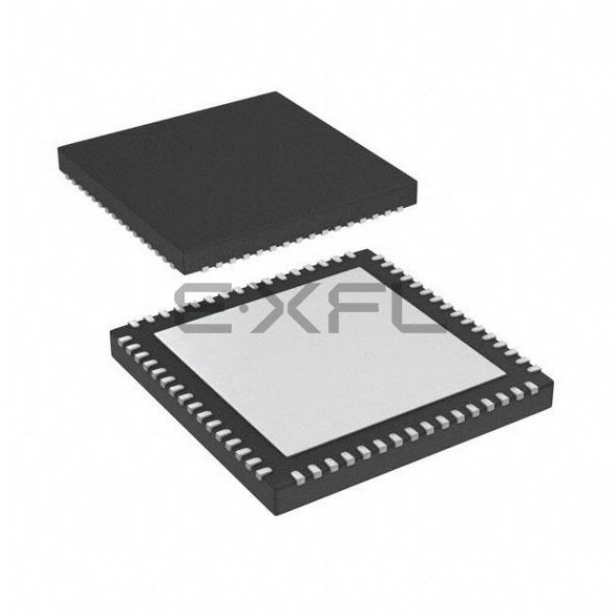
Microchip Technology - DSPIC33CK256MP206T-I/MR Datasheet



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	100MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, QEI, WDT
Number of I/O	53
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	24K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 20x12b; D/A 3x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-QFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33ck256mp206t-i-mr

3.4.1 PROGRAMMER'S MODEL

The programmer's model for the dsPIC33CK256MP508 family is shown in Figure 3-2. All registers in the programmer's model are memory-mapped and can be manipulated directly by instructions. Table 3-1 lists a description of each register.

In addition to the registers contained in the programmer's model, the dsPIC33CK256MP508 devices contain control registers for Modulo Addressing, Bit-Reversed Addressing and interrupts. These registers are described in subsequent sections of this document.

All registers associated with the programmer's model are memory-mapped, as shown in Figure 3-2.

TABLE 3-1: PROGRAMMER'S MODEL REGISTER DESCRIPTIONS

Register(s) Name	Description
W0 through W15(1)	Working Register Array
W0 through W14(1)	Alternate Working Register Array 1
W0 through W14(1)	Alternate Working Register Array 2
W0 through W14(1)	Alternate Working Register Array 3
W0 through W14(1)	Alternate Working Register Array 4
ACCA, ACCB	40-Bit DSP Accumulators (Additional 4 Alternate Accumulators)
PC	23-Bit Program Counter
SR	ALU and DSP Engine STATUS Register
SPLIM	Stack Pointer Limit Value Register
TBLPAG	Table Memory Page Address Register
DSRPAG	Extended Data Space (EDS) Read Page Register
RCOUNT	REPEAT Loop Counter Register
DCOUNT	DO Loop Counter Register
DOSTARTH, DOSTARTL(2)	DO Loop Start Address Register (High and Low)
DOENDH, DOENDL	DO Loop End Address Register (High and Low)
CORCON	Contains DSP Engine, DO Loop Control and Trap Status bits

Note 1: Memory-mapped W0 through W14 represent the value of the register in the currently active CPU context.

2: The DOSTARTH and DOSTARTL registers are read-only.

4.1.3 UNIQUE DEVICE IDENTIFIER (UDID)

All dsPIC33CK256MP508 family devices are individually encoded during final manufacturing with a Unique Device Identifier or UDID. The UDID cannot be erased by a bulk erase command or any other user-accessible means. This feature allows for manufacturing traceability of Microchip Technology devices in applications where this is a requirement. It may also be used by the application manufacturer for any number of things that may require unique identification, such as:

- Tracking the device
- Unique serial number
- Unique security key

The UDID comprises five 24-bit program words. When taken together, these fields form a unique 120-bit identifier.

The UDID is stored in five read-only locations, located between 0x801200 and 0x801208 in the device configuration space. Table 4-1 lists the addresses of the identifier words and shows their contents

TABLE 4-1: UDID ADDRESSES

UDID	Address	Description
UDID1	0x801200	UDID Word 1
UDID2	0x801202	UDID Word 2
UDID3	0x801204	UDID Word 3
UDID4	0x801206	UDID Word 4
UDID5	0x801208	UDID Word 5

4.2 Data Address Space

The dsPIC33CK256MP508 family CPU has a separate 16-bit wide data memory space. The Data Space is accessed using separate Address Generation Units (AGUs) for read and write operations. The data memory maps are shown in Figure 4-7, Figure 4-8 and Figure 4-9.

All Effective Addresses (EAs) in the data memory space are 16 bits wide and point to bytes within the Data Space. This arrangement gives a base Data Space address range of 64 Kbytes or 32K words.

The lower half of the data memory space (i.e., when EA<15> = 0) is used for implemented memory addresses, while the upper half (EA<15> = 1) is reserved for the Program Space Visibility (PSV).

The dsPIC33CK256MP508 family devices implement up to 16 Kbytes of data memory. If an EA points to a location outside of this area, an all-zero word or byte is returned.

4.2.1 DATA SPACE WIDTH

The data memory space is organized in byte-addressable, 16-bit wide blocks. Data is aligned in data memory and registers as 16-bit words, but all Data Space EAs resolve to bytes. The Least Significant Bytes (LSBs) of each word have even addresses, while the Most Significant Bytes (MSBs) have odd addresses.

