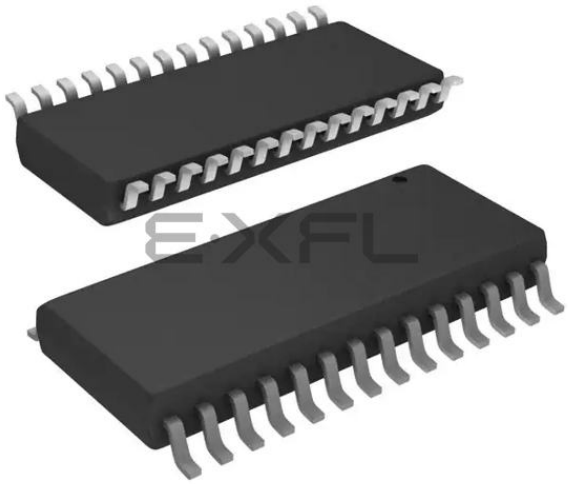
Microchip Technology - DSPIC33EV32GM102T-I/SO Datasheet



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	70 MIPs
Connectivity	CANbus, I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, WDT
Number of I/O	21
Program Memory Size	32KB (11K x 24)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 11x10/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33ev32gm102t-i-so

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' by the hardware.

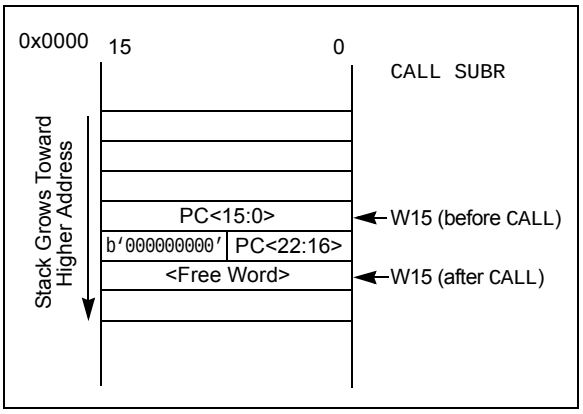
W15 is initialized to 0x1000 during all Resets. This address ensures that the SSP points to valid RAM in all dsPIC33EVXXXGM00X/10X family devices and permits stack availability for non-maskable trap exceptions. These can occur before the SSP is initialized by the user software. You can reprogram the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

In addition, DMA transfers can be triggered by timers as well as external interrupts. Each DMA channel is unidirectional. Two DMA channels must be allocated to read and write to a peripheral. If more than one channel receives a request to transfer data, a simple fixed priority scheme, based on channel number, dictates which channel completes the transfer and which channel or channels are left pending. Each DMA channel moves a block of data, after which, it generates an interrupt to the CPU to indicate that the block is available for processing.

The DMA Controller provides these functional capabilities:

- Four DMA Channels
- Register Indirect with Post-Increment Addressing mode
- Register Indirect without Post-Increment Addressing mode
- Peripheral Indirect Addressing mode (peripheral generates destination address)
- CPU Interrupt after Half or Full Block Transfer Complete
- Byte or Word Transfers
- Fixed Priority Channel Arbitration
- Manual (software) or Automatic (peripheral DMA requests) Transfer Initiation
- One-Shot or Auto-Repeat Block Transfer modes
- Ping-Pong mode (automatic switch between two SRAM start addresses after each block transfer complete)
- DMA Request for Each Channel can be Selected from any Supported Interrupt Source
- Debug Support Features

The peripherals that can utilize DMA are listed in Table 8-1.

TABLE 8-1: DMA CHANNEL TO PERIPHERAL ASSOCIATIONS

Peripheral to DMA Association	DMAxREQ Register IRQSEL<7:0> Bits	DMAxPAD Register (Values to Read from Peripheral)	DMAxPAD Register (Values to Write to Peripheral)
External Interrupt 0 (INT0)	00000000	—	—
Input Capture 1 (IC1)	00000001	0x0144 (IC1BUF)	—
Input Capture 2 (IC2)	00000101	0x014C (IC2BUF)	—
Input Capture 3 (IC3)	00100101	0x0154 (IC3BUF)	—
Input Capture 4 (IC4)	00100110	0x015C (IC4BUF)	—
Output Compare 1 (OC1)	00000010	—	0x0906 (OC1R) 0x0904 (OC1RS)
Output Compare 2 (OC2)	00000110	—	0x0910 (OC2R) 0x090E (OC2RS)
Output Compare 3 (OC3)	00011001	—	0x091A (OC3R) 0x0918 (OC3RS)
Output Compare 4 (OC4)	00011010	—	0x0924 (OC4R) 0x0922 (OC4RS)
Timer2 (TMR2)	00000111	—	—
Timer3 (TMR3)	00001000	—	—
Timer4 (TMR4)	00011011	—	—
Timer5 (TMR5)	00011100	—	—
SPI1 Transfer Done	00001010	0x0248 (SPI1BUF)	0x0248 (SPI1BUF)
SPI2 Transfer Done	00100001	0x0268 (SPI2BUF)	0x0268 (SPI2BUF)
UART1 Receiver (UART1RX)	00001011	0x0226 (U1RXREG)	—
UART1 Transmitter (UART1TX)	00001100	—	0x0224 (U1TXREG)
UART2 Receiver (UART2RX)	00011110	0x0236 (U2RXREG)	—
UART2 Transmitter (UART2TX)	00011111	—	0x0234 (U2TXREG)
RX Data Ready (CAN1)	00100010	0x0440 (C1RXD)	—
TX Data Request (CAN1)	01000110	—	0x0442 (C1TXD)
ADC1 Convert Done (ADC1)	00001101	0x0300 (ADC1BUF0)	—

