INTERRUPT CONTROL REGISTER 4

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	U-0	U-0	U-0	R/C-0	R/C-0
—	—	—	—	—	—	ECCDBE	SGHT
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 15-2 **Unimplemented:** Read as ‘0’

bit 1 **ECCDBE:** ECC Double-Bit Error Trap bit
1 = ECC Double-Bit Error trap has occurred
0 = ECC Double-Bit Error trap has not occurred

bit 0 **SGHT:** Software Generated Hard Trap Status bit
1 = Software generated hard trap has occurred
0 = Software generated hard trap has not occurred

REGISTER 9-4: OSCDIV: OSCILLATOR DIVISOR REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DIV<14:8>						
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-1
DIV<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 **Unimplemented:** Read as ‘0’

bit 14-0 **DIV<14:0>:** Reference Clock Divider bits

Specifies the 1/2 period of the reference clock in the source clocks
(ex: Period of ref_clk_output = [Reference Source * 2] * DIV<14:0>).
111111111111111 = Oscillator frequency divided by 65,534 (32,767 * 2)
111111111111110 = Oscillator frequency divided by 65,532 (32,766 * 2)
•
•
•
000000000000011 = Oscillator frequency divided by 6 (3 * 2)
000000000000010 = Oscillator frequency divided by 4 (2 * 2)
000000000000001 = Oscillator frequency divided by 2 (1 * 2) (default)
000000000000000 = Oscillator frequency is unchanged (no divider)

11.1.1 I/O PORT WRITE/READ TIMING

One instruction cycle is required between a port direction change or port write operation and a read operation of the same port. Typically, this instruction would be a NOP.

11.1.2 OPEN-DRAIN CONFIGURATION

In addition to the PORTx, LATx and TRISx registers for data control, each port pin can also be individually configured for either a digital or open-drain output. This is controlled by the Open-Drain Control register, ODCx, associated with each port. Setting any of the bits configures the corresponding pin to act as an open-drain output.

The open-drain feature allows the generation of outputs higher than VDD (e.g., 5V) on any desired digital only pins by using external pull-up resistors. The maximum open-drain voltage allowed is the same as the maximum VIH specification.

11.2 Configuring Analog Port Pins (ANSx)

The ANSx and TRISx registers control the operation of the pins with analog function. Each port pin with analog function is associated with one of the ANSx bits, which decide if the pin function should be analog or digital. Refer to Table 11-1 for detailed behavior of the pin for different ANSx and TRISx bit settings.

When reading the PORTx register, all pins configured as analog input channels will read as cleared (a low level).

11.2.1 ANALOG INPUT PINS AND VOLTAGE CONSIDERATIONS

The voltage tolerance of pins used as device inputs is dependent on the pin's input function. Most input pins are able to handle DC voltages of up to 5.5V, a level typical for digital logic circuits. However, several pins can only tolerate voltages up to VDD. Voltage excursions beyond VDD on these pins should always be avoided.

Table 11-2 summarizes the different voltage tolerances. For more information, refer to **Section 32.0 "Electrical Characteristics"** for more details.

TABLE 11-1: CONFIGURING ANALOG/DIGITAL FUNCTION OF AN I/O PIN

Pin Function	ANSx Setting	TRISx Setting	Comments
Analog Input	1	1	It is recommended to keep ANSx = 1.
Analog Output	1	1	It is recommended to keep ANSx = 1.
Digital Input	0	1	Firmware must wait at least one instruction cycle after configuring a pin as a digital input before a valid input value can be read.
Digital Output	0	0	Make sure to disable the analog output function on the pin if any is present.

