MCP19122/3

Digitally Enhanced Power Analog Controller with Integrated Synchronous Driver

Synchronous Buck Features

- Input Voltage: 4.5V to 40V (operating), 48V (non-operating)
- Output Voltage: 0.3V to 16V
  - 0.1% typical output voltage accuracy
  - Greater than 16V requires external divider
- Switching Frequency: 100 kHz to 1.6 MHz
- Shutdown Quiescent Current: 50 µA Typical
- High-Drive:
  - +5V Gate Drive
  - 2A Source Current
  - 2A Sink Current
- Low-Drive:
  - +5V Gate Drive
  - 2A Source Current
  - 4A Sink Current
- Emulated Average Current Mode Control
- Differential Remote Output Sense
- Multi-Phase Systems:
  - Master or Slave
  - Frequency Synchronized
  - Common Current Sense Signal
- Multiple Output Systems:
  - Master or Slave
  - Frequency Synchronized
- AEC-Q100 Qualified
- Configureable Parameters:
  - Overcurrent Limit
  - Input Undervoltage Lockout
  - Input Overvoltage
  - Output Overvoltage
  - Output Undervoltage
  - Internal Analog Compensation
  - Soft Start Profile
  - Synchronous Driver Dead Time
  - Switching Frequency
- Thermal Shutdown

Microcontroller Features

- Precision 8 MHz Internal Oscillator Block:
  - Factory Calibrated
- Interrupt Capable
  - Firmware
  - Interrupt-on-Change Pins
- Only 35 Instructions to Learn
- 4096 Words On-Chip Program Memory
- High Endurance Flash:
  - 100,000 Write Flash Endurance
  - Flash Retention: >40 years
- Watchdog Timer (WDT) with Independent Oscillator for Reliable Operation
- Programmable Code Protection
- In-Circuit Debug (ICD) via Two Pins (MCP19123)
- In-Circuit Serial Programming™ (ICSP™) via Two Pins
- 12 I/O Pins and One Input-Only Pin (MCP19122)
  - 3 Open Drain Pins
  - 2 Weak Current Source Pins
- 16 I/O Pins and One Input-Only Pin (MCP19123)
  - 3 Open Drain Pins
  - 2 Weak Current Source Pins
- Analog-to-Digital Converter (ADC):
  - 10-bit Resolution
  - 24 Internal Channels
  - 8 External Channels
- Timer0: 8-bit Timer/Counter with 8-Bit Prescaler
- Enhanced Timer1:
  - 16-bit Timer/Counter with Prescaler
  - 2 Selectable Clock Sources
  - External Gate Input Mode
- Timer2: 8-Bit Timer/Counter with Prescaler
  - 8-bit Period Register
- Capture, Compare Module
- I^2C™ Communication:
  - 7-bit Address Masking
  - 2 Dedicated Address Registers
  - SMBus/PMBus™ Compatibility

Pin Diagram – 24-Pin 4X4 QFN (MCP19122)

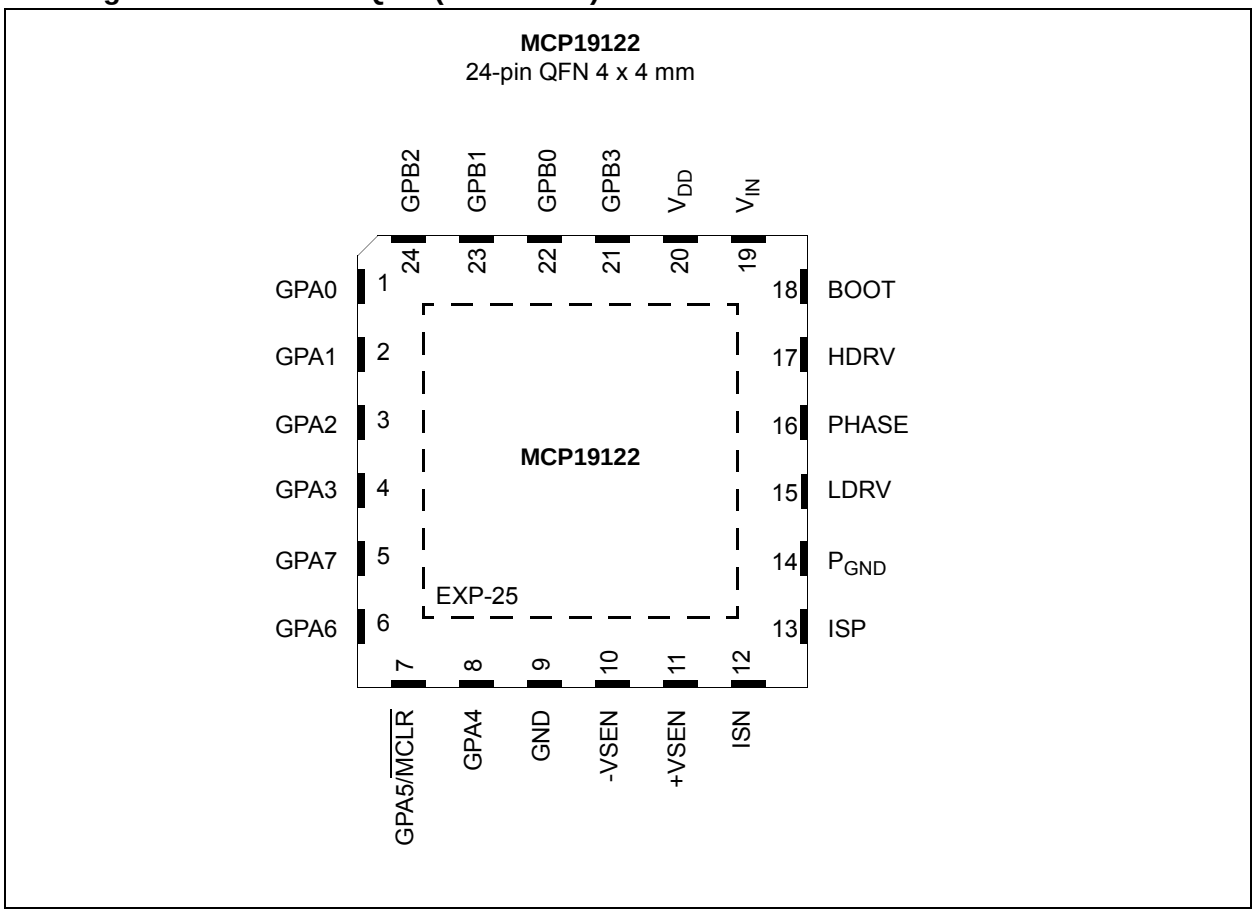


TABLE 1: 24-PIN QFN (MCP19122) SUMMARY

I/O	24-Pin QFN	ANSEL	A/D	Timers	MSSP	Interrupt	Pull-up	Basic	Additional
GPA0	1	Y	AN0	—	—	IOC	Y	—	Analog Debug Output (1)
GPA1	2	Y	AN1	—	—	IOC	Y	—	Sync Signal In/Out (2, 3)
GPA2	3	Y	AN2	T0CKI	—	IOC INT	—	—	Weak Current Source
GPA3	4	Y	AN3		—	IOC	—	—	Weak Current Source Timer1 Gate Input 1
GPA4	8	N	—	—	—	IOC	N	—	—
GPA5	7	N	—	—	—	IOC(4)	Y(5)	$\overline{\text{MCLR}}$	—
GPA6	6	N	—	—	—	IOC	N	ICSPDAT	—
GPA7	5	N	—	—	SCL	IOC	N	ICSPCLK	—
GPB0	22	N	—	—	SDA	IOC	N	—	—
GPB1	23	Y	AN4	—	—	IOC	Y	—	Current Sense Output Current Reference Input (3)
GPB2	24	Y	AN5	—	—	IOC	Y	—	Timer1 Gate Input 2
GPB3	21	N	—	—	—	IOC	Y	—	Clock Signal In/Out (2, 3)
V_{IN}	19	N	—	—	—	—	—	V_{IN}	Device Input Voltage
V_{DD}	20	N	—	—	—	—	—	V_{DD}	Internal Regulator Output
GND	9	N	—	—	—	—	—	GND	Small Signal Ground
P_{GND}	14	N	—	—	—	—	—	—	Large Signal Ground
LDRV	15	N	—	—	—	—	—	—	Low-Side MOSFET Connection
HDRV	17	N	—	—	—	—	—	—	High-Side MOSFET Connection
PHASE	16	N	—	—	—	—	—	—	Switch Node
BOOT	18	N	—	—	—	—	—	—	Floating Bootstrap Supply
$+V_{SEN}$	11	N	—	—	—	—	—	—	Output Voltage Differential Sense
$-V_{SEN}$	10	N	—	—	—	—	—	—	Output Voltage Differential Sense
ISP	13	N	—	—	—	—	—	—	Current Sense Input
ISN	12	N	—	—	—	—	—	—	Current Sense Input
EP	—	—	—	—	—	—	—	—	Exposed Pad

Note 1: The Analog Debug Output is selected when the BUFFCON<BNCHEN> bit is set.

2: Selected when device is functioning as multiple output master or slave by proper configuration of the MSC<2:0> bits in the MODECON register.

3: Selected when device is functioning as multi-phase master or slave by proper configuration of the MSC<2:0> bits in the MODECON register.

4: The IOC is disabled when $\overline{\text{MCLR}}$ is enabled.

5: Weak pull-up always enabled when $\overline{\text{MCLR}}$ is enabled, otherwise the pull-up is under user control.

Pin Diagram – 28-Pin 5X5 QFN (MCP19123)

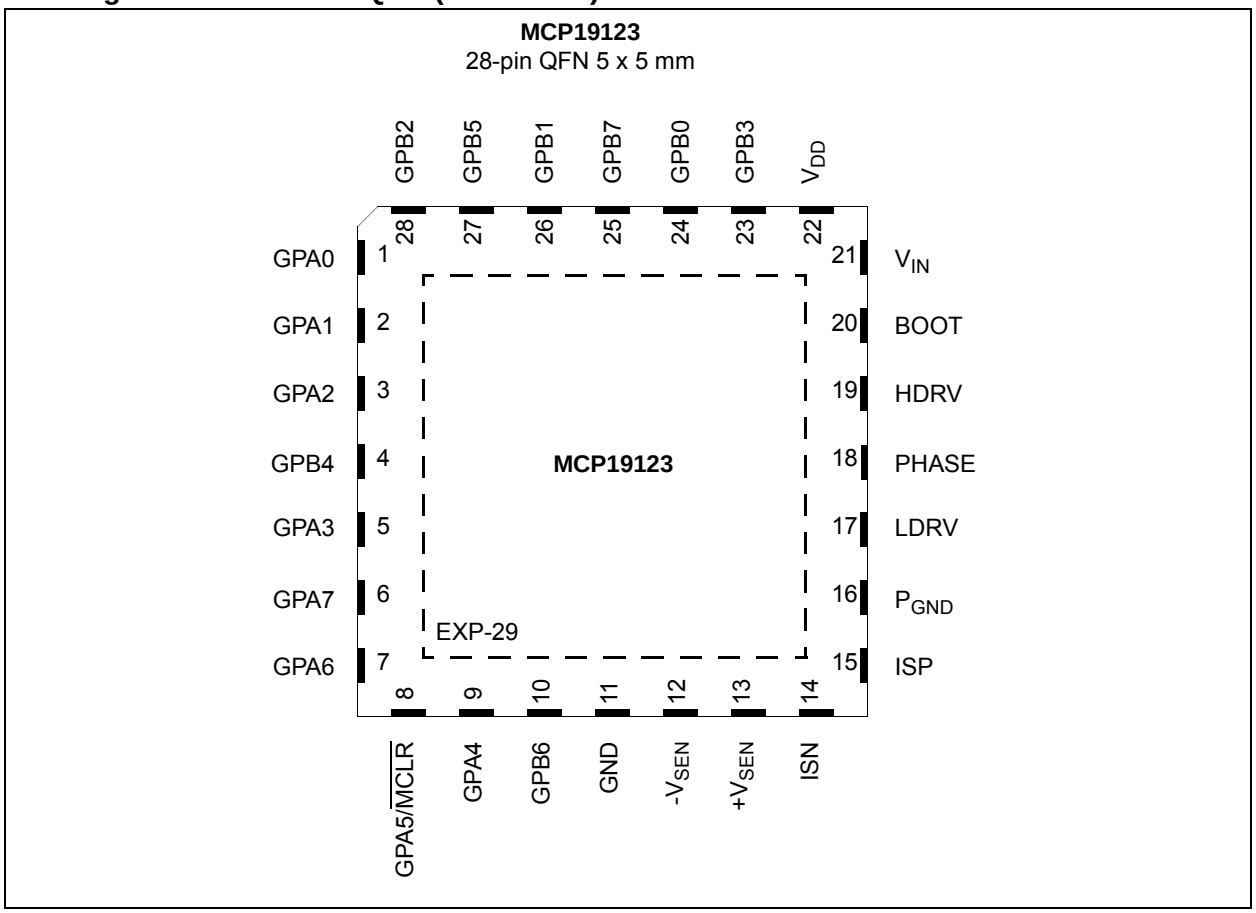


TABLE 2: 28-PIN QFN (MCP19123) SUMMARY

I/O	28-Pin QFN	ANSEL	A/D	Timers	MSSP	Interrupt	Pull-up	Basic	Additional
GPA0	1	Y	AN0	—	—	IOC	Y	—	Analog Debug Output (1)
GPA1	2	Y	AN1	—	—	IOC	Y	—	Sync Signal In/Out (2, 3)
GPA2	3	Y	AN2	T0CKI	—	IOC INT	Y	—	Weak Current Source
GPA3	5	Y	AN3	—	—	IOC	Y	—	Weak Current Source Timer1 Gate Input 1
GPA4	9	N	—	—	—	IOC	N	—	—
GPA5	8	N	—	—	—	IOC(4)	Y(5)	$\overline{\text{MCLR}}$	—
GPA6	7	N	—	—	—	IOC	N	—	CCD Input 1
GPA7	6	N	—	—	SCL	IOC	N	—	—
GPB0	24	N	—	—	SDA	IOC	N	—	—
GPB1	26	Y	AN4	—	—	IOC	Y	—	Current Sense Output Current Reference Input(3)
GPB2	28	Y	AN5	—	—	IOC	Y	—	Timer1 Gate Input 2
GPB3	23	N	—	—	—	IOC	Y	—	Clock Signal In/Out (2, 3)
GPB4	4	Y	AN6	—	—	IOC	Y	ICSPDAT ICDDAT	—
GPB5	27	Y	AN7	—	—	IOC	Y	ICSPCLK ICDCLK	—
GPB6	10	N	—	—	—	IOC	Y	—	CCD Input 2
GPB7	25	N	—	—	—	IOC	Y	—	External A/D Reference
V_{IN}	21	N	—	—	—	—	—	V_{IN}	Device Input Voltage
V_{DD}	22	N	—	—	—	—	—	V_{DD}	Internal Regulator Output
GND	11	N	—	—	—	—	—	GND	Small Signal Ground
P_{GND}	16	N	—	—	—	—	—	—	Large Signal Ground
LDRV	17	N	—	—	—	—	—	—	Low-Side MOSFET Connection
HDRV	19	N	—	—	—	—	—	—	High-Side MOSFET Connection
PHASE	18	N	—	—	—	—	—	—	Switch Node
BOOT	20	N	—	—	—	—	—	—	Floating Bootstrap Supply
$+V_{SEN}$	13	N	—	—	—	—	—	—	Output Voltage Differential Sense
$-V_{SEN}$	12	N	—	—	—	—	—	—	Output Voltage Differential Sense
ISP	15	N	—	—	—	—	—	—	Current Sense Input
ISN	14	N	—	—	—	—	—	—	Current Sense Input
EP	—	—	—	—	—	—	—	—	Exposed Pad

Note 1: The Analog Debug Output is selected when the BUFFCON<BNCHEN> bit is set.

2: Selected when device is functioning as multiple output master or slave by proper configuration of the MSC<2:0> bits in the MODECON register.

3: Selected when device is functioning as multi-phase master or slave by proper configuration of the MSC<2:0> bits in the MODECON register.

4: The IOC is disabled when $\overline{\text{MCLR}}$ is enabled.

5: Weak pull-up always enabled when $\overline{\text{MCLR}}$ is enabled, otherwise the pull-up is under user control.

Table of Contents

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via e-mail at **docerrors@microchip.com** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Web site; **http://www.microchip.com**
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our web site at **www.microchip.com** to receive the most current information on all of our products.

NOTES:

1.0 DEVICE OVERVIEW

The MCP19122/3 is a stand-alone mixed signal synchronous buck pulse-width modulated (PWM) current mode controller that features an integrated microcontroller core, high-endurance flash memory, communication and configurable analog circuitry. It features integrated synchronous drivers, bootstrap device, internal linear regulator and 4k words of nonvolatile memory. The devices are capable of efficiently converting 4.5V-40V to 0.3V-16V.

Since the MCP19122/3 uses traditional analog control circuitry to regulate the output of the DC/DC converter, the integration of the PIC® microcontroller mid-range core is used to provide complete customization of device operating parameters, start-up and shut-down profiles, protection levels and fault handling procedures.

After initial device configuration using Microchip's MPLAB® X Integrated Development Environment (IDE) software, PMBus commands or I^2C can be used by a host to communicate with, or modify, the operation of the MCP19122/3.