REGISTER 9-5: REFOCON: REFERENCE OSCILLATOR CONTROL REGISTER

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 = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15 **ROON:** Reference Oscillator Output Enable bit
1 = Reference oscillator output is enabled on the REFCLK pin(2)
0 = Reference oscillator output is disabled

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

bit 13 **ROSSLP:** Reference Oscillator Run in Sleep bit
1 = Reference oscillator output continues to run in Sleep mode
0 = Reference oscillator output is disabled in Sleep mode

bit 12 **ROSEL:** Reference Oscillator Source Select bit
1 = Oscillator crystal is used as the reference clock
0 = System clock is used as the reference clock

bit 11-8 **RODIV<3:0>:** Reference Oscillator Divider bits(1)
1111 = Reference clock divided by 32,768
1110 = Reference clock divided by 16,384
1101 = Reference clock divided by 8,192
1100 = Reference clock divided by 4,096
1011 = Reference clock divided by 2,048
1010 = Reference clock divided by 1,024
1001 = Reference clock divided by 512
1000 = Reference clock divided by 256
0111 = Reference clock divided by 128
0110 = Reference clock divided by 64
0101 = Reference clock divided by 32
0100 = Reference clock divided by 16
0011 = Reference clock divided by 8
0010 = Reference clock divided by 4
0001 = Reference clock divided by 2
0000 = Reference clock

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

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.

10.2.1 SLEEP MODE

The following events occur 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 before entering Sleep mode.
- Some device features or peripherals can continue to operate. This includes items such as the Input Change Notification (ICN) 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 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 internal regulator and the Flash regulator can be configured to go into Standby mode when Sleep mode is entered by clearing the VREGS (RCON<8>) and VREGSF (RCON<11>) bits (default configuration).

If the application requires a faster wake-up time, and can accept higher current requirements, the VREGS (RCON<8>) and VREGSF (RCON<11>) bits can be set to keep the internal regulator and the Flash regulator active during Sleep mode.

10.2.2 IDLE MODE

The following events occur 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 10.4 “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 Interrupt Service Routine (ISR).

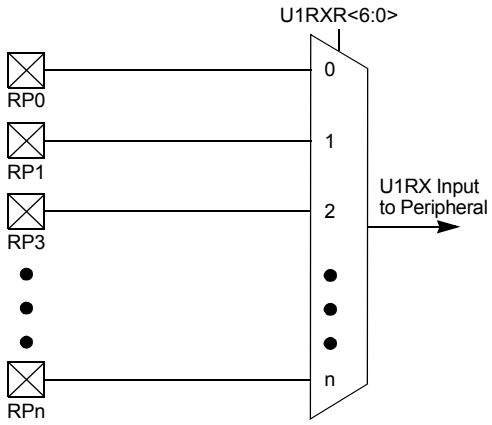
All peripherals also have the option to discontinue operation when Idle mode is entered to allow for increased power savings. This option is selectable in the control register of each peripheral; for example, the TSIDL bit in the Timer1 Control register (T1CON<13>).

10.2.3 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 either from Sleep mode or Idle mode.

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

FIGURE 11-2: REMAPPABLE INPUT FOR U1RX



Note: For input only, PPS functionality does not have priority over TRISx settings. Therefore, when configuring an RPn pin for input, the corresponding bit in the TRISx register must also be configured for input (set to '1').

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, RPI176-RPI181. 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 inter-peripheral 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.

REGISTER 17-12: TRGCONx: PWMx TRIGGER CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
TRGDIV3	TRGDIV2	TRGDIV1	TRGDIV0	—	—	—	—
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
—	—	TRGSTRT5(1)	TRGSTRT4(1)	TRGSTRT3(1)	TRGSTRT2(1)	TRGSTRT1(1)	TRGSTRT0(1)
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 15-12 **TRGDIV<3:0>:** Trigger Output Divider bits

1111 = Triggers output for every 16th trigger event
1110 = Triggers output for every 15th trigger event
1101 = Triggers output for every 14th trigger event
1100 = Triggers output for every 13th trigger event
1011 = Triggers output for every 12th trigger event
1010 = Triggers output for every 11th trigger event
1001 = Triggers output for every 10th trigger event
1000 = Triggers output for every 9th trigger event
0111 = Triggers output for every 8th trigger event
0110 = Triggers output for every 7th trigger event
0101 = Triggers output for every 6th trigger event
0100 = Triggers output for every 5th trigger event
0011 = Triggers output for every 4th trigger event
0010 = Triggers output for every 3rd trigger event
0001 = Triggers output for every 2nd trigger event
0000 = Triggers output for every trigger event

bit 11-6 **Unimplemented:** Read as '0'

bit 5-0 **TRGSTRT<5:0>:** Trigger Postscaler Start Enable Select bits(1)

111111 = Waits 63 PWM cycles before generating the first trigger event after the module is enabled
•
•
•
000010 = Waits 2 PWM cycles before generating the first trigger event after the module is enabled
000001 = Waits 1 PWM cycle before generating the first trigger event after the module is enabled
000000 = Waits 0 PWM cycles before generating the first trigger event after the module is enabled

Note 1: The secondary PWM generator cannot generate PWMx trigger interrupts.

18.0 SERIAL PERIPHERAL INTERFACE (SPI)

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 **"Serial Peripheral Interface (SPI)"** (DS70005185) 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. Refer to **Section 4.0 "Memory Organization"** in this data sheet for device-specific register and bit information.

The Serial Peripheral Interface (SPI) module is a synchronous serial interface, useful for communicating with other peripheral or microcontroller devices. These peripheral devices can be serial EEPROMs, shift registers, display drivers, ADC Converters, etc. The SPI module is compatible with the Motorola® SPI and SIOP interfaces.