4.2.2 DATA MEMORY ORGANIZATION AND ALIGNMENT

To maintain backward compatibility with PIC® MCU devices and improve Data Space memory usage efficiency, the dsPIC33CK256MP508 family instruction set supports both word and byte operations. As a consequence of byte accessibility, all Effective Address calculations are internally scaled to step through word-aligned memory. For example, the core recognizes that Post-Modified Register Indirect Addressing mode [Ws++] results in a value of Ws + 1 for byte operations and Ws + 2 for word operations.

A data byte read, reads the complete word that contains the byte, using the LSb of any EA to determine which byte to select. The selected byte is placed onto the LSB of the data path. That is, data memory and registers are organized as two parallel, byte-wide entities with shared (word) address decode, but separate write lines. Data byte writes only write to the corresponding side of the array or register that matches the byte address.

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

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

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

TABLE 4-14: SFR BLOCK D00h

Register	Address	All Resets	Register	Address	All Resets	Register	Address	All Resets
PPS			RPINR21	D2E	1111111111111111	RPOR4	D88	--000000--000000
RPCON	D00	----0----------	RPINR22	D30	1111111111111111	RPOR5	D8A	--000000--000000
RPINR0	D04	11111111--------	RPINR23	D32	--------11111111	RPOR6	D8C	--000000--000000
RPINR1	D06	1111111111111111	RPINR26	D38	--------11111111	RPOR7	D8E	--000000--000000
RPINR2	D08	11111111--------	RPINR27	D3A	1111111111111111	RPOR8	D90	--000000--000000
RPINR3	D0A	1111111111111111	RPINR29	D3E	1111111111111111	RPOR9	D92	--000000--000000
RPINR4	D0C	1111111111111111	RPINR30	D40	--------11111111	RPOR10	D94	--000000--000000
RPINR5	D0E	1111111111111111	RPINR32	D44	11111111--------	RPOR11	D96	--000000--000000
RPINR6	D10	1111111111111111	RPINR33	D46	--------11111111	RPOR12	D98	--000000--000000
RPINR7	D12	1111111111111111	RPINR37	D4E	1111111111111111	RPOR13	D9A	--000000--000000
RPINR8	D14	1111111111111111	RPINR38	D50	--------11111111	RPOR14	D9C	--000000--000000
RPINR9	D16	1111111111111111	RPINR42	D58	1111111111111111	RPOR15	D9E	--000000--000000
RPINR10	D18	1111111111111111	RPINR43	D5A	1111111111111111	RPOR16	DA0	--000000--000000
RPINR11	D1A	1111111111111111	RPINR44	D5C	1111111111111111	RPOR17	DA2	--000000--000000
RPINR12	D1C	1111111111111111	RPINR45	D5E	1111111111111111	RPOR18	DA4	--000000--000000
RPINR13	D1E	1111111111111111	RPINR46	D60	1111111111111111	RPOR19	DA6	--000000--000000
RPINR14	D20	1111111111111111	RPINR47	D62	1111111111111111	RPOR20	DA8	--000000--000000
RPINR15	D22	1111111111111111	RPINR48	D64	1111111111111111	RPOR21	DAA	--000000--000000
RPINR16	D24	1111111111111111	RPINR49	D66	--------11111111	RPOR22	DAC	--000000--000000
RPINR17	D26	1111111111111111	RPOR0	D80	--000000--000000	RPOR23	DAE	--000000--000000
RPINR18	D28	1111111111111111	RPOR1	D82	--000000--000000	RPOR24	DB0	--000000--000000
RPINR19	D2A	1111111111111111	RPOR2	D84	--000000--000000	RPOR25	DB2	--000000--000000
RPINR20	D2C	1111111111111111	RPOR3	D86	--000000--000000	RPOR26	DB4	--000000--000000

Legend: x = unknown or indeterminate value; "-" = unimplemented bits. Address values are in hexadecimal. Reset values are in binary.

7.0 INTERRUPT CONTROLLER

Note 1: This data sheet summarizes the features of the dsPIC33CK256MP508 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Interrupts"** (DS70000600) in the *"dsPIC33/PIC24 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.

The dsPIC33CK256MP508 family interrupt controller reduces the numerous peripheral interrupt request signals to a single interrupt request signal to the dsPIC33CK256MP508 family CPU.

The interrupt controller has the following features:

- Six Processor Exceptions and Software Traps
- Seven User-Selectable Priority Levels
- Interrupt Vector Table (IVT) with a Unique Vector for each Interrupt or Exception Source
- Fixed Priority within a Specified User Priority Level
- Fixed Interrupt Entry and Return Latencies
- Alternate Interrupt Vector Table (AIVT) for Debug Support

7.1 Interrupt Vector Table

The dsPIC33CK256MP508 family Interrupt Vector Table (IVT), shown in Figure 7-1, resides in program memory, starting at location, 000004h. The IVT contains six non-maskable trap vectors and up to 246 sources of interrupts. In general, each interrupt source has its own vector. Each interrupt vector contains a 24-bit wide address. The value programmed into each interrupt vector location is the starting address of the associated Interrupt Service Routine (ISR).

Interrupt vectors are prioritized in terms of their natural priority. This priority is linked to their position in the vector table. Lower addresses generally have a higher natural priority. For example, the interrupt associated with Vector 0 takes priority over interrupts at any other vector address.

7.1.1 ALTERNATE INTERRUPT VECTOR TABLE

The Alternate Interrupt Vector Table (AIVT), shown in Figure 7-2, is available only when the Boot Segment (BS) is defined and the AIVT has been enabled. To enable the Alternate Interrupt Vector Table, the Configuration bit, AIVTDIS in the FSEC register, must be programmed and the AIVTEN bit must be set (INTCON2<8> = 1). When the AIVT is enabled, all interrupt and exception processes use the alternate vectors instead of the default vectors. The AIVT begins at the start of the last page of the Boot Segment, defined by BSLIM<12:0>. The second half of the page is no longer usable space. The Boot Segment must be at least 2 pages to enable the AIVT.

