Microchip Technology - DSPIC33CH128MP508-E/PT Datasheet

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit Dual-Core
Speed	180MHz, 200MHz
Connectivity	CANbus, I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, QEI, WDT
Number of I/O	69
Program Memory Size	152KB (152K x 8)
Program Memory Type	FLASH, PRAM
EEPROM Size	-
RAM Size	20K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 34x12b; D/A 4x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	80-TQFP
Supplier Device Package	80-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33ch128mp508-e-pt

3.1.5 PROGRAMMER'S MODEL

The programmer's model for the dsPIC33CH128MP508 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 dsPIC33CH128MP508 devices contain control registers for Modulo Addressing, Bit-Reversed Addressing and interrupts. These registers are described in subsequent sections of this document.

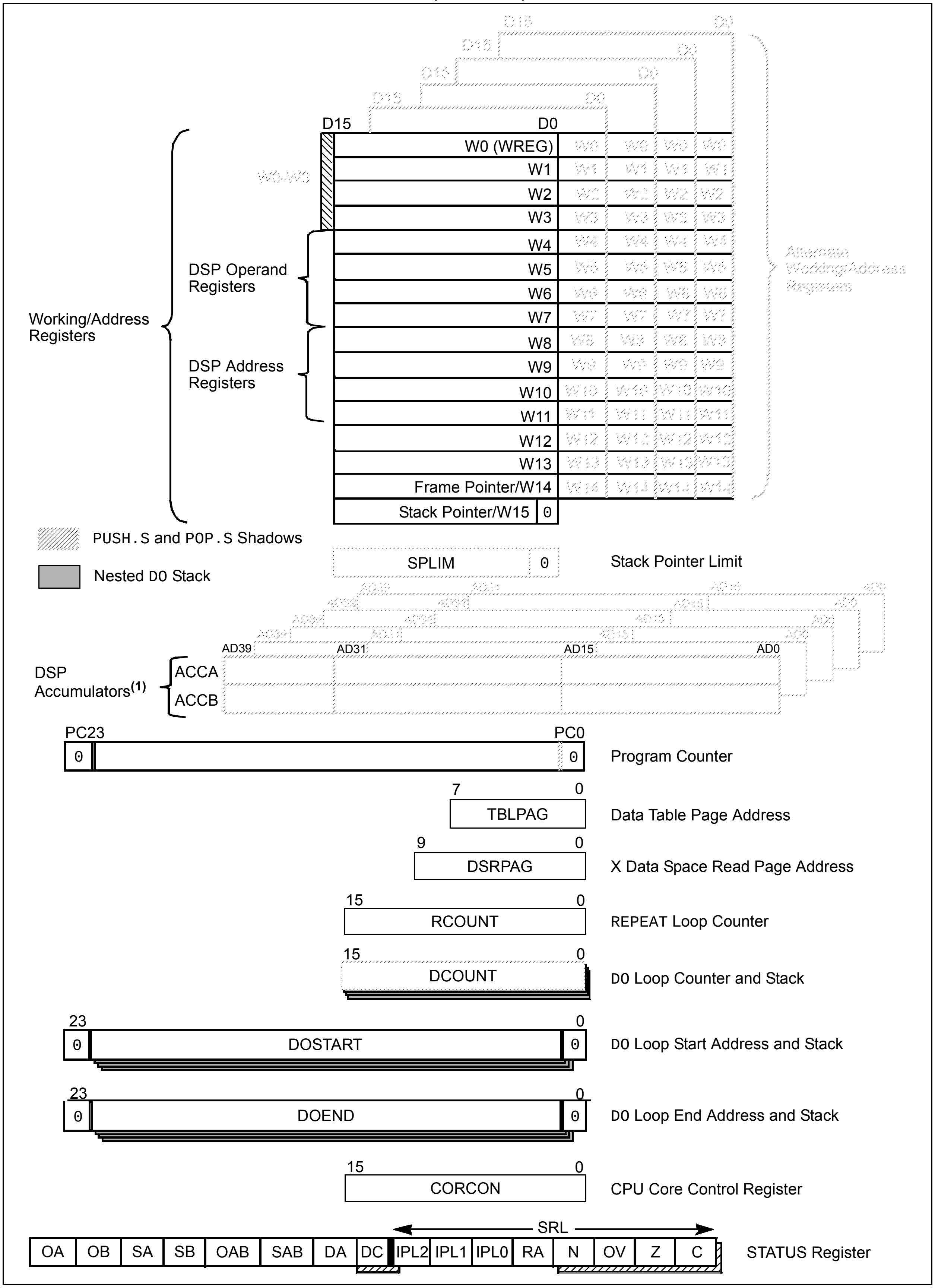
All registers associated with the programmer's model are memory-mapped, as shown in Figure 3-3 and Figure 3-4.