The dsPIC33EVXXXGM00X/10X device family offers two SPI modules on a single device, SPI1 and SPI2, that are functionally identical. Each SPI module includes an eight-word FIFO buffer and allows DMA bus connections. When using the SPI module with DMA, FIFO operation can be disabled.

Note: In this section, the SPI modules are referred to together as SPIx, or separately as SPI1 and SPI2. Special Function Registers follow a similar notation. For example, SPIxCON refers to the control register for the SPI1 and SPI2 modules.

The SPI1 module uses dedicated pins which allow for a higher speed when using SPI1. The SPI2 module takes advantage of the Peripheral Pin Select (PPS) feature to allow for greater flexibility in pin configuration of this module, but results in a lower maximum speed. See **Section 30.0 "Electrical Characteristics"** for more information.

The SPIx serial interface consists of the following four pins:

- SDIx: Serial Data Input
- SDOx: Serial Data Output
- SCKx: Shift Clock Input or Output
- $\overline{SS}$x/FSYNCx: Active-Low Slave Select or Frame Synchronization I/O Pulse

Note: All of the 4 pins of the SPIx serial interface must be configured as digital in the ANSELx registers.

The SPIx module can be configured to operate with two, three or four pins. In 3-pin mode, $\overline{SS}$x is not used. In 2-pin mode, neither SDOx nor $\overline{SS}$x is used.

Figure 18-1 illustrates the block diagram of the SPIx module in Standard and Enhanced modes.

REGISTER 18-2: SPIxCON1: SPIx CONTROL REGISTER 1 (CONTINUED)

bit 4-2 **SPRE<2:0>:** Secondary Prescale bits (Master mode)(3)
111 = Secondary prescale 1:1
110 = Secondary prescale 2:1
•
•
•
000 = Secondary prescale 8:1

bit 1-0 **PPRE<1:0>:** Primary Prescale bits (Master mode)(3)
11 = Primary prescale 1:1
10 = Primary prescale 4:1
01 = Primary prescale 16:1
00 = Primary prescale 64:1

Note 1: The CKE bit is not used in Framed SPI modes. Program this bit to '0' for Framed SPI modes (FRMEN = 1).

2: This bit must be cleared when FRMEN = 1.

3: Do not set both primary and secondary prescalers to the value of 1:1.

REGISTER 21-2: UxSTA: UARTx STATUS AND CONTROL REGISTER (CONTINUED)

bit 5 **ADDEN:** Address Character Detect bit (bit 8 of received data = 1)
1 = Address Detect mode is enabled; if 9-bit mode is not selected, this does not take effect
0 = Address Detect mode is disabled

bit 4 **RIDLE:** Receiver Idle bit (read-only)
1 = Receiver is Idle
0 = Receiver is active

bit 3 **PERR:** Parity Error Status bit (read-only)
1 = Parity error has been detected for the current character (character at the top of the receive FIFO)
0 = Parity error has not been detected

bit 2 **FERR:** Framing Error Status bit (read-only)
1 = Framing error has been detected for the current character (character at the top of the receive FIFO)
0 = Framing error has not been detected

bit 1 **OERR:** Receive Buffer Overrun Error Status bit (clear/read-only)
1 = Receive buffer has overflowed
0 = Receive buffer has not overflowed; clearing a previously set OERR bit (1 → 0 transition) resets the receive buffer and the UxRSR to the empty state

bit 0 **URXDA:** UARTx Receive Buffer Data Available bit (read-only)
1 = Receive buffer has data, at least one more character can be read
0 = Receive buffer is empty

Note 1: Refer to **"Universal Asynchronous Receiver Transmitter (UART)"** (DS70000582) in the *"dsPIC33/PIC24 Family Reference Manual"* for information on enabling the UART module for transmit operation.

24.2 ADC Helpful Tips

1. The SMPIx control bits in the ADxCON2 registers:
   a) Determine when the ADC interrupt flag is set and an interrupt is generated, if enabled.
   b) When the CSCNA bit in the ADxCON2 register is set to '1', this determines when the ADC analog scan channel list, defined in the ADxCSSL/ADxCSSH registers, starts over from the beginning.
   c) When the DMA peripheral is not used (ADDMAEN = 0), this determines when the ADC Result Buffer Pointer to ADC1BUF0-ADC1BUFF gets reset back to the beginning at ADC1BUF0.
   d) When the DMA peripheral is used (ADDMAEN = 1), this determines when the DMA Address Pointer is incremented after a sample/conversion operation. ADC1BUF0 is the only ADC buffer used in this mode. The ADC Result Buffer Pointer to ADC1BUF0-ADC1BUFF gets reset back to the beginning at ADC1BUF0. The DMA address is incremented after completion of every 32nd sample/conversion operation. Conversion results are stored in the ADC1BUF0 register for transfer to RAM using the DMA peripheral.
2. When the DMA module is disabled (ADDMAEN = 0), the ADC has 16 result buffers. ADC conversion results are stored sequentially in ADC1BUF0-ADC1BUFF, regardless of which analog inputs are being used subject to the SMPIx bits and the condition described in 1.c) above. There is no relationship between the ANx input being measured and which ADC buffer (ADC1BUF0-ADC1BUFF) that the conversion results will be placed in.
3. When the DMA module is enabled (ADDMAEN = 1), the ADC module has only 1 ADC result buffer (i.e., ADCxBUF0) per ADC peripheral and the ADC conversion result must be read, either by the CPU or DMA Controller, before the next ADC conversion is complete to avoid overwriting the previous value.
4. The DONE bit (ADxCON1<0>) is only cleared at the start of each conversion and is set at the completion of the conversion, but remains set indefinitely, even through the next sample phase until the next conversion begins. If application code is monitoring the DONE bit in any kind of software loop, the user must consider this behavior because the CPU code execution is faster than the ADC. As a result, in Manual Sample mode, particularly where the user's code is setting the SAMP bit (ADxCON1<1>), the DONE bit should also be cleared by the user application just before setting the SAMP bit.
5. Enabling op amps, comparator inputs and external voltage references can limit the availability of analog inputs (ANx pins). For example, when Op Amp 2 is enabled, the pins for AN0, AN1 and AN2 are used by the op amp's inputs and output. This negates the usefulness of Alternate Input mode since the MUX A selections use AN0-AN2. Carefully study the ADC block diagram to determine the configuration that will best suit your application. For configuration examples, refer to **"Analog-to-Digital Converter (ADC)"** (DS70621) in the *"dsPIC33/PIC24 Family Reference Manual"*.