FIGURE 1-1: TYPICAL APPLICATION CIRCUIT

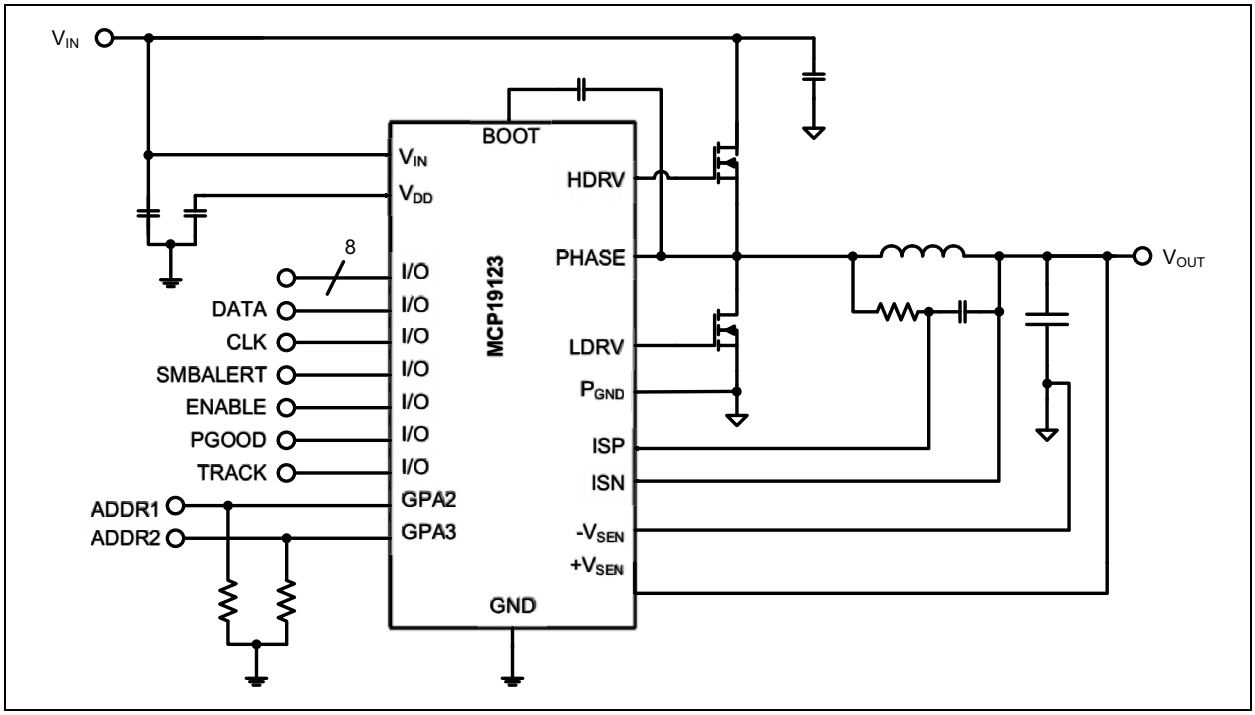


FIGURE 1-2: MCP19122/3 SYNCHRONOUS BUCK BLOCK DIAGRAM

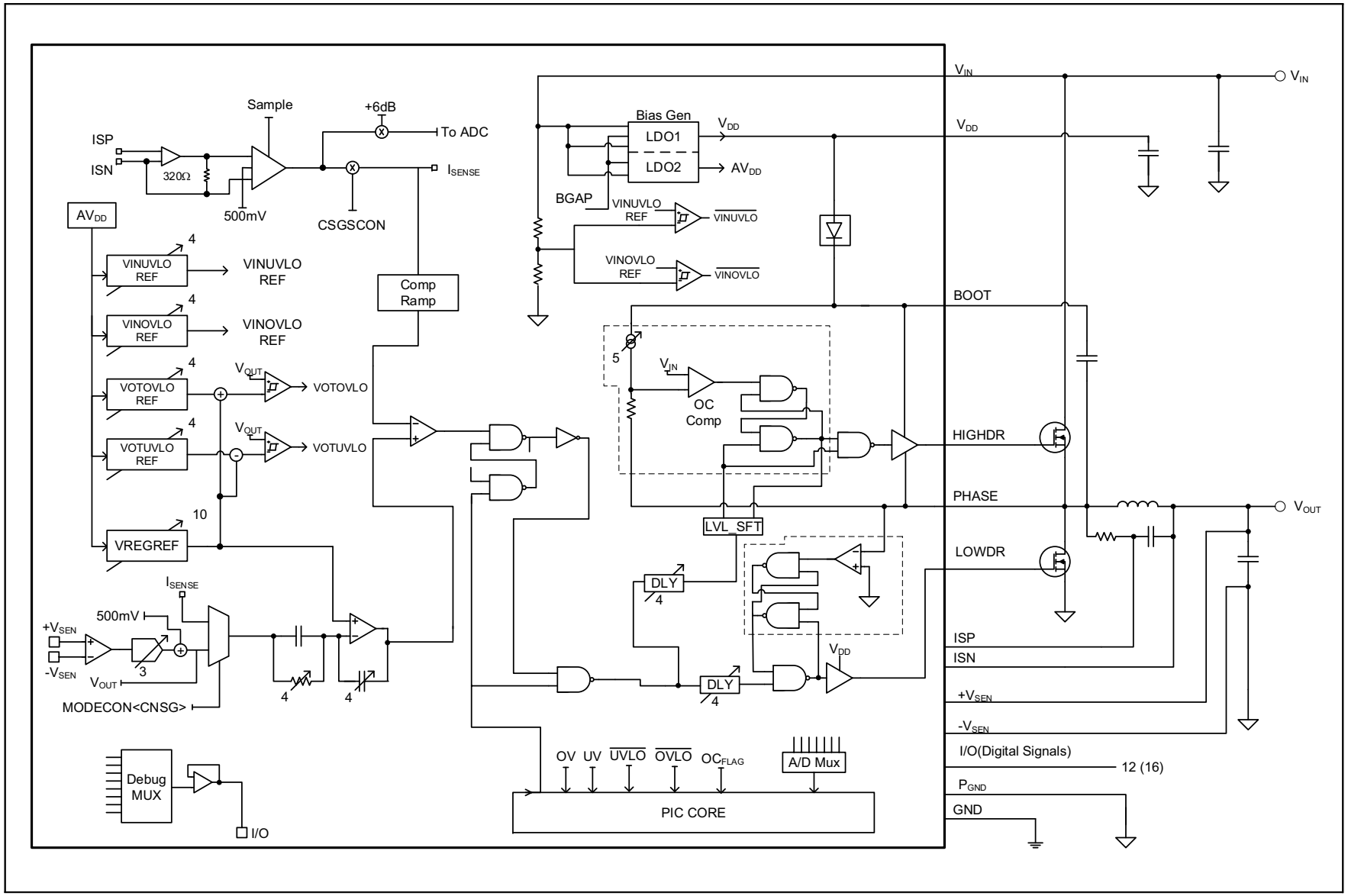
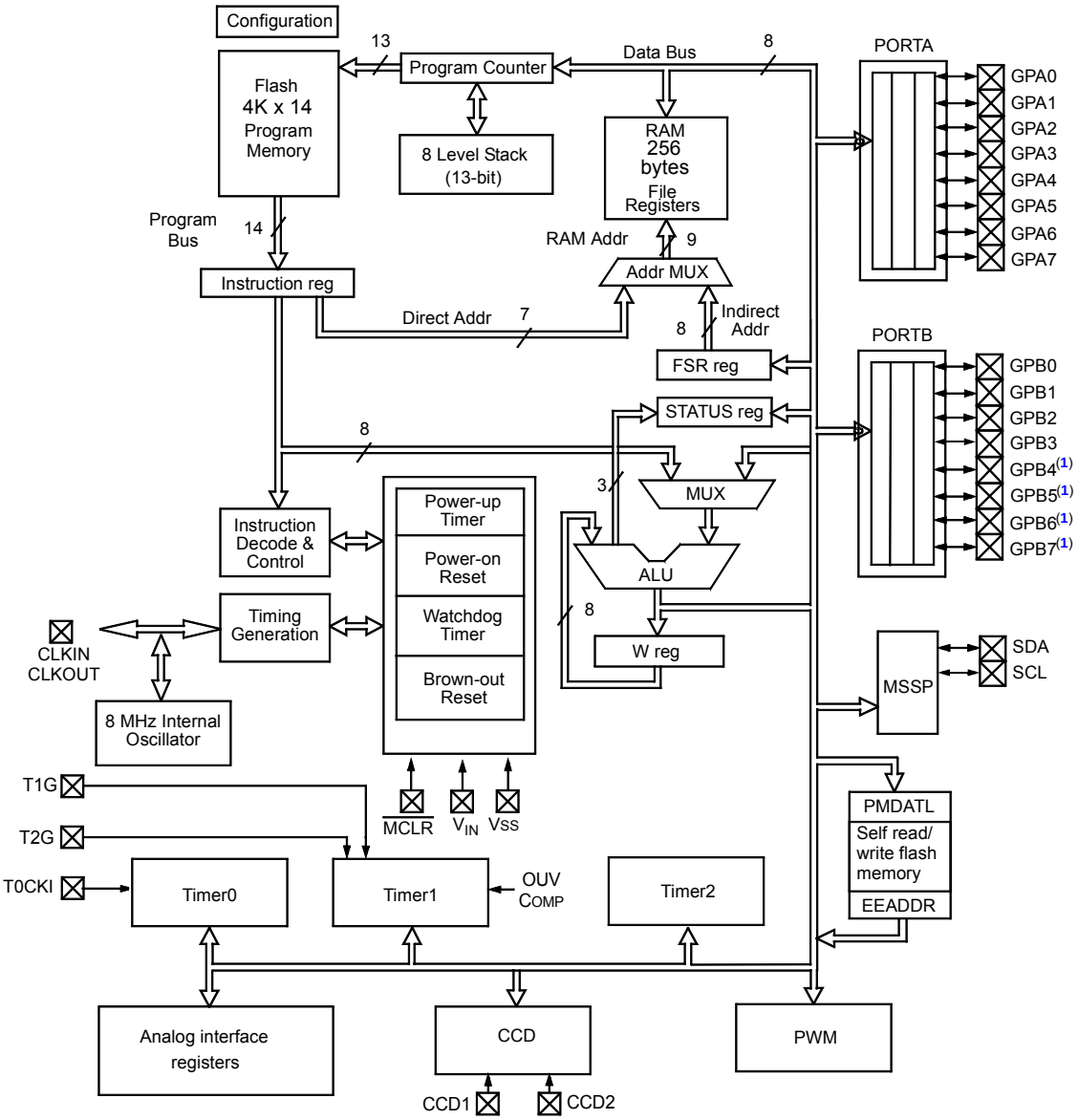


FIGURE 1-3: MICROCONTROLLER CORE BLOCK DIAGRAM



Note 1: Not implemented on the MCP19122.

2.0 PIN DESCRIPTION

The MCP19122/3 family of devices features pins that have multiple functions associated with each pin. Table 2-1 provides a description of the different functions. See **Section 2.1 "Detailed Pin Description"** for more detailed information.

TABLE 2-1: MCP19122/3 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
GPA0/AN0/ANALOG_TEST	GPA0	TTL	CMOS	General purpose I/O
	AN0	AN	—	A/D Channel 0 input.
	ANALOG_TEST	—	—	Internal analog signal multiplexer output (1)
GPA1/AN1/SYC_SIGNAL	GPA1	TTL	CMOS	General purpose I/O
	AN1	AN	—	A/D Channel 1 input
	SYC_SIGNAL	—	—	Switching clock synchronization signal input and output (2,3)
GPA2/AN2/T0CKI/INT	GPA2	TTL	CMOS	General purpose I/O
	AN2	AN	—	A/D Channel 2 input
	T0CKI	ST	—	Timer0 clock input
	INT	ST	—	External interrupt
GPA3/AN3/T1G1	GPA3	TTL	CMOS	General purpose I/O
	AN3	AN	—	A/D Channel 3 input
	T1G1	ST	—	Timer1 gate input 1
GPA4	GPA4	TTL	OD	General purpose I/O
GPA5/$\overline{\text{MCLR}}$	GPA5	TTL	—	General purpose input only
	$\overline{\text{MCLR}}$	ST	—	Master Clear with internal pull-up
GPA6/CCD1(4)/ICSPDAT(5)	GPA6	ST	CMOS	General purpose I/O
	CCD1	ST	CMOS	Capture/Compare input 1 (4)
	ICSPDAT		CMOS	Serial Programming Data I/O (5)
GPA7/SCL/ICSPCLK(5)	GPA7	ST	OD	General purpose open drain I/O
	SCL	I^2C	OD	I^2C clock
	ICSPCLK	ST	—	Serial Programming Clock (5)
GPB0/SDA	GPB0	TTL	OD	General purpose I/O
	SDA	I^2C	OD	I^2C data input/output
GPB1/AN4/CON_SIGNAL	GPB1	TTL	CMOS	General purpose I/O
	AN4	AN	—	A/D Channel 4 input
	CON_SIGNAL	—	—	Current sense output or current reference input (3)

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain
TTL = TTL compatible input ST =Schmitt Trigger input with CMOS levels I^2C = Schmitt Trigger input with I^2C

Note 1: Analog Test is selected when the BUFFCON<BNCHEN> bit is set.

2: Selected when device is functioning as multiple output master or slave by proper configuration of the MSC<2:0> bits in the MODECON register.

3: Selected when device is functioning as multi-phase master or slave by proper configuration of the MSC<2:0> bits in the MODECON register.

4: Feature only available on the MCP19123.

5: Feature only available on the MCP19122.

TABLE 2-1: MCP19122/3 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
GPB2/AN5/T1G2	GPB2	TTL	CMOS	General purpose I/O
	AN5	AN	—	A/D Channel 5 input
	T1G2	ST	—	Timer1 gate input 2
GPB3/CLOCK	GPB3	TTL	CMOS	General purpose I/O
	CLOCK	—	—	Clock signal input/output (2,3)
GPB4(4)/AN6(4)/ICSPDAT(4)/ ICDDAT(4)	GPB4	TTL	CMOS	General purpose I/O(4)
	AN6	AN	—	A/D Channel 6 input(4)
	ICSPDAT	ST	—	Serial Programming Data I/O(4)
	ICDDAT	ST	—	In-circuit debug data(4)
GPB5(4)/AN7(4)/ICSPCLK(4)/ ICDCLK(4)	GPB5	TTL	CMOS	General purpose I/O(4)
	AN7	AN	—	A/D Channel 7 input(4)
	ISCPCLK	ST	—	Serial Programming Clock(4)
	ICDCLK	ST	—	In-circuit debug clock(4)
GPB6/CCD2(4)	GPB6	TTL	CMOS	General purpose I/O
	CCD2	ST	CMOS	Capture/Compare input 2(4)
GPB7/VADC(4)	GPB7	TTL	CMOS	General purpose I/O
	VADC	AN		External voltage reference for A/D(4)
V_{IN}	V_{IN}	—	—	Device input supply voltage
V_{DD}	V_{DD}	—	—	Internal +5V LDO output pin
GND	GND	—	—	Small signal quiet ground
P_{GND}	P_{GND}	—	—	Large signal power ground
LDRV	LDRV	—	—	High-current drive signal connected to the gate of the low-side MOSFET
HDRV	HDRV	—	—	Floating high-current drive signal connected to the gate of the high-side MOSFET
PHASE	PHASE	—	—	Synchronous buck switch node connection
BOOT	BOOT	—	—	Floating bootstrap supply
$+V_{SEN}$	$+V_{SEN}$	—	—	Positive input of the output voltage sense differential amplifier
$-V_{SEN}$	$-V_{SEN}$	—	—	Negative input of the output voltage sense differential amplifier
ISP	ISP	—	—	Current sense input
ISN	ISN	—	—	Current sense input
EP	—	—	—	Exposed Thermal Pad

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain
TTL = TTL compatible input ST =Schmitt Trigger input with CMOS levels I^2C = Schmitt Trigger input with I^2C

Note 1: Analog Test is selected when the BUFFCON<BNCHEN> bit is set.

2: Selected when device is functioning as multiple output master or slave by proper configuration of the MSC<2:0> bits in the MODECON register.

3: Selected when device is functioning as multi-phase master or slave by proper configuration of the MSC<2:0> bits in the MODECON register.

4: Feature only available on the MCP19123.

5: Feature only available on the MCP19122.

2.1 Detailed Pin Description

2.1.1 GPA0 PIN

GPA0 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPA. An internal weak pull-up and interrupt-on-change are also available.

AN0 is an input to the A/D. To configure this pin to be read by the A/D on channel 0, bits TRISA0 and ANSA0 must be set.

When the BUFFCON<BNCHEN> bit is set, this pin is configured as the ANALOG_TEST function. It is a buffered output of the internal analog and digital signal multiplexer. Analog signals present on this pin are controlled by the ADCON0 register; see Register 19-1. Digital signals present on this pin are controlled by the BUFFCON register; see Register 7-1.

2.1.2 GPA1 PIN

GPA1 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPA. An internal weak pull-up and interrupt-on-change are also available.

AN1 is an input to the A/D. To configure this pin to be read by the A/D on channel 1, bits TRISA1 and ANSA1 must be set.

When the MCP19122/3 is configured as a multiple output or multi-phase MASTER or SLAVE, this pin is configured to be the switching frequency synchronization input or output, SYN_SIGNAL. See **Section 3.12 "System Configuration Control"** for more information.

2.1.3 GPA2 PIN

GPA2 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPA. An internal weak current source and interrupt-on-change are also available.

AN2 is an input to the A/D. To configure this pin to be read by the A/D on channel 2, bits TRISA2 and ANSA2 must be set.

When bit T0CS is set, the T0CKI function is enabled. See **Section 21.0 "Timer0 Module"** for more information.

GPA2 can also be configured as an external interrupt by setting of the INTE bit. See **Section 13.0.1 "GPA2/INT Interrupt"** for more information.

2.1.4 GPA3 PIN

GPA3 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPA. An internal weak current source and interrupt-on-change are also available.

AN3 is an input to the A/D. To configure this pin to be read by the A/D on channel 3, bits TRISA3 and ANSA3 must be set.

T1G1 is an input to the TIMER1 gate. To configure this pin to be an external source to the TIMER1 gate circuitry, see **Section 22.0 "Timer1 Module With Gate Control"**.

2.1.5 GPA4 PIN

GPA4 is a true open drain general purpose pin whose data direction is controlled in TRISGPA. There is no internal connection between this pin and device V_{DD}, making this pin ideal to be used as an SMBus Alert pin. This pin does not have a weak pull-up, but interrupt-on-change is available.

2.1.6 GPA5 PIN

GPA5 is a general purpose TTL input-only pin. An internal weak pull-up and interrupt-on-change are also available.

For programming purposes, this pin is to be connected to the $\overline{MCLR}$ pin of the serial programmer. See **Section 29.0 "In-Circuit Serial Programming™ (ICSP™)"** for more information.

2.1.7 GPA6 PIN

GPA6 is a general purpose CMOS input/output pin whose data direction is controlled in TRISGPA. An interrupt-on-change is also available.

On the MCP19122, the ISCPDAT is the primary serial programming data input function. This is used in conjunction with ICSPCLK to serial program the device. This pin function is only implemented on the MCP19122.

On the MCP19123, this pin can be configured as an input to the CCD module. For more information refer to **Section 24.0 "Dual Capture/Compare (CCD) Module"**.

2.1.8 GPA7 PIN

GPA7 is a true open drain general purpose pin whose data direction is controlled in TRISGPA. There is no internal connection between this pin and device V_{DD}. This pin does not have a weak pull-up, but interrupt-on-change is available.

When the MCP19122/3 is configured for I^2C communication (see **Section 27.2 "I^2C Mode Overview"**), GPA7 functions as the I^2C clock, SCL.

On the MCP19122, the ISCPCLK is the serial programming clock function. This is used in conjunction with ICSPDAT to serial program the device. This pin function is only implemented on the MCP19122.

2.1.9 GPB0 PIN

GPB0 is a true open drain general purpose pin whose data direction is controlled in TRISGPB. There is no internal connection between this pin and device V_{DD}. This pin does not have a weak pull-up, but interrupt-on-change is available.

When the MCP19122/3 is configured for I^2C communication (see **Section 27.2 "I^2C Mode Overview"**), GPB0 functions as the I^2C clock, SDA.

2.1.10 GPB1 PIN

GPB1 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPB. An internal weak pull-up and interrupt-on-change are also available.

AN4 is an input to the A/D. To configure this pin to be read by the A/D on channel 4, bits TRISB1 and ANSB1 must be set.

When the MCP19122/3 is configured as a multi-phase MASTER or SLAVE, this pin is configured to be the sensed current input or output signal. On a device configured to be a MASTER, this is an output signal of the sensed current that is to be shared with the SLAVE devices. On a device configured as a SLAVE, this is an input signal used to as a current regulation point. See **Section 3.12 "System Configuration Control"**, for more information.

2.1.11 GPB2 PIN

GPB2 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPB. An internal weak pull-up and interrupt-on-change are also available.

AN5 is an input to the A/D. To configure this pin to be read by the A/D on channel 5, bits TRISB2 and ANSB2 must be set.

T1G2 is an input to the TIMER1 gate. To configure this pin to be an external source to the TIMER1 gate circuitry, see **Section 22.0 "Timer1 Module With Gate Control"**.

2.1.12 GPB3 PIN

GPB3 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPB. An internal weak pull-up and interrupt-on-change are also available.

When the MCP19122/3 is configured as a multiple output or multi-phase Master or Slave, this pin is configured to be the switching frequency clock input or output. See **Section 3.12 "System Configuration Control"**.

2.1.13 GPB4 PIN

This pin and associated functions are only available on the MCP19123 device.

GPB4 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPB. An internal weak pull-up and interrupt-on-change are also available.

AN6 is an input to the A/D. To configure this pin to be read by the A/D on channel 6, bits TRISB4 and ANSB4 must be set.

On the MCP19123, the ISCPDAT is the primary serial programming data input function. This is used in conjunction with ICSPCLK to serial program the device.

The ICDDAT is the in-circuit debug data function. This pin function is only implemented on the MCP19123. See **Section 29.2 "In-Circuit Debugger"**

2.1.14 GBP5 PIN

This pin and associated functions is only available on the MCP19123 device.

GPB5 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPB. An internal weak pull-up and interrupt-on-change are also available.

AN7 is an input to the A/D. To configure this pin to be read by the A/D on channel 7, bits TRISB5 and ANSB5 must be set.

On the MCP19123, the ISCPCLK is the primary serial programming clock function. This is used in conjunction with ICSPDAT to serial program the device.

The ICDDLK is the in-circuit debug clock function. This pin function is only implemented on the MCP19123. See **Section 29.2 "In-Circuit Debugger"**

2.1.15 GPB6 PIN

This pin and associated functions is only available on the MCP19123 device.

GPB6 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPB. An internal weak pull-up and interrupt-on-change are also available.

CCD2 is an input to the CCD module. For more information refer to **Section 24.0 "Dual Capture/Compare (CCD) Module"**.

2.1.16 GPB7 PIN

This pin and associated functions is only available on the MCP19123 device.

GPB7 is a general purpose TTL input or CMOS output pin whose data direction is controlled in TRISGPB. An internal weak pull-up and interrupt-on-change are also available.

VADC is an external A/D reference voltage input. See **Section 19.0 "Analog-to-Digital Converter (ADC) Module"**.

2.1.17 V_{IN} PIN

Device input power connection pin. It is recommended that capacitance be placed between this pin and the GND pin of the device.

2.1.18 V_{DD} PIN

The output of the internal +5.0V regulator is connected to this pin. It is recommended that a 1.0 µF bypass capacitor be connected between this pin and the GND pin of the device. The bypass capacitor should be placed physically close to the device.

2.1.19 GND PIN

GND is the small signal ground connection pin. This pin should be connected to the exposed pad, on the bottom of the package.

2.1.20 P_{GND} PIN

Connect all large signal level ground returns to P_{GND}. These large-signal level ground traces should have a small loop area and minimal length to prevent coupling of switching noise to sensitive traces.

2.1.21 LDRV PIN

The gate of the low-side or rectifying MOSFET is connected to LDRV. The PCB trace connecting LDRV to the gate must be of minimal length and appropriate width to handle the high peak drive currents and fast voltage transitions.

2.1.22 HDRV PIN

The gate of the high-side MOSFET is connected to HDRV. This is a floating driver referenced to PHASE. The PCB trace connecting HDRV to the gate must be of minimal length and appropriate width to handle the high peak drive current and fast voltage transitions.

2.1.23 PHASE PIN

The PHASE pin provides the return path for the high-side gate driver. The source of the high-side MOSFET, drain of the low-side MOSFET and the inductor are connected to this pin.

2.1.24 BOOT PIN

The BOOT pin is the floating bootstrap supply pin for the high-side gate driver. A capacitor is connected between this pin and the PHASE pin to provide the necessary charge to turn on the high-side MOSFET.

2.1.25 $+V_{SEN}$ PIN

The non-inverting input of the unity gain amplifier used for output voltage remote sensing is connected to the $+V_{SEN}$ pin.

2.1.26 $-V_{SEN}$ PIN

The inverting input of the unity gain amplifier used for output voltage remote sensing is connected to the $–V_{SEN}$ pin.