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.

FIGURE 3-2: PROGRAMMER'S MODEL (MASTER)



3.2.3 DATA ADDRESS SPACE (MASTER)

The dsPIC33CH128MP508 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 map is shown in Figure 3-6.

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 dsPIC33CH128MP508 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.

3.2.3.1 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.

3.2.3.2 Data Memory Organization and Alignment

To maintain backward compatibility with PIC® MCU devices and improve Data Space memory usage efficiency, the dsPIC33CH128MP508 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.

3.2.3.3 SFR Space

The first 4 Kbytes of the Near Data Space, from 0x0000 to 0x0FFF, is primarily occupied by Special Function Registers (SFRs). These are used by the dsPIC33CH128MP508 family core and peripheral modules for controlling the operation of the device.

SFRs are distributed among the modules that they control and are generally grouped together by module. Much of the SFR space contains unused addresses; these are read as '0'.

Note:	The actual set of peripheral features and interrupts varies by the device. Refer to the corresponding device tables and pinout diagrams for device-specific information.

3.2.3.4 Near Data Space

The 8-Kbyte area, between 0x0000 and 0x1FFF, is referred to as the Near Data Space. Locations in this space are directly addressable through a 13-bit absolute address field within all memory direct instructions. Additionally, the whole Data Space is addressable using MOV instructions, which support Memory Direct Addressing mode with a 16-bit address field, or by using Indirect Addressing mode using a Working register as an Address Pointer.

REGISTER 3-18: INTCON1: INTERRUPT CONTROL REGISTER 1 (CONTINUED)

bit 6	**DIV0ERR:** Divide-by-Zero Error Status bit 1 = Math error trap was caused by a divide-by-zero 0 = Math error trap was not caused by a divide-by-zero
bit 5	**DMACERR:** DMA Controller Trap Status bit 1 = DMAC error trap has occurred 0 = DMAC error trap has not occurred
bit 4	**MATHERR:** Math Error Status bit 1 = Math error trap has occurred 0 = Math error trap has not occurred
bit 3	**ADDRERR:** Address Error Trap Status bit 1 = Address error trap has occurred 0 = Address error trap has not occurred
bit 2	**STKERR:** Stack Error Trap Status bit 1 = Stack error trap has occurred 0 = Stack error trap has not occurred
bit 1	**OSCFAIL:** Oscillator Failure Trap Status bit 1 = Oscillator failure trap has occurred 0 = Oscillator failure trap has not occurred
bit 0	**Unimplemented:** Read as ‘0’

REGISTER 3-63: RPINR43: PERIPHERAL PIN SELECT INPUT REGISTER 43

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCI15R7	PCI15R6	PCI15R5	PCI15R4	PCI15R3	PCI15R2	PCI15R1	PCI15R0
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
PCI14R7	PCI14R6	PCI14R5	PCI14R4	PCI14R3	PCI14R2	PCI14R1	PCI14R0
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 **PCI15R<7:0>:** Assign PWM Input 15 (PCI15) to the Corresponding RPn Pin bits
See Table 3-30.

bit 7-0 **PCI14R<7:0>:** Assign PWM Input 14 (PCI14) to the Corresponding RPn Pin bits
See Table 3-30.