REGISTER 24-2: ADxCON2: ADCx CONTROL REGISTER 2

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0
VCFG2(1)	VCFG1(1)	VCFG0(1)	—	—	CSCNA	CHPS1	CHPS0
bit 15							bit 8

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BUFS	SMPI4	SMPI3	SMPI2	SMPI1	SMPI0	BUFM	ALTS
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 **VCFG<2:0>:** Converter Voltage Reference Configuration bits(1)

Value	VREFH	VREFL
xxx	AVDD	AVSS

bit 12-11 **Unimplemented:** Read as '0'

bit 10 **CSCNA:** Input Scan Select bit
1 = Scans inputs for CH0+ during Sample MUX A
0 = Does not scan inputs

bit 9-8 **CHPS<1:0>:** Channel Select bits
In 12-Bit Mode (AD21B = 1), CHPS<1:0> bits are Unimplemented and are Read as '0':
1x = Converts CH0, CH1, CH2 and CH3
01 = Converts CH0 and CH1
00 = Converts CH0

bit 7 **BUFS:** Buffer Fill Status bit (only valid when BUFM = 1)
1 = ADCx is currently filling the second half of the buffer; the user application should access data in the first half of the buffer
0 = ADCx is currently filling the first half of the buffer; the user application should access data in the second half of the buffer

bit 6-2 **SMPI<4:0>:** Increment Rate bits
When ADDMAEN = 0:
x1111 = Generates interrupt after completion of every 16th sample/conversion operation
x1110 = Generates interrupt after completion of every 15th sample/conversion operation
•
•
•
x0001 = Generates interrupt after completion of every 2nd sample/conversion operation
x0000 = Generates interrupt after completion of every sample/conversion operation
When ADDMAEN = 1:
11111 = Increments the DMA address after completion of every 32nd sample/conversion operation
11110 = Increments the DMA address after completion of every 31st sample/conversion operation
•
•
•
00001 = Increments the DMA address after completion of every 2nd sample/conversion operation
00000 = Increments the DMA address after completion of every sample/conversion operation

Note 1: The ADCx VREFH Input is connected to AVDD and the VREFL input is connected to AVSS.

REGISTER 25-5: CMxMSKCON: COMPARATOR x MASK GATING CONTROL REGISTER (CONTINUED)

bit 3 **ABEN:** AND Gate B Input Enable bit
1 = MBI is connected to AND gate
0 = MBI is not connected to AND gate

bit 2 **ABNEN:** AND Gate B Input Inverted Enable bit
1 = Inverted MBI is connected to AND gate
0 = Inverted MBI is not connected to AND gate

bit 1 **AAEN:** AND Gate A Input Enable bit
1 = MAI is connected to AND gate
0 = MAI is not connected to AND gate

bit 0 **AANEN:** AND Gate A Input Inverted Enable bit
1 = Inverted MAI is connected to AND gate
0 = Inverted MAI is not connected to AND gate

REGISTER 27-1: DEVID: DEVICE ID REGISTER

R	R	R	R	R	R	R	R
DEVID<23:16>(1)							
bit 23							bit 16

R	R	R	R	R	R	R	R
DEVID<15:8>(1)							
bit 15							bit 8

R	R	R	R	R	R	R	R
DEVID<7:0>(1)							
bit 7							bit 0

Legend:	R = Read-Only bit	U = Unimplemented bit

bit 23-0 **DEVID<23:0>:** Device Identifier bits(1)

Note 1: Refer to *"dsPIC33EVXXXGM00X/10X Families Flash Programming Specification"* (DS70005137) for the list of Device ID values.

REGISTER 27-2: DEVREV: DEVICE REVISION REGISTER

R	R	R	R	R	R	R	R
DEVREV<23:16>(1)							
bit 23							bit 16

R	R	R	R	R	R	R	R
DEVREV<15:8>(1)							
bit 15							bit 8

R	R	R	R	R	R	R	R
DEVREV<7:0>(1)							
bit 7							bit 0

Legend:	R = Read-only bit	U = Unimplemented bit

bit 23-0 **DEVREV<23:0>:** Device Revision bits(1)

Note 1: Refer to *"dsPIC33EVXXXGM00X/10X Families Flash Programming Specification"* (DS70005137) for the list of device revision values.