2.1.27 ISP PIN

The non-inverting input of the current sense amplifier is connected to the ISP pin.

2.1.28 ISN PIN

The inverting input of the current sense amplifier is connected to the ISN pin.

2.1.29 EXPOSED PAD (EP)

There is no internal connection to the Exposed Thermal Pad. The EP should be connected to the GND pin and to the GND PCB plane to aid in the removal of the heat.

3.0 FUNCTIONAL DESCRIPTION

3.1 Internal Supplies

The operating input voltage of the MCP19122/3 ranges from 4.5V to 40V. There are two internal Low Dropout (LDO) voltage regulators. A 5V LDO (V_{DD}) is used to power the internal microcontroller, the internal gate driver circuitry and provide a 5V output for external use. It is recommended that a 1 µF ceramic capacitor be placed between the V_{DD} pin and the P_{GND} pin.

The MODECON<VDDEN> bit controls the state of the 5V V_{DD} LDO when the SLEEP command is issued to the MCP19122/3. See **Section 3.12.3 "VDD LDO Control"** for more information.

The gate drive current required to drive the external power MOSFETs must be added to the MCP19122/3 quiescent current $I_{Q(max)}$. This total current must be less than the maximum current, I_{DD-OUT}, available from V_{DD} that is specified in **Section 4.0 "Electrical Characteristics"**.

A second 4V LDO (AV_{DD}) is used to power the internal analog circuitry. The AV_{DD} is not available externally. AV_{DD} is calibrated to 4.096V and is the default ADC reference voltage.

EQUATION 3-1: TOTAL REGULATOR CURRENT

$$I_{DD-OUT} > (I_Q + I_{DRIVE} + I_{EXT})$$

Where:

- I_{DD-OUT} is the total current available from V_{DD}
- I_Q is the device quiescent current
- I_{DRIVE} is the current required to drive the external MOSFETs
- I_{EXT} is the amount of current used to power additional external circuitry.

EQUATION 3-2: GATE DRIVE CURRENT

$$I_{DRIVE} = (Q_{gHIGH} + Q_{gLOW}) \times F_{SW}$$

Where:

- I_{DRIVE} is the current required to drive the external MOSFETs
- Q_{gHIGH} is the total gate charge of the high-side MOSFET
- Q_{gLOW} is the total gate charge of the low-side MOSFET
- F_{SW} is the switching frequency

3.2 Switching Frequency

The switching frequency is configurable over the range of 100 kHz to 1.6 MHz. The Timer2 module is used to generate the HDRV/LDRV switching frequency. Refer to Section 26.0, Enhanced PWM Module for more information. Example 3-1 shows how to configure the MCP19122/3 for a switching frequency of 300 kHz.

EXAMPLE 3-1: CONFIGURING F_{SW}

```
BANKSEL     T2CON
CLRF        T2CON   ;Turn off Timer2
CLRF        TMR2    ;Initialize module
MOVLW       0x19    ;Fsw=300 kHz
MOVWF       PR2
MOVLW       0x0A    ;Max duty cycle=40%
MOVWF       PWMRL
MOVWF       PWMRH
MOVLW       0x00    ;No phase shift
MOVWF       PWMPHL
MOVWF       PWMPHH
MOVLW       0x04    ;Turn on Timer2
MOVWF       T2CON
```

3.3 Input Voltage Monitoring

The input voltage to the MCP19122/3 is monitored to determine an input undervoltage or an input overvoltage. It can also be measured by the ADC and reported as telemetry data.

3.3.1 INPUT UNDERVOLTAGE LOCKOUT

The VINUVLO register contains the digital value that sets the input under voltage lockout. When the input voltage on the V_{IN} pin to the MCP19122/3 is below this programmed level, the PIR2<UVLOIF> status flag will be set. This bit is automatically cleared when the MCP19122/3 V_{IN} voltage rises above this programmed level. The VINUVLO shall operate on a rising or falling input voltage. Hysteresis shall exist between the rising threshold that clears the flag and the failing threshold that sets the flag.

A hardware under voltage lockout path can be enabled by setting the VINCON<UVLOEN> bit. When this bit is set and the voltage on the V_{IN} pin is below the threshold set by the VINUVLO register, hardware will keep the high-side and low-side MOSFET drivers off. Once the voltage on the V_{IN} pin is greater than the threshold set by the VINUVLO register, the high-side and low-side MOSFET drivers are enabled.

To function properly, the V_{IN} under voltage lockout setting must be lower than the V_{IN} over voltage lockout setting. The state of the VINUVLO and VINOVLO registers are unknown at power-up. Therefore if only the V_{IN} under voltage lockout is desired, the V_{IN} over voltage lockout threshold still must be set in the VINOVLO register.

Note:	The UVLOIF interrupt flag bit is set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register.
Note:	The UVLOIF interrupt flag bit is set when an interrupt condition occurs regardless of the state of the VINCON<UVLOEN> bit.

REGISTER 3-1: VINUVLO: INPUT UNDER VOLTAGE LOCKOUT CONTROL REGISTER

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	UVLO3	UVLO2	UVLO1	UVLO0
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 7-4 **Unimplemented:** Read as ‘0’

bit 3-0 **UVLO<3:0>:** Under Voltage Lockout Configuration bits

0000 = 4.0V
0001 = 6.0V
0010 = 8.0V
0011 = 10.0V
0100 = 12.0V
0101 = 14.0V
0110 = 16.0V
0111 = 18.0V
1000 = 20.0V
1001 = 22.0V
1010 = 24.0V
1011 = 26.0V
1100 = 28.0V
1101 = 30.0V
1110 = 32.0V
1111 = 34.0V

3.3.2 INPUT OVER VOLTAGE LOCKOUT

The VINOVLO register contains the digital value that sets the input over voltage lockout. When the input voltage on the V_{IN} pin to the MCP19122/3 is above this programmed level, the PIR2<OVLOIF> status flag will be set. This bit is automatically cleared when the MCP19122/3 V_{IN} voltage falls below this programmed level. The VINOVLO shall operate on a rising or falling input voltage. Hysteresis shall exist between the rising threshold that sets the flag and the failing threshold that clears the flag.

A hardware over voltage lockout path can be enabled by setting the VINCON<OVLOEN> bit. When this bit is set and the voltage on the V_{IN} pin is above the threshold set by the VINOVLO register, hardware will keep the high-side and low-side MOSFET drivers off. Once the voltage on the V_{IN} pin is lower than the threshold set by the VINOVLO register, the high-side and low-side MOSFET drivers are enabled.

To function properly, the V_{IN} overvoltage lockout setting must be lower than the V_{IN} undervoltage lockout setting. The state of the VINUVLO and VINOVLO registers are unknown at power-up. Therefore if only the V_{IN} overvoltage lockout is desired, the V_{IN} undervoltage lockout threshold still must be set in the VINUVLO register.

Note:	The OVLOIF interrupt flag bit is set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register.
Note:	The OVLOIF interrupt flag bit is set when an interrupt condition occurs regardless of the state of the VINCON<OVLOEN> bit.

REGISTER 3-2: VINOVLO: INPUT OVERVOLTAGE LOCKOUT CONTROL REGISTER

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	OVLO3	OVLO2	OVLO1	OVLO0
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 7-4 **Unimplemented:** Read as ‘0’

bit 3-0 **OVLO<3:0>:** Overvoltage Lockout Configuration bits
0000 = 12.0V
0001 = 14.0V
0010 = 16.0V
0011 = 18.0V
0100 = 20.0V
0101 = 22.0V
0110 = 24.0V
0111 = 26.0V
1000 = 28.0V
1001 = 30.0V
1010 = 32.0V
1011 = 34.0V
1100 = 36.0V
1101 = 38.0V
1110 = 40.0V
1111 = 42.0V

3.3.3 INPUT UNDER/OVERVOLTAGE CONTROL REGISTER

The VINCON register is the comparator control register for both the input undervoltage lockout and input overvoltage lockout. It contains the enable bits, the polarity edge detection bits and the status output bits for both protection circuits. The interrupt flags <UVLOIF> and <OVLOIF> in the PIR2 register are independent of the enable <UVLOEN> and <OVLOEN> bits in the VINCON register. The <UVLOOUT> undervoltage lockout status output bit in the VINCON register indicates if an UVLO event has occurred. The <OVLOOUT> overvoltage lockout status output bit in the VINCON register indicates if an OVLO event has occurred.

When the input voltage on the V_{IN} pin to the MCP19122/3 is below the threshold programmed by the VINUVLO register and the <UVLOEN> bit is set, both the HDRV and LDRV gate drivers are disabled.

When the input voltage on the V_{IN} pin to the MCP19122/3 is above the threshold programmed by the VINOVLO register and the <OVLOEN> bit is set, both the HDRV and LDRV gate drivers are disabled.

REGISTER 3-3: VINCON: INPUT VOLTAGE UVLO AND OVLO CONTROL REGISTER

R/W-0	R-0	R/W-0	R/W-0	R/W-0	R-0	R/W-0	R/W-0
UVLOEN	UVLOOUT	UVLOINTP	UVLOINTN	OVLOEN	OVLOOUT	OVLOINTP	OVLOINTN
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 7 **UVLOEN:** UVLO Comparator Module Logic Enable bit
1 = UVLO Comparator Module Logic enabled
0 = UVLO Comparator Module Logic disabled

bit 6 **UVLOOUT:** Undervoltage Lock Out Status bit
1 = UVLO event has occurred
0 = UVLO event has not occurred

bit 5 **UVLOINTP:** UVLO Comparator Interrupt-on-Positive Going Edge Enable bit
1 = UVLOIF will be set upon a positive going edge of the UVLO
0 = No UVLOIF will be set upon a positive going edge of the UVLO

bit 4 **UVLOINTN:** UVLO Comparator Interrupt on Negative Going Edge Enable bit
1 = UVLOIF will be set upon a negative going edge of the UVLO
0 = No UVLOIF will be set upon a negative going edge of the UVLO

bit 3 **OVLOEN:** OVLO Comparator Module Logic Enable bit
1 = OVLO Comparator Module Logic enabled
0 = OVLO Comparator Module Logic disabled

bit 2 **OVLOOUT:** Overvoltage Lock Out Status bit
1 = OVLO event has occurred
0 = OVLO event has not occurred

bit 1 **OVLOINTP:** OVLO Comparator Interrupt on Positive Going Edge Enable bit
1 = OVLOIF will be set upon a positive going edge of the OVLO
0 = No OVLOIF will be set upon a positive going edge of the OVLO

bit 0 **OVLOINTN:** OVLO Comparator Interrupt on Negative Going Edge Enable bit
1 = OVLOIF will be set upon a negative going edge of the OVLO
0 = No OVLOIF will be set upon a negative going edge of the OVLO

3.4 Output Overcurrent

The MCP19122/3 features a cycle-by-cycle peak current limit. By monitoring the OCIF interrupt flag, custom over current fault handling can be implemented.

To detect an output overcurrent, the MCP19122/3 senses the voltage drop across the high-side MOSFET while it is conducting. Leading-edge blanking is incorporated to mask the overcurrent measurement for a given amount of time. This helps prevent false overcurrent readings.

When an output overcurrent is sensed, the OCIF flag is set and the high-side drive signal is immediately terminated. Without any custom overcurrent handling implemented, the high-side drive signal will be asserted high at the beginning of the next clock cycle. If the overcurrent condition still exists, the high-drive signal will again be terminated.

The OCIF interrupt flag must be cleared in software. It can only be cleared once a switching cycle without an overcurrent condition has occurred.

Register OCCON contains the bits used to configure both the output overcurrent limit and the amount of leading edge blanking (see Register 3-4).

The OCCON<OCEN> bit must be set to enable the input overcurrent circuitry.

Note:	The OCIF interrupt flag bit is set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register.

REGISTER 3-4: OCCON: OUTPUT OVERCURRENT CONTROL REGISTER

R/W-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
OCEN	OCLEB1	OCLEB0	OOC4	OOC3	OOC2	OOC1	OOC0
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 7 **OCEN:** Output Overcurrent Control bit
1 = Output Overcurrent comparator is enabled
0 = Output Overcurrent comparator is disabled

bit 6-5 **OCLEB<1:0>:** Leading Edge Blanking
00 = 110 ns blanking
01 = 200 ns blanking
10 = 380 ns blanking
11 = 740 ns blanking

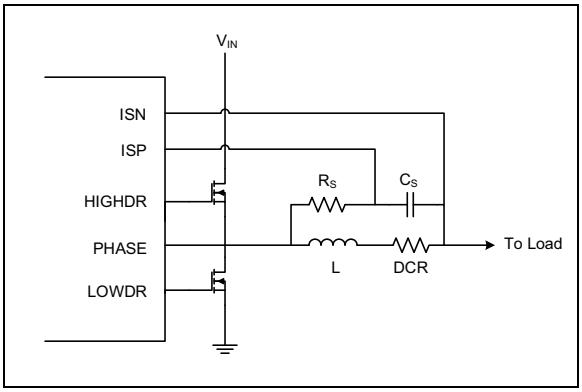
bit 4-0 **OOC<4:0>:** Output Overcurrent Configuration bits
00000 = 91 mV drop
00001 = 112 mV drop
00010 = 134 mV drop
00011 = 155 mV drop
00100 = 177 mV drop
00101 = 198 mV drop
00110 = 220 mV drop
00111 = 241 mV drop
01000 = 263 mV drop
01001 = 284 mV drop
01010 = 306 mV drop
01011 = 327 mV drop
01100 = 350 mV drop
01101 = 370 mV drop
01110 = 392 mV drop
01111 = 413 mV drop
10000 = 435 mV drop
10001 = 456 mV drop
10010 = 478 mV drop
10011 = 500 mV drop
10100 = 521 mV drop
10101 = 542 mV drop
10111 = 585 mV drop
11000 = 607 mV drop
11001 = 628 mV drop
11010 = 650 mV drop
11011 = 671 mV drop
11100 = 693 mV drop
11101 = 714 mV drop
11110 = 736 mV drop
11111 = 757 mV drop

3.5 Current Sensing

The system output current can be sensed by using either a low value resistor placed in series with the output or for applications that require the highest possible efficiency the series resistance (DCR) of the inductor.

For applications that use DCR sensing, a resistor in series with a capacitor are placed around the inductor, as shown in Figure 3-1. If the value of R_S and C_S are chosen so the RC time constant matches the inductor time constant, the voltage appearing across C_S will equal the voltage across the DCR and therefore the current flowing through the inductor. Equation 3-3 can be used to select R_S and C_S.

FIGURE 3-1: INDUCTOR CURRENT SENSE FILTER



EQUATION 3-3: CALCULATING FILTER VALUES

$$\frac{L}{DCR} = (R_S \times C_S)$$

Where:

- L is the inductance value of the output inductor
- DCR is the series resistance of the output inductor
- R_S is the current sense filter resistor
- C_S is the current sense filter capacitor

3.5.1 CURRENT SENSE GAIN

The entire current sense path has a fixed gain of 32. Additional gain or attenuation can be added. The amount added is controlled by the CSGSCON register, Register 3-5. The gain added to this current sense signal does not change the +6 dB of current sense gain added before being read by the A/D.

3.5.2 INDUCTOR OR SENSE RESISTOR SELECTION

The DCR of the inductor or the value of the sense resistor are to be selected so the output of the internal current sense amplifier output does not exceed 3.0V at full load current. The internal current sense amplifier has a fixed gain of 32. See Equation 3-4.

EQUATION 3-4: SENSE ELEMENT RESISTANCE

$$R_{SENSE} = \frac{AMP_{VOUT}}{AMP_{GAIN} \times I_{MAX}}$$

Where:

- R_{SENSE} is the resistance of the sense element
- AMP_{VOUT} is the maximum output voltage of the current sense amplifier
- AMP_{GAIN} is the fixed gain of the current sense amplifier
- I_{MAX} is the maximum application load current

3.5.3 MEASURING SYSTEM LOAD CURRENT

The system load current can be measured by the internal ADC. Before being measured by the ADC, the sampled current is gained by a fixed +6 dB. It is recommended that multiple ADC readings of the sampled current be taken and averaged together to provide a more uniform measurement.

REGISTER 3-5: CSGSCON: CURRENT SENSE GAIN CONTROL REGISTER

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	CSGS4	CSGS3	CSGS2	CSGS1	CSGS0
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 7-5 **Unimplemented:** Read as '0'

bit 4-0 **CSGS<4:0>:** Current Sense Gain Setting bits

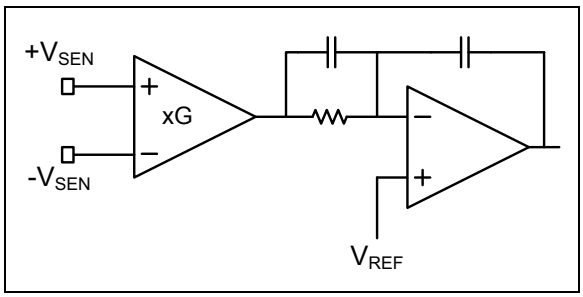
000000 = -3.0 dB
000001 = -2.8 dB
000010 = -2.6 dB
000011 = -2.4 dB
000100 = -2.2 dB
000101 = -2.0 dB
000110 = -1.8 dB
000111 = -1.6 dB
001000 = -1.4 dB
001001 = -1.2 dB
001010 = -1.0 dB
001011 = -0.8 dB
001100 = -0.6 dB
001101 = -0.4 dB
001110 = -0.2 dB
001111 = 0.0 dB
010000 = 0.2 dB
010001 = 0.4 dB
010010 = 0.6 dB
010011 = 0.8 dB
010100 = 1.0 dB
010101 = 1.2 dB
010110 = 1.4 dB
010111 = 1.6 dB
011000 = 1.8 dB
011001 = 2.0 dB
011010 = 2.2 dB
011011 = 2.4 dB
011100 = 2.6 dB
011101 = 2.8 dB
011110 = 3.0 dB
011111 = 3.2 dB

3.6 Control Parameters

3.6.1 COMPENSATION SETTING

The MCP19122/3 is an emulated current mode controller with integrated compensation. The desired response of the overall loop can be tuned by proper placement of the compensation zero frequency and gain. The CMPZCON register, Register 3-6, is used to adjust the compensation zero frequency and gain. Figure 3-2 shows a simplified drawing of the internal compensation with and the adjustable gain differential amplifier.

FIGURE 3-2: SIMPLIFIED COMPENSATION



REGISTER 3-6: CMPZCON: COMPENSATION SETTING CONTROL REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CMPZF3	CMPZF2	CMPZF1	CMPZF0	CMPZG3	CMPZG2	CMPZG1	CMPZG0
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 7-4 **CMPZF<3:0>:** Compensation Zero Frequency Setting bits
0000 = 1500 Hz
0001 = 1850 Hz
0010 = 2300 Hz
0011 = 2840 Hz
0100 = 3460 Hz
0101 = 4300 Hz
0110 = 5300 Hz
0111 = 6630 Hz
1000 = 8380 Hz
1001 = 9950 Hz
1010 = 12200 Hz
1011 = 14400 Hz
1100 = 18700 Hz
1101 = 23000 Hz
1110 = 28400 Hz
1111 = 35300 Hz

bit 3-0 **CMPZG<3:0>:** Compensation Gain Setting bits
0000 = 30.13 dB
0001 = 27.73 dB
0010 = 24.66 dB
0011 = 22.41 dB
0100 = 20.08 dB
0101 = 17.78 dB
0110 = 15.42 dB
0111 = 13.06 dB
1000 = 10.75 dB
1001 = 8.30 dB
1010 = 6.02 dB
1011 = 3.52 dB
1100 = 1.21 dB
1101 = –1.41 dB
1110 = –3.74 dB
1111 = –6.02 dB

3.6.2 SLOPE COMPENSATION RAMP

The difference between the average inductor current and the DC value of the sampled inductor current can cause instability for certain operating conditions. This instability occurs when the inductor ripple current does not return to its initial value by the start of the next switching cycle. Adding slope compensation ramp to the current sense signal prevents this oscillation. The amount of slope added is controlled by the RAMPCON register, Register 3-7.

Note 1: To enable the slope compensation circuitry, the RAMPCON<RAMPEN> bit must be cleared.

REGISTER 3-7: RAMPCON: COMPENSATION RAMP CONTROL REGISTER

R/W-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RMPEN	—	—	RMP4	RMP3	RMP2	RMP1	RMP0
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 7 **RMPEN:** Compensation Ramp Disable bit
1 = Compensation ramp is disabled
0 = Compensation ramp is enabled

bit 6-5 **Unimplemented:** Read as '0'

bit 4-0 **RMP<4:0>:** Compensation Ramp Configuration bits
RMP<4:0> = (dV/dt * 200/V_{IN}); Where dV/dt is in V/µs

3.7 Determining System Output Voltage

3.7.1 REFERENCE VOLTAGE CONFIGURATION

The control system reference voltage is determined by the setting contained in the 10-bit V_{REF} DAC. The system reference is adjustable in 2 mV typical increments. The configuring of this DAC is accomplished by the settings contained in the VOUTH and VOUTL registers. See Equation 3-5 for more information.

Note 1: To enable the slope compensation circuitry, the RAMPCON<RAMPEN> bit must be cleared.

See Section 4.0, Electrical Characteristics for more information regarding the DAC specification.