TABLE 11-2: INPUT VOLTAGE LEVELS FOR PORT OR PIN TOLERATED DESCRIPTION INPUT

Port or Pin	Tolerated Input	Description
PORTB<11:10,8:5>	5.5V	Tolerates input levels above VDD; useful for most standard logic.
PORTC<9:6>		
PORTA<14:7,4:0>	VDD	Only VDD input levels are tolerated.
PORTB<15:12,9,4:0>		
PORTC<5:0>		

REGISTER 14-2: ICxCON2: INPUT CAPTURE x CONTROL REGISTER 2 (CONTINUED)

bit 4-0 **SYNCSEL<4:0>:** Synchronization/Trigger Source Selection bits

11111 = Not used
11110 = Not used
11101 = Not used
11100 = CTMU trigger(1)
11011 = A/D interrupt(1)
11010 = CMP3 trigger(1)
11001 = CMP2 trigger(1)
11000 = CMP1 trigger(1)
10111 = Not used
10110 = MCCP4 IC/OC interrupt
10101 = MCCP3 IC/OC interrupt
10100 = MCCP2 IC/OC interrupt
10011 = MCCP1 IC/OC interrupt
10010 = IC3 interrupt(2)
10001 = IC2 interrupt(2)
10000 = IC1 interrupt(2)
01111 = Not used
01110 = Not used
01101 = Timer3 match event
01100 = Timer2 match event
01011 = Timer1 match event
01010 = Not used
01001 = Not used
01000 = Not used
00111 = MCCP4 Sync/Trigger out
00110 = MCCP3 Sync/Trigger out
00101 = MCCP2 Sync/Trigger out
00100 = MCCP1 Sync/Trigger out
00011 = OC3 Sync/Trigger out
00010 = OC2 Sync/Trigger out
00001 = OC1 Sync/Trigger out
00000 = Off

Note 1: Use these inputs as Trigger sources only and never as Sync sources.
2: Never use an Input Capture x module as its own Trigger source by selecting this mode.

NOTES:

16.5 Auxiliary Output

The MCCPx modules have an auxiliary (secondary) output that provides other peripherals access to internal module signals. The auxiliary output is intended to connect to other MCCPx modules, or other digital peripherals, to provide these types of functions:

- Time Base Synchronization
- Peripheral Trigger and Clock Inputs
- Signal Gating

The type of output signal is selected using the AUXOUT<1:0> control bits (CCPxCON2H<4:3>). The type of output signal is also dependent on the module operating mode.

On the PIC24FJ256GA705 family of devices, only the CTMU discharge trigger has access to the auxiliary output signal.

TABLE 16-4: AUXILIARY OUTPUT

AUXOUT<1:0>	CCSEL	MOD<3:0>	Comments	Signal Description
00	x	xxxx	Auxiliary Output Disabled	No Output
01	0	0000	Time Base Modes	Time Base Period Reset or Rollover
10				Special Event Trigger Output
11				No Output
01	0	0001 through 1111	Output Compare Modes	Time Base Period Reset or Rollover
10				Output Compare Event Signal
11				Output Compare Signal
01	1	xxxx	Input Capture Modes	Time Base Period Reset or Rollover
10				Reflects the Value of the ICDIS bit
11				Input Capture Event Signal

REGISTER 16-1: CCPxCON1L: CCPx CONTROL 1 LOW REGISTERS

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CCPON	—	CCPSIDL	CCPSLP	TMRSYNC	CLKSEL2	CLKSEL1	CLKSEL0
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
TMRPS1	TMRPS0	T32	CCSEL	MOD3	MOD2	MOD1	MOD0
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 **CCPON:** CCPx Module Enable bit
1 = Module is enabled with an operating mode specified by the MOD<3:0> control bits
0 = Module is disabled