Note: Although the Boot Segment must be enabled in order to enable the AIVT, application code does not need to be present inside of the Boot Segment. The AIVT (and IVT) will inherit the Boot Segment code protection.

The AIVT supports debugging by providing a means to switch between an application and a support environment without requiring the interrupt vectors to be reprogrammed. This feature also enables switching between applications for evaluation of different software algorithms at run time.

7.2 Reset Sequence

A device Reset is not a true exception because the interrupt controller is not involved in the Reset process. The dsPIC33CK256MP508 family devices clear their registers in response to a Reset, which forces the PC to zero. The device then begins program execution at location, 0x000000. A GOTO instruction at the Reset address can redirect program execution to the appropriate start-up routine.

Note: Any unimplemented or unused vector locations in the IVT should be programmed with the address of a default interrupt handler routine that contains a RESET instruction.

REGISTER 8-12: CNFx: INTERRUPT CHANGE NOTIFICATION FLAG FOR PORTx REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNFx<15:8>							
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
CNFx<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- **CNFx<15:0>:** Interrupt Change Notification Flag for PORTx bits

When CNSTYLE (CNCONx<11>) = 1:

1 = An enabled edge event occurred on the PORTx[n] pin

0 = An enabled edge event did not occur on the PORTx[n] pin

REGISTER 8-54: RPOR0: PERIPHERAL PIN SELECT OUTPUT REGISTER 0

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP33R5	RP33R4	RP33R3	RP33R2	RP33R1	RP33R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP32R5	RP32R4	RP32R3	RP32R2	RP32R1	RP32R0
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 **RP33R<5:0>:** Peripheral Output Function is Assigned to RP33 Output Pin bits (see Table 8-7 for peripheral function numbers)

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **RP32R<5:0>:** Peripheral Output Function is Assigned to RP32 Output Pin bits (see Table 8-7 for peripheral function numbers)

REGISTER 8-55: RPOR1: PERIPHERAL PIN SELECT OUTPUT REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP35R5	RP35R4	RP35R3	RP35R2	RP35R1	RP35R0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	RP34R5	RP34R4	RP34R3	RP34R2	RP34R1	RP34R0
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 **RP35R<5:0>:** Peripheral Output Function is Assigned to RP35 Output Pin bits (see Table 8-7 for peripheral function numbers)

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **RP34R<5:0>:** Peripheral Output Function is Assigned to RP34 Output Pin bits (see Table 8-7 for peripheral function numbers)

10.0 DIRECT MEMORY ACCESS (DMA) CONTROLLER

Note 1: This data sheet summarizes the features of this group of dsPIC33 devices. It is not intended to be a comprehensive reference source. For more information, refer to **"Direct Memory Access Controller (DMA)"** (DS39742) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

The Direct Memory Access (DMA) Controller is designed to service high data throughput peripherals operating on the SFR bus, allowing them to access data memory directly and alleviating the need for CPU-intensive management. By allowing these data-intensive peripherals to share their own data path, the main data bus is also deloaded, resulting in additional power savings.

The DMA Controller functions both as a peripheral and a direct extension of the CPU. It is located on the microcontroller data bus, between the CPU and DMA-enabled peripherals, with direct access to SRAM. This partitions the SFR bus into two buses, allowing the DMA Controller access to the DMA-capable peripherals located on the new DMA SFR bus. The controller serves as a Master device on the DMA SFR bus, controlling data flow from DMA-capable peripherals.

The controller also monitors CPU instruction processing directly, allowing it to be aware of when the CPU requires access to peripherals on the DMA bus and automatically relinquishing control to the CPU as needed. This increases the effective bandwidth for handling data without DMA operations, causing a processor Stall. This makes the controller essentially transparent to the user.

The DMA Controller has these features:

- Four Independently Programmable Channels
- Concurrent Operation with the CPU (no DMA caused Wait states)
- DMA Bus Arbitration
- Five Programmable Address modes
- Four Programmable Transfer modes
- Four Flexible Internal Data Transfer modes
- Byte or Word Support for Data Transfer
- 16-Bit Source and Destination Address Register for each Channel, Dynamically Updated and Reloadable
- 16-Bit Transaction Count Register, Dynamically Updated and Reloadable
- Upper and Lower Address Limit Registers
- Counter Half-Full Level Interrupt
- Software Triggered Transfer
- Null Write mode for Symmetric Buffer Operations

A simplified block diagram of the DMA Controller is shown if Figure 10-1.

REGISTER 11-25: C1TXREQH: CAN TRANSMIT REQUEST REGISTER HIGH

S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0
TXREQ<31:24>							
bit 15							bit 8

S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0
TXREQ<23:16>							
bit 7							bit 0

Legend:	S = Settable bit	HC = Hardware 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 15-0 **TXREQ<31:16>:** Unimplemented

REGISTER 11-26: C1TXREQL: CAN TRANSMIT REQUEST REGISTER LOW

S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0
TXREQ<15:8>							
bit 15							bit 8

S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0	S/HC-0
TXREQ<7:1>							TXREQ0
bit 7							bit 0

Legend:	S = Settable bit	HC = Hardware 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 15-8 **TXREQ<15:8>:** Unimplemented

bit 7-1 **TXREQ<7:1>:** Message Send Request bits

TXEN = 1 (object configured as a transmit object):
Setting this bit to '1' requests sending a message. The bit will automatically clear when the message(s) queued in the object is (are) successfully sent. This bit can NOT be used for aborting a transmission.