REGISTER 3-8: VOUTL: OUTPUT VOLTAGE SET POINT LSB CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
VOUT7	VOUT6	VOUT5	VOUT4	VOUT3	VOUT2	VOUT1	VOUT0
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 7-0 **VOUT<7:0>:** Output Voltage Set Point LSB Configuration bits

REGISTER 3-9: VOUTH: OUTPUT VOLTAGE SET POINT MSB CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	VOUT9	VOUT8
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 7-2 **Unimplemented:** Read as '0'

bit 1-0 **OVOUT<9:8>:** Output Voltage Set Point MSB Configuration bits

3.7.2 DIFFERENTIAL AMPLIFIER GAIN CONTROL

The MCP19122/3 contains a low offset programmable gain differential amplifier used for remote sensing of the output voltage. Connect the $+V_{SEN}$ and $-V_{SEN}$ pins directly at the load for better load regulation. The $+V_{SEN}$ and $-V_{SEN}$ are the positive and negative inputs, respectively, of the programmable gain differential amplifier.

The programmable gain settings are controlled by the DAGCON register, Register 3-10.

EQUATION 3-5: SYSTEM OUTPUT VOLTAGE

$$V_{REFDAC} = \frac{V_{OUT}}{DA_{GAIN} \times DAC_{STEP}}$$

Where:

- V_{REFDAC} is the concatenated decimal value of VOUTH and VOUTL
- DA_{GAIN} is the programmable gain of the differential amplifier.
- DAC_{Step} is the volts/step of the reference voltage DAC, typically 2 mV/step
- V_{OUT} is the desired output voltage

Note 1: If the hexadecimal V_{REFDAC} value calculated is larger than 10-bits, the programmable gain differential amplifier gain must be adjusted.

REGISTER 3-10: DAGCON: DIFFERENTIAL AMPLIFIER GAIN CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	DAG2	DAG1	DAG0
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 7-3 **Unimplemented:** Read as '0'

bit 2-0 **DAG<2:0>:** Differential Amplifier Gain control bit

000 = Gain of 1
001 = Gain of 1/2
010 = Gain of 1/4
011 = Gain of 1/8
100 = Gain of 2
101 = Gain of 4
110 = Gain of 8
111 = Gain of 1

3.8 System Output Voltage Protection

The MCP19122/3 provides the option for hardware and/or software protection for a system output under voltage as well as a system output over voltage.

3.8.1 OUTPUT UNDERVOLTAGE

The output voltage is monitored and compared to an adjustable undervoltage (UV) reference. When the output voltage is below UV reference the PIR2<UVIF> flag is set. If the hardware UV accelerator response circuitry (see Register 3-14) is enabled, the high-side MOSFET is turned on until the maximum programmed duty cycle is reached. Then the low-side MOSFET is turned on for the remainder of the switching period.

Once the output voltage is above the UV reference the MCP19122/3 returns to normal operation. The UVIF flag must be cleared in software.

The output undervoltage reference is controlled by the VOTUVLO register, Register 3-11. A fixed voltage is subtracted from the adjustable system output voltage reference.

Note 1: The system output voltage reference is determined by the setting in the VOUTH and VOUTL registers.

2: The UVIF interrupt flag bit is set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register.

REGISTER 3-11: VOTUVLO: OUTPUT UNDER VOLTAGE DETECT LEVEL CONTROL REGISTER

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	OUV3	OUV2	OUV1	OUV0
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 7-4 **Unimplemented:** Read as '0'

bit 3-0 **OUV<3:0>:** Output Under Voltage Detect Level Configuration bits

0000 = V_{REF} - 50 mV
0001 = V_{REF} - 80 mV
0010 = V_{REF} - 110 mV
0011 = V_{REF} - 140 mV
0100 = V_{REF} - 170 mV
0101 = V_{REF} - 200 mV
0110 = V_{REF} - 230 mV
0111 = V_{REF} - 260 mV
1000 = V_{REF} - 290 mV
1001 = V_{REF} - 320 mV
1010 = V_{REF} - 350 mV
1011 = V_{REF} - 380 mV
1100 = V_{REF} - 410 mV
1101 = V_{REF} - 440 mV
1110 = V_{REF} - 470 mV
1111 = V_{REF} - 500 mV

3.8.2 OUTPUT OVER VOLTAGE

The output voltage is monitored and compared to an adjustable over voltage (OV) reference. When the output voltage is above OV reference the PIR2<OVIF> flag is set. If the hardware OV accelerator response circuitry, see Register 3-14, is enabled the high-side and low-side MOSFETs are turned off until the output voltage fails below the output over voltage reference. Once the output voltage is below the OV reference the MCP19122/3 returns to normal operation. The OVIF flag must be cleared in software.

The output under voltage reference is controlled by the VOTOVLO register, Register 3-12. A fixed voltage is added to the adjustable system output voltage reference.

Note 1: The system output voltage reference is determined by the setting in the VOUTH and VOUTL registers.

2: The OVIF interrupt flag bit is set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register.

REGISTER 3-12: VOTOVLO: OUTPUT OVER VOLTAGE DETECT LEVEL CONTROL REGISTER

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	OOV3	OOV2	OOV1	OOV0
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 7-4 **Unimplemented:** Read as ‘0’

bit 3-0 **OOV<3:0>:** Output Over Voltage Detect Level Configuration bits

0000 = V_{REF} + 50 mV
0001 = V_{REF} + 80 mV
0010 = V_{REF} + 110 mV
0011 = V_{REF} + 140 mV
0100 = V_{REF} + 170 mV
0101 = V_{REF} + 200 mV
0110 = V_{REF} + 230 mV
0111 = V_{REF} + 260 mV
1000 = V_{REF} + 290 mV
1001 = V_{REF} + 320 mV
1010 = V_{REF} + 350 mV
1011 = V_{REF} + 380 mV
1100 = V_{REF} + 410 mV
1101 = V_{REF} + 440 mV
1110 = V_{REF} + 470 mV
1111 = V_{REF} + 500 mV

3.9 Internal Synchronous Driver

The internal synchronous driver is capable of driving two N-Channel MOSFETs in a synchronous rectified buck converter topology. The gate of the floating MOSFET is connected to the HDRV pin. The source of this MOSFET is connected to the PHASE pin. This pin is capable is sourcing and sinking 2A of peak current.

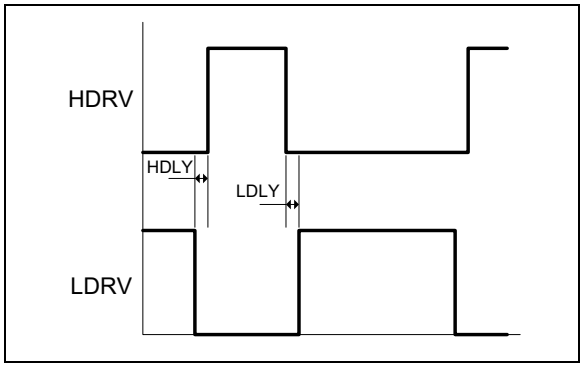
The MOSFET connected to the LDRV pin is not floating. The low-side MOSFET gate is connected to the LDRV pin and the source of this MOSFET is connected to P_{GND}. This pin is capable of sourcing a peak current of 2A. The peak sink current is 4A. This helps keep the low-side MOSFET off when the high-side MOSFET is turning on.

3.9.1 MOSFET DRIVER DEAD TIME

The MOSFET driver dead time is defined as the time between one drive signal going low and the complimentary drive signal going high. Refer to Figure 3-3. The MCP19122/3 has the capability to adjust both the high-side and low-side driver dead time independently. The adjustment of the driver dead time is controlled by the DEADCON register.

Note 1:	The DEADCON register controls the amount of dead time added to the HDRV or LDRV signal.

FIGURE 3-3: MOSFET DRIVER DEAD TIME



3.9.2 MOSFET DRIVER CONTROL

The MCP19122/3 has the ability to independently disable high-side or low-side driver circuitry. This control of the HDRV or LDRV signal is accomplished by setting or clearing the HIDIS or LODIS bits in the PE1 register. When either driver is disabled, the output signal is set low. The default power-on or reset state is to have the high-side and low-side drivers disabled.

3.10 High-Side MOSFET Driver Supply

A floating voltage is required by the high-side driver. An external bootstrap capacitor connected between the BOOT and PHASE pins supplies this gate drive voltage. This capacitor is charged by internally connecting the BOOT pin to V_{DD} when the PHASE pin is low.

The selection of the bootstrap capacitor is based upon the total charge of the high-side power MOSFET and the allowable droop in the voltage applied to the gate of the high-side power MOSFET.

EQUATION 3-6: BOOTSTRAP CAPACITOR

$$C_{BOOT} = \frac{Q_{G(Total)}}{\Delta V_{DROOP}}$$

Where:

- $Q_{G(Total)}$ = High-side MOSFET Total Gate Charge (C)
- V_{DROOP} = Allowable Gate Drive Voltage Droop (V)

REGISTER 3-13: DEADCON: DRIVER DEAD TIME CONTROL REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
HDLY3	HDLY2	HDLY1	HDLY0	LDLY3	LDLY2	LDLY1	LDLY0
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 7-4 **HDLY<3:0>:** High-Side Dead Time Configuration bits
0000 = 14 ns delay
0001 = 18 ns delay
0010 = 22 ns delay
0011 = 26 ns delay
0100 = 30 ns delay
0101 = 34 ns delay
0110 = 38 ns delay
0111 = 42 ns delay
1000 = 46 ns delay
1001 = 50 ns delay
1010 = 54 ns delay
1011 = 58 ns delay
1100 = 62 ns delay
1101 = 66 ns delay
1110 = 70 ns delay
1111 = 74 ns delay

bit 3-0 **LDLY<3:0>:** Low-Side Dead Time Configuration bits
0000 = -4 ns delay
0001 = 0 ns delay
0010 = 4 ns delay
0011 = 8 ns delay
0100 = 12 ns delay
0101 = 16 ns delay
0110 = 20 ns delay
0111 = 24 ns delay
1000 = 28 ns delay
1001 = 32 ns delay
1010 = 36 ns delay
1011 = 40 ns delay
1100 = 44 ns delay
1101 = 48 ns delay
1110 = 52 ns delay
1111 = 56 ns delay

3.11 Analog Peripheral Control

The MCP19122/3 has various analog peripherals. These peripherals can be configured to allow customizable operation. Refer to Register 3-14 more information.

3.11.1 DIODE EMULATION MODE

The MCP19122/3 can operate in either diode emulation or synchronous rectification mode. When operating in diode emulation mode, the LDRV signal is terminated when the voltage across the low-side MOSFET is approximately 0V. This provides better light load efficiency by preventing reverse current from flowing through the inductor. Both the HDRV and LDRV signals are low until the beginning of the next switching cycle. At that time, the HDRV signal is asserted high, turning on the high-side MOSFET.

The PE1<DECON> bit controls the diode emulation operating mode of the MCP19122/3.

3.11.2 HIGH-SIDE DRIVER CONTROL

The high-side driver is enabled by clearing the PE1<HIDIS> bit. Setting this bit disables the high-side driver.

Note:	The HIDIS bit is reset to '1' so the high-side driver is in a known state after reset. This bit must be cleared by software for normal operation.

3.11.3 LOW-SIDE DRIVE CONTROL

The low-side driver is enabled by clearing the PE1<LODIS> bit. Setting this bit disables the low-side driver.

Note:	The LODIS bit is reset to '1' so the low-side driver is in a known state after reset. This bit must be cleared by software for normal operation.

3.11.4 OUTPUT UNDERVOLTAGE ACCELERATOR

The MCP19122/3 has additional control circuitry to allow it to respond quickly to an output undervoltage condition. The enabling of this circuitry is handled by the PE1<UVTEE> bit. When this bit is set, the MCP19122/3 will respond to an output under voltage condition by turning on the high-side MOSFET until the output voltage is above the output undervoltage threshold set by the OUVCON register. The low-side MOSFET is off during this time.

The undervoltage reference is controlled by the VOTUVLO register, Register 3-11.

3.11.5 OUTPUT OVER VOLTAGE ACCELERATOR

The MCP19122/3 has additional control circuitry to allow it to respond quickly to an output overvoltage condition. The enabling of this circuitry is handled by the PE1<OVTEE> bit. When this bit is set, the MCP19122/3 will respond to an output overvoltage condition by turning off the high-side and low-side MOSFETs until the output voltage is below the output overvoltage threshold set by the OOVCON register.

The overvoltage reference is controlled by the VOTOVLO register, Register 3-12.

3.11.6 RELATIVE EFFICIENCY RAMP MEASUREMENT CONTROL

The PE1<SPAN> bit determines what portion of the Relative Efficiency Measurement timing ramp is connected to A/D channel 0x08h. When the PE1<MEASEN> bit is low and the PE1<SPAN> bit is low, the RELEFF channel of the A/D (channel 0x08h) will measure the valley of the relative efficiency timing ramp. When the PE1<MEASEN> is low and the PE1<SPAN> bit is hit, the RELEFF channel of the A/D will measure the peak of the relative efficiency timing ramp.

REGISTER 3-14: PE1: ANALOG PERIPHERAL ENABLE 1 CONTROL REGISTER

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
DECON	TOPO	HIDIS	LODIS	MEASEN	SPAN	UVTEE	OVTEE
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 7 **DECON:** Diode Emulation Mode bit
1 = Diode emulation mode enabled
0 = Synchronous rectification mode enabled

bit 6 **TOPO:** Topology selection control bit
1 = Boost topology is enabled
0 = Buck topology is enabled

bit 5 **HIDIS:** High-side driver control bit
1 = High-side driver is disabled
0 = High-side driver is enabled

bit 4 **LODIS:** Low-side driver control bit
1 = Low-side driver is disabled
0 = Low-side driver is enabled

bit 3 **MEASEN:** Relative efficiency measurement control bit
1 = Initiate relative efficiency measurement
0 = Relative efficiency measurement not in progress

bit 2 **SPAN:** Relative efficiency ramp measurement control bit
1 = A/D channel 0x08h measures the peak of the RELEFF signal
0 = A/D channel 0x08h measures the valley of the RELEFF signal

bit 1 **UVTEE:** Output Undervoltage Accelerator Enable bit
1 = Output undervoltage accelerator is enabled
0 = Output undervoltage accelerator is disabled

bit 0 **OVTEE:** Output Overvoltage Accelerator Enable bit
1 = Output overvoltage accelerator is enabled
0 = Output overvoltage accelerator is disabled

3.12 System Configuration Control

The MCP19122/3 is capable of operating in a variety of different configurations. The MODECON register controls the system configuration of the MCP19122/3.

3.12.1 ERROR AMPLIFIER CLAMP

The internal error amplifier is a rail-to-rail amplifier. However, the output of the error amplifier can be clamped to an adjustable level. The MODECON<EACLMP> is the error amplifier clamp control bit. Setting this bit enables the error amplifier clamp. The bit is cleared on a reset making the error amplifier clamp disabled by default.

The error amplifier clamp source is controlled by the MODECON<CLMPSEL> bit. When this bit is set, the voltage present on GPA3 will set the error amplifier clamp voltage.

The clamp voltage can also be determined by a combination of the GPA3 pin voltage and the sampled output current. This is achieved by clearing the MODECON<CLMPSEL> bit.

Note:	The GPA3 pin can be configured as a weak current source by setting the WPUGPA<WCS1> bit. See Register 17-3.

3.12.2 INTERNAL PEDESTAL VOLTAGE

To improve accuracy at low voltages, a pedestal voltage is implemented throughout the analog control loop. This voltage is typically 500 mV and can be enabled or disabled by the MODECON<VGNDEN> bit. This pedestal voltage is enabled on any reset. The VGNDEN bit must be cleared to disable the pedestal.

When the MCP19122/3 is disabled, the system output voltage may float up. If the pedestal voltage is also disabled under this condition, the output voltage will not float up. It is recommended that the pedestal voltage always be enabled while operating the MCP19122/3. Operating with the pedestal voltage disabled may cause the MCP19122/3 to not meet all of the electrical specifications contained in the specification table. See **Section 4.0 "Electrical Characteristics"**.

3.12.3 VDD LDO CONTROL

The VDDEN bit controls the state of the internal 5V VDD LDO when the SLEEP command is issued to the microcontroller core. If the VDDEN bit is set and the SLEEP command is issued the 5V VDD LDO will remain operational and capable of supporting a load.

When the SLEEP command is issued and the VDDEN bit is clear, the 5V VDD LDO will be commanded to a low current consumption state. The voltage will drop to approximately 3V and will not be able to supply any external current.

3.12.4 VOLTAGE/CURRENT SOURCE

The system output voltage or load current can be controlled by the MCP19122/3. The MODECON<CNSG> is the control system configuration bit. When the CNSG bit is cleared, the MCP19122/3 functions as a voltage source. The system output voltage is regulated by comparing the sensed voltage to the adjustable reference voltage.

When the CNSG bit is set, the MCP19122/3 is configured to control the system output current. The output current is regulated by adjusting the high-side duty cycle according to the amount of error that exists between the sampled load current and the adjustable reference.

3.12.5 MULTIPLE OUTPUT SYSTEM CONFIGURATION

The MCP19122/3 is capable of being configured as a Master or Slave in a multiple output system. The device configuration is set by the MODECON<MSC2:0> bits.

3.12.5.1 Multiple Output System Master

When configured as a Master, the GPA1 pin is automatically set to output the switching frequency synchronization signal. The frequency of this synchronization signal sets the converters switching frequency. The GPB3 pin is automatically set to output the system clock signal frequency of 8 MHz.

3.12.5.2 Multiple Output System Slave

The are two different multiple output Slave configurations. The first configuration, MSC<2:0> = 010, requires the Slave to receive a switching frequency synchronization signal and system clock signal from a Master device. For this configuration GPA1 and GPB3 are automatically set as an input. The switching frequency synchronization signal from the Master is connected to the GPA1 pin. Phase shift from this synchronization signal can be applied. See Equation 26-2. The 8 MHz system clock from the Master is connected to the GPB3 pin. This multiple output Slave mode results in less switching waveform frequency jitter when compared to the Master's switching waveforms.

The second multiple output system Slave configuration, MSC<2:0> = 110, requires the Slave to only receive a switching frequency synchronization signal from the Master. GPA1 is automatically set as an input and the switching frequency synchronization signal from the Master is connected to this pin. Phase shift

from the synchronization signal can be applied. See Equation 26-2. No system clock is required in this mode.

Note 1:	The TMR2 register should be initialized to 0 to allow proper synchronization.
2:	The PWMPHL and PWMPHH register control the amount of phase shift applied.

3.12.6 MULTI-PHASE SYSTEM

The MCP19122/3 is capable of being configured as a Master or Slave in a multi-phase system. The MODECON<MSC2:0> bits determine the device configuration.

3.12.6.1 Multi-phase System Master

When configured as a Master, the GPA1 pin is automatically set to output the switching frequency synchronization signal. The frequency of this synchronization signal sets the converters switching frequency. The GPB3 pin is automatically set to output the system clock signal frequency of 8 MHz. The GPB1 pin is automatically set to output the sampled current sense signal.

3.12.6.2 Multi-phase System Slave

When configured as a Slave, GPA1 is set to be the switching frequency synchronization signal input pin. The Master's switching frequency synchronization signal is to be connected to this pin. Phase shift from the synchronization signal can be applied. See Register 26-2.

To ensure proper synchronization between the Master and Slave, GPB3 of the Slave is set to be the system clock input pin. Both the Master and Slave GPB3 pins must be connected together.

To properly balance the system output current between the phases, all devices need to regulate to the same current. On the Slave devices, GPB1 is set to be the current sense input signal from the Master. Both the Master and Slave GPB1 pins must be connected together. The MODECON<CNSG> must also be set to a '1' so the Slave device is set to control the system output current.

Note 1:	The TMR2 register should be initialized to 0 to allow proper synchronization.
2:	The PWMPHL and PWMPHH register control the amount of phase shift applied.

REGISTER 3-15: MODECON: SYSTEM CONFIGURATION CONTROL REGISTER

R/W-0	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CLMPSEL	VGNDEN	VDDEN	CNSG	EACLMP	MSC2	MSC1	MSC0
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 7 **CLMPSEL:** Error Amplifier Clamp Configuration bit
1 = EA Clamp current set by GPA3 pin voltage
0 = EA Clamp current set by GPA3 pin voltage and average output current

bit 6 **VGNDEN:** Virtual Ground Control bit
1 = Virtual Ground is enabled
0 = Virtual Ground is disabled

bit 5 **VDDEN:** V_{DD} LDO control bit
0 = V_{DD} LDO is disabled when SLEEP command issued
1 = V_{DD} LDO remains enabled when SLEEP command issued

bit 4 **CNSG:** Control Signal configuration bit
0 = Device set to control system output voltage
1 = Device set to control system output current

bit 3 **EACLMP:** Error Amplifier Clamp control bit
1 = Error amplifier output clamp enabled
0 = Error amplifier output clamp disabled

bit 2-0 **MSC<2:0>:** System configuration control bit
000 = Device set as a stand alone unit
001 = Device set as multiple output Master (GPB3: clock signal out, GPA1: synchronization signal out)
010 = Device set as multiple output Slave (GPB3: clock signal in, GPA1:synchronization signal in)
011 = Device set as multi-phase Master (GPB3: clock signal out,GPA1: synchronization signal out, GPB1: demand out)
100 = Device set as multi-phase Slave (GPB3: clock signal in, GPA1: synchronization signal in, GPB1: demand in)
101 = Device set as multiple output Master (GPA1: synchronization signal out)
110 = Device set as multiple output Slave (GPA1: synchronization signal in)
111 = Unimplemented

3.13 Miscellaneous Features

3.13.1 OVERTEMPERATURE

The MCP19122/3 features a hardware overtemperature shutdown protection typically set at +160°C. No firmware fault-handling procedure is required to shutdown the MCP19122/3 for an over temperature condition.