REGISTER 3-64: RPINR44: PERIPHERAL PIN SELECT INPUT REGISTER 44

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SENT1R7	SENT1R6	SENT1R5	SENT1R4	SENT1R3	SENT1R2	SENT1R1	SENT1R0
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
PCI16R7	PCI16R6	PCI16R5	PCI16R4	PCI16R3	PCI16R2	PCI16R1	PCI16R0
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 **SENT1R<7:0>:** Assign SENT1 Input (SENT1) to the Corresponding RPn Pin bits
See Table 3-30.

bit 7-0 **PCI16<7:0>:** Assign PWM Input 16 (PCI16) to the Corresponding RPn Pin bits
See Table 3-30.

REGISTER 3-131: C1TXQCONL: CAN TRANSMIT QUEUE CONTROL REGISTER LOW

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	FRESET	TXREQ	UINC
bit 15							bit 8

R-0	U-0	U-0	HS/C-0	U-0	R/W-0	U-0	R/W-0
TXEN	—	—	TXATIE	—	TXQEIE	—	TXQNIE
bit 7							bit 0

Legend:	HS = Hardware Settable bit	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 15-11 **Unimplemented:** Read as ‘0’

bit 10 **FRESET:** FIFO Reset bit
1 = FIFO will be reset when bit is set, cleared by hardware when FIFO is reset; user should poll whether this bit is clear before taking any action
0 = No effect

bit 9 **TXREQ:** Message Send Request bit
1 = Requests sending a message; the bit will automatically clear when all the messages queued in the TXQ are successfully sent
0 = Clearing the bit to ‘0’ while set (‘1’) will request a message abort

bit 8 **UINC:** Increment Head/Tail bit
When this bit is set, the FIFO head will increment by a single message.

bit 7 **TXEN:** TX Enable bit

bit 6-5 **Unimplemented:** Read as ‘0’

bit 4 **TXATIE:** Transmit Attempts Exhausted Interrupt Enable bit
1 = Enables interrupt
0 = Disables interrupt

bit 3 **Unimplemented:** Read as ‘0’

bit 2 **TXQEIE:** Transmit Queue Empty Interrupt Enable bit
1 = Interrupt is enabled for TXQ empty
0 = Interrupt is disabled for TXQ empty

bit 1 **Unimplemented:** Read as ‘0’

bit 0 **TXQNIE:** Transmit Queue Not Full Interrupt Enable bit
1 = Interrupt is enabled for TXQ not full
0 = Interrupt is disabled for TXQ not full

REGISTER 3-138: C1TEFSTA: CAN TRANSMIT EVENT FIFO STATUS REGISTER

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	S/HC-0	R-0	R-0	R-0
—	—	—	—	TEFOVIF	TEFFIF(1)	TEFHIF(1)	TEFNEIF(1)
bit 7							bit 0

Legend:	HC = Hardware Clearable bit	S = Settable by ‘1’ 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 15-4 **Unimplemented:** Read as ‘0’

bit 3 **TEFOVIF:** Transmit Event FIFO Overflow Interrupt Flag bit
1 = Overflow event has occurred
0 = No overflow event has occurred

bit 2 **TEFFIF:** Transmit Event FIFO Full Interrupt Flag bit(1)
1 = FIFO is full
0 = FIFO is not full

bit 1 **TEFHIF:** Transmit Event FIFO Half Full Interrupt Flag bit(1)
1 = FIFO is ≥ half full
0 = FIFO is < half full

bit 0 **TEFNEIF:** Transmit Event FIFO Not Empty Interrupt Flag bit(1)
1 = FIFO is not empty
0 = FIFO is empty

Note 1: These bits are read-only and reflect the status of the FIFO.