TXEN = 0 (object configured as a receive object):
This bit has no effect.

bit 0 **TXREQ0:** Transmit Queue Message Send Request bit

Setting this bit to '1' requests sending a message. The bit will automatically clear when the message(s) queued in the object is (are) successfully sent. This bit can NOT be used for aborting a transmission.

REGISTER 12-13: PGxCONH: PWM GENERATOR x CONTROL REGISTER HIGH

R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
MDCSEL	MPERSEL	MPHSEL	—	MSTEN	UPMOD2	UPMOD1	UPMOD0
bit 15							bit 8

r-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TRGMOD	—	—	SOCS3(1,2,3)	SOCS2(1,2,3)	SOCS1(1,2,3)	SOCS0(1,2,3)
bit 7							bit 0

Legend:	r = Reserved 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 **MDCSEL:** Master Duty Cycle Register Select bit
1 = PWM Generator uses MDC register
0 = PWM Generator uses PGxDC register

bit 14 **MPERSEL:** Master Period Register Select bit
1 = PWM Generator uses MPER register
0 = PWM Generator uses PGxPER register

bit 13 **MPHSEL:** Master Phase Register Select bit
1 = PWM Generator uses MPHASE register
0 = PWM Generator uses PGxPHASE register

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

bit 11 **MSTEN:** Master Update Enable bit
1 = PWM Generator broadcasts software set/clear of the UPDREQ status bit and EOC signal to other PWM Generators
0 = PWM Generator does not broadcast the UPDREQ status bit state or EOC signal

bit 10-8 **UPMOD<2:0>:** PWM Buffer Update Mode Selection bits
011 = Slaved immediate update
Data registers immediately, or as soon as possible, when a Master update request is received. A Master update request will be transmitted if MSTEN = 1 and UPDATE = 1 for the requesting PWM Generator.
010 = Slaved SOC update
Data registers at start of next cycle if a Master update request is received. A master update request will be transmitted if MSTEN = 1 and UPDATE = 1 for the requesting PWM Generator.
001 = Immediate update
Data registers immediately, or as soon as possible, if UPDATE = 1. The UPDATE status bit will be cleared automatically after the update occurs (UPDATE = 1). The UPDATE status bit will be cleared automatically after the update occurs.
000 = SOC update
Data registers at start of next PWM cycle if UPDATE = 1. The UPDATE status bit will be cleared automatically after the update occurs.(1)

bit 7 **Reserved:** Maintain as '0'

Note 1: The PCI selected Sync signal is always available to be OR'd with the selected SOC signal per the SOCS<3:0> bits if the PCI Sync function is enabled.

2: The source selected by the SOCS<3:0> bits MUST operate from the same clock source as the local PWM Generator. If not, the source must be routed through the PCI Sync logic so the trigger signal may be synchronized to the PWM Generator clock domain.

3: PWM Generators are grouped into groups of four: PG1-PG4 and PG5-PG8, if available. Any generator within a group of four may be used to trigger another generator within the same group.

REGISTER 13-8: ADCON4H: ADC CONTROL REGISTER 4 HIGH

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	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	C1CHS1	C1CHS0	C0CHS1	C0CHS0
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 15-4 **Unimplemented:** Read as ‘0’

bit 3-2 **C1CHS<1:0>:** Dedicated ADC Core 1 Input Channel Selection bits
11 = Reserved
10 = Reserved
01 = ANA1
00 = AN1

bit 1-0 **C0CHS<1:0>:** Dedicated ADC Core 0 Input Channel Selection bits
11 = Reserved
10 = Reserved
01 = ANA0
00 = AN0

REGISTER 14-1: DACCTRL1L: DAC CONTROL 1 LOW REGISTER

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
DACON	—	DACSIDL	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
CLKSEL1(1,3)	CLKSEL0(1,3)	CLKDIV1(1,3)	CLKDIV0(1,3)	—	FCLKDIV2(2)	FCLKDIV1(2)	FCLKDIV0(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

bit 15 **DACON:** Common DAC Module Enable bit
1 = Enables DAC modules
0 = Disables DAC modules and disables FSCM clocks to reduce power consumption; any pending Slope mode and/or underflow conditions are cleared

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

bit 13 **DACSIDL:** DAC Stop in Idle Mode bit
1 = Discontinues module operation when device enters Idle mode
0 = Continues module operation in Idle mode

bit 12-8 **Unimplemented:** Read as '0'

bit 7-6 **CLKSEL<1:0>:** DAC Clock Source Select bits(1,3)
11 = FPLLO
10 = AFPLLO
01 = FVCO/2
00 = AFVCO/2

bit 5-4 **CLKDIV<1:0>:** DAC Clock Divider bits(1,3)
11 = Divide-by-4
10 = Divide-by-3 (non-uniform duty cycle)
01 = Divide-by-2
00 = 1x

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

bit 2-0 **FCLKDIV<2:0>:** Comparator Filter Clock Divider bits(2)
111 = Divide-by-8
110 = Divide-by-7
101 = Divide-by-6
100 = Divide-by-5
011 = Divide-by-4
010 = Divide-by-3
001 = Divide-by-2
000 = 1x

Note 1: These bits should only be changed when DACON = 0 to avoid unpredictable behavior.
2: The input clock to this divider is the selected clock input, CLKSEL<1:0>, and then divided by 2.
3: Clock source and dividers should yield an effective DAC clock input of 500 MHz.