3.13.2 DEVICE ADDRESSING

The communication address of the MCP19122/3 is stored in the SSPADD register. This value can be loaded when the device firmware is programmed or configured by external components. By reading a voltage on a GPIO with the ADC, a device specific address can be stored into the SSPADD register.

The MCP19122/3 contains a second address register, SSPADD2. This is a 7-bit address that can be used as the SMBus alert address when PMBus communication is used. See Section 27.0, Master Synchronous Serial Port (MSSP) Module for more information.

3.13.3 DEVICE ENABLE

A GPIO pin can be configured to be a device enable pin. By configuring the pin as an input, the PORT register or the interrupt on change (IOC) can be used to enable the device. Example 3-2 shows how to configure a GPIO as an enable pin by testing the PORT register.

EXAMPLE 3-2: CONFIGURING GPA3 AS DEVICE ENABLE

```
    BANKSEL     TRISGPA
    BSF         TRISGPA, 3      ;Set GPA3 as input
    BANKSEL     ANSELA
    BCF         ANSELA, 3       ;Set GPA3 as digital input
    :
    :                           ;Insert additional user code here
    :
WAIT_ENABLE:
    BANKSEL     PORTGPA
    BTFSS       PORTGPA, 3      ;Test GPA3 to see if pulled high
                                ;A high on GPA3 indicated device to be enabled
    GOTO        WAIT_ENABLE     ;Stay in loop waiting for device enable
    BANKSEL     ATSTCON
    BSF         ATSTCON, 0      ;Enable the device by enabling drivers
    :
    :                           ;Insert additional code here
    :
```

3.13.4 OUTPUT POWER GOOD

The output voltage measured between the $+V_{SEN}$ and $-V_{SEN}$ pins can be monitored by the internal ADC. In firmware, when this ADC reading matches a user-defined power good value, a GPIO can be toggled to indicate the system output voltage is within a specified range. Delays, hysteresis and time out values can all be configured in firmware.

3.13.5 OUTPUT VOLTAGE SOFT-START

During start-up, soft start of the output voltage is accomplished in firmware. By using one of the internal timers and incrementing the OVCCON or OVFCON register on a timer overflow, very long soft start times can be achieved.

3.13.6 OUTPUT VOLTAGE TRACKING

The MCP19122/3 can be configured to track another voltage signal at start-up or shutdown. The ADC is configured to read a GPIO that has the desired tracking voltage applied to it. The firmware then handles the tracking of the internal output voltage reference to this ADC reading.

4.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings (†)

$V_{IN} – V_{GND}$ .....–0.3V to +43V
$V_{IN} – V_{GND}$ (non-switching).....–0.3V to +48V
$V_{BOOT} – V_{IN}$ .....–0.3V to $+V_{DDMAX}$V
V_{PHASE} (continuous) ..... GND – 0.3V to $+V_{IN}$V
V_{PHASE} (transient < 100 ns)..... GND – 5.0V to $+V_{IN}$V
V_{DD} internally generated.....+5V ±8%
V_{HDRV}, HDRV Pin.....$+V_{PHASE}$ – 0.3V to V_{BOOT} + 0.3V
V_{LDRV}, LDRV Pin..... $+(V_{GND} – 0.3V)$ to $(V_{DD} + 0.3V)$
Voltage on $\overline{MCLR}$ with respect to GND .....–0.3V to +13.5V
$+V_{SEN}$, ISP, ISN pins.....$(V_{GND}– 0.3V)$ to +16V
Maximum Voltage: any other pin..... $+(V_{GND} – 0.3V)$ to $(V_{DD} + 0.3V)$
Maximum output current sunk by any single I/O pin .....25 mA
Maximum output current sourced by any single I/O pin .....25 mA
Maximum current sunk by all GPIO .....65 mA
Maximum current sourced by all GPIO .....45 mA
ESD protection on all pins (HBM) ..... 1.0 kV
ESD protection on all pins (MM)..... 200 V

† **Notice:** Stresses above those listed under “Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

Electrical Specifications: Unless otherwise noted, V_{IN} = 12V, F_{SW} = 300 kHz, T_A = +25°C. **Boldface** specifications apply over the T_A range of –40°C to +125°C

Parameter	Sym.	Min.	Typ.	Max.	Units	Conditions
Input						
Input Voltage	V_{IN}	**4.5**	—	**40**	V	
Input Quiescent Current	I_Q	—	5	**10**	mA	Not Switching, $+V_{SEN} > V_{REF}$ MODECON<VGNDEN> = 0
Input Shutdown Current	I_{SHDN}	—	50	**150**	µA	SLEEP Command, V_{DD} disabled
Internal Regulator V_{DD}						
Bias Voltage	V_{DD}	**4.6**	5.0	**5.4**	V	Vin = 6V to 40V
Maximum external V_{DD} output current	I_{DD_OUT}	**35**			mA	Vin = 6V to 40V
Line Regulation	$\Delta V_{DD-OUT}/(V_{DD-OUT}*\Delta V_{IN})$	—	0.1	**0.45**	%/V	$(V_{DD-OUT}+1.0V) \leq V_{IN} \geq 40V$ Note 2
Load Regulation	$\Delta V_{DD-OUT}/V_{DD-OUT}$	**–0.5**	±0.1	**+0.5**	%	I_{DD-OUT} = 1 mA to 20 mA Note 2
Output Short Circuit Current	I_{DD-OUT_SC}	—	45	—	mA	$V_{IN} = (V_{DD-OUT} + 1.0V)$ Note 2
Dropout Voltage	$V_{IN} - V_{DD-OUT}$	—	0.5	**1**	V	I_{DD-OUT} = 35 mA, $V_{IN} = V_{DD-OUT} + 1.0V$ Note 2
Power Supply Rejection Ratio	$PSRR_{LDO}$	—	60	—	dB	f ≤ 1000 Hz, I_{DD-OUT} = 35mA C_{IN} = 0 µF, C_{DD-OUT} = 1 µF
Band Gap						
Band Gap Voltage	BG	**1.205**	1.230	**1.254**	V	
Internal Regulator AV_{DD}						
Internal Analog Supply Voltage	AV_{DD}	**3.97**	4.096	**4.23**	V	
Overcurrent						
Adjustable Overcurrent Range	OC_{MIN}	**90**	—	**755**	mV	
Overcurrent Step Size	OC_{STEP_SIZE}	**15**	21	**25**	mV	
Adjustable Leading Edge Blanking Time	LEB	**90**	110	**135**	ns	
		150	200	**250**	ns	
		250	380	**480**	ns	
		500	740	**950**	ns	
Current Sense						
Current Sense Fixed Gain	I_{SENSE_GAIN}	**30**	32	**34**	V/V	
Current Sense Amplifier Offset	I_{SENSE_OFFSET}	**–10**	**0**	**10**	mV	Fixed gain removed

Note 1: These parameters are characterized but not production tested.

2: V_{DD-OUT} is the voltage present at the V_{DD} pin. V_{DD} is the internally generated bias voltage.

3: This is the voltage measured between the PHASE pin and GND. When measured voltage is between 2.5 mV and +2.5 mV, the LOWDR signal is to be pulled low

4: This is the total source current for all GPIO pins combined. Individually each pin can source a maximum of 15 mA.

5: System output voltage tolerance specified when 1.024V ≤ VREF ≤ 2.046V.

ELECTRICAL CHARACTERISTICS

Electrical Specifications: Unless otherwise noted, V_{IN} = 12V, F_{SW} = 300 kHz, T_A = +25°C. **Boldface** specifications apply over the T_A range of –40°C to +125°C

Parameter	Sym.	Min.	Typ.	Max.	Units	Conditions
Pedestal Voltage	$V_{PEDESTAL}$	—	500	—	mV	
Voltage Reference						
Adjustable V_{OUT} Range	V_{OUT_RANGE}	0.5	—	16	V	V_{OUT} range with no external voltage divider
Reference Voltage Step Size	$VREF_{STEP}$	—	2	—	mV	
System Output Voltage Tolerance	V_{TOL}	**–3%**		**+3%**		Differential amplifier gain setting of 1/8, 1/4, 1/2, 1 and 2 Note 5
System Output Voltage Tolerance	V_{TOL}	**–5%**		**+5%**		Differential amplifier gain setting of 4 Note 5
System Output Voltage Tolerance	V_{TOL}	**–6.5%**		**+6.5%**		Differential amplifier gain setting of 8 Note 5
Output Over Voltage Reference						
Output Over Voltage Step Size	OV_{STEP}	**27**	31	**35**	mV	
Output Under Voltage DAC						
Output Under Voltage Step Size	UV_{STEP}	**27**	31	**35**	mV	
Remote Sense Differential Amplifier						
Input Impedance	R_{IN}	—	12	—	kOhms	
Common Mode Voltage Range	V_{CMR}	–0.5	—	+0.5	V	
Differential Feedback Voltage Range	V_{DIFF}	0		16	V	
Oscillator/PWM						
Internal Oscillator Frequency	F_{OSC}	**7.60**	8.00	**8.40**	MHz	
Switching Frequency	F_{SW}	—	F_{OSC}/N	—	kHz	
Switching Frequency Range Select	N	**4**	—	**80**	—	F_{SW} = 100 kHz to 2 MHz
Maximum Duty Cycle	—	—	(N-1)/N	—	%/100	
Dead Time Adjustment						
LDRV Dead Time Adjustable Range	DT_{RANGE_L}	**-4**	—	**56**	ns	Labeled LDLY in Figure 3-3
HDRV Dead Time Adjustable Range	DT_{RANGE_H}	**14**	—	**74**	ns	Labeled HDLY in Figure 3-3
Dead Time Step Size	DT_{STEP}	—	4	—	ns	For both HDLY and LDLY
HDRV Output Driver						
HDRV Source Resistance	$R_{HIDRV-SCR}$	—	1.4	**2.7**	Ω	Measured at 100mA Note 1

Note 1: These parameters are characterized but not production tested.

2: V_{DD-OUT} is the voltage present at the V_{DD} pin. V_{DD} is the internally generated bias voltage.

3: This is the voltage measured between the PHASE pin and GND. When measured voltage is between 2.5 mV and +2.5 mV, the LOWDR signal is to be pulled low

4: This is the total source current for all GPIO pins combined. Individually each pin can source a maximum of 15 mA.

5: System output voltage tolerance specified when 1.024V ≤ VREF ≤ 2.046V.

ELECTRICAL CHARACTERISTICS

Electrical Specifications: Unless otherwise noted, V_{IN} = 12V, F_{SW} = 300 kHz, T_A = +25°C. **Boldface** specifications apply over the T_A range of –40°C to +125°C

Parameter	Sym.	Min.	Typ.	Max.	Units	Conditions
HDRV Sink Resistance	$R_{HIDRV-SINK}$	—	1.2	**2.5**	Ω	Measured at 100mA Note 1
HDRV Source Current	$I_{HIDRV-SCR}$	—	2	—	A	Note 1
HDRV Sink Current	$I_{HIDRV-SINK}$	—	2	—	A	Note 1
HDRV Minimum On Time	t_{MIN}		75	**95**	ns	Note 1
LDRV Output Driver						
LDRV Source Resistance	$R_{LODRV-SCR}$	—	1.8	**3.2**	Ω	Measured at 100mA Note 1
LDRV Sink Resistance	$R_{LODRV-SINK}$	—	0.6	**2**	Ω	Measured at 100mA Note 1
LDRV Source Current	$I_{LODRV-SCR}$	—	2	—	A	Note 1
LDRV Sink Current	$I_{LODRV-SINK}$	—	4	—	A	Note 1
LDRV Minimum On Time	t_{MIN}	—	167	—	ns	Note 1
Bootstrap Blocking Device						
Blocking Device Resistance	R_{BLOCK}		20	**40**	Ω	Note 1
GPIO Pins						
GPIO Weak Current Source	—	**45**	50.0	**55**	µA	Selected current source on GPA2, GPA3.
Maximum GPIO Sink Current	I_{SINK_GPIO}	—	—	35	mA	Note 1,Note 4
Maximum GPIO Source Current	I_{SOURCE_GPIO}	—	—	35	mA	Note 1,Note 4
GPIO Weak Pull-up Current	$I_{PULL-UP\_GPIO}$	50	250	400	µA	V_{DD} = 5V
GPIO Input Leakage Current	GPIO_I_{IL}	—	±0.1	±1	µA	Negative current is defined as current sourced by the pin. T_A = +90°C
GPIO Input Low Voltage	V_{IL}	GND	—	0.8	V	I/O Port with TTL buffer V_{DD} = 5V, T_A = +90°C
		GND	—	$0.2V_{DD}$	V	I/O Port with Schmitt Trigger buffer, V_{DD} = 5V, T_A = +90°C
		GND	—	$0.2V_{DD}$	V	$\overline{MCLR}$, T_A = +90°C
GPIO Input High Voltage	V_{IH}	2.0	—	V_{DD}	V	I/O Port with TTL buffer V_{DD} = 5V, T_A = +90°C
		$0.8V_{DD}$	—	V_{DD}	V	I/O Port with Schmitt Trigger buffer, V_{DD} = 5V, T_A = +90°C
		$0.8V_{DD}$	—	V_{DD}	V	$\overline{MCLR}$, T_A = +90°C
Thermal Shutdown						
Thermal Shutdown	T_{SHD}	—	160	—	°C	

Note 1: These parameters are characterized but not production tested.

2: V_{DD-OUT} is the voltage present at the V_{DD} pin. V_{DD} is the internally generated bias voltage.

3: This is the voltage measured between the PHASE pin and GND. When measured voltage is between 2.5 mV and +2.5 mV, the LOWDR signal is to be pulled low

4: This is the total source current for all GPIO pins combined. Individually each pin can source a maximum of 15 mA.

5: System output voltage tolerance specified when 1.024V ≤ VREF ≤ 2.046V.

ELECTRICAL CHARACTERISTICS

Electrical Specifications: Unless otherwise noted, V_{IN} = 12V, F_{SW} = 300 kHz, T_A = +25°C. **Boldface** specifications apply over the T_A range of –40°C to +125°C

Parameter	Sym.	Min.	Typ.	Max.	Units	Conditions
Thermal Shutdown Hysteresis	T_{SHD_HYS}	—	20	—	°C	

Note 1: These parameters are characterized but not production tested.

2: V_{DD-OUT} is the voltage present at the V_{DD} pin. V_{DD} is the internally generated bias voltage.

3: This is the voltage measured between the PHASE pin and GND. When measured voltage is between 2.5 mV and +2.5 mV, the LOWDR signal is to be pulled low

4: This is the total source current for all GPIO pins combined. Individually each pin can source a maximum of 15 mA.

5: System output voltage tolerance specified when 1.024V ≤ VREF ≤ 2.046V.

THERMAL SPECIFICATIONS

Parameter	Symbol	Min	Typ	Max	Units	Conditions
Temperature Ranges						
Specified Temperature Range	T_A	–40	—	+125	°C	
Operating Temperature Range	T_A	–40	—	+125	°C	
Maximum Junction Temperature	T_J	—	—	+150	°C	
Storage Temperature Range	T_A	-65	—	+150	°C	
Thermal Package Resistances						
Thermal Resistance, 24L-QFN 4x4	θ_{JA}	—	42	—	°C/W	Note 1
Thermal Resistance, 28L-QFN 5x5	θ_{JA}	—	35.3	—	°C/W	

Note 1: The parameter is determined using a High K 2S2P 4-layer board as described in JESD51-7, as well as JESD 51-5 for packages with exposed pads.

NOTES:

5.0 DIGITAL ELECTRICAL CHARACTERISTICS

5.1 Timing Parameter Symbology

The timing parameter symbols have been created with one of the following formats:

1. TppS2ppS		3. $T_{CC:ST}$	(I^2C specifications only)
2. TppS		4. Ts	(I^2C specifications only)

T			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

pp			
cc	CCP1	osc	OSC1
ck	CLKOUT	rd	$\overline{RD}$
cs	$\overline{CS}$	rw	$\overline{RD}$ or $\overline{WR}$
di	SDI	sc	SCK
do	SDO	ss	$\overline{SS}$
dt	Data in	t0	T0CKI
io	I/O port	t1	T1CKI
mc	$\overline{MCLR}$	wr	$\overline{WR}$

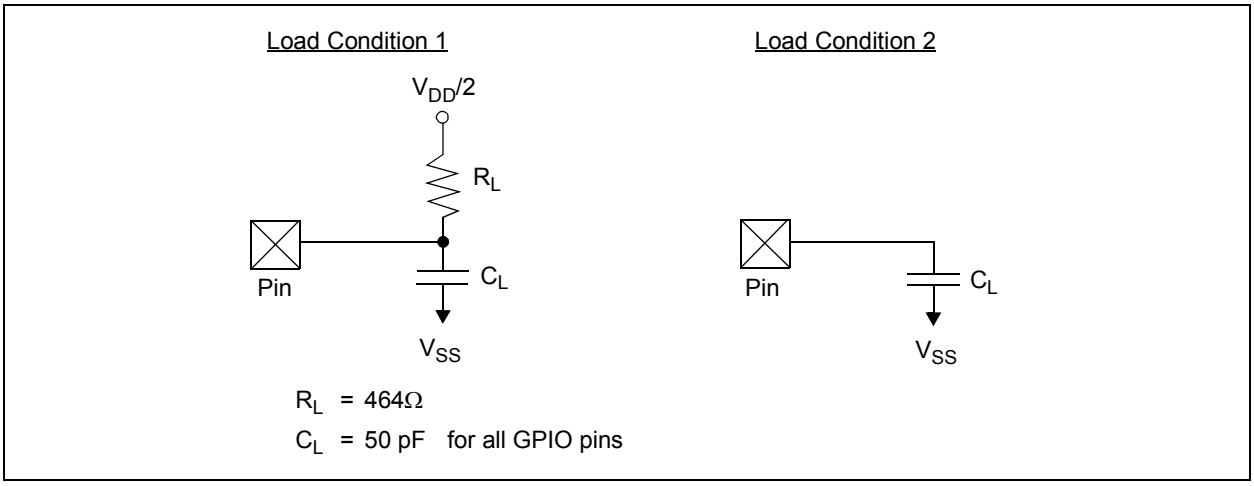
Uppercase letters and their meanings:

S			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (high-impedance)	V	Valid
L	Low	Z	High-impedance
I^2C only			
AA	output access	High	High
BUF	Bus free	Low	Low

TCC:ST (I^2C specifications only)

CC			
HD	Hold	SU	Setup
ST			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

FIGURE 5-1: LOAD CONDITIONS



5.2 AC Characteristics: MCP19122/3

FIGURE 5-2: EXTERNAL CLOCK TIMING

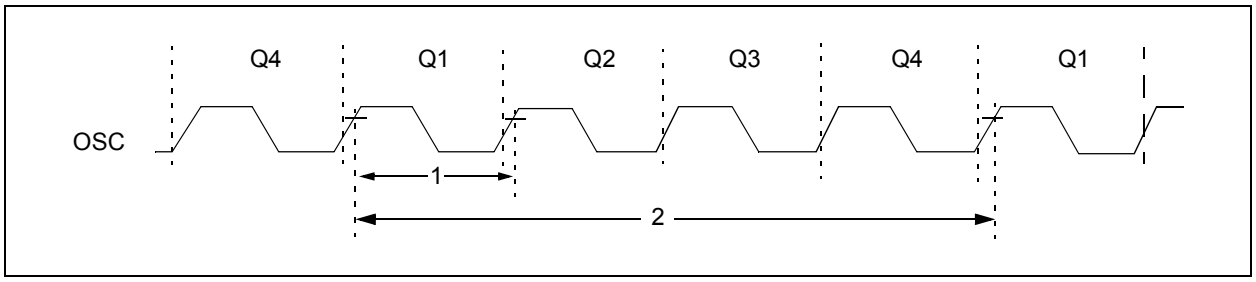


TABLE 5-1: EXTERNAL CLOCK TIMING REQUIREMENTS

Param No.	Sym.	Characteristic	Min.	Typ.†	Max.	Units	Conditions
	F_{OSC}	Oscillator Frequency(1)	—	8	—	MHz	
1	T_{OSC}	Oscillator Period(1)	—	250	—	ns	
2	T_{CY}	Instruction Cycle Time(1)	—	1000	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at V_{IN} = 12V (V_{DD} = 5V), +25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Instruction cycle period (T_{CY}) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code.