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

bit 13 **CCPSIDL:** CCPx Stop in Idle Mode Bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12 **CCPSLP:** CCPx Sleep Mode Enable bit
1 = Module continues to operate in Sleep modes
0 = Module does not operate in Sleep modes

bit 11 **TMRSYNC:** Time Base Clock Synchronization bit
1 = Module time base clock is synchronized to the internal system clocks; timing restrictions apply
0 = Module time base clock is not synchronized to the internal system clocks

bit 10-8 **CLKSEL<2:0>:** CCPx Time Base Clock Select bits
111 = TCKIA pin
110 = TCKIB pin
101 = PLL clock
100 = 2x peripheral clock
010 = SOSC clock
001 = Reference clock output
000 = System clock
For MCCP1:
011 = CLC1 output
For MCCP2:
011 = CLC2 output

bit 7-6 **TMRPS<1:0>:** Time Base Prescale Select bits
11 = 1:64 Prescaler
10 = 1:16 Prescaler
01 = 1:4 Prescaler
00 = 1:1 Prescaler

bit 5 **T32:** 32-Bit Time Base Select bit
1 = Uses 32-bit time base for timer, single edge output compare or input capture function
0 = Uses 16-bit time base for timer, single edge output compare or input capture function

bit 4 **CCSEL:** Capture/Compare Mode Select bit
1 = Input capture peripheral
0 = Output compare/PWM/timer peripheral (exact function is selected by the MOD<3:0> bits)

19.0 UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER (UART)

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the *"dsPIC33/PIC24 Family Reference Manual"*, **"UART"** (DS39708), which is available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

The Universal Asynchronous Receiver Transmitter (UART) module is one of the serial I/O modules available in the PIC24F device family. The UART is a full-duplex, asynchronous system that can communicate with peripheral devices, such as personal computers, LIN/J2602, RS-232 and RS-485 interfaces. The module also supports a hardware flow control option with the $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ pins. The UART module includes an IrDA® encoder/decoder unit.

The PIC24FJ256GA705 family devices are equipped with two UART modules, referred to as UART1 and UART2.

The primary features of the UARTx modules are:

- Full-Duplex, 8 or 9-Bit Data Transmission through the UxTX and UxRX Pins
- Even, Odd or No Parity Options (for 8-bit data)
- One or Two Stop bits
- Hardware Flow Control Option with the $\overline{\text{UxCTS}}$ and $\overline{\text{UxRTS}}$ Pins
- Fully Integrated Baud Rate Generator with 16-Bit Prescaler
- Baud Rates Range from up to 1 Mbps and Down to 15 Hz at 16 MIPS in 16x mode
- Baud Rates Range from up to 4 Mbps and Down to 61 Hz at 16 MIPS in 4x mode
- 4-Deep, First-In-First-Out (FIFO) Transmit Data Buffer
- 4-Deep FIFO Receive Data Buffer
- Parity, Framing and Buffer Overrun Error Detection
- Support for 9-Bit mode with Address Detect (9^{th} bit = 1)
- Separate Transmit and Receive Interrupts
- Loopback mode for Diagnostic Support
- Polarity Control for Transmit and Receive Lines
- Support for Sync and Break Characters
- Supports Automatic Baud Rate Detection
- IrDA® Encoder and Decoder Logic
- Includes DMA Support
- 16x Baud Clock Output for IrDA Support

A simplified block diagram of the UARTx module is shown in Figure 19-1. The UARTx module consists of these key important hardware elements:

- Baud Rate Generator
- Asynchronous Transmitter
- Asynchronous Receiver

Note: Throughout this section, references to register and bit names that may be associated with a specific UART module are referred to generically by the use of 'x' in place of the specific module number. Thus, "UxSTA" might refer to the Status register for either UART1 or UART2.