3.9.3 ADC CONTROL/STATUS REGISTERS

REGISTER 3-157: ADCON1L: ADC CONTROL REGISTER 1 LOW

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
ADON[(1)]	—	ADSIDL	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-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 **ADON:** ADC Enable bit[(1)]
1 = ADC module is enabled
0 = ADC module is off

bit 14 **Unimplemented:** Read as ‘0’

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

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

Note 1: Set the ADON bit only after the ADC module has been configured. Changing ADC Configuration bits when ADON = 1 will result in unpredictable behavior.

REGISTER 3-159: ADCON2L: ADC CONTROL REGISTER 2 LOW

R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
REFCIE	REFERCIE	—	EIEN	—	SHREISEL2(1)	SHREISEL1(1)	SHREISEL0(1)
bit 15							bit 8

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	SHRADCS6	SHRADCS5	SHRADCS4	SHRADCS3	SHRADCS2	SHRADCS1	SHRADCS0
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 **REFCIE:** Band Gap and Reference Voltage Ready Common Interrupt Enable bit
1 = Common interrupt will be generated when the band gap becomes ready
0 = Common interrupt is disabled for the band gap ready event

bit 14 **REFERCIE:** Band Gap or Reference Voltage Error Common Interrupt Enable bit
1 = Common interrupt will be generated when a band gap or reference voltage error is detected
0 = Common interrupt is disabled for the band gap and reference voltage error event

bit 13 **Unimplemented:** Read as ‘0’

bit 12 **EIEN:** Early Interrupts Enable bit
1 = The early interrupt feature is enabled for the input channel interrupts (when the EISTATx flag is set)
0 = The individual interrupts are generated when conversion is done (when the ANxRDY flag is set)

bit 11 **Unimplemented:** Read as ‘0’

bit 10-8 **SHREISEL<2:0>:** Shared Core Early Interrupt Time Selection bits(1)
111 = Early interrupt is set and interrupt is generated 8 TADCORE clocks prior to when the data is ready
110 = Early interrupt is set and interrupt is generated 7 TADCORE clocks prior to when the data is ready
101 = Early interrupt is set and interrupt is generated 6 TADCORE clocks prior to when the data is ready
100 = Early interrupt is set and interrupt is generated 5 TADCORE clocks prior to when the data is ready
011 = Early interrupt is set and interrupt is generated 4 TADCORE clocks prior to when the data is ready
010 = Early interrupt is set and interrupt is generated 3 TADCORE clocks prior to when the data is ready
001 = Early interrupt is set and interrupt is generated 2 TADCORE clocks prior to when the data is ready
000 = Early interrupt is set and interrupt is generated 1 TADCORE clock prior to when the data is ready

bit 7 **Unimplemented:** Read as ‘0’

bit 6-0 **SHRADCS<6:0>:** Shared ADC Core Input Clock Divider bits
These bits determine the number of TCORESRC (Source Clock Periods) for one shared TADCORE (Core Clock Period).
1111111 = 254 Source Clock Periods
...
0000011 = 6 Source Clock Periods
0000010 = 4 Source Clock Periods
0000001 = 2 Source Clock Periods
0000000 = 2 Source Clock Periods

Note 1: For the 6-bit shared ADC core resolution (SHRRES<1:0> = 00), the SHREISEL<2:0> settings, from ‘100’ to ‘111’, are not valid and should not be used. For the 8-bit shared ADC core resolution (SHRRES<1:0> = 01), the SHREISEL<2:0> settings, ‘110’ and ‘111’, are not valid and should not be used.