FIGURE 5-3: I/O TIMING

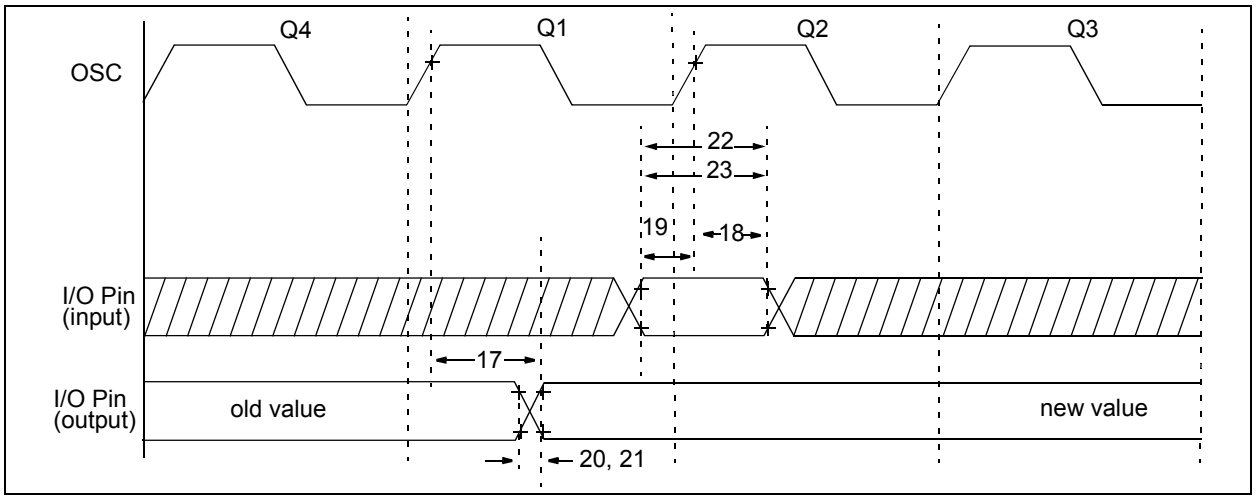


TABLE 5-2: I/O TIMING REQUIREMENTS

Param No.	Sym.	Characteristic	Min.	Typ.†	Max.	Units	Conditions
17	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid	—	50	70*	ns	
18	TosH2ioI	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	50	—	—	ns	
19	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	20	—	—	ns	
20	TioR	Port output rise time	—	32	40	ns	
21	TioF	Port output fall time	—	15	30	ns	
22*	Tinp	INT pin high or low time	25	—	—	ns	
23*	Trbp	GPIO Interrupt-on-change new input level time	T_{CY}	—	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at V_{IN} = 12V (V_{DD} = 5V), +25°C unless otherwise stated.

FIGURE 5-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

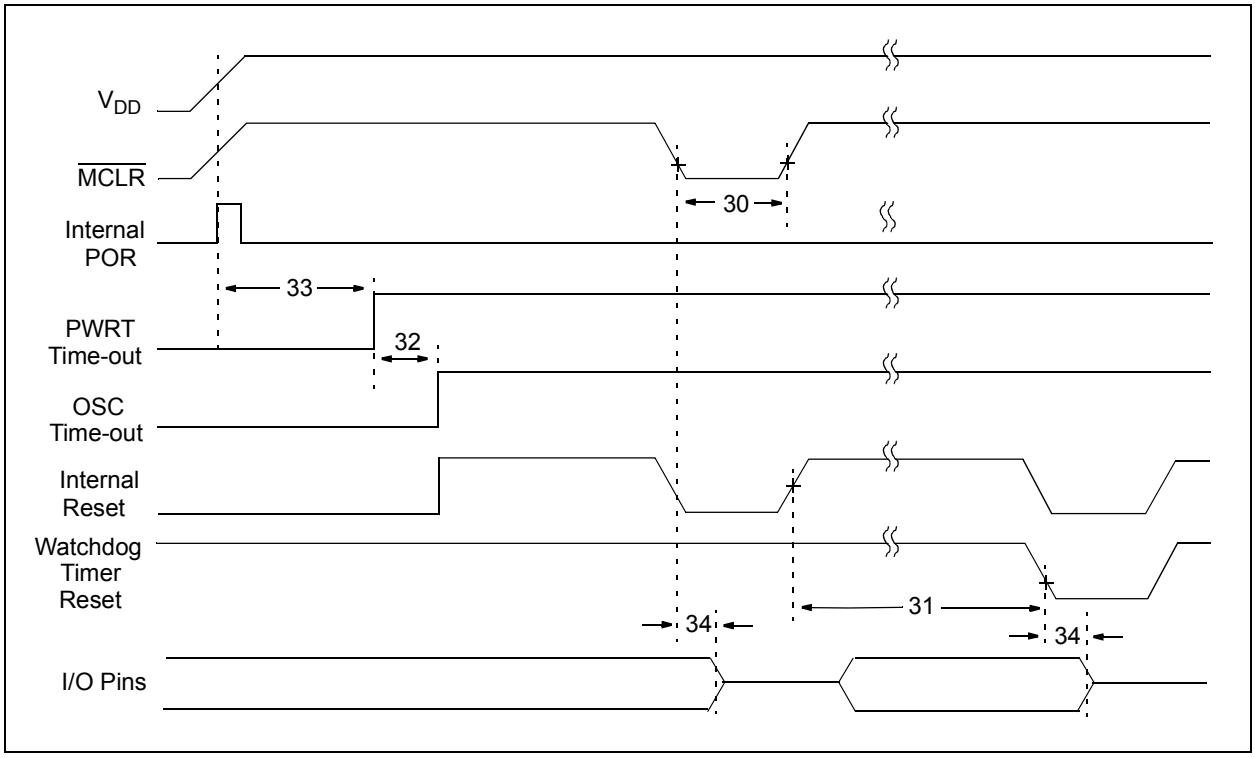


FIGURE 5-5: BROWN-OUT RESET TIMING AND CHARACTERISTICS

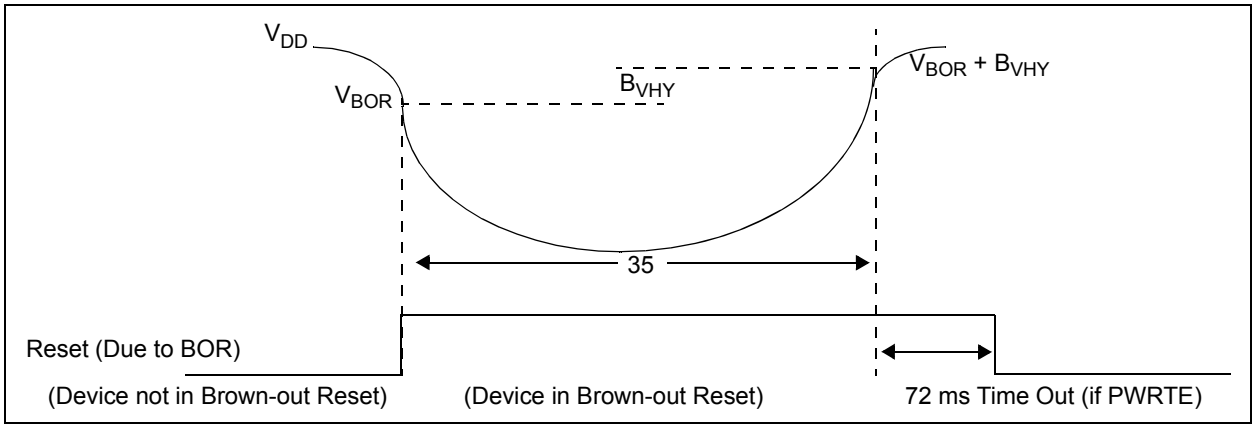


TABLE 5-3: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, AND POWER-UP TIMER REQUIREMENTS

Param No.	Sym.	Characteristic	Min.	Typ.†	Max.	Units	Conditions
30	T_{MCL}	$\overline{MCLR}$ Pulse Width (low)	2	—	—	µs	V_{DD} = 5V, –40°C to +85°C
31	T_{WDT}	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	V_{DD} = 5V, –40°C to +85°C
32	T_{OST}	Oscillation Start-up Timer Period	—	$1024T_{OSC}$	—	—	T_{OSC} = OSC1 period
33*	T_{PWRT}	Power-up Timer Period (4 x T_{WDT})	28	72	132	ms	V_{DD} = 5V, –40°C to +85°C
34	T_{IOZ}	I/O high-impedance from $\overline{MCLR}$ Low or Watchdog Timer Reset	—	—	2.0	µs	
*	V_{POR}	Power-on Reset Voltage	—	2.13	—	V	V_{DD} Rising
*	V_{POR_HYS}	Power-on Reset Voltage Hysteresis	—	100	—	mV	
*	V_{BOR}	Brown-out Reset voltage	—	2.7	—	V	V_{DD} Falling
*	B_{VHY}	Brown-out Hysteresis	—	100	—	mV	
35	T_{BCR}	Brown-out Reset pulse width	100*	—	—	µs	$V_{DD} \leq V_{BOR}$ (D005)
48	$T_{CKEZ-TMR}$	Delay from clock edge to timer increment	$2T_{OSC}$	—	$7T_{OSC}$		

* These parameters are characterized but not tested.

† Data in "Typ." column is at V_{IN} = 12V (V_{DD} = 5V), +25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 5-6: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS

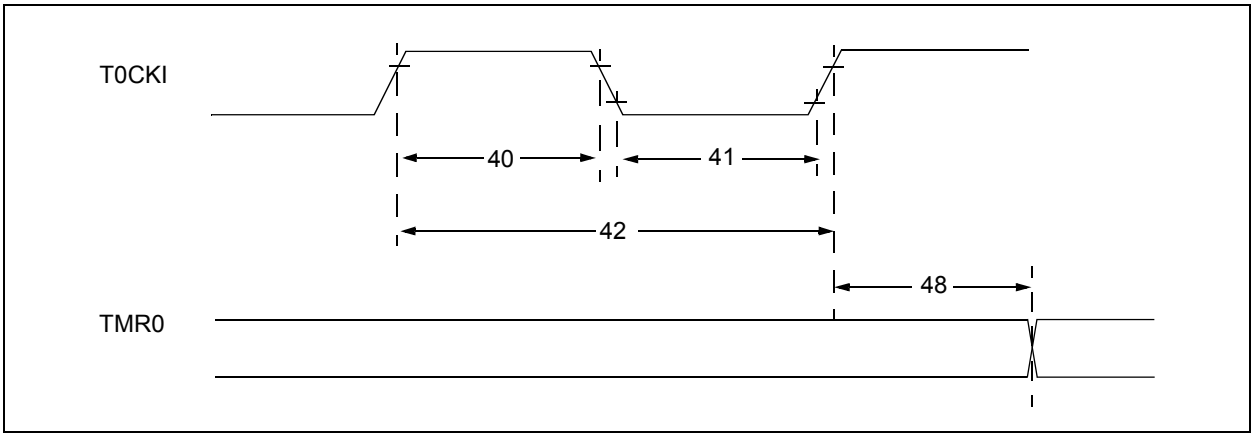
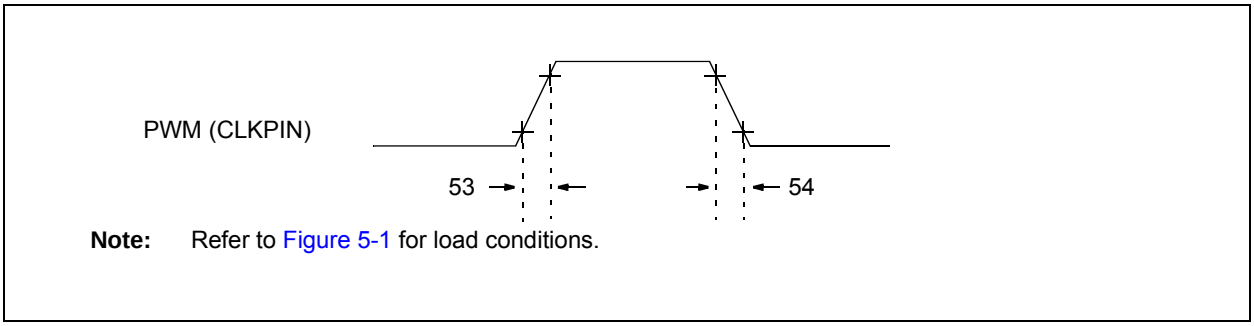


FIGURE 5-7: PNW TIMING



Note: Refer to Figure 5-1 for load conditions.

TABLE 5-4: TABLE5-4: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

Param No.	Sym.	Characteristic		Min.	Typ.†	Max.	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	$0.5T_{CY}$ + 20	—	—	ns	
			With Prescaler	10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	$0.5T_{CY}$ + 20	—	—	ns	
			With Prescaler	10	—	—	ns	
42*	Tt0P	T0CKI Period		Greater of: 20 or $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (2, 4, ..., 256)

* These parameters are characterized but not tested.

† Data in “Typ” column is at V_{IN} = 12V (V_{DD} = 5V), +25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

TABLE 5-5: MCP19122/3 A/D CONVERTER (ADC) CHARACTERISTICS:

Standard Operating Conditions (unless otherwise stated) Operating temperature –40°C ≤ TA ≤ +125°C							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
AD01*	N_R	Resolution	—	—	10	bit	
AD02*	E_{IL}	Integral Error	—	—	±1	LSb	$V_{REF\_ADC}=AV_{DD}$ $V_{REF\_ADC}=V_{DD}$
AD03*	E_{DL}	Differential Error	—	—	±1	LSb	No missing codes to 10 bits $V_{REF\_ADC}=AV_{DD}$ $V_{REF\_ADC}=V_{DD}$
AD04	E_{OFF}	Offset Error	—	+3.0	+7.0	LSb	$V_{REF\_ADC}=AV_{DD}$ $V_{REF\_ADC}=V_{DD}$
AD07	E_{GN}	Gain Error	—	±2	±6	LSb	$V_{REF\_ADC}=AV_{DD}$ $V_{REF\_ADC}=V_{DD}$
AD07*	V_{AIN}	Full-Scale Range	GND	—	AV_{DD}	V	AV_{DD} Selected as ADC Reference
			GND	—	V_{DD}	V	V_{DD} Selected as ADC Reference
AD08*	Z_{AIN}	Recommended Impedance of Analog Voltage Source	—	—	10	kΩ	

* These parameters are characterized but not tested.

† Data in 'Typ' column is at V_{IN} = 12V (AV_{DD} = 4.096V), +25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Total Absolute Error includes integral, differential, offset and gain errors.

2: The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.

3: When ADC is off, it will not consume any current other than leakage current. The power-down current specification includes any such leakage from the ADC module. To minimize Sleep current the ADC Reference must be set to the (default) AV_{DD}.

TABLE 5-6: MCP19122/3 A/D CONVERSION REQUIREMENTS

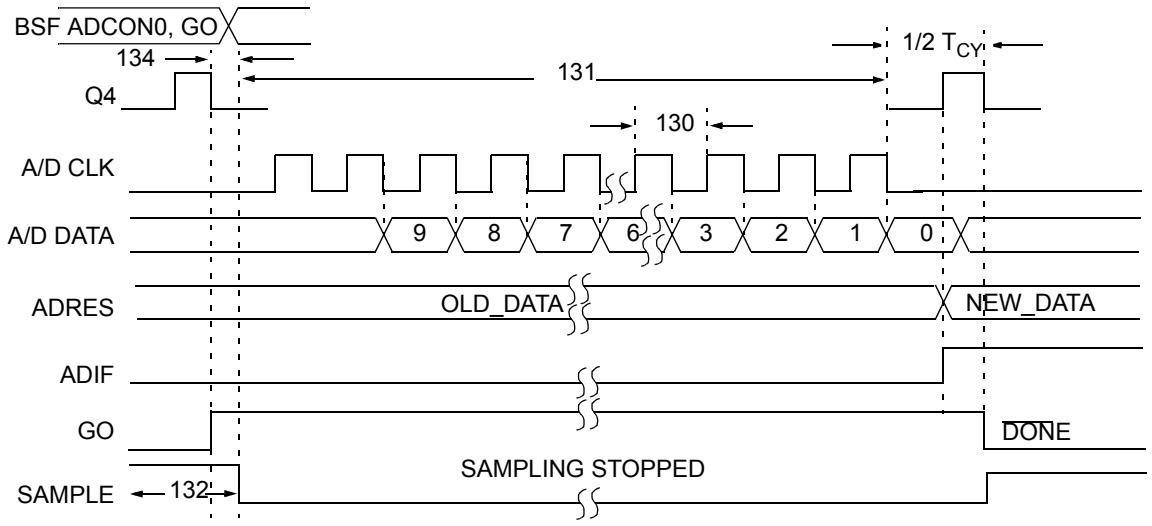
Standard Operating Conditions (unless otherwise stated) Operating temperature –40°C ≤ T_A ≤ +125°C							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
AD130*	T_{AD}	A/D Clock Period	1.6	—	9.0	µs	T_{OSC}-based
		A/D Internal RC Oscillator Period	1.6	4.0	6.0	µs	ADCS<1:0> = 11 (ADRC mode)
AD131	T_{CNV}	Conversion Time (not including Acquisition Time)(1)	—	11	—	T_{AD}	Set GO/$\overline{DONE}$ bit to new data in A/D Result register
AD132*	T_{ACQ}	Acquisition Time	—	11.5	—	µs	
AD133*	T_{AMP}	Amplifier Settling Time	—	—	5	µs	
AD134	T_{GO}	Q4 to A/D Clock Start	—	T_{OSC}/2	—	—	

* These parameters are characterized but not tested.

† Data in 'Typ' column is at V_{IN} = 12V (V_{DD} = 5V), +25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: ADRESH and ADRESL registers may be read on the following T_{CY} cycle.

FIGURE 5-8: A/D CONVERSION TIMING



Note 1: If the A/D clock source is selected as RC, a time of T_{CY} is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

NOTES:

6.0 TYPICAL PERFORMANCE CURVES.

Note: The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore, outside the warranted range.

FIGURE 6-1: I_Q VS. TEMPERATURE (V_{IN} = 12.0V)

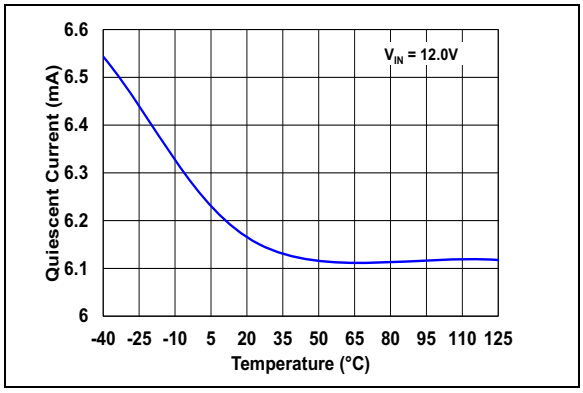


FIGURE 6-2: I_{SHDN} VS. TEMPERATURE (V_{IN} = 12.0V)

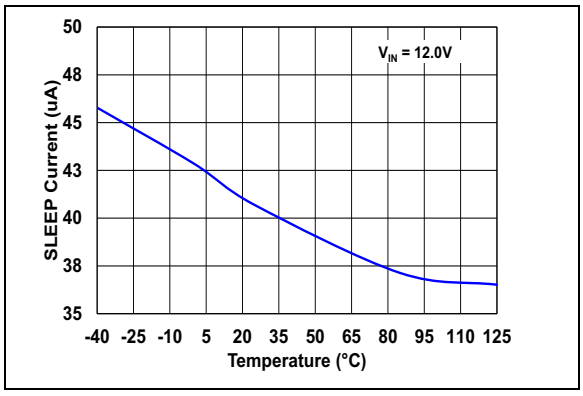


FIGURE 6-3: V_{DD} VS. V_{IN}

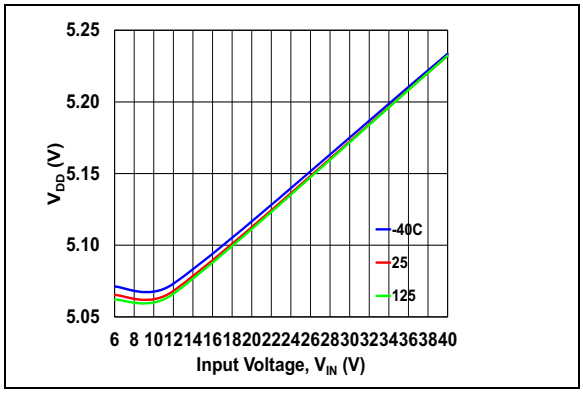


FIGURE 6-4: V_{REF} VS. TEMPERATURE (V_{REF} = 0.512V)

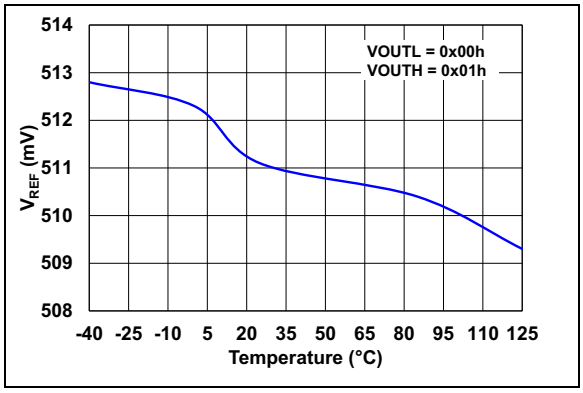


FIGURE 6-5: V_{REF} VS. TEMPERATURE (V_{REF} = 1.024V)