TABLE 20-2: ENHANCED PARALLEL MASTER PORT PIN DESCRIPTIONS

Pin Name (Alternate Function)	Type	Description
PMA<22:16>	O	Address Bus bits<22:16>
PMA14	O	Address Bus bit 14
	I/O	Data Bus bit 14 (16-bit port with Multiplexed Addressing)
(PMCS1)	O	Chip Select 1 (alternate location)
PMA<13:8>	O	Address Bus bits<13:8>
	I/O	Data Bus bits<13:8> (16-bit port with Multiplexed Addressing)
PMA<7:3>	O	Address Bus bits<7:3>
PMA2 (PMALU)	O	Address Bus bit 2
	O	Address Latch Upper Strobe for Multiplexed Address
PMA1 (PMALH)	I/O	Address Bus bit 1
	O	Address Latch High Strobe for Multiplexed Address
PMA0 (PMALL)	I/O	Address Bus bit 0
	O	Address Latch Low Strobe for Multiplexed Address
PMD<15:8>	I/O	Data Bus bits<15:8> (Demultiplexed Addressing)
PMD<7:4>	I/O	Data Bus bits<7:4>
	O	Address Bus bits<7:4> (4-bit port with 1-Phase Multiplexed Addressing)
PMD<3:0>	I/O	Data Bus bits<3:0>
PMCS1	O	Chip Select 1
PMCS2	O	Chip Select 2
PMWR	I/O	Write Strobe(1)
(PMENB)	I/O	Enable Signal(1)
PMRD	I/O	Read Strobe(1)
(PMRD/$\overline{\text{PMWR}}$)	I/O	Read/$\overline{\text{Write}}$ Signal(1)
PMBE1	O	Byte Indicator
PMBE0	O	Nibble or Byte Indicator
PMACK1	I	Acknowledgment Signal 1
PMACK2	I	Acknowledgment Signal 2

Note 1: Signal function depends on the setting of the MODE<1:0> and SM bits (PMCON1<9:8> and PMCSxCF<8>).

21.3.2 RTCVAL REGISTER MAPPINGS

REGISTER 21-4: RTCCON2H: RTCC CONTROL REGISTER 2 (HIGH)(1)

R/W-0	R/W-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
DIV<15:8>							
bit 15							bit 8

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
DIV<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 15-0 **DIV<15:0>:** Clock Divide bits
Sets the period of the clock divider counter; value should cause a nominal 1/2 second underflow.

Note 1: A write to this register is only allowed when WRLOCK = 1.

22.1.3 DATA SHIFT DIRECTION

The LENDIAN bit (CRCCON1<3>) is used to control the shift direction. By default, the CRC will shift data through the engine, MSb first. Setting LENDIAN (= 1) causes the CRC to shift data, LSb first. This setting allows better integration with various communication schemes and removes the overhead of reversing the bit order in software. Note that this only changes the direction the data is shifted into the engine. The result of the CRC calculation will still be a normal CRC result, not a reverse CRC result.

22.1.4 INTERRUPT OPERATION

The module generates an interrupt that is configurable by the user for either of two conditions.

If CRCISEL is '0', an interrupt is generated when the VWORD<4:0> bits make a transition from a value of '1' to '0'. If CRCISEL is '1', an interrupt will be generated after the CRC operation finishes and the module sets the CRCGO bit to '0'. Manually setting CRCGO to '0' will not generate an interrupt. Note that when an interrupt occurs, the CRC calculation would not yet be complete. The module will still need (PLENx + 1)/2 clock cycles after the interrupt is generated until the CRC calculation is finished.

22.1.5 TYPICAL OPERATION

To use the module for a typical CRC calculation:

1. Set the CRCEN bit to enable the module.
2. Configure the module for desired operation:
   a) Program the desired polynomial using the CRCXOR registers and PLEN<4:0> bits.
   b) Configure the data width and shift direction using the DWIDTH<4:0> and LENDIAN bits.
3. Set the CRCGO bit to start the calculations.
4. Set the desired CRC non-direct initial value by writing to the CRCWDAT registers.
5. Load all data into the FIFO by writing to the CRCDAT registers as space becomes available (the CRCFUL bit must be zero before the next data loading).
6. Wait until the data FIFO is empty (CRCMPT bit is set).
7. Read the result:
   If the data width (DWIDTH<4:0> bits) is more than the polynomial length (PLEN<4:0> bits):
   a) Wait (DWIDTH<4:0> + 1)/2 instruction cycles to make sure that shifts from the shift buffer are finished.
   b) Change the data width to the polynomial length (DWIDTH<4:0> = PLEN<4:0>).
   c) Write one dummy data word to the CRCDAT registers.
   d) Wait 2 instruction cycles to move the data from the FIFO to the shift buffer and (PLEN<4:0> + 1)/2 instruction cycles to shift out the result.