REGISTER 3-160: ADCON2H: ADC CONTROL REGISTER 2 HIGH

HSC/R-0	HSC/R-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
REFRDY	REFERR	—	—	—	—	SHRSAMC9	SHRSAMC8
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
SHRSAMC7	SHRSAMC6	SHRSAMC5	SHRSAMC4	SHRSAMC3	SHRSAMC2	SHRSAMC1	SHRSAMC0
bit 7							bit 0

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

bit 15 **REFRDY:** Band Gap and Reference Voltage Ready Flag bit
1 = Band gap is ready
0 = Band gap is not ready

bit 14 **REFERR:** Band Gap or Reference Voltage Error Flag bit
1 = Band gap was removed after the ADC module was enabled (ADON = 1)
0 = No band gap error was detected

bit 13-10 **Unimplemented:** Read as ‘0’

bit 9-0 **SHRSAMC<9:0>:** Shared ADC Core Sample Time Selection bits
These bits specify the number of shared ADC Core Clock Periods (TADCORE) for the shared ADC core sample time (Sample Time = (SHRSAMC<9:0> + 2) * TADCORE).
1111111111 = 1025 TADCORE
...
0000000001 = 3 TADCORE
0000000000 = 2 TADCORE

REGISTER 3-169: ADEISTATL: ADC EARLY INTERRUPT STATUS REGISTER LOW

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EISTAT<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
EISTAT<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 **EISTAT<15:0>:** Early Interrupt Status for Corresponding Analog Input bits
1 = Early interrupt was generated
0 = Early interrupt was not generated since the last ADCBUFx read

REGISTER 3-170: ADEISTATH: ADC EARLY INTERRUPT STATUS REGISTER 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	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	EISTAT<20:16>				
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 4-0 **EISTAT<20:16>:** Early Interrupt Status for Corresponding Analog Input bits
1 = Early interrupt was generated
0 = Early interrupt was not generated since the last ADCBUFx read

4.1.6 CPU RESOURCES

Many useful resources are provided on the main product page of the Microchip web site for the devices listed in this data sheet. This product page contains the latest updates and additional information.

4.1.6.1 Key Resources

- **“dsPIC33E Enhanced CPU”** (DS70005158) in the *“dsPIC33/PIC24 Family Reference Manual”*
- Code Samples
- Application Notes
- Software Libraries
- Webinars
- All related *“dsPIC33/PIC24 Family Reference Manual”* Sections
- Development Tools

REGISTER 4-101: ADMOD0L: ADC INPUT MODE CONTROL REGISTER 0 LOW

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
—	—	—	—	DIFF1	SIGN1	DIFF0	SIGN0
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 and bit 1 (odd)	**DIFF<1:0>:** Differential-Mode for Corresponding Analog Inputs bits 1 = Channel is differential 0 = Channel is single-ended
bit 2 and bit 0 (even)	**SIGN<1:0>:** Output Data Sign for Corresponding Analog Inputs bits 1 = Channel output data is signed 0 = Channel output data is unsigned

REGISTER 6-2: CLKDIV: CLOCK DIVIDER REGISTER (MASTER)

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
ROI	DOZE2(1)	DOZE1(1)	DOZE0(1)	DOZEN(2,3)	FRCDIV2	FRCDIV1	FRCDIV0
bit 15							bit 8

U-0	U-0	r-0	r-0	R/W-0	R/W-0	R/W-0	R/W-1
—	—	—	—	PLLPRE3(4)	PLLPRE2(4)	PLLPRE1(4)	PLLPRE0(4)
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 **ROI:** Recover on Interrupt bit
1 = Interrupts will clear the DOZEN bit and the processor clock, and the peripheral clock ratio is set to 1:1
0 = Interrupts have no effect on the DOZEN bit

bit 14-12 **DOZE<2:0>:** Processor Clock Reduction Select bits(1)
111 = FP divided by 128
110 = FP divided by 64
101 = FP divided by 32
100 = FP divided by 16
011 = FP divided by 8 (default)
010 = FP divided by 4
001 = FP divided by 2
000 = FP divided by 1

bit 11 **DOZEN:** Doze Mode Enable bit(2,3)
1 = DOZE<2:0> field specifies the ratio between the peripheral clocks and the processor clocks
0 = Processor clock and peripheral clock ratio is forced to 1:1

bit 10-8 **FRCDIV<2:0>:** Internal Fast RC Oscillator Postscaler bits
111 = FRC divided by 256
110 = FRC divided by 64
101 = FRC divided by 32
100 = FRC divided by 16
011 = FRC divided by 8
010 = FRC divided by 4
001 = FRC divided by 2
000 = FRC divided by 1 (default)