15.0 QUADRATURE ENCODER INTERFACE (QEI)

Note 1: This data sheet summarizes the features of the dsPIC33CK256MP508 family of devices. It is not intended to be a comprehensive resource. For more information, refer to **"Quadrature Encoder Interface (QEI)"** (DS70000601) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

The Quadrature Encoder Interface (QEI) module provides the interface to incremental encoders for obtaining mechanical position data. The dsPIC33CK256MP508 family implements 2 instances of the QEI. Quadrature Encoders, also known as incremental encoders or optical encoders, detect position and speed of rotating motion systems. Quadrature Encoders enable closed-loop control of motor control applications, such as Switched Reluctance (SR) and AC Induction Motors (ACIM).

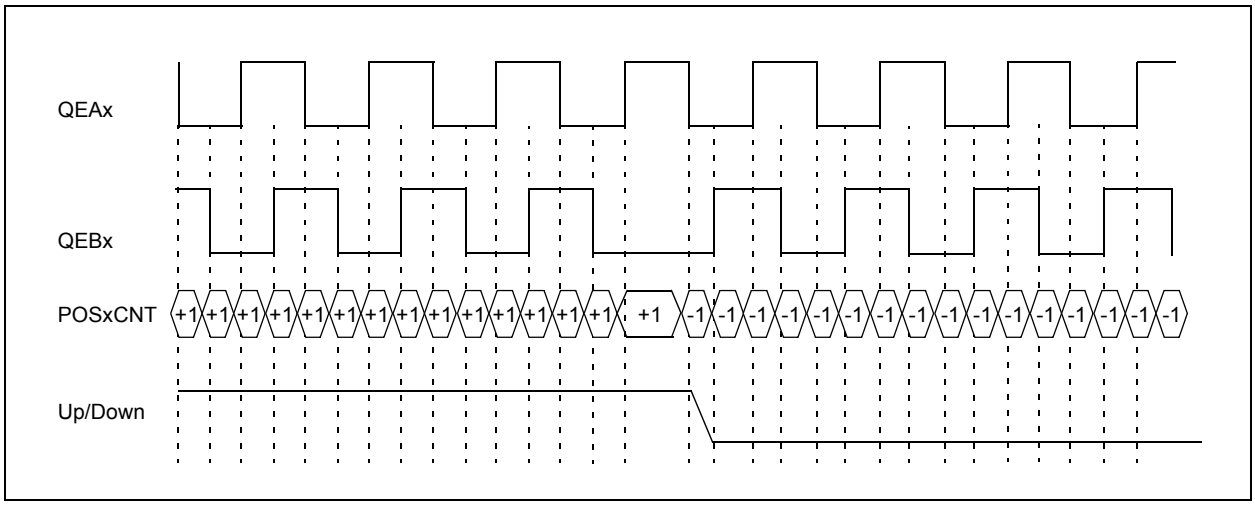
A typical Quadrature Encoder includes a slotted wheel attached to the shaft of the motor and an emitter/ detector module that senses the slots in the wheel. Typically, three output channels, Phase A (QEAx), Phase B (QEBx) and Index (INDXx), provide information on the movement of the motor shaft, including distance and direction.

The two channels, Phase A (QEAx) and Phase B (QEBx), are typically 90 degrees out of phase with respect to each other. The Phase A and Phase B channels have a unique relationship. If Phase A leads Phase B, the direction of the motor is deemed positive or forward. If Phase A lags Phase B, the direction of the motor is deemed negative or reverse. The Index pulse occurs once per mechanical revolution and is used as a reference to indicate an absolute position. Figure 15-1 illustrates the Quadrature Encoder Interface signals.

The Quadrature signals from the encoder can have four unique states ('01', '00', '10' and '11') that reflect the relationship between QEAx and QEBx. Figure 15-1 illustrates these states for one count cycle. The order of the states get reversed when the direction of travel changes.

The Quadrature Decoder increments or decrements the 32-bit up/down Position x Counter (POSxCNTH/L) registers for each Change-of-State (COS). The counter increments when QEAx leads QEBx and decrements when QEBx leads QEAx.

FIGURE 15-1: QUADRATURE ENCODER INTERFACE SIGNALS



21.0 TIMER1

Note 1: This data sheet summarizes the features of the dsPIC33CK256MP508 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **"Timers"** (DS70362) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

The Timer1 module is a 16-bit timer that can operate as a free-running interval timer/counter.

The Timer1 module has the following unique features over other timers:

- Can be Operated in Asynchronous Counter mode
- Asynchronous Timer
- Operational during CPU Sleep mode
- Software Selectable Prescalers 1:1, 1:8, 1:64 and 1:256
- External Clock Selection Control
- The Timer1 External Clock Input (T1CK) can Optionally be Synchronized to the Internal Device Clock and the Clock Synchronization is Performed after the Prescaler