   Or, if the data width (DWIDTH<4:0> bits) is less than the polynomial length (PLEN<4:0> bits):
   1. Clear the CRC Interrupt Selection bit (CRCISEL = 0) to get the interrupt when all shifts are done. Clear the CRC interrupt flag. Write dummy data in the CRCDAT registers and wait until the CRC interrupt flag is set.
   2. Read the final CRC result from the CRCWDAT registers.
   3. Restore the data width (DWIDTH<4:0> bits) for further calculations (OPTIONAL). If the data width (DWIDTH<4:0> bits) is equal to, or less than, the polynomial length (PLEN<4:0> bits):
      a) Clear the CRC Interrupt Selection bit (CRCISEL = 0) to get the interrupt when all shifts are done.
      b) Suspend the calculation by setting CRCGO = 0.
      c) Clear the CRC interrupt flag.
      d) Write the dummy data with the total data length equal to the polynomial length in the CRCDAT registers.
      e) Resume the calculation by setting CRCGO = 1.
      f) Wait until the CRC interrupt flag is set.
      g) Read the final CRC result from the CRCWDAT registers.

There are eight registers used to control programmable CRC operation:

- CRCCON1
- CRCCON2
- CRCXORL
- CRCXORH
- CRCDATL
- CRCDATH
- CRCWDATL
- CRCWDATH

The CRCCON1 and CRCCON2 registers (Register 22-1 and Register 22-2) control the operation of the module and configure the various settings.

The CRCXOR registers (Register 22-3 and Register 22-4) select the polynomial terms to be used in the CRC equation. The CRCDAT and CRCWDAT registers are each register pairs that serve as buffers for the double-word input data, and CRC processed output, respectively.

29.0 SPECIAL FEATURES

Note: This data sheet summarizes the features of this group of PIC24F devices. It is not intended to be a comprehensive reference source. For more information, refer to the following sections of the *"dsPIC33/PIC24 Family Reference Manual"*, which are available from the Microchip web site (www.microchip.com). The information in this data sheet supersedes the information in the FRM.

- **"Watchdog Timer (WDT)"** (DS39697)
- **"High-Level Device Integration"** (DS39719)
- **"Programming and Diagnostics"** (DS39716)

PIC24FJ256GA705 family devices include several features intended to maximize application flexibility and reliability, and minimize cost through elimination of external components. These are:

- Flexible Configuration
- Watchdog Timer (WDT)
- Code Protection
- JTAG Boundary Scan Interface
- In-Circuit Serial Programming™
- In-Circuit Emulation

29.1 Configuration Bits

The Configuration bits are stored in the last page location of implemented program memory. These bits can be set or cleared to select various device configurations. There are two types of Configuration bits: system operation bits and code-protect bits. The system operation bits determine the power-on settings for system-level components, such as the oscillator and the Watchdog Timer. The code-protect bits prevent program memory from being read and written.

29.1.1 CONSIDERATIONS FOR CONFIGURING PIC24FJ256GA705 FAMILY DEVICES

In PIC24FJ256GA705 family devices, the Configuration bytes are implemented as volatile memory. This means that configuration data must be programmed each time the device is powered up. Configuration data is stored in the three words at the top of the on-chip program memory space, known as the Flash Configuration Words. Their specific locations are shown in Table 29-1. The configuration data is automatically loaded from the Flash Configuration Words to the proper Configuration registers during device Resets.

Note: Configuration data is reloaded on all types of device Resets.

When creating applications for these devices, users should always specifically allocate the location of the Flash Configuration Word for configuration data. This is to make certain that program code is not stored in this address when the code is compiled.