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

bit 5-4 **Reserved:** Read as ‘0’

Note 1: The DOZE<2:0> bits can only be written to when the DOZEN bit is clear. If DOZEN = 1, any writes to DOZE<2:0> are ignored.
2: This bit is cleared when the ROI bit is set and an interrupt occurs.
3: The DOZEN bit cannot be set if DOZE<2:0> = 000. If DOZE<2:0> = 000, any attempt by user software to set the DOZEN bit is ignored.
4: PLLPRE<3:0> may be updated while the PLL is operating, but the VCO may overshoot.

REGISTER 8-3: DMAINTn: DMA CHANNEL n INTERRUPT REGISTER

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
DBUFWF[1]	CHSEL6	CHSEL5	CHSEL4	CHSEL3	CHSEL2	CHSEL1	CHSEL0
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
HIGHIF[1,2]	LOWIF[1,2]	DONEIF[1]	HALFIF[1]	OVRUNIF[1]	—	—	HALFEN
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 **DBUFWF:** DMA Buffered Data Write Flag bit[1]
1 = The content of the DMA buffer has not been written to the location specified in DMADSTn or DMASRCn in Null Write mode
0 = The content of the DMA buffer has been written to the location specified in DMADSTn or DMASRCn in Null Write mode

bit 14-8 **CHSEL<6:0>:** DMA Channel Trigger Selection bits
See Table 8-2 for a complete list.

bit 7 **HIGHIF:** DMA High Address Limit Interrupt Flag bit[1,2]
1 = The DMA channel has attempted to access an address higher than DMAH or the upper limit of the data RAM space
0 = The DMA channel has not invoked the high address limit interrupt

bit 6 **LOWIF:** DMA Low Address Limit Interrupt Flag bit[1,2]
1 = The DMA channel has attempted to access the DMA SFR address lower than DMAL, but above the SFR range (07FFh)
0 = The DMA channel has not invoked the low address limit interrupt

bit 5 **DONEIF:** DMA Complete Operation Interrupt Flag bit[1]
If CHEN = 1:
1 = The previous DMA session has ended with completion
0 = The current DMA session has not yet completed
If CHEN = 0:
1 = The previous DMA session has ended with completion
0 = The previous DMA session has ended without completion

bit 4 **HALFIF:** DMA 50% Watermark Level Interrupt Flag bit[1]
1 = DMACNTn has reached the halfway point to 0000h
0 = DMACNTn has not reached the halfway point

bit 3 **OVRUNIF:** DMA Channel Overrun Flag bit[1]
1 = The DMA channel is triggered while it is still completing the operation based on the previous trigger
0 = The overrun condition has not occurred

bit 2-1 **Unimplemented:** Read as ‘0’

bit 0 **HALFEN:** Halfway Completion Watermark bit
1 = Interrupts are invoked when DMACNTn has reached its halfway point and at completion
0 = An interrupt is invoked only at the completion of the transfer

Note 1: Setting these flags in software does not generate an interrupt.
2: Testing for address limit violations (DMASRCn or DMADSTn is either greater than DMAH or less than DMAL) is NOT done before the actual access.

REGISTER 12-19: INDXxHLDL: INDEX x COUNTER HOLD REGISTER LOW

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INDXHLD<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
INDXHLD<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 **INDXHLD<15:0>:** Hold for Reading/Writing Index x Counter Register (INDXCNT) bits

REGISTER 12-20: INDXxHLDH: INDEX x COUNTER HOLD REGISTER HIGH

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INDXHLD<31:24>							
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
INDXHLD<23:16>							
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 **INDXHLD<31:16>:** Hold for Reading/Writing Index x Counter Register (INDXCNT) bits

15.0 INTER-INTEGRATED CIRCUIT (I^2C)

Note 1: This data sheet summarizes the features of the dsPIC33CH128MP508 family of devices. It is not intended to be a comprehensive reference source. For more information, refer to **"Inter-Integrated Circuit (I^2C)"** (DS70000195) in the *"dsPIC33/PIC24 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com).