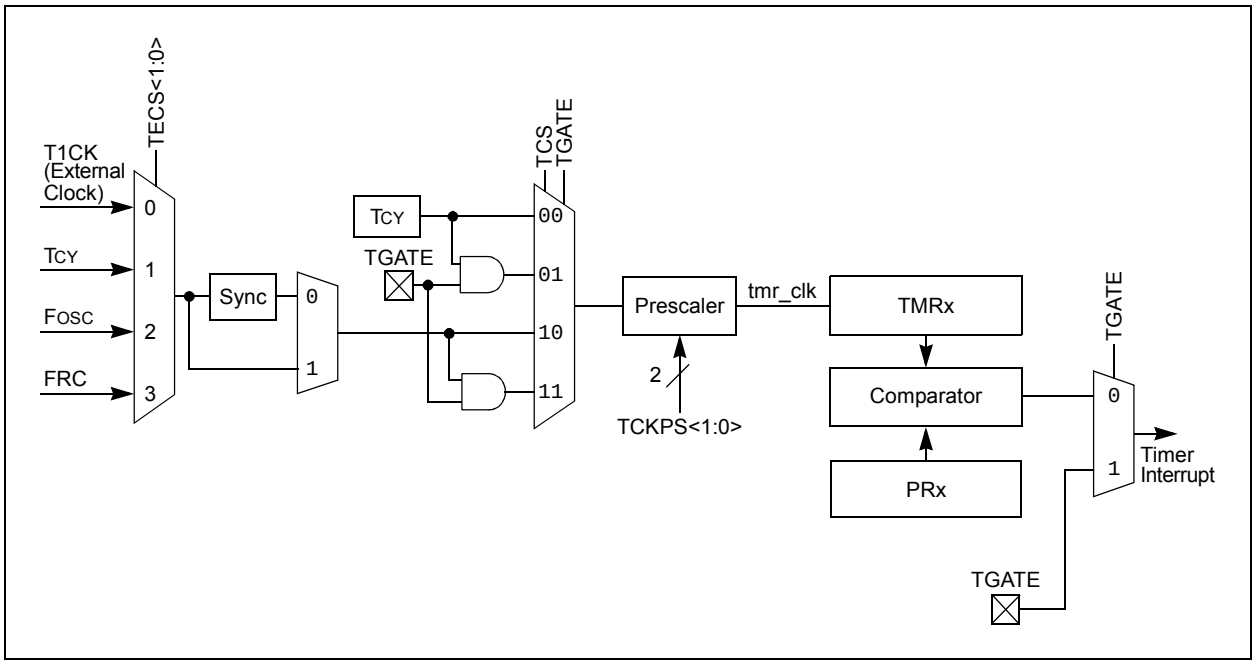
If Timer1 is used for SCCP, the timer should be running in Synchronous mode.

The Timer1 module can operate in one of the following modes:

- Timer mode
- Gated Timer mode
- Synchronous Counter mode
- Asynchronous Counter mode

A block diagram of Timer1 is shown in Figure 21-1.

FIGURE 21-1: 16-BIT TIMER1 MODULE BLOCK DIAGRAM



REGISTER 22-2: CCPxCON1H: CCPx CONTROL 1 HIGH REGISTERS

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
OPSSRC(1)	RTRGEN(2)	—	—	OPS3(3)	OPS2(3)	OPS1(3)	OPS0(3)
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
TRIGEN	ONESHOT	ALTSYNC	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0
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 **OPSSRC:** Output Postscaler Source Select bit(1)
1 = Output postscaler scales module trigger output events
0 = Output postscaler scales time base interrupt events

bit 14 **RTRGEN:** Retrigger Enable bit(2)
1 = Time base can be retriggered when TRIGEN bit = 1
0 = Time base may not be retriggered when TRIGEN bit = 1

bit 13-12 **Unimplemented:** Read as '0'

bit 11-8 **OPS3<3:0>:** CCPx Interrupt Output Postscale Select bits(3)
1111 = Interrupt every 16th time base period match
1110 = Interrupt every 15th time base period match
...
0100 = Interrupt every 5th time base period match
0011 = Interrupt every 4th time base period match or 4th input capture event
0010 = Interrupt every 3rd time base period match or 3rd input capture event
0001 = Interrupt every 2nd time base period match or 2nd input capture event
0000 = Interrupt after each time base period match or input capture event

bit 7 **TRIGEN:** CCPx Trigger Enable bit
1 = Trigger operation of time base is enabled
0 = Trigger operation of time base is disabled

bit 6 **ONESHOT:** One-Shot Trigger Mode Enable bit
1 = One-Shot Trigger mode is enabled; trigger duration is set by OSCNT<2:0>
0 = One-Shot Trigger mode is disabled

bit 5 **ALTSYNC:** CCPx Clock Select bits
1 = An alternate signal is used as the module synchronization output signal
0 = The module synchronization output signal is the Time Base Reset/rollover event

bit 4-0 **SYNC<4:0>:** CCPx Synchronization Source Select bits
See Table 22-5 for the definition of inputs.

Note 1: This control bit has no function in Input Capture modes.
2: This control bit has no function when TRIGEN = 0.
3: Output postscale settings, from 1:5 to 1:16 (0100-1111), will result in a FIFO buffer overflow for Input Capture modes.

23.1 Control Registers

The CLCx module is controlled by the following registers:

- CLCxCONL
- CLCxCONH
- CLCxSEL
- CLCxGLSL
- CLCxGLSH

The CLCx Control registers (CLCxCONL and CLCxCONH) are used to enable the module and interrupts, control the output enable bit, select output polarity and select the logic function. The CLCx Control registers also allow the user to control the logic polarity of not only the cell output, but also some intermediate variables.

The CLCx Input MUX Select register (CLCxSEL) allows the user to select up to 4 data input sources using the 4 data input selection multiplexers. Each multiplexer has a list of 8 data sources available.

The CLCx Gate Logic Input Select registers (CLCxGLSL and CLCxGLSH) allow the user to select which outputs from each of the selection MUXes are used as inputs to the input gates of the logic cell. Each data source MUX outputs both a true and a negated version of its output. All of these 8 signals are enabled, ORed together by the logic cell input gates.