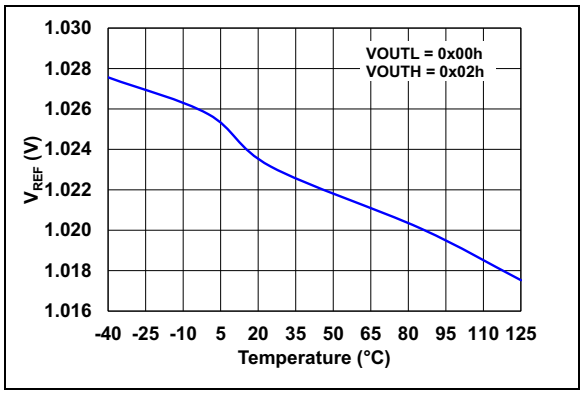


FIGURE 6-6: V_{REF} VS. TEMPERATURE (V_{REF} = 2.048V)

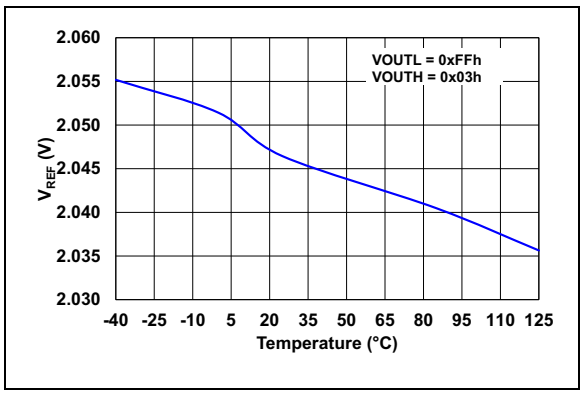


FIGURE 6-7: HDRV RDS-ON VS. TEMPERATURE

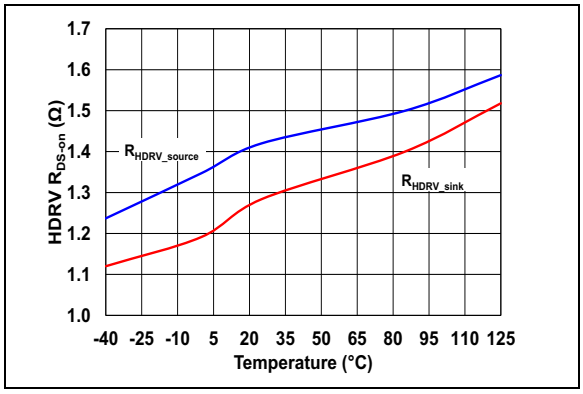


FIGURE 6-8: LDRV RDS-ON VS. TEMPERATURE

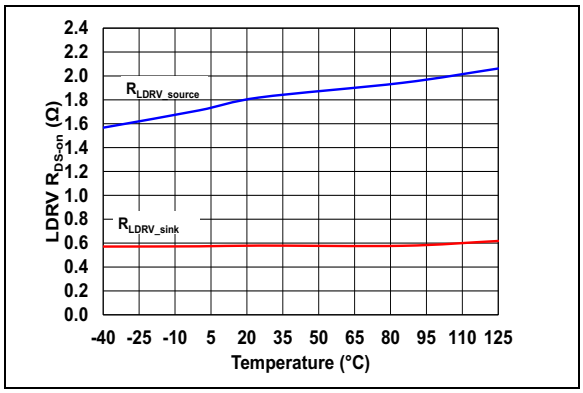


FIGURE 6-9: OSCILLATOR FREQUENCY VS. TEMPERATURE

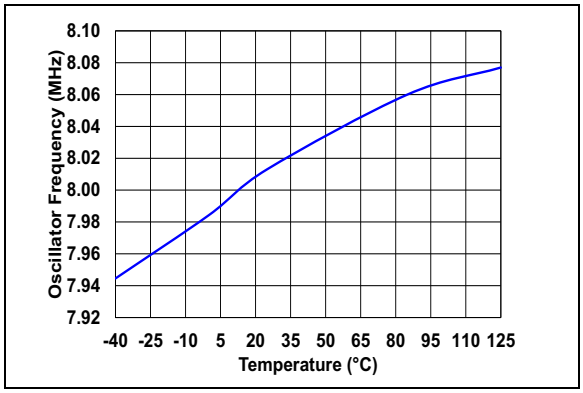


FIGURE 6-10: CURRENT SENSE GAIN VS. TEMPERATURE

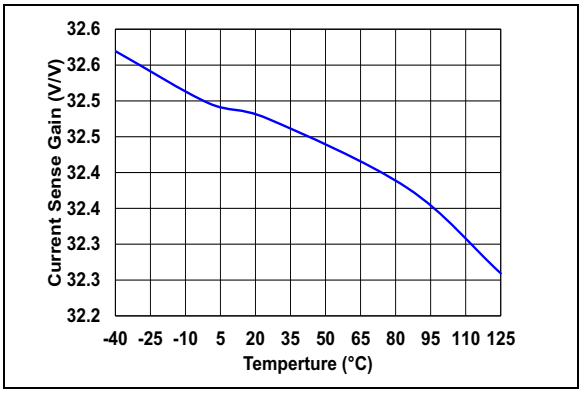
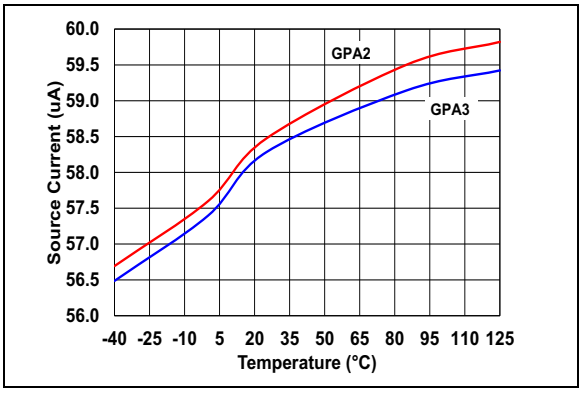


FIGURE 6-11: GPA2/3 SOURCE CURRENT VS. TEMPERATURE



7.0 TEST MUX CONTROL

To allow for easier system design and bench testing, the MCP19122/3 feature a multiplexer used to output various internal analog and digital signals. These signals can be measured on the GPA0 pin through a unity gain buffer. The configuration control of the GPA0 pin is found in the BUFFCON register.

When the BUFFCON<BNCHEN> bit is set, the analog multiplexer output is connected to the GPA0 pin and the CHS<4:0> bits of the ADCON0 register determine which internal analog signal can be measured on the GPA0 pin. Refer to the ADCON0 register (Register 19-1) for analog signals that can be view on GPA0 while BNCHEN is set.

When the BUFFCON<DIGOEN> bit is set, the digital multiplexer output is connected to the GPA0 pin. The DSEL<4:0> bits of the BUFFCON register determine which internal digital signal can be measured on the GPA0 pin.

If a conflict exist where both the BNCHEN and DIGOEN bits are set, the DIGOEN bit takes priority.

When measuring signals with the unity gain buffer, the buffer offset must be added to the measured signal. The factory measured buffer offset can be read from memory location 2087h. Refer to **Section 10.1.1 "Reading Program Memory as Data"** for more information.

REGISTER 7-1: BUFFCON: TEST MUX CONTROL REGISTER

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BNCHEN	DIGOEN	—	DSEL4	DSEL3	DSEL2	DSEL1	DSEL0
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 7 **BNCHEN:** GPA0 analog multiplexer configuration control bit
1 = GPA0 is configured to be analog multiplexer output
0 = GPA0 is configured for normal operation

bit 6 **DIGOEN:** GPA0 digital multiplexer configuration control bit
1 = GPA0 is configured to be digital multiplexer output
0 = GPA0 is configured for normal operation

bit 5 **Unimplemented:** Read as '0'

bit 4-0 **DSEL<4:0>:** Multiplexer output control bit
00000 = 50% period signal
00001 = System clock
00010 = Inductor current SAMPLE signal
00011 = OV Comparator Output
00100 = UV Comparator Output
00101 = OVLO Comparator Output
00110 = UVLO Comparator Output
00111 = OC Comparator Output
01000 = high_on signal
01001 = Low-side on signal before the delay block
•
•
•
10000 = Output of PWM Comparator
10001 = SWFRQ Signal
10010 = T2_EQ_PR2 Signal
10011 = PWM_OUT Signal
10100 = Clock Select / Switchover Waveform
10101 = DEM Comparator Output
10111 = DEM Blanking Time
10111 = Auto Zero Time Or'ed Signal

NOTES:

8.0 RELATIVE EFFICIENCY MEASUREMENT

With a constant input voltage, output voltage and load current, any change in the high-side MOSFET on-time represents a change in the system efficiency. The MCP19122/3 is capable of measuring the on-time of the high-side MOSFET. Therefore, the relative efficiency of the system can be measured and optimized by changing the system parameters' driver dead time, such as switching frequency.

8.1 Relative Efficiency Measurement Procedure

To measure the relative efficiency, the RELEFF register, PE1<MEASEN> bit, and the ADC RELEFF input are used. The following steps outlines the measurement process:

1. Clear the PE1<MEASEN> bit.
2. Set the PE1<SPAN> bit.
3. With the ADC, read the RELEFF channel and store this reading as the High.
4. Clear the PE1<SPAN> bit.
5. With the ADC, read the RELEFF channel and store this reading as the Low.
6. Set the PE1<MEASEN> bit to initiate a measurement cycle.
7. Monitor the RELEFF<MSDONE> bit. When set, it indicates the measurement is complete.
8. When the measurement is complete, use the ADC to read the RELEFF channel. This value becomes the Fractional variable in Equation 10 1. This reading should be accomplished within 50ms of the RELEFF<MSDONE> bit is set.
9. Read the value of the RE<6:0> bits in the RELEFF register and store the reading as Whole.
10. Clear the PE1<MEASEN> bit.
11. The relative efficiency is then calculated by the following equation:

EQUATION 8-1:

$$Duty\_Cycle = \frac{\left(Whole + \frac{(Fractional - Low)}{(High - Low)}\right)}{(PR2 + 1)}$$

Where:

Whole = Value obtained in Step 9 of the measurement procedure

Fractional = Value obtained in Step 8 of the measurement procedure

High = Value obtained in Step 3 of the measurement procedure

Low = Value obtained in Step 5 of the measurement procedure

Note 1: The RELEFF<MSDONE> bit is set and cleared automatically.

REGISTER 8-1: RELEFF: RELATIVE EFFICIENCY MEASUREMENT REGISTER

R-0	R-x	R-x	R-x	R-x	R-x	R-x	R-x
MSDONE	RE6	RE5	RE4	RE3	RE2	RE1	RE0
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 7 **MSDONE:** Relative efficiency measurement done bit
1 = Relative efficiency measurement is complete
0 = Relative efficiency measurement is not complete

bit 6-0 **RE<6:0>:** Whole clock counts for relative efficiency measurement result

NOTES:

9.0 DEVICE CALIBRATION

Read-only memory locations 2080h through 208Fh contain factory calibration data. Refer to **Section 20.0 "Flash Program Memory Control"** for information on how to read from these memory locations.

9.1 Calibration Word 1

The TTA<3:0> bits at memory location 2080h calibrate the over temperature shutdown threshold point. Firmware must read these values and write them to the TTACAL register for proper calibration.

The FCAL<6:0> bits at memory location 2080h set the internal oscillator calibration. Firmware must read these values and write them to the OSCCAL register for proper calibration.

REGISTER 9-1: CALWD1: CALIBRATION WORD 1 REGISTER

U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1
—	—	TTA3	TTA2	TTA1	TTA0
bit 13					bit 8

U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	FCAL6	FCAL5	FCAL4	FCAL3	FCAL2	FCAL1	FCAL0
bit 7							bit 0

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

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

bit 11-8 **TTA<3:0>:** Overtemperature shutdown calibration bits.

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

bit 6-0 **FCAL<6:0>:** Internal oscillator calibration bits

9.2 Calibration Word 2

The BGT<3:0> bits at memory location 2081h calibrate the internal band gap over temperature. Firmware must read these values and write them to the BGTCAL register for proper calibration.

The BGR<5:0> bits at memory location 2081h calibrate the internal band gap. Firmware must read these values and write them to the BGRCAL register for proper calibration.

REGISTER 9-2: CALWD2: CALIBRATION WORD 2 REGISTER

U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1
—	—	BGT3	BGT2	BGT1	BGT0
bit 13					bit 8

U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	BGR5	BGR4	BGR3	BGR2	BGR1	BGR0
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 13-12 **Unimplemented:** Read as ‘0’

bit 11-8 **BGT<3:0>:** Internal band gap temperature calibration bits.

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

bit 5-0 **BGR<5:0>:** Internal band gap calibration bits.

9.3 Calibration Word 3

The AVDD<3:0> bits at memory location 2082h calibrate the internal 4.096V bias voltage. Firmware must read these values and write them to the AVDDCAL register for proper calibration.

The VOUR<4:0> bits at memory location 2082h calibrate the output overvoltage and undervoltage reference. Firmware must read these values and write them to the VOURCAL register for proper calibration.

REGISTER 9-3: CALWD3: CALIBRATION WORD 3 REGISTER

U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1
—	—	AVDD3	AVDD2	AVDD1	AVDD0
bit 13					bit 8

U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	VOUR4	VOUR3	VOUR2	VOUR1	VOUR0
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 13-12 **Unimplemented:** Read as ‘0’

bit 11-8 **AVDD<3:0>:** Internal 4V bias voltage calibration bits.

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **VOUR<4:0>:** Output overvoltage and undervoltage reference calibration bits.

9.4 Calibration Word 4

The DOV<5:0> bits at memory location 2083h set the offset calibration for the output voltage remote sense differential amplifier. Firmware must read these values and write them to the DOVCAL register for proper calibration.

The VEAO<4:0> bits at memory location 2083h calibrate the offset of the error amplifier. Firmware must read these values and write them to the VEAOCAL register for proper calibration.

REGISTER 9-4: CALWD4: CALIBRATION WORD 4 REGISTER

U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	DOV4	DOV3	DOV2	DOV1	DOV0
bit 13					bit 8

U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	VEAO4	VEAO3	VEAO2	VEAO1	VEAO0
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 13 **Unimplemented:** Read as ‘0’

bit 12-8 **DOV<4:0>:** Output voltage remote sense differential amplifier offset calibration bits

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **VEAO<4:0>:** Error amplifier offset voltage calibration bits

9.5 Calibration Word 5

The VREF<4:0> bits at memory location 2084h calibrate the reference to the DAC that sets the output voltage reference. Firmware must read these values and write them to the VREFCAL register for proper calibration.

The VRFS<4:0> bits at memory location 2084h calibrate the full scale range of reference to the DAC that sets the output voltage reference. Firmware must read these values and write them to the VRFSCAL register for proper calibration. The VRFS<4:0> bits are to be loaded in to the VRFSCAL register when the DACCON = 0x00h or 0x07h.

REGISTER 9-5: CALWD5: CALIBRATION WORD 5 REGISTER

U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	VREF4	VREF3	VREF2	VREF1	VREF0
bit 13					bit 8

U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	VRFS4	VRFS3	VRFS2	VRFS1	VRFS0
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 13 **Unimplemented:** Read as ‘0’

bit 12-8 **VREF<4:0>:** Output voltage reference DAC calibration bits

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **VRFS<4:0>:** Output voltage reference DAC full scale range calibration bits

9.6 Calibration Word 6

The RAMP<4:0> bits at memory location 2085h calibrate the span of the slope compensation ramp. Firmware must read these values and write them to the RAMPCAL register for proper calibration.

The CSR<4:0> bits at memory location 2085h are used to calibrate the gain of the current sense amplifier. Firmware must read these values and write them to the CSRCAL register for proper calibration.

REGISTER 9-6: CALWD6: CALIBRATION WORD 6 REGISTER

U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	RAMP4	RAMP3	RAMP2	RAMP1	RAMP0
bit 13					bit 8

U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	CSR4	CSR3	CSR2	CSR1	CSR0
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 13 **Unimplemented:** Read as ‘0’

bit 12-8 **RAMP<4:0>:**Slope compensation ramp span calibration bits

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **CSR<4:0>:** Current sense amplifier gain calibration bits

9.7 Calibration Word 7

Calibration Word 7 at memory location 2086h contains the offset calibration for the output overvoltage and undervoltage comparators.

The OVCO<3:0> bits contain the output overvoltage comparator offset voltage calibration.

The UVCO<3:0> bits contain the output undervoltage comparator offset voltage calibration.

Firmware must read these values and write them to the OVUVCAL register for proper operation.

REGISTER 9-7: CALWD7: CALIBRATION WORD 7 REGISTER

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—
bit 13					bit 8

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
UVCO3	UVCO2	UVCO1	UVCO0	OVCO3	OVCO2	OVCO1	OVCO0
bit 7							bit 0

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

bit 13-8 **Unimplemented:** Read as '0'

bit 7-4 **UVCO<3:0>:** Output Undervoltage Comparator Offset calibration bits

bit 3-0 **OVCO<3:0>:** Output Overvoltage Comparator Offset calibration bits

9.8 Calibration Word 8

The DEMOV<4:0> bits at memory location 2087h contain the diode emulation mode comparator offset voltage. Firmware must read these values and write them to the DEMCAL register for proper operation.

REGISTER 9-8: CALWD8: CALIBRATION WORD 8 REGISTER

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—
bit 13					bit 8

U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	DEMOV4	DEMOV3	DEMOV2	DEMOV1	DEMOV0
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 13-5 **Unimplemented:** Read as ‘0’

bit 4-0 **DEMOV<4:0>:** Diode Emulation Mode Comparator Offset Voltage calibration bits

9.9 Calibration Word 9

The HCSOV<4:0> bits at memory location 2088h contain the calibration values for the offset voltage on the high-side current sense amplifier. Firmware must read these values and write them to the HCSOVCAL register for proper operation.

REGISTER 9-9: CALWD9: CALIBRATION WORD 9 REGISTER

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—
bit 13					bit 8

U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	HCSOV6	HCSOV5	HCSOV4	HCSOV3	HCSOV2	HCSOV1	HCSOV0
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 13-7 **Unimplemented:** Read as ‘0’

bit 6-0 **HCSOV<6:0>:** High-side Current Amplifier Offset Voltage calibration bits

9.10 Calibration Word 10

The TANA<7:0> bits at memory location 2089h contain the ADC reading from the internal temperature sensor when the silicon temperature is at +25°C.

This 10-bit reading can be used to calculate the silicon die temperature. See Section 25.0 "Internal Temperature Indicator Module" for more details.

REGISTER 9-10: CALWD10: CALIBRATION WORD 10 REGISTER

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	TANA9	TANA8
bit 13					bit 8

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
TANA7	TANA6	TANA5	TANA4	TANA3	TANA2	TANA1	TANA0
bit 7							bit 0

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

bit 13-8 **Unimplemented:** Read as '0'

bit 7-0 **TANA<9:0>:** ADC internal temperature sensor at +25°C calibration bits

9.11 Calibration Word 11

The BUFF<7:0> bits at memory location 208Ah represent the offset voltage of the unity gain buffer in units of millivolts. This is an 8-bit two's complement number. The MSB is the sign bit. If the MSB is set to 1, the resulting number is negative.

REGISTER 9-11: CALWD11: CALIBRATION WORD 11 REGISTER

U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—
bit 13					bit 8

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
BUFF7	BUFF6	BUFF5	BUFF4	BUFF3	BUFF2	BUFF1	BUFF0
bit 7							bit 0

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

bit 13-8 **Unimplemented:** Read as '0'

bit 7-0 **BUFF<7:0>:** Unity gain buffer offset voltage calibration bits

9.12 Calibration Word 12

The ADCCAL<13:0> bits at memory location 208Bh contain the calibration bits for the A/D converter. Calibration Word 12 contains the factory measurement of the full scale ADC reference. The value represents the number of A/D converter counts per volt. ADCC<4:0> represent the fraction of an A/D converter count, which can provide additional precision when oversampling the ADC for enhanced resolution. This calibration word can be used to calibrate signals read by the A/D converter.

REGISTER 9-12: CALWD12: CALIBRATION WORD 12 REGISTER

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
ADCC13	ADCC12	ADCC11	ADCC10	ADCC9	ADCC8
bit 13					bit 8

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
ADCC7	ADCC6	ADCC5	ADCC4	ADCC3	ADCC2	ADCC1	ADCC0
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 13-5 **ADCC<13:5>:** Whole number of A/D converter counts
111111111 = 511
•
•
•
000000000 = 0

bit 4-0 **ADCC<4:0>:** Fraction number of A/D converter counts
1111 = 0.96875
•
•
•
0001 = 0.03125
0000 = 0

9.13 Calibration Word 13

Calibration Word 13 at memory location 208Ch contain the offset of the A/D converter. This calibration word can be used to calibrate signals read by the A/D converter.

REGISTER 9-13: CALWD13: CALIBRATION WORD 13 REGISTER

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
ADCO13	ADCO12	ADCO11	ADCO10	ADCO9	ADCO8
bit 13					bit 8

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
ADCO7	ADCO6	ADCO5	ADCO4	ADCO3	ADCO2	ADCO1	ADCO0
bit 7							bit 0

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

bit 13-0 **ADCO<13:0>:** A/D converter offset

9.14 Calibration Words 14-16

Memory locations 208Dh to 208Fh contain reference voltage span adjustment (VRFSCAL) calibration values to be used with different settings of the differential amplifier gain. The appropriate VRFSCAL value must be used to achieve the system output voltage tolerance specification listed in the **Section "Electrical characteristics"**.