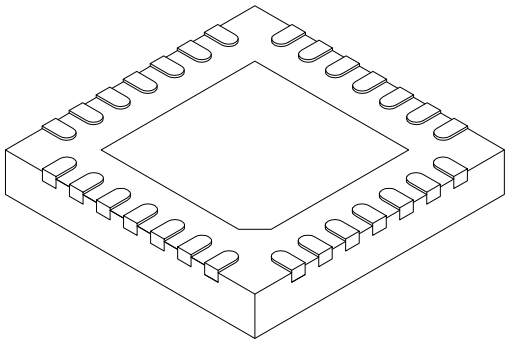
The upper byte of all Flash Configuration Words in program memory should always be '0000 0000'. This makes them appear to be NOP instructions in the remote event that their locations are ever executed by accident. Since Configuration bits are not implemented in the corresponding locations, writing '0's to these locations has no effect on device operation.

TABLE 29-1: CONFIGURATION WORD ADDRESSES

Configuration Register	PIC24FJ256GA70X	PIC24FJ128GA70X	PIC24FJ64GA70X
FSEC	02AF00h	015F00h	00AF00h
FBSLIM	02AF10h	015F10h	00AF10h
FSIGN	02AF14h	015F14h	00AF14h
FOSCSEL	02AF18h	015F18h	00AF18h
FOSC	02AF1Ch	015F1Ch	00AF1Ch
FWDT	02AF20h	015F20h	00AF20h
FPOR	02AF24h	015F24h	00AF24h
FICD	02AF28h	015F28h	00AF28h
FDEVOPT1	02AF2Ch	015F2Ch	00AF2Ch

28-Lead Plastic Quad Flat, No Lead Package (ML) - 6x6 mm Body [QFN] With 0.55 mm Terminal Length

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



	Units	MILLIMETERS		
Dimension	Limits	MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		0.65 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3		0.20 REF	
Overall Width	E		6.00 BSC	
Exposed Pad Width	E2	3.65	3.70	4.20
Overall Length	D		6.00 BSC	
Exposed Pad Length	D2	3.65	3.70	4.20
Terminal Width	b	0.23	0.30	0.35
Terminal Length	L	0.50	0.55	0.70
Terminal-to-Exposed Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M.

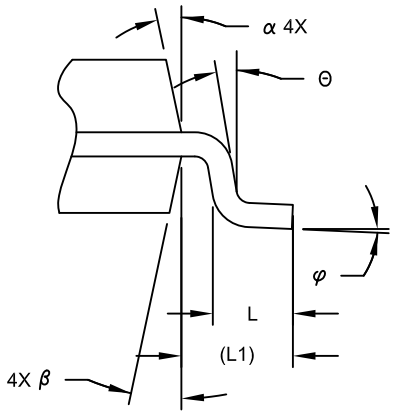
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

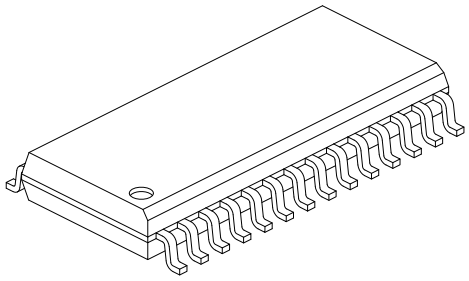
Microchip Technology Drawing C04-105C Sheet 2 of 2

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



VIEW C



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		1.27 BSC	
Overall Height	A	-	-	2.65
Molded Package Thickness	A2	2.05	-	-
Standoff §	A1	0.10	-	0.30
Overall Width	E		10.30 BSC	
Molded Package Width	E1		7.50 BSC	
Overall Length	D		17.90 BSC	
Chamfer (Optional)	h	0.25	-	0.75
Foot Length	L	0.40	-	1.27
Footprint	L1		1.40 REF	
Lead Angle	Θ	0°	-	-
Foot Angle	φ	0°	-	8°
Lead Thickness	c	0.18	-	0.33
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic
3. Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.
5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing C04-052C Sheet 2 of 2