REGISTER 23-1: CLCxCONL: CLCx CONTROL REGISTER (LOW)

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0	U-0
LCEN	—	—	—	INTP	INTN	—	—
bit 15							bit 8

R-0	R-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
LCOE	LCOUT	LCPOL	—	—	MODE2	MODE1	MODE0
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 **LCEN:** CLCx Enable bit
1 = CLCx is enabled and mixing input signals
0 = CLCx is disabled and has logic zero outputs

bit 14-12 **Unimplemented:** Read as ‘0’

bit 11 **INTP:** CLCx Positive Edge Interrupt Enable bit
1 = Interrupt will be generated when a rising edge occurs on LCOUT
0 = Interrupt will not be generated

bit 10 **INTN:** CLCx Negative Edge Interrupt Enable bit
1 = Interrupt will be generated when a falling edge occurs on LCOUT
0 = Interrupt will not be generated

bit 9-8 **Unimplemented:** Read as ‘0’

bit 7 **LCOE:** CLCx Port Enable bit
1 = CLCx port pin output is enabled
0 = CLCx port pin output is disabled

bit 6 **LCOUT:** CLCx Data Output Status bit
1 = CLCx output high
0 = CLCx output low

bit 5 **LCPOL:** CLCx Output Polarity Control bit
1 = The output of the module is inverted
0 = The output of the module is not inverted

bit 4-3 **Unimplemented:** Read as ‘0’

REGISTER 24-2: PTGCON: PTG CONTROL/STATUS HIGH REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTGCLK2	PTGCLK1	PTGCLK0	PTGDIV4	PTGDIV3	PTGDIV2	PTGDIV1	PTGDIV0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
PTGPWD3	PTGPWD2	PTGPWD1	PTGPWD0	—	PTGWDT2	PTGWDT1	PTGWDT0
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 15-13 **PTGCLK<2:0>:** PTG Module Clock Source Selection bits
111 = CLC1
110 = PLL VCO DIV 4 output
101 = PTG module clock source will be SCCP7
100 = PTG module clock source will be SCCP8
011 = Input from Timer1 Clock pin, T1CK
010 = PTG module clock source will be ADC clock
001 = PTG module clock source will be FOSC
000 = PTG module clock source will be FOSC/2 (FP)

bit 12-8 **PTGDIV<4:0>:** PTG Module Clock Prescaler (Divider) bits
11111 = Divide-by-32
11110 = Divide-by-31
...
00001 = Divide-by-2
00000 = Divide-by-1

bit 7-4 **PTGPWD<3:0>:** PTG Trigger Output Pulse-Width (in PTG clock cycles) bits
1111 = All trigger outputs are 16 PTG clock cycles wide
1110 = All trigger outputs are 15 PTG clock cycles wide
...
0001 = All trigger outputs are 2 PTG clock cycles wide
0000 = All trigger outputs are 1 PTG clock cycle wide

bit 3 **Unimplemented:** Read as ‘0’

bit 2-0 **PTGWDT<2:0>:** PTG Watchdog Timer Time-out Selection bits
111 = Watchdog Timer will time out after 512 PTG clocks
110 = Watchdog Timer will time out after 256 PTG clocks
101 = Watchdog Timer will time out after 128 PTG clocks
100 = Watchdog Timer will time out after 64 PTG clocks
011 = Watchdog Timer will time out after 32 PTG clocks
010 = Watchdog Timer will time out after 16 PTG clocks
001 = Watchdog Timer will time out after 8 PTG clocks
000 = Watchdog Timer is disabled

29.2.1 SLEEP MODE

The following occurs in Sleep mode:

- The system clock source is shut down. If an on-chip oscillator is used, it is turned off.
- The device current consumption is reduced to a minimum, provided that no I/O pin is sourcing current.
- The Fail-Safe Clock Monitor does not operate, since the system clock source is disabled.
- The LPRC clock continues to run in Sleep mode if the WDT is enabled.
- The WDT, if enabled, is automatically cleared prior to entering Sleep mode.
- Some device features or peripherals can continue to operate. This includes items such as the Input Change Notification on the I/O ports or peripherals that use an External Clock input.
- Any peripheral that requires the system clock source for its operation is disabled.

The device wakes up from Sleep mode on any of the these events:

- Any interrupt source that is individually enabled
- Any form of device Reset
- A WDT time-out

On wake-up from Sleep mode, the processor restarts with the same clock source that was active when Sleep mode was entered.

For optimal power savings, the regulators can be configured to go into standby when Sleep mode is entered by clearing the VREGS (RCON<8>) bit (default configuration).

If the application requires a faster wake-up time, and can accept higher current requirements, the VREGS (RCON<8>) bit can be set to keep the regulators active during Sleep mode. The available Low-Power Sleep modes are shown in Table 29-1. Additional regulator information is available in **Section 30.4 "On-Chip Voltage Regulators"**.

TABLE 29-1: LOW-POWER SLEEP MODES

Relative Power	LPWREN	VREGS	MODE
Highest	0	1	Full power, active
—	0	0	Full power, standby
—	1(1)	1	Low power, active
Lowest	1(1)	0	Low power, standby

Note 1: Low-Power modes, when LPWREN = 1, can only be used in the industrial temperature range.

29.2.2 IDLE MODE

The following occurs in Idle mode:

- The CPU stops executing instructions.
- The WDT is automatically cleared.
- The system clock source remains active. By default, all peripheral modules continue to operate normally from the system clock source, but can also be selectively disabled (see **Section 29.4 "Peripheral Module Disable"**).
- If the WDT or FSCM is enabled, the LPRC also remains active.