9.14.1 CALIBRATION WORD 14

The VRS18<4:0> bits in CALWD 14 at memory location 208Dh are to be loaded into the VRFSCAL register when the DAGCON = 0x03h. The VRS14<4:0> bits of CALWD14 are to be loaded into the VRFSCAL register when the DAGCON = 0x02h.

REGISTER 9-14: CALWD14: CALIBRATION WORD 14 REGISTER

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	VRS144	VRS143	VRS142	VRS141	VRS140
bit 13					bit 8

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	VRS184	VRS183	VRS182	VRS181	VRS180
bit 7							bit 0

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

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

bit 12-8 **VRS14<4:0>**: VRFSCAL calibration with Differential Amplifier gain of 1/4 (DAGCON = 0x02h)

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **VRS18<4:0>**: VRFSCAL calibration with Differential Amplifier gain of 1/8 (DAGCON = 0x03h)

9.14.2 CALIBRATION WORD 15

The VSR12<4:0> bits in CALWD 15 at memory location 208Eh are to be loaded into the VRFSCAL register when the DAGCON = 0x01h. The VRS2<4:0> bits of CALWD15 are to be loaded into the VRSFCAL register when the DAGCON = 0x04h.

REGISTER 9-15: CALWD15: CALIBRATION WORD 15 REGISTER

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	VRS24	VRS23	VRS22	VRS21	VRS20
bit 13					bit 8

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	VRS124	VRS123	VRS122	VRS121	VRS120
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 13 **Unimplemented:** Read as ‘0’

bit 12-8 **VRS2<4:0>**: VRFSCAL calibration with Differential Amplifier gain of 2 (DAGCON = 0x04h)

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **VRS12<4:0>**: VRFSCAL calibration with Differential Amplifier gain of 1/2 (DAGCON = 0x01h)

9.14.3 CALIBRATION WORD 16

The VSR4<4:0> bits in CALWD 16 at memory location 208Fh are to be loaded into the VRFSCAL register when the DAGCON = 0x05h. The VRS8<4:0> bits of CALWD16 are to be loaded into the VRSFCAL register when the DAGCON = 0x06h.

REGISTER 9-16: CALWD16: CALIBRATION WORD 16 REGISTER

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	VRS84	VRS83	VRS82	VRS81	VRS80
bit 13					bit 8

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	VRS44	VRS43	VRS42	VRS84	VRS40
bit 7							bit 0

Legend:			
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 13 **Unimplemented:** Read as ‘0’

bit 12-8 **VRS8<4:0>**: VRFSCAL calibration with Differential Amplifier gain of 8 (DAGCON = 0x06h)

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **VRS4<4:0>**: VRFSCAL calibration with Differential Amplifier gain of 4 (DAGCON = 0x05h)

10.0 MEMORY ORGANIZATION

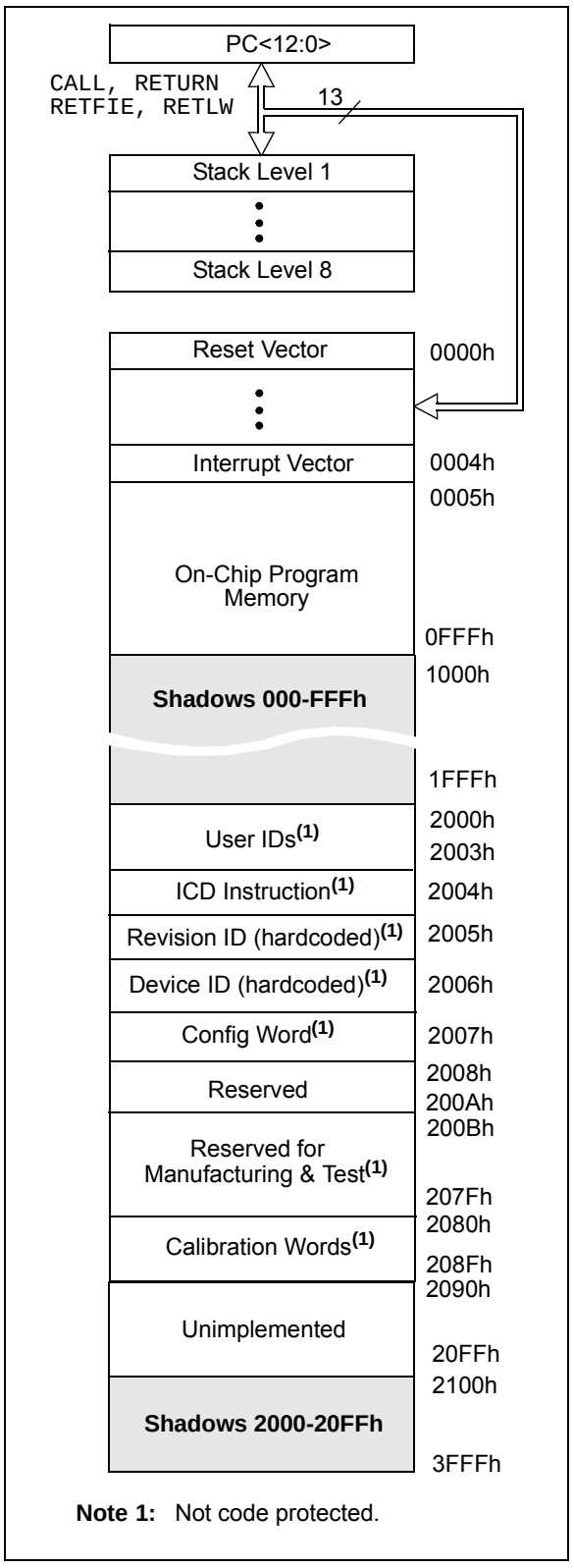
There are two types of memory in the MCP19122/3:

- Program Memory
- Data Memory
  - Special Function Registers (SFRs)
  - General-Purpose RAM

10.1 Program Memory Organization

The MCP19122/3 has a 13-bit program counter capable of addressing an 8K x 14 program memory space. Only the first 4K x 14 (0000h-0FFFh) is physically implemented. Addressing a location above this boundary will cause a wrap-around within the first 4K x 14 space. The Reset vector is at 0000h and the interrupt vector is at 0004h (see Figure 10-1). The width of the program memory bus (instruction word) is 14-bits. Since all instructions are a single word, the MCP19122/3 has space for 4K of instructions.

FIGURE 10-1: PROGRAM MEMORY MAP AND STACK FOR MCP19122/3



Note 1: Not code protected.

10.1.1 READING PROGRAM MEMORY AS DATA

There are two methods of accessing constants in program memory. The first method is to use tables of RETLW instructions. The second method is to set a Files Select Register (FSR) to point to the program memory.

10.1.1.1 RETLW Instruction

The RETLW instruction can be used to provide access to tables of constants. The recommended way to create such a table is shown in Example 10-1.

EXAMPLE 10-1: RETLW INSTRUCTION

```
constants
    ADDWF_PCL
    RETLW DATA0         ;Index0 data
    RETLW DATA1         ;Index1 data
    RETLW DATA2
    RETLW DATA3


my_function
    ;… LOTS OF CODE…
    MOVLW       DATA_INDEX
    call constants
    ;… THE CONSTANT IS IN W
```

10.2 Data Memory Organization

The data memory (see Table 10-1) is partitioned into four banks, which contain the General Purpose Registers (GPR) and the Special Function Registers (SFR). The Special Function Registers are located in the first 32 locations of each bank. Register locations 20h-7Fh in Bank 0, A0h-EFh in Bank 1 and 120h-16Fh in Bank 2 are General Purpose Registers, implemented as static RAM. All other RAM is unimplemented and returns '0' when read. The RP<1:0> bits of the STATUS register are the bank select bits.

RP1	RP0	
0	0	-> Bank 0 is selected
0	1	-> Bank 1 is selected
1	0	-> Bank 2 is selected
1	1	-> Bank 3 is selected

To move values from one register to another register, the value must pass throught the W register. This means that for all register-to-register moves, two instruction cycles are required.

The entire data memory can be accessed either directly or indirectly. Direct addressing may require the use of the RP<1:0> bits. Indirect addressing uses the Indirect REgister Pointer (IRP) bit inthe STATUS register for access to the Bank0/Bank1 or the Bank2/Bank3 areas of data memory.

10.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 64 x 8 in the MCP19122/3. Each register is accessed, either directly or indirectly, through the FSR (refer to **Section 10.5 "Indirect Addressing, INDF and FSR Registers"**).

10.2.2 CORE REGISTERS

The core registers contain the registers that directly affect the basic operation. The core registers can be addressed from any bank. These registers are listed in Table 10-1. For detailed information refer to Table 10-2.

TABLE 10-1: CORE REGISTERS

Addresses	BANKx
x00h, x80h, x100h, or x180h	INDF
x02h, x82h, x102h, or x182h	PCL
x03h, x83h, x103h, or x183h	STATUS
x04h, x84h, x104h, or x184h	FSR
x0Ah, x8Ah, x10Ah, or x18Ah	PCLATH
x0Bh, x8Bh, x10Bh, or x18Bh	INTCON

10.2.3 STATUS REGISTER

The STATUS register, shown in Register 10-1, contains:

- the arithmetic status of the ALU
- the Reset status
- the bank select bits for data memory (RAM)

The STATUS register can be the destination for any instruction, like any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{TO}$ and $\overline{PD}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, CLRF STATUS will clear the upper three bits and set the Z bit. This leaves the STATUS register as '000u u1uu' (where u = unchanged).

Therefore, it is recommended that only BCF, BSF, SWAPF and MOVWF instructions are used to alter the STATUS register, because these instructions do not affect any Status bits. For other instructions not affecting any Status bits, see the **Section 28.0 "Instruction Set Summary"**.

Note 1: The C and DC bits operate as $\overline{Borrow}$ and Digit $\overline{Borrow}$ out bits, respectively, in subtraction.

REGISTER 10-1: STATUS: STATUS REGISTER

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC(1)	C(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 7 **IRP:** Register Bank Select bit (used for Indirect addressing)
1 = Bank 2 & 3 (100h - 1FFh)
0 = Bank 0 & 1 (00h - FFh)

bit 6-5 **RP<1:0>:** Register Bank Select bits (used for Direct addressing)
00 = Bank 0 (00h - 7Fh)
01 = Bank 1 (80h - FFh)
10 = Bank 2 (100h - 17Fh)
11 = Bank 3 (180h - 1FFh)

bit 4 **$\overline{TO}$:** Time-out bit
1 = After power-up, CLRWDT instruction or SLEEP instruction
0 = A WDT time-out occurred

bit 3 **$\overline{PD}$:** Power-down bit
1 = After power-up or by the CLRWDT instruction
0 = By execution of the SLEEP instruction

bit 2 **Z:** Zero bit
1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

bit 1 **DC:** Digit Carry/$\overline{\text{Digit Borrow}}$ bit(1) (ADDWF, ADDLW, SUBLW, SUBWF instructions)
1 = A carry-out from the 4th low-order bit of the result occurred
0 = No carry-out from the 4th low-order bit of the result

bit 0 **C:** Carry/$\overline{\text{Borrow}}$ bit(1) (ADDWF, ADDLW, SUBLW, SUBWF instructions)(1)
1 = A carry-out from the Most Significant bit of the result occurred
0 = No carry-out from the Most Significant bit of the result occurred

Note 1: For $\overline{\text{Borrow}}$, the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (RRF, RLF) instructions, this bit is loaded with either the high-order or low-order bit of the source register.

10.2.4 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device (see Table 10-2). These registers are static RAM.

The special registers can be classified into two sets:

- core
- peripheral

The Special Function Registers associated with the microcontroller core are described in this section. Those related to the operation of the peripheral features are described in the section of that peripheral feature.

10.3 DATA MEMORY

TABLE 10-2: MCP19122/3 DATA MEMORY MAP

	File Address		File Address		File Address		File Address
Indirect addr.(1)	00h	Indirect addr. (1)	80h	Indirect addr.(1)	100h	Indirect addr. (1)	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTGPA	05h	TRISGPA	85h	WPUGPA	105h	IOCA	185h
PORTGPB	06h	TRISGPB	86h	WPUGPB	106h	IOCB	186h
PIR1	07h	PIE1	87h	PE1	107h	ANSELA	187h
PIR2	08h	PIE2	88h	MODECON	108h	ANSELB	188h
PCON	09h		89h		109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
TMR1L	0Ch		8Ch		10Ch	PORTICD(2)	18Ch
TMR1H	0Dh		8Dh		10Dh	TRISICD(2)	18Dh
T1CON	0Eh		8Eh		10Eh	ICKBUG(2)	18Eh
TMR2	0Fh		8Fh		10Fh	BIGBUG(2)	18Fh
T2CON	10h	VINUVLO	90h	SSPADD	110h	PMCON1	190h
PR2	11h	VINOVLO	91h	SSPBUF	111h	PMCON2	191h
T1GCON	12h	VINCON	92h	SSPCON1	112h	PMADRL	192h
PWMPHL	13h	DAGCON	93h	SSPCON2	113h	PMADRH	193h
PWMPHH	14h	VOUTL	94h	SSPCON3	114h	PMDATL	194h
PWMRL	15h	VOUTH	95h	SSPMSK	115h	PMDATH	195h
PWMRH	16h	OSCTUNE	96h	SSPSTAT	116h	TTACAL	196h
CC1RL(3)	17h		97h	SSPADD2	117h	OSCCAL	197h
CC1RH(3)	18h	CMPZCON	98h	SSPMSK2	118h	BGTCAL	198h
CC2RL(3)	19h	VOTUVLO	99h	VREFCAL	119h	BGRCAL	199h
CC2RH(3)	1Ah	VOTOVLO	9Ah	VRFSCAL	11Ah	AVDDCAL	19Ah
CCDCON(3)	1Bh	DEADCON	9Bh	RAMPCAL	11Bh	VOURCAL	19Bh
ADRESL	1Ch	RAMPCON	9Ch	CSRCAL	11Ch	DOVCAL	19Ch
ADRESH	1Dh	OCCON	9Dh	OVUVCAL	11Dh	VEAOCAL	19Dh
ADCON0	1Eh	CSGSCON	9Eh	DEMCAL	11Eh	BUFFCON	19Eh
ADCON1	1Fh	RELEFF	9Fh	HCSOVCAL	11Fh	Reserved	19Fh
General Purpose Register 96 Bytes	20h	General Purpose Register 80 Bytes	A0h	General Purpose Register 80 bytes	120h		1A0h
			EFh		16F		1EF
		Accesses Bank 0	F0h	Accesses Bank 0	170h	Accesses Bank 0	1F0h
	7Fh		FFh		17Fh		1FFh
Bank 0		Bank 1		Bank2		Bank3	

Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.

2: Only accessible when DBGEN = 0 and ICKBUG<INBUG> = 1.

3: Only in MCP19123 and DSTEMP.

TABLE 10-3: MCP19122/3 SPECIAL REGISTERS SUMMARY BANK 0

Addr.	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR Reset BOR Reset	Value on all other resets(1)
Bank 0											
00h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
01h	TMR0	Timer0 Module's Register								xxxx xxxx	uuuu uuuu
02h	PCL	Program Counter's (PC) Least Significant byte								0000 0000	0000 0000
03h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTGPA	GPA7	GPA6	GPA5	GPA4	GPA3	GPA2	GPA1	GPA0	xxxx xxxx	uuuu uuuu
06h	PORTGPB	GPB7	GPB6	GPB5	GPB4	GPB3	GPB2	GPB1	GPB0	xxxx xxxx	uuuu uuuu
07h	PIR1	TMR1GIF	ADIF	BCLIF	SSPIF	CC2IF	CC1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
08h	PIR2	UVIF	OTIF	OCIF	OVIF	—	—	OVLOIF	UVLOIF	0000 --00	0000 --00
09h	PCON	—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$	---- --qq	---- --uu
0Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of program counter					---0 0000	---0 0000
0Bh	INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF(2)	0000 000x	0000 000u
0Ch	TMR1L	Holding register for the Least Significant byte of the 16-bit TMR1								xxxx xxxx	uuuu uuuu
0Dh	TMR1H	Holding register for the Most Significant byte of the 16-bit TMR1								xxxx xxxx	uuuu uuuu
0Eh	T1CON	—	—	T1CKPS1	T1CKPS0	—	—	TMR1CS	TMR1ON	--00 --00	--uu --uu
0Fh	TMR2	Timer2 Module Register								xxxx xxxx	uuuu uuuu
10h	T2CON	—	—	—	—	—	TMR2ON	T2CKPS1	T2CKPS0	---- -000	---- -000
11h	PR2	Timer2 Module Period Register								1111 1111	1111 1111
12h	T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/DONE	T1GVAL	T1GSS1	T1GSS0	0000 0x00	uuuu uuuu
13h	PWMPHL	SLAVE Phase Shift Register								xxxx xxxx	uuuu uuuu
14h	PWMPHH	SLAVE Phase Shift Register								xxxx xxxx	uuuu uuuu
15h	PWMRL	PWM Register Low Byte								xxxx xxxx	uuuu uuuu
16h	PWMRH	PWM Register High Byte								xxxx xxxx	uuuu uuuu
17h	CC1RL	Capture1/Compare1 Register 1 x Low Byte (LSB)								xxxx xxxx	uuuu uuuu
18h	CC1RH	Capture1/Compare1 Register 1 x High Byte (MSB)								xxxx xxxx	uuuu uuuu
19h	CC2RL	Capture2/Compare2 Register 2 x Low Byte (LSB)								xxxx xxxx	uuuu uuuu
1Ah	CC2RH	Capture2/Compare2 Register 2 x High Byte (MSB)								xxxx xxxx	uuuu uuuu
1Bh	CCDCON	CC2M3	CC2M2	CC2M1	CC2M0	CC1M3	CC1M2	CC1M1	CC1M0	0000 0000	0000 0000
1Ch	ADRESL	Least significant 8 bits of the right-shifted result (3)								xxxx xxxx	uuuu uuuu
1Dh	ADRESH	Most significant 2 bits of right-shifted result (3)								---- --xx	uuuu uuuu
1Eh	ADCON0	CHS5	CHS4	CHS3	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	ADON	0000 0000	0000 0000
1Fh	ADCON1	—	ADCS2	ADCS1	ADCS0	—	ADFM	VCFG1	VCFG0	-000 -000	-000 -000

Legend: — = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition shaded = unimplemented

Note 1: Other (non power-up) resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.

2: $\overline{MCLR}$ and WDT reset does not affect the previous value data latch. The IOCF bit will be cleared upon reset but will set again if the mismatch exists.

3: Results of ADC reading maybe left- or right-shifted. Results shown here are for right-shifted results.

TABLE 10-4: MCP19122/3 SPECIAL REGISTERS SUMMARY BANK 1

Addr.	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR Reset	Values on all other resets(1)
Bank 1											
80h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
81h	OPTION_REG	$\overline{RAPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h	PCL	Program Counter's (PC) Least Significant byte								0000 0000	0000 0000
83h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
84h	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISGPA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111
86h	TRISGPB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
87h	PIE1	TMR1GIE	ADIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
88h	PIE2	UVIE	OTIE	OCIE	OVIE	—	—	OVLOIE	UVLOIE	0000 --00	000 --00
89h											
8Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of program counter					---0 0000	---0 0000
8Bh	INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF(3)	0000 000x	0000 000u
8Ch	—	Unimplemented								–	–
8Dh	—	Unimplemented								–	–
8Eh	—	Unimplemented								–	–
8Fh	—	Unimplemented								–	–
90h	VINUVLO	—	—	—	—	UVLO3	UVLO2	UVLO1	UVLO0	---- xxxx	---- uuuu
91h	VINOVLO	—	—	—	—	OVLO3	OVLO2	OVLO1	OVLO0	---- xxxx	---- uuuu
92h	VINCON	UVLOEN	UVLOOUT	UVLOINTP	UVLOINTN	OVLOEN	OVLOOUT	OVLOINTP	OVLOINTN	0000 0000	0000 0000
93h	DAGCON	—	—	—	—	—	DAG2	DAG1	DAG0	---- -000	---- -000
94h	VOUTL	VOUT7	VOUT6	VOUT5	VOUT4	VOUT3	VOUT2	VOUT1	VOUT0	0000 0000	0000 0000
95h	VOUTH	—	—	—	—	—	—	VOUT9	VOUT8	---- --00	---- --00
96h	OSCTUNE	—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0	---0 0000	---0 0000
97h	—	Unimplemented								–	–
98h	CMPZCON	CMPZF3	CMPZF2	CMPZF1	CMPZF0	CMPZG3	CMPZG2	CMPZG1	CMPZG0	xxxx xxxx	uuuu uuuu
99h	VOTUVLO	—	—	—	—	OUV3	OUV2	OUV1	OUV0	---- xxxx	---- uuuu
9Ah	VOTOVLO	—	—	—	—	OOV3	OOV2	OOV1	OOV0	---- xxxx	---- uuuu
9Bh	DEADCON	HDLY3	HDLY2	HDLY1	HDLY0	LDLY3	LDLY2	LDLY1	LDLY0	1111 1111	1111 1111
9Ch	RAMPCON	RMPEN	—	—	RMP4	RMP3	RMP2	RMP1	RMP0	x--x xxxx	u--u uuuu
9Dh	OCCON	OCEN	OCLEB1	OCLEB0	OOC4	OOC3	OOC2	OOC1	OOC0	0xxx xxxx	0uuu uuuu