TABLE 30-7: DC CHARACTERISTICS: IDLE CURRENT (IIDLE)

DC CHARACTERISTICS			Standard Operating Conditions: 4.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended			
Parameter No.	**Typ.(2)**	**Max.**	**Units**	**Conditions**		
Idle Current (IIDLE)(1)						
DC40d	1.25	2	mA	-40°C	5.0V	10 MIPS
DC40a	1.25	2	mA	+25°C		
DC40b	1.5	2.6	mA	+85°C		
DC40c	1.5	2.6	mA	+125°C		
DC42d	2.3	3	mA	-40°C	5.0V	20 MIPS
DC42a	2.3	3	mA	+25°C		
DC42b	2.6	3.45	mA	+85°C		
DC42c	2.6	3.85	mA	+125°C		
DC44d	6.9	8	mA	-40°C	5.0V	70 MIPS
DC44a	6.9	8	mA	+25°C		
DC44b	7.25	8.6	mA	+85°C		

Note 1: Base Idle current (IIDLE) is measured as follows:

- CPU core is off, oscillator is configured in EC mode and external clock is active, OSC1 is driven with external square wave from rail-to-rail (EC clock overshoot/undershoot < 250 mV required)
- CLKO is configured as an I/O input pin in the Configuration Word
- All I/O pins are configured as outputs and driving low
- $\overline{MCLR}$ = VDD, WDT and FSCM are disabled
- No peripheral modules are operating or being clocked (defined PMDx bits are all ones)
- The NVMSIDL bit (NVMCON<12>) = 1 (i.e., Flash regulator is set to standby while the device is in Idle mode)
- The VREGSF bit (RCON<11>) = 0 (i.e., Flash regulator is set to standby while the device is in Sleep mode)

2: Data in “Typ.” column is at 5.0V, +25°C unless otherwise stated.

TABLE 30-9: DC CHARACTERISTICS: DOZE CURRENT (I_{DOZE})

DC CHARACTERISTICS			Standard Operating Conditions: 4.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Parameter No.	**Typ.(2)**	**Max.**	**Doze Ratio**	**Units**	**Conditions**		
Doze Current (I_{DOZE})(1)							
DC73a	16.0	18.25	1:2	mA	-40°C	5.0V	70 MIPS
DC73g	7.1	8.0	1:128	mA			
DC70a	16.25	18.5	1:2	mA	+25°C	5.0V	70 MIPS
DC70g	7.3	8.2	1:128	mA			
DC71a	17.0	19.0	1:2	mA	+85°C	5.0V	70 MIPS
DC71g	7.5	8.9	1:128	mA			
DC72a	17.75	19.95	1:2	mA	+125°C	5.0V	60 MIPS
DC72g	8.25	9.32	1:128	mA			

Note 1: I_{DOZE} is primarily a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption. The test conditions for all I_{DOZE} measurements are as follows:

- Oscillator is configured in EC mode and external clock is active, OSC1 is driven with external square wave from rail-to-rail (EC clock overshoot/undershoot < 250 mV required)
- CLKO is configured as an I/O input pin in the Configuration Word
- All I/O pins are configured as outputs and driving low
- $\overline{MCLR}$ = V_{DD}, WDT and FSCM are disabled
- CPU, SRAM, program memory and data memory are operational
- No peripheral modules are operating or being clocked (defined PMDx bits are all ones)
- CPU executing

```
while(1)
{
NOP();
}
```

2: Data in “Typ.” column is at 5.0V, +25°C unless otherwise stated.

TABLE 30-10: DC CHARACTERISTICS: I/O PIN INPUT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 4.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ.(1)**	**Max.**	**Units**	**Conditions**
DI10	VIL	**Input Low Voltage** I/O Pins	VSS	—	0.2 VDD	V	
DI20	VIH	**Input High Voltage** I/O Pins	0.75 VDD	—	5.5	V	
DI30	ICNPU	**Change Notification Pull-up Current**	200	375	600	μA	VDD = 5.0V, VPIN = VSS
DI31	ICNPD	**Change Notification Pull-Down Current(7)**	175	400	625	μA	VDD = 5.0V, VPIN = VDD
DI50	IIL	**Input Leakage Current(2,3)** I/O Pins	-100	—	100	nA	VSS ≤ VPIN ≤ VDD, pin at high-impedance
DI55		$\overline{\text{MCLR}}$	-700	—	700	nA	VSS ≤ VPIN ≤ VDD
DI56		OSC1	-200	—	200	nA	VSS ≤ VPIN ≤ VDD, XT and HS modes
DI60a	IICL	**Input Low Injection Current**	0	—	-5(4,6)		All pins except VDD, VSS, AVDD, AVSS, $\overline{\text{MCLR}}$, VCAP and RB7
DI60b	IICH	**Input High Injection Current**	0	—	+5(5,6)	mA	All pins except VDD, VSS, AVDD, AVSS, $\overline{\text{MCLR}}$, VCAP, RB7 and all 5V tolerant pins(5)
DI60c	∑IICT	**Total Input Injection Current** (sum of all I/O and control pins)	-20(7)	—	+20(7)		Absolute instantaneous sum of all ± input injection currents from all I/O pins (| IICL |+ | IICH |) ≤ ∑IICT

Note 1: Data in "Typ." column is at 5.0V, +25°C unless otherwise stated.

2: The leakage current on the $\overline{\text{MCLR}}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current can be measured at different input voltages.

3: Negative current is defined as current sourced by the pin.

4: VIL source < (VSS – 0.3). Characterized but not tested.

5: Digital 5V tolerant pins cannot tolerate any "positive" input injection current from input sources > 5.5V.

6: Non-zero injection currents can affect the ADC results by approximately 4-6 counts.

7: Any number and/or combination of I/O pins not excluded under IICL or IICH conditions are permitted, provided the mathematical "absolute instantaneous" sum of the input injection currents from all pins do not exceed the specified limit. Characterized but not tested.

FIGURE 30-30: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (SLAVE MODE)

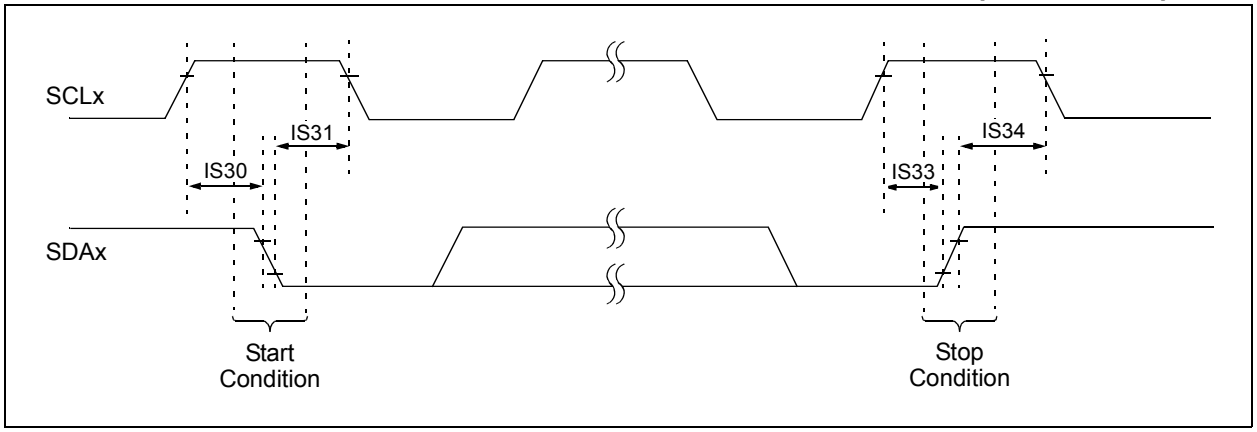


FIGURE 30-31: I2Cx BUS DATA TIMING CHARACTERISTICS (SLAVE MODE)

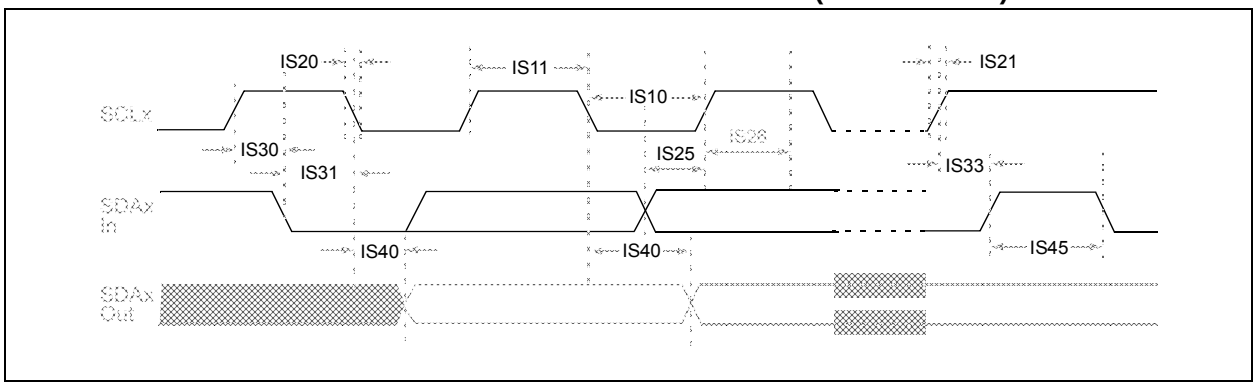
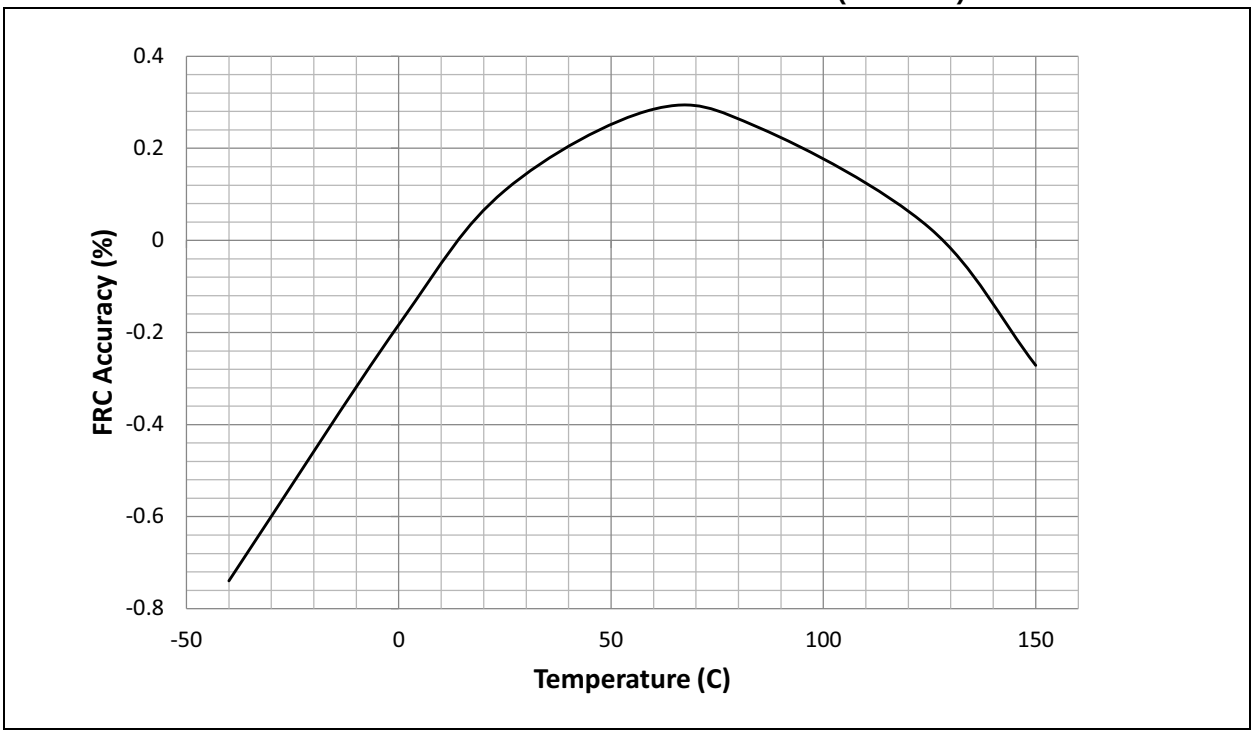
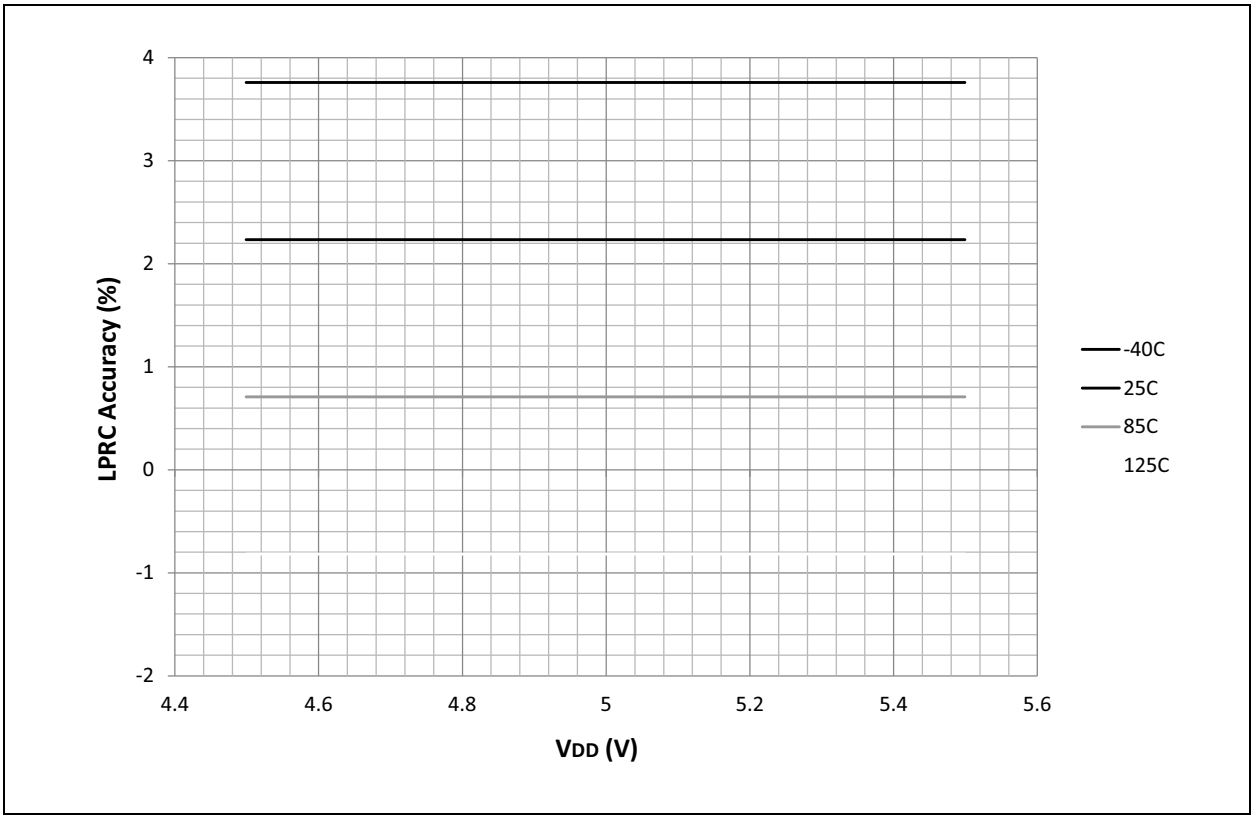


FIGURE 32-21: TYPICAL FRC ACCURACY vs. TEMPERATURE (5.5V VDD)



32.6 LPRC

FIGURE 32-22: TYPICAL LPRC ACCURACY vs. VDD



32.8 Pull-up and Pull-Down Current

FIGURE 32-27: TYPICAL PULL-UP CURRENT ($V_{PIN} = V_{SS}$) vs. TEMPERATURE

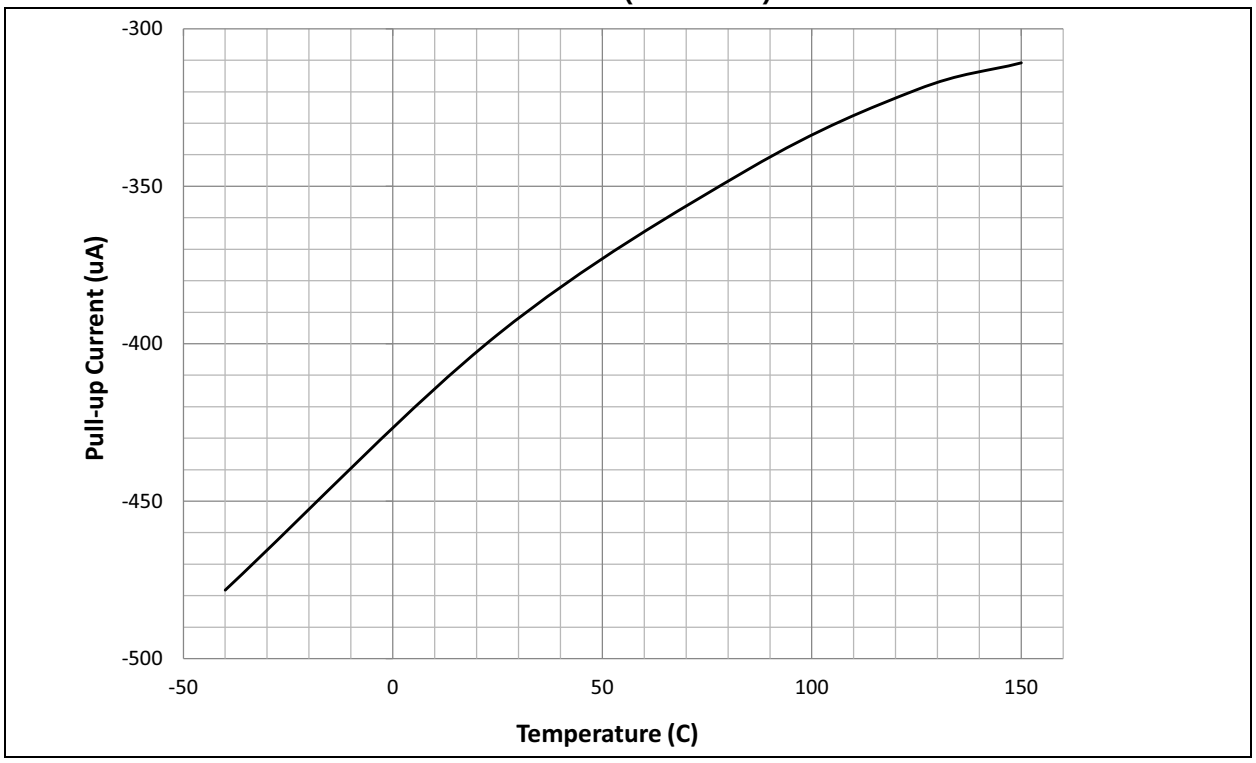
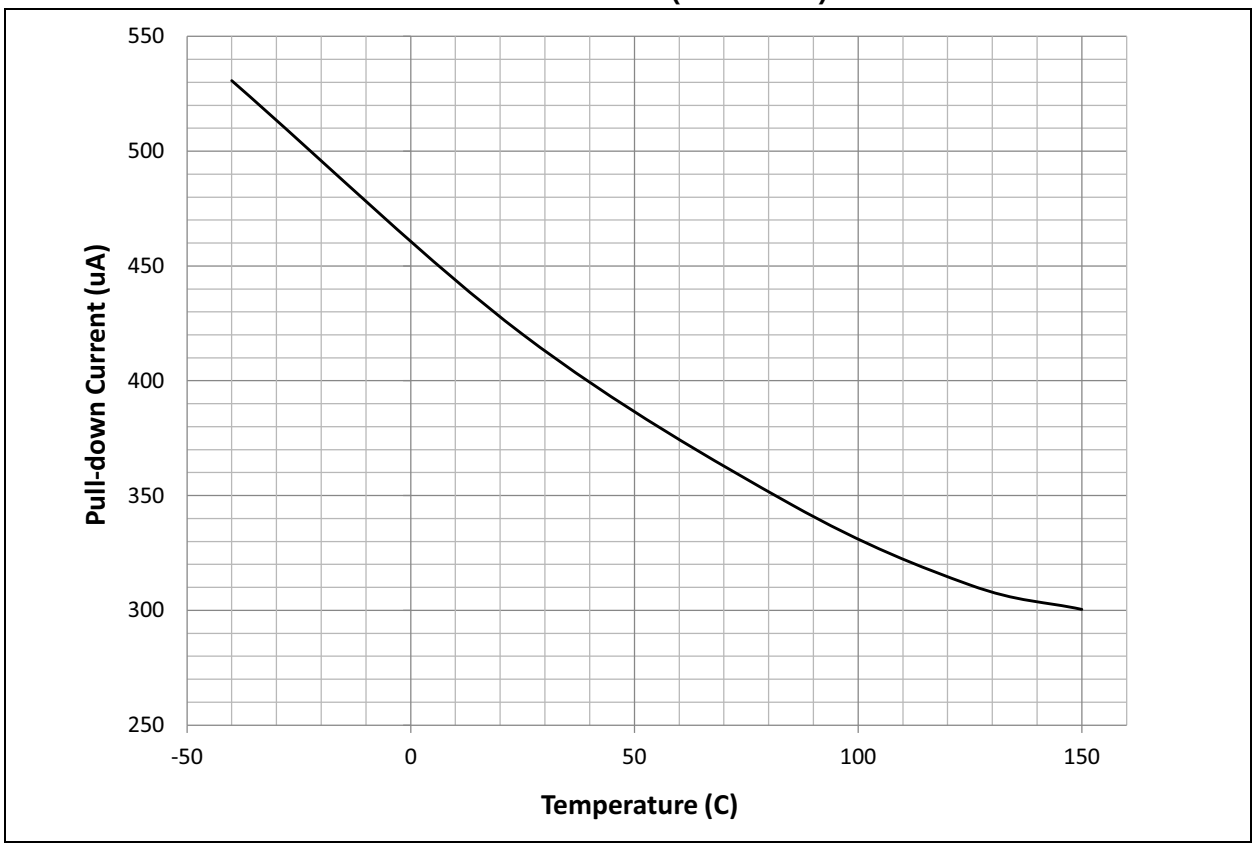


FIGURE 32-28: TYPICAL PULL-DOWN CURRENT (V_{PIN} = 5.5V) vs. TEMPERATURE



33.10 Voltage Output Low (V_{OL}) – Voltage Output High (V_{OH})

FIGURE 33-26: TYPICAL V_{OH} 8x DRIVER PINS vs. I_{OH} (GENERAL PURPOSE I/Os, TEMPERATURES AS NOTED)