2: The I^2C is identical for both Master core and Slave core. The x is common for both Master and Slave (where the x represents the number of the specific module being addressed). The number of I^2C modules available on the Master and Slave is different and they are located in different SFR locations.

3: All associated register names are the same on the Master core and the Slave core. The Slave code will be developed in a separate project in MPLAB® X IDE with the device selection, dsPIC33CH128MP508**S1**, where the **S1** indicates the Slave device. The Master I^2C is I2C1 and I2C2, and the Slave is I2C1.

The Inter-Integrated Circuit (I^2C) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, display drivers, A/D Converters, etc.

The I^2C module supports these features:

- Independent Master and Slave Logic
- 7-Bit and 10-Bit Device Addresses
- General Call Address as Defined in the I^2C Protocol
- Clock Stretching to Provide Delays for the Processor to Respond to a Slave Data Request
- Both 100 kHz and 400 kHz Bus Specifications
- Configurable Address Masking
- Multi-Master modes to Prevent Loss of Messages in Arbitration
- Bus Repeater mode, Allowing the Acceptance of All Messages as a Slave, regardless of the Address
- Automatic SCL

A block diagram of the module is shown in Figure 15-1.

15.1 Communicating as a Master in a Single Master Environment

The details of sending a message in Master mode depends on the communication protocol for the device being communicated with. Typically, the sequence of events is as follows:

1. Assert a Start condition on SDAx and SCLx.
2. Send the I^2C device address byte to the Slave with a write indication.
3. Wait for and verify an Acknowledge from the Slave.
4. Send the first data byte (sometimes known as the command) to the Slave.
5. Wait for and verify an Acknowledge from the Slave.
6. Send the serial memory address low byte to the Slave.
7. Repeat Steps 4 and 5 until all data bytes are sent.
8. Assert a Repeated Start condition on SDAx and SCLx.
9. Send the device address byte to the Slave with a read indication.
10. Wait for and verify an Acknowledge from the Slave.
11. Enable Master reception to receive serial memory data.
12. Generate an ACK or NACK condition at the end of a received byte of data.
13. Generate a Stop condition on SDAx and SCLx.

REGISTER 17-1: T1CON: TIMER1 CONTROL REGISTER (CONTINUED)

bit 5-4	**TCKPS<1:0>:** Timer1 Input Clock Prescale Select bits 11 = 1:256 10 = 1:64 01 = 1:8 00 = 1:1
bit 3	**Unimplemented:** Read as '0'
bit 2	**TSYNC:** Timer1 External Clock Input Synchronization Select bit(1) When TCS = 1: 1 = Synchronizes the External Clock input 0 = Does not synchronize the External Clock input When TCS = 0: This bit is ignored.
bit 1	**TCS:** Timer1 Clock Source Select bit(1) 1 = External Clock source selected by TECS<1:0> 0 = Internal peripheral clock (FP)
bit 0	**Unimplemented:** Read as '0'

Note 1: When Timer1 is enabled in External Synchronous Counter mode (TCS = 1, TSYNC = 1, TON = 1), any attempts by user software to write to the TMR1 register are ignored.

REGISTER 21-28: FS1POR CONFIGURATION REGISTER (SLAVE)

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
—	—	—	—	—	—	—	—
bit 23							bit 16

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
—	—	—	—	—	—	—	—
bit 15							bit 8

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1
—	—	—	—	—	—	—	—
bit 7							bit 0

Legend:	PO = Program Once 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 23-0 **Unimplemented:** Read as ‘1’

D