The device wakes from Idle mode on any of these events:

- Any interrupt that is individually enabled
- Any device Reset
- A WDT time-out

On wake-up from Idle mode, the clock is reapplied to the CPU and instruction execution will begin (2-4 clock cycles later), starting with the instruction following the PWRSAV instruction or the first instruction in the ISR.

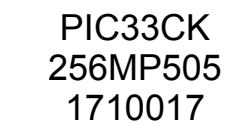
All peripherals also have the option to discontinue operation when Idle mode is entered to allow for increased power savings. This option is selectable in the control register of each peripheral; for example, the SIDL bit in the Timer1 Control register (T1CON<13>).

29.2.3 INTERRUPTS COINCIDENT WITH POWER SAVE INSTRUCTIONS

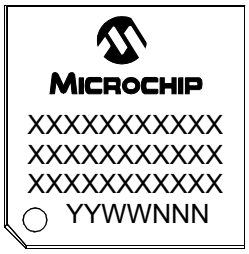
Any interrupt that coincides with the execution of a PWRSAV instruction is held off until entry into Sleep or Idle mode has completed. The device then wakes up from Sleep or Idle mode.

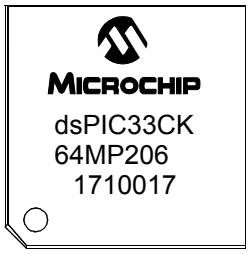
34.1 Package Marking Information (Continued)

48-Lead UQFN (6x6 mm) Example

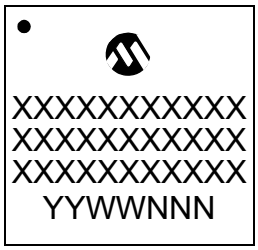


64-Lead TQFP (10x10x1 mm) Example

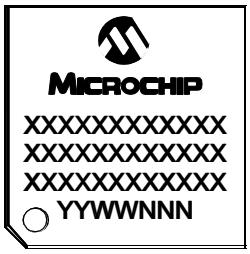


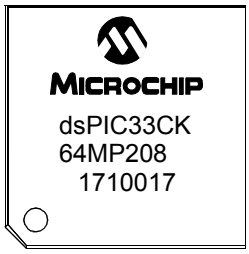


64-Lead QFN (9x9x0.9 mm) Example



80-Lead TQFP (12x12x1 mm) Example

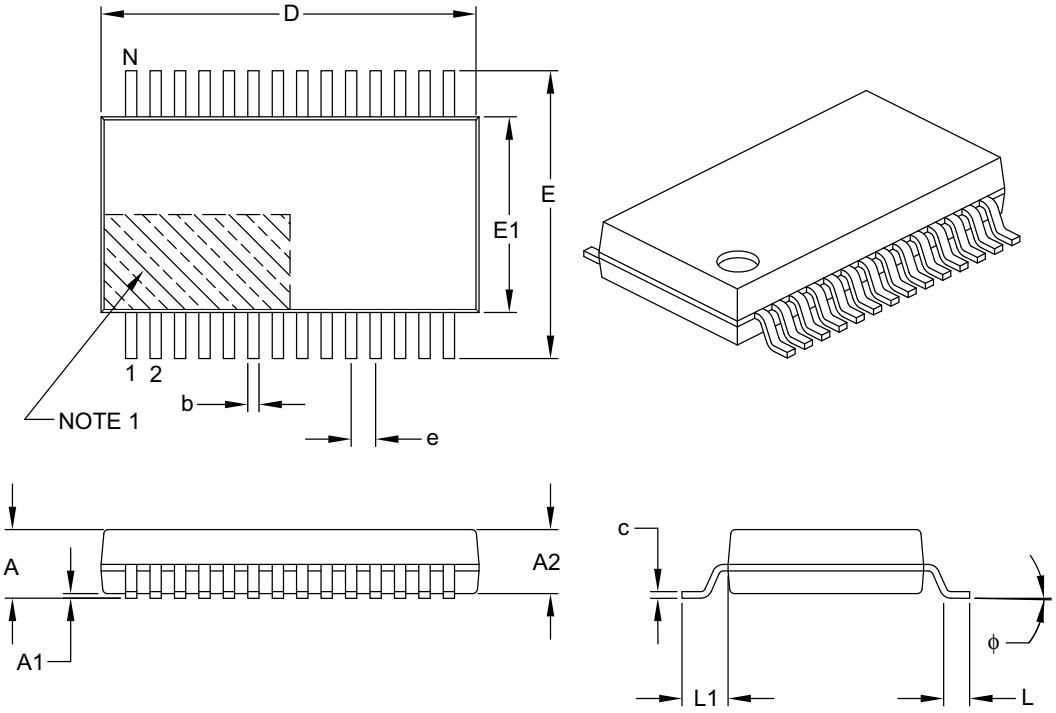




34.2 Package Details

28-Lead Plastic Shrink Small Outline (SS) – 5.30 mm Body [SSOP]

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	–	–	2.00
Molded Package Thickness	A2	1.65	1.75	1.85
Standoff	A1	0.05	–	–
Overall Width	E	7.40	7.80	8.20
Molded Package Width	E1	5.00	5.30	5.60
Overall Length	D	9.90	10.20	10.50
Foot Length	L	0.55	0.75	0.95
Footprint	L1		1.25 REF	
Lead Thickness	c	0.09	–	0.25
Foot Angle	ϕ	0°	4°	8°
Lead Width	b	0.22	–	0.38

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
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 C04-073B

NOTES:

