

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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

E·XFl

Product Status	Obsolete
Core Processor	dsPIC
Core Size	16-Bit
Speed	40 MIPs
Connectivity	I ² C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, WDT
Number of I/O	53
Program Memory Size	128KB (43K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 36x10/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 150°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-VQFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33ev128gm006-h-mr

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong





TANK CAPACITORS 2.2.1

On boards with power traces running longer than six inches in length, it is suggested to use a tank capacitor for integrated circuits including DSCs to supply a local power source. The value of the tank capacitor should be determined based on the trace resistance that connects the power supply source to the device, and the maximum current drawn by the device in the application. In other words, select the tank capacitor so that it meets the acceptable voltage sag at the device. Typical values range from 4.7 µF to 47 µF.

2.3 **CPU Logic Filter Capacitor** Connection (VCAP)

A low-ESR (<1 Ohms) capacitor is required on the VCAP pin, which is used to stabilize the internal voltage regulator output. The VCAP pin must not be connected to VDD, and must have a capacitor greater than 4.7 µF (10 µF is recommended), with at least a 16V rating connected to the ground. The type can be ceramic or tantalum. See Section 30.0 "Electrical Characteristics" for additional information.

The placement of this capacitor should be close to the VCAP pin. It is recommended that the trace length should not exceed one-quarter inch (6 mm).

2.4 Master Clear (MCLR) Pin

The MCLR pin provides two specific device functions:

- · Device Reset
- Device Programming and Debugging

During device programming and debugging, the resistance and capacitance that can be added to the pin must be considered. Device programmers and debuggers drive the MCLR pin. Consequently, specific voltage levels (VIH and VIL) and fast signal transitions must not be adversely affected. Therefore, specific values of R and C will need to be adjusted based on the application and PCB requirements.

For example, as shown in Figure 2-1, it is recommended that the capacitor, C, be isolated from the MCLR pin during programming and debugging operations.

Place the components as shown in Figure 2-2 within one-quarter inch (6 mm) from the MCLR pin.



EXAMPLE OF MCLR PIN CONNECTIONS



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

bit 7-5	IPL<2:0>: CPU Interrupt Priority Level Status bits ^(1,2)
	<pre>111 = CPU Interrupt Priority Level is 7 (15); user interrupts are disabled 110 = CPU Interrupt Priority Level is 6 (14) 101 = CPU Interrupt Priority Level is 5 (13) 100 = CPU Interrupt Priority Level is 4 (12) 011 = CPU Interrupt Priority Level is 3 (11) 010 = CPU Interrupt Priority Level is 2 (10) 001 = CPU Interrupt Priority Level is 1 (9) 000 = CPU Interrupt Priority Level is 0 (8)</pre>
bit 4	RA: REPEAT Loop Active bit
	1 = REPEAT loop is in progress 0 = REPEAT loop is not in progress
bit 3	N: MCU ALU Negative bit
	1 = Result was negative0 = Result was non-negative (zero or positive)
bit 2	OV: MCU ALU Overflow bit
	This bit is used for signed arithmetic (2's complement). It indicates an overflow of the magnitude that causes the sign bit to change state. 1 = Overflow occurred for signed arithmetic (in this arithmetic operation) 0 = Overflow has not occurred for signed arithmetic
bit 1	Z: MCU ALU Zero bit
	 1 = An operation that affects the Z bit has set it at some time in the past 0 = The most recent operation that affects the Z bit has cleared it (i.e., a non-zero result)
bit 0	C: MCU ALU Carry/Borrow bit
	 1 = A carry-out from the Most Significant bit (MSb) of the result occurred 0 = No carry-out from the Most Significant bit of the result occurred
Note 1.	The IPI <2:0> hits are concatenated with the IPI 3 hit (CORCON<3>) to form the CPI I Interrupt Priority

- **Note 1:** The IPL<2:0> bits are concatenated with the IPL3 bit (CORCON<3>) to form the CPU Interrupt Priority Level. The value in parentheses indicates the IPL if IPL3 = 1. User interrupts are disabled when IPL3 = 1.
 - 2: The IPL<2:0> Status bits are read-only when the NSTDIS bit (INTCON1<15>) = 1.
 - **3:** A data write to the SR register can modify the SA and SB bits by either a data write to SA and SB or by clearing the SAB bit. To avoid a possible SA or SB bit write race condition, the SA and SB bits should not be modified using the bit operations.

3.7 Arithmetic Logic Unit (ALU)

The dsPIC33EVXXXGM00X/10X family ALU is 16 bits wide and is capable of addition, subtraction, bit shifts and logic operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. Depending on the operation, the ALU can affect the values of the Carry (C), Zero (Z), Negative (N), Overflow (OV) and Digit Carry (DC) Status bits in the SR register. The C and DC Status bits operate as Borrow and Digit Borrow bits, respectively, for subtraction operations.

The ALU can perform 8-bit or 16-bit operations, depending on the mode of the instruction that is used. The data for the ALU operation can come from the W register array or from the data memory, depending on the addressing mode of the instruction. Similarly, the output data from the ALU can be written to the W register array or a data memory location.

For information on the SR bits affected by each instruction, refer to the *"16-bit MCU and DSC Programmer's Reference Manual"* (DS70157).

The core CPU incorporates hardware support for both multiplication and division. This includes a dedicated hardware multiplier and support hardware for 16-bit divisor division.

3.7.1 MULTIPLIER

Using the high-speed, 17-bit x 17-bit multiplier, the ALU supports unsigned, signed or mixed-sign operation in several MCU multiplication modes:

- 16-bit x 16-bit signed
- 16-bit x 16-bit unsigned
- 16-bit signed x 5-bit (literal) unsigned
- 16-bit signed x 16-bit unsigned
- 16-bit unsigned x 5-bit (literal) unsigned
- 16-bit unsigned x 16-bit signed
- 8-bit unsigned x 8-bit unsigned

3.7.2 DIVIDER

The divide block supports 32-bit/16-bit and 16-bit/16-bit signed and unsigned integer divide operations with the following data sizes:

- 32-bit signed/16-bit signed divide
- 32-bit unsigned/16-bit unsigned divide
- 16-bit signed/16-bit signed divide
- 16-bit unsigned/16-bit unsigned divide

The quotient for all divide instructions ends up in W0 and the remainder in W1. The 16-bit signed and unsigned DIV instructions can specify any W register for both the 16-bit divisor (Wn) and any W register (aligned) pair (W(m + 1):Wm) for the 32-bit dividend. The divide algorithm takes the single-cycle per bit of the divisor, so both 32-bit/16-bit and 16-bit/16-bit instructions take the same number of cycles to execute.

3.8 DSP Engine

The DSP engine consists of a high-speed, 17-bit x 17-bit multiplier, a 40-bit barrel shifter and a 40-bit adder/ subtracter (with two target accumulators, round and saturation logic).

The DSP engine can also perform inherent accumulatorto-accumulator operations that require no additional data. These instructions are ADD, SUB and NEG.

The DSP engine has options selected through bits in the CPU Core Control register (CORCON) as follows:

- Fractional or Integer DSP Multiply (IF)
- Signed, Unsigned or Mixed-Sign DSP Multiply (US)
- Conventional or Convergent Rounding (RND)
- · Automatic Saturation On/Off for ACCA (SATA)
- Automatic Saturation On/Off for ACCB (SATB)
- Automatic Saturation On/Off for Writes to Data Memory (SATDW)
- Accumulator Saturation mode Selection (ACCSAT)

TABLE 3-2:DSP INSTRUCTIONSSUMMARY

Instruction	Algebraic Operation	ACC Write Back
CLR	A = 0	Yes
ED	$A = (x - y)^2$	No
EDAC	$A = A + (x - y)^2$	No
MAC	$A = A + (x \bullet y)$	Yes
MAC	$A = A + x^2$	No
MOVSAC	No change in A	Yes
MPY	$A = x \bullet y$	No
MPY	$A = x^2$	No
MPY.N	$A = -x \bullet y$	No
MSC	$A = A - x \bullet y$	Yes

TABLE 4-26: DMAC REGISTER MAP (CONTINUED)

SFR Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
DMALCA	0BF6	I	_	—	_	—	—	—	—	—	—	—	—		LSTC	H<3:0>		000F
DSADRL	0BF8									DSADF	R<15:0>							0000
DSADRH	0BFA	_	_	_	_	_	_		-				DSADF	<23:16>				0000

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

TABLE 4-27: PWM REGISTER MAP

SFR Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PTCON	0C00	PTEN	_	PTSIDL	SESTAT	SEIEN	EIPU	SYNCPOL	SYNCOEN	SYNCEN	SYNCSRC2	SYNCSRC1	SYNCSRC0	SEVTPS3	SEVTPS2	SEVTPS1	SEVTPS0	0000
PTCON2	0C02	_		—	—	-	—	_	_	_	_	_	_	_	F	PCLKDIV<2:0	>	0000
PTPER	0C04									PTPER•	<15:0>							FFF8
SEVTCMP	0C06									SEVTCM	P<15:0>							0000
MDC	0C0A									MDC<	15:0>							0000
CHOP	0C1A	CHPCLKEN		—	—	-	—	CHOPCLK9	CHOPCLK8	CHOPCLK7	CHOPCLK6	CHOPCLK5	CHOPCLK4	CHOPCLK3	CHOPCLK2	CHOPCLK1	CHOPCLK0	0000
PWMKEY	0C1E									PWMKEY	/<15:0>							0000

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

TABLE 4-28: PWM GENERATOR 1 REGISTER MAP

SFR Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PWMCON1	0C20	FLTSTAT	CLSTAT	TRGSTAT	FLTIEN	CLIEN	TRGIEN	ITB	MDCS	DTC1	DTC0	DTCP	_	—	CAM	XPRES	IUE	0000
IOCON1	0C22	PENH	PENL	POLH	POLL	PMOD1	PMOD0	OVRENH	OVRENL	OVRDAT1	OVRDAT0	FLTDAT1	FLTDAT0	CLDAT1	CLDAT0	SWAP	OSYNC	0000
FCLCON1	0C24	_	CLSRC4	CLSRC3	CLSRC2	CLSRC1	CLSRC0	CLPOL	CLMOD	FLTSRC4	FLTSRC3	FLTSRC2	FLTSRC1	FLTSRC0	FLTPOL	FLTMOD1	FLTMOD0	0000
PDC1	0C26								PDC	1<15:0>								0000
PHASE1	0C28								PHAS	E1<15:0>								0000
DTR1	0C2A	_	_							DTR1	<13:0>							0000
ALTDTR1	0C2C	_	_							ALTDTI	R1<13:0>							0000
TRIG1	0C32								TRGC	MP<15:0>								0000
TRGCON1	0C34	TRGDIV3	TRGDIV2	TRGDIV1	TRGDIV0	_	_	_	-	_	_	TRGSTRT5	TRGSTRT4	TRGSTRT3	TRGSTRT2	TRGSTRT1	TRGSTRT0	0000
PWMCAP1	0C38								PWMC	AP1<15:0>								0000
LEBCON1	0C3A	PHR	PHF	PLR	LR PLF FLTLEBEN CLLEBEN — — — — BCH BCL BPHH BPHL BPLH BPLL 0000							0000						
LEBDLY1	0C3C	_	_		LEB<11:0> 0000								0000					
AUXCON1	0C3E	_	_		—	BLANKSEL3	BLANKSEL2	BLANKSEL1	BLANKSEL0	—	_	CHOPSEL3	CHOPSEL2	CHOPSEL1	CHOPSEL0	CHOPHEN	CHOPLEN	0000

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

4.3.4 SOFTWARE STACK

The W15 register serves as a dedicated Software Stack Pointer (SSP) and is automatically modified by exception processing, subroutine calls and returns; however, W15 can be referenced by any instruction in the same manner as all other W registers. This simplifies reading, writing and manipulating the SSP (for example, creating stack frames).

Note:	To protect against misaligned	stack
	accesses, W15<0> is fixed to '0' b	y the
	hardware.	

W15 is initialized to 0x1000 during all Resets. This address ensures that the SSP points to valid RAM in all dsPIC33EVXXXGM00X/10X family devices and permits stack availability for non-maskable trap exceptions. These can occur before the SSP is initialized by the user software. You can reprogram the SSP during initialization to any location within the Data Space.

The SSP always points to the first available free word and fills the software stack, working from lower toward higher addresses. Figure 4-14 illustrates how it predecrements for a stack pop (read) and post-increments for a stack push (writes).

When the PC is pushed onto the stack, PC<15:0> are pushed onto the first available stack word, then PC<22:16> are pushed into the second available stack location. For a PC push during any CALL instruction, the MSB of the PC is zero-extended before the push, as shown in Figure 4-14. During exception processing, the MSB of the PC is concatenated with the lower 8 bits of the CPU STATUS Register (SR). This allows the contents of SRL to be preserved automatically during interrupt processing.

- Note 1: To maintain system SSP (W15) coherency, W15 is never subject to (EDS) paging, and is therefore, restricted to an address range of 0x0000 to 0xFFFF. The same applies to the W14 when used as a Stack Frame Pointer (SFA = 1).
 - 2: As the stack can be placed in, and can access X and Y spaces, care must be taken regarding its use, particularly with regard to local automatic variables in a 'C' development environment.

FIGURE 4-14:

CALL STACK FRAME



4.4 Instruction Addressing Modes

The addressing modes shown in Table 4-45 form the basis of the addressing modes optimized to support the specific features of the individual instructions. The addressing modes provided in the MAC class of instructions differ from those in the other instruction types.

4.4.1 FILE REGISTER INSTRUCTIONS

Most file register instructions use a 13-bit address field (f) to directly address data present in the first 8192 bytes of data memory (Near Data Space). Most file register instructions employ a Working register, W0, which is denoted as WREG in these instructions. The destination is typically either the same file register or WREG (with the exception of the MUL instruction), which writes the result to a register or register pair. The MOV instruction allows additional flexibility and can access the entire Data Space.

4.4.2 MCU INSTRUCTIONS

The three-operand MCU instructions are of the form:

Operand 3 = Operand 1 <function> Operand 2

where, Operand 1 is always a Working register (that is, the addressing mode can only be Register Direct), which is referred to as Wb. Operand 2 can be a W register fetched from data memory or a 5-bit literal. The result location can be either a W register or a data memory location. The following addressing modes are supported by MCU instructions:

- Register Direct
- Register Indirect
- · Register Indirect Post-Modified
- Register Indirect Pre-Modified
- 5-Bit or 10-Bit Literal
- Note: Not all instructions support all of the addressing modes given above. Individual instructions can support different subsets of these addressing modes.

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ROON	—	ROSSLP	ROSEL	RODIV3 ⁽¹⁾	RODIV2 ⁽¹⁾	RODIV1 ⁽¹⁾	RODIV0 ⁽¹⁾
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 = Unimpler	nented bit, read	as '0'	
-n = Value at F	POR	'1' = Bit is set		'0' = Bit is cle	ared	x = Bit is unkr	ıown
bit 15	ROON: Refer	ence Oscillato	r Output Enab	ole bit			
	1 = Reference	e oscillator out	out is enabled	on the REFCL	₋K pin ⁽²⁾		
	0 = Reference	e oscillator out	out is disabled	k	-		
bit 14	Unimplemen	ted: Read as '	0'				
bit 13	ROSSLP: Re	ference Oscilla	itor Run in Sle	eep bit			
	1 = Reference	e oscillator outp	out continues	to run in Sleep	mode		
	0 = Reference	e oscillator outp	out is disabled	t in Sleep mode	e		
bit 12	ROSEL: Refe	erence Oscillato	or Source Sel	ect bit			
	1 = Oscillator	crystal is used	as the refere	nce clock			
	0 = System c	lock is used as	the reference	e clock			
bit 11-8	RODIV<3:0>:	Reference Os	cillator Divide	er bits ⁽¹⁾			
	1111 = Refer	ence clock divi	ded by 32,76	8			
	1110 = Refer	ence clock divi	ded by 16,38	4			
	1100 = Refer	ence clock divi	ded by 4.096				
	1011 = Refer	ence clock divi	ded by 2,048				
	1010 = Refe r	ence clock divi	ded by 1,024				
	1001 = Refer	ence clock divi	ded by 512				
	1000 = Refer	ence clock divi	ded by 256				
	0110 = Refer	ence clock divi	ded by 64				
	0101 = Refe r	ence clock divi	ded by 32				
	0100 = Refer	ence clock divi	ded by 16				
	0011 = Refer	ence clock divi	ded by 8				
	0010 = Refer	ence clock divi	ded by 4				
	0000 = Refer	ence clock					
bit 7-0	Unimplemen	ted: Read as '	0'				
	-						

REGISTER 9-5: REFOCON: REFERENCE OSCILLATOR CONTROL REGISTER

- **Note 1:** The reference oscillator output must be disabled (ROON = 0) before writing to these bits.
 - 2: This pin is remappable. See Section 11.5 "Peripheral Pin Select (PPS)" for more information.

NOTES:

For example, Figure 11-2 shows the remappable pin selection for the U1RX input.

FIGURE 11-2: REMAPPABLE INPUT FOR U1RX



11.5.4.1 Virtual Connections

dsPIC33EVXXXGM00X/10X family devices support virtual (internal) connections to the output of the op amp/comparator module (see Figure 25-1 in Section 25.0 "Op Amp/Comparator Module").

These devices provide six virtual output pins (RPV0-RPV5) that correspond to the outputs of six peripheral pin output remapper blocks (RP176-RP181). The six virtual remapper outputs (RP176-RP181) are not connected to actual pins. The six virtual pins may be read by any of the input remappers as inputs, RP176-RP181. These virtual pins can be used to connect the internal peripherals, whose signals are of significant use to the other peripherals, but these output signals are not present on the device pin.

Virtual connections provide a simple way of interperipheral connection without utilizing a physical pin. For example, by setting the FLT1R<7:0> bits of the RPINR12 register to the value of 'b0000001', the output of the analog comparator, C1OUT, will be connected to the PWM Fault 1 input, which allows the analog comparator to trigger PWM Faults without the use of an actual physical pin on the device.

Input Name ⁽¹⁾	Function Name	Register	Configuration Bits
External Interrupt 1	INT1	RPINR0	INT1R<7:0>
External Interrupt 2	INT2	RPINR1	INT2R<7:0>
Timer2 External Clock	T2CK	RPINR3	T2CKR<7:0>
Input Capture 1	IC1	RPINR7	IC1R<7:0>
Input Capture 2	IC2	RPINR7	IC2R<7:0>
Input Capture 3	IC3	RPINR8	IC3R<7:0>
Input Capture 4	IC4	RPINR8	IC4R<7:0>
Output Compare Fault A	OCFA	RPINR11	OCFAR<7:0>
PWM Fault 1	FLT1	RPINR12	FLT1R<7:0>
PWM Fault 2	FLT2	RPINR12	FLT2R<7:0>
UART1 Receive	U1RX	RPINR18	U1RXR<7:0>
UART2 Receive	U2RX	RPINR19	U2RXR<7:0>
SPI2 Data Input	SDI2	RPINR22	SDI2R<7:0>
SPI2 Clock Input	SCK2	RPINR22	SCK2R<7:0>
SPI2 Slave Select	SS2	RPINR23	SS2R<7:0>
CAN1 Receive	C1RX	RPINR26	C1RXR<7:0>
PWM Sync Input 1	SYNCI1	RPINR37	SYNCI1R<7:0>
PWM Dead-Time Compensation 1	DTCMP1	RPINR38	DTCMP1R<7:0>
PWM Dead-Time Compensation 2	DTCMP2	RPINR39	DTCMP2R<7:0>
PWM Dead-Time Compensation 3	DTCMP3	RPINR39	DTCMP3R<7:0>
SENT1 Input	SENT1R	RPINR44	SENT1R<7:0>
SENT2 Input	SENT2R	RPINR45	SENT2R<7:0>

Note 1: Unless otherwise noted, all inputs use the Schmitt Trigger input buffers.

REGISTER 16-1: OCxCON1: OUTPUT COMPARE x CONTROL REGISTER 1 (CONTINUED)

- bit 2-0 OCM<2:0>: Output Compare x Mode Select bits
 - 111 = Center-Aligned PWM mode: Output sets high when OCxTMR = OCxR and sets low when OCxTMR = OCxRS⁽¹⁾
 - 110 = Edge-Aligned PWM mode: Output sets high when OCxTMR = 0 and sets low when OCxTMR = $OCxR^{(1)}$
 - 101 = Double Compare Continuous Pulse mode: Initializes OCx pin low, toggles OCx state continuously on alternate matches of OCxR and OCxRS
 - 100 = Double Compare Single-Shot mode: Initializes OCx pin low, toggles OCx state on matches of OCxR and OCxRS for one cycle
 - 011 = Single Compare mode: Compare event with OCxR, continuously toggles OCx pin
 - 010 = Single Compare Single-Shot mode: Initializes OCx pin high, compare event with OCxR, forces OCx pin low
 - 001 = Single Compare Single-Shot mode: Initializes OCx pin low, compare event with OCxR, forces OCx pin high
 - 000 = Output compare channel is disabled
- Note 1: OCxR and OCxRS are double-buffered in PWM mode only.

dsPIC33EVXXXGM00X/10X FAMILY



U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	_	—	—	—	—	—
bit 15							bit 8
R-0	R-0	R-0	R-0	R-0	R/C-0	R-0	R/W-0, HC
PAUSE	NIB2	NIB1	NIB0	CRCERR	FRMERR	RXIDLE	SYNCTXEN ⁽¹⁾
bit 7							bit 0
Legend:		C = Clearable	bit	HC = Hardwa	are Clearable b	oit	
R = Readable	bit	W = Writable	bit	U = Unimplei	mented bit, rea	d as '0'	
-n = Value at F	POR	'1' = Bit is set		'0' = Bit is cle	eared	x = Bit is unkr	Iown
bit 15-8	Unimplemen	ted: Read as '	0'				
bit 7	PAUSE: Paus	se Period Statu	is bit				
	1 = The modu	ule is transmitti	ng/receiving a	pause period			
	0 = The modu	ule is not transr	mitting/receivir	ng a pause per	riod		
bit 6-4	NIB<2:0>: Ni	bble Status bit					
	Module in Tra	nsmit Mode (R	<u> CVEN = 0):</u>				
	111 = Module	e is transmitting	a CRC nibble	9			
	101 = Module	e is transmitting	Data Nibble	5			
	100 = Module	e is transmitting	Data Nibble	4			
	011 = Module	e is transmitting	g Data Nibble 3	3			
	010 = Module	e is transmitting	g Data Nibble 2	2			
	001 = Module	e is transmitting	j Dala Nibble	I le or nause ne	eriod or is not t	ransmitting	
	Module in Re	ceive Mode (R	CVEN = 1)			ransmitting	
	111 = Module	e is receiving a	CRC nibble of	r was receiving	g this nibble wh	nen an error oc	curred
	110 = Module	e is receiving D	ata Nibble 6 o	r was receivin	g this nibble wl	nen an error oo	curred
	101 = Module	e is receiving D	ata Nibble 5 o	r was receivin	g this nibble wi	nen an error oc	curred
	100 = Module	e is receiving D	ata Nibble 4 0 ata Nibble 3 o	r was receiving	g this nibble wi	nen an error oc	curred
	010 = Module	e is receiving D	ata Nibble 2 o	r was receiving	g this nibble wi	hen an error og	curred
	001 = Module	e is receiving D	ata Nibble 1 o	r was receivin	g this nibble wi	nen an error oo	curred
	000 = Module	e is receiving a	status nibble	or waiting for S	Sync		
bit 3	CRCERR: CF	RC Status bit (F	Receive mode	only)			
	1 = A CRC er	ror occurred fo	r the 1-6 data	nibbles in SEN	NTxDATH/L		
	0 = A CRC er	ror has not occ	curred				
bit 2	FRMERR: Fra	aming Error Sta	atus bit (Recei	ve mode only))		
	\perp = A data nit	ble was receiv	ed with less th	han 12 tick per	loas or greater	than 27 tick pe	eriods
bit 1		Ty Possiver L	dla Statua hit (Docoivo modo			
	1 - The SEN	n i X rtetteliveli I(Tv data bus ba	ne Status Dil (s boon Idio /bi	neceive mode		X<15.05 or are	ator
	1 = The SEN	Tx data bus ha	not Idle	ign) ior a pello		~~10.0~ 01 gre	alei
							

REGISTER 20-2: SENTxSTAT: SENTx STATUS REGISTER

Note 1: In Receive mode (RCVEN = 1), the SYNCTXEN bit is read-only.

22.0 CONTROLLER AREA NETWORK (CAN) MODULE (dsPIC33EVXXXGM10X DEVICES ONLY)

- Note 1: This data sheet summarizes the features of the dsPIC33EVXXXGM00X/10X family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to "Enhanced Controller Area Network (ECAN™)" (DS70353) in the "dsPIC33/PIC24 Family Reference Manual", which is available from the Microchip web site (www.microchip.com).
 - Some registers and associated bits described in this section may not be available on all devices. Refer to Section 4.0 "Memory Organization" in this data sheet for device-specific register and bit information.

22.1 Overview

The Controller Area Network (CAN) module is a serial interface, useful for communicating with other CAN modules or microcontroller devices. This interface/ protocol was designed to allow communications within noisy environments. The dsPIC33EVXXXGM10X devices contain one CAN module.

The CAN module is a communication controller implementing the CAN 2.0 A/B protocol, as defined in the BOSCH CAN specification. The module supports CAN 1.2, CAN 2.0A, CAN 2.0B Passive and CAN 2.0B Active versions of the protocol. The module implementation is a full CAN system. The CAN specification is not covered within this data sheet. The reader can refer to the BOSCH CAN specification for further details. The CAN module features are as follows:

- Implementation of the CAN Protocol, CAN 1.2, CAN 2.0A and CAN 2.0B
- · Standard and Extended Data Frames
- 0 to 8-Byte Data Length
- Programmable Bit Rate, up to 1 Mbit/sec
- Automatic Response to Remote Transmission Requests
- Up to Eight Transmit Buffers with Application Specified Prioritization and Abort Capability (each buffer can contain up to 8 bytes of data)
- Up to 32 Receive Buffers (each buffer can contain up to 8 bytes of data)
- Up to 16 Full (Standard/Extended Identifier) Acceptance Filters
- Three Full Acceptance Filter Masks
- DeviceNet[™] Addressing Support
- Programmable Wake-up Functionality with Integrated Low-Pass Filter
- Programmable Loopback Mode Supports Self-Test Operation
- Signaling through Interrupt Capabilities for All CAN Receiver and Transmitter Error States
- · Programmable Clock Source
- Programmable Link to Input Capture 2 (IC2) module for Timestamping and Network Synchronization
- · Low-Power Sleep and Idle Modes

The CAN bus module consists of a protocol engine and message buffering/control. The CAN protocol engine handles all functions for receiving and transmitting messages on the CAN bus. Messages are transmitted by first loading the appropriate data registers. Status and errors can be checked by reading the appropriate registers. Any message detected on the CAN bus is checked for errors, and then matched against filters to see if it should be received and stored in one of the Receive registers.

Figure 22-1 shows a block diagram of the CANx module.

dsPIC33EVXXXGM00X/10X FAMILY

FIGURE 22-1: CANX MODULE BLOCK DIAGRAM



22.2 Modes of Operation

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

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

Modes are requested by setting the REQOP<2:0> bits (CxCTRL1<10:8>). Entry into a mode is Acknowledged by monitoring the OPMODE<2:0> bits (CxCTRL1<7:5>). The module does not change the mode and the OPMODEx bits until a change in mode is acceptable, generally during bus Idle time, which is defined as at least 11 consecutive recessive bits.

22.3 CAN Control Registers

REGISTER 22-1: CxCTRL1: CANx CONTROL REGISTER 1

		B # 1 / 2				54446	
U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0
—	—	CSIDL	ABAT	CANCKS	REQOP2	REQOP1	REQOP0
bit 15							bit 8
R-1	R-0	R-0	U-0	R/W-0	U-0	U-0	R/W-0
OPMODE2	OPMODE1	OPMODE0	_	CANCAP	_	_	WIN
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	CSIDL: CANx Stop in Idle Mode bit
	1 = Discontinues module operation when the device enters Idle mode
	0 = Continues module operation in Idle mode
bit 12	ABAT: Abort All Pending Transmissions bit
	1 = Signals all transmit buffers to abort transmission
	0 = Module will clear this bit when all transmissions are aborted
bit 11	CANCKS: CANx Module Clock (FCAN) Source Select bit
	1 = FCAN is equal to 2 * FP
	0 = FCAN is equal to FP
bit 10-8	REQOP<2:0>: Request Operation Mode bits
	111 = Sets Listen All Messages mode
	110 = Reserved
	101 = Reserved 100 = Sets Configuration mode
	011 = Sets Listen Only mode
	010 = Sets Loopback mode
	001 = Sets Disable mode
	000 = Sets Normal Operation mode
bit 7-5	OPMODE<2:0>: Operation Mode bits
	111 = Module is in Listen All Messages mode
	110 = Reserved
	101 - Reserved 100 = Module is in Configuration mode
	011 = Module is in Listen Only mode
	010 = Module is in Loopback mode
	001 = Module is in Disable mode
	000 = Module is in Normal Operation mode
bit 4	Unimplemented: Read as '0'
bit 3	CANCAP: CANx Message Receive Timer Capture Event Enable bit
	1 = Enables input capture based on CAN message receive
	0 = Disables CAN capture
bit 2-1	Unimplemented: Read as '0'
bit 0	WIN: SFR Map Window Select bit
	1 = Uses filter window
	0 = Uses buffer window

Most instructions are a single word. Certain double-word instructions are designed to provide all the required information in these 48 bits. In the second word, the 8 MSbs are '0's. If this second word is executed as an instruction (by itself), it executes as a NOP.

The double-word instructions execute in two instruction cycles.

Most single-word instructions are executed in a single instruction cycle, unless a conditional test is true, or the Program Counter is changed as a result of the instruction, or a PSV or Table Read is performed. In these cases, the execution takes multiple instruction cycles with the additional instruction cycle(s) executed as a NOP. Certain instructions that involve skipping over the subsequent instruction require either two or three cycles if the skip is performed, depending on whether the instruction being skipped is a single-word or two-word instruction. Moreover, double-word moves require two cycles.

Note: For more details on the instruction set, refer to the *"16-bit MCU and DSC Programmer's Reference Manual"* (DS70157).

Field	Description		
#text	Means literal defined by "text"		
(text)	Means "content of text"		
[text]	Means "the location addressed by text"		
{}	Optional field or operation		
a ∈ {b, c, d}	a is selected from the set of values b, c, d		
<n:m></n:m>	Register bit field		
.b	Byte mode selection		
.d	Double-Word mode selection		
.S	Shadow register select		
.w	Word mode selection (default)		
Acc	One of two accumulators {A, B}		
AWB	Accumulator Write-Back Destination Address register \in {W13, [W13]+ = 2}		
bit4	4-bit bit selection field (used in word-addressed instructions) $\in \{015\}$		
C, DC, N, OV, Z	MCU Status bits: Carry, Digit Carry, Negative, Overflow, Sticky Zero		
Expr	Absolute address, label or expression (resolved by the linker)		
f	File register address ∈ {0x00000x1FFF}		
lit1	1-bit unsigned literal $\in \{0,1\}$		
lit4	4-bit unsigned literal $\in \{015\}$		
lit5	5-bit unsigned literal $\in \{031\}$		
lit8	8-bit unsigned literal \in {0255}		
lit10	10-bit unsigned literal \in {0255} for Byte mode, {0:1023} for Word mode		
lit14	14-bit unsigned literal $\in \{016384\}$		
lit16	16-bit unsigned literal ∈ {065535}		
lit23	23-bit unsigned literal \in {08388608}; LSb must be '0'		
None	Field does not require an entry, can be blank		
OA, OB, SA, SB	DSP Status bits: ACCA Overflow, ACCB Overflow, ACCA Saturate, ACCB Saturate		
PC	Program Counter		
Slit10	10-bit signed literal ∈ {-512511}		
Slit16	16-bit signed literal ∈ {-3276832767}		
Slit6	6-bit signed literal \in {-1616}		
Wb	Base W register ∈ {W0W15}		
Wd	Destination W register \in { Wd, [Wd], [Wd++], [Wd], [++Wd], [Wd] }		
Wdo	Destination W register ∈ { Wnd, [Wnd], [Wnd++], [Wnd], [++Wnd], [Wnd], [Wnd+Wb] }		
Wm Wn	Dividend Divisor Working register pair (Direct Addressing)		

TABLE 28-1: SYMBOLS USED IN OPCODE DESCRIPTIONS



FIGURE 32-29: TYPICAL VIH/VIL vs. TEMPERATURE (GENERAL PURPOSE I/Os)



32.10 Voltage Output Low (VOL) – Voltage Output High (VOH)













34.2 Package Details

The following sections give the technical details of the packages.

28-Lead Skinny Plastic Dual In-Line (SP) – 300 mil Body [SPDIP]

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



Units		INCHES		
Dim	nension Limits	MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	.100 BSC		
Top to Seating Plane	A	-	-	.200
Molded Package Thickness	A2	.120	.135	.150
Base to Seating Plane	A1	.015	-	-
Shoulder to Shoulder Width	E	.290	.310	.335
Molded Package Width	E1	.240	.285	.295
Overall Length	D	1.345	1.365	1.400
Tip to Seating Plane	L	.110	.130	.150
Lead Thickness	С	.008	.010	.015
Upper Lead Width	b1	.040	.050	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	_	-	.430

Notes:

- 1. Pin 1 visual index feature may vary, but must be located within the hatched area.
- 2. § Significant Characteristic.
- 3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- 4. Dimensioning and tolerancing per ASME Y14.5M.

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

Microchip Technology Drawing C04-070B

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN or VQFN]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Optional Center Pad Width	X2			6.60
Optional Center Pad Length	Y2			6.60
Contact Pad Spacing	C1		8.00	
Contact Pad Spacing	C2		8.00	
Contact Pad Width (X44)	X1			0.35
Contact Pad Length (X44)	Y1			0.85
Contact Pad to Contact Pad (X40)	G1	0.30		
Contact Pad to Center Pad (X44)	G2	0.28		
Thermal Via Diameter	V		0.33	
Thermal Via Pitch	EV		1.20	

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing No. C04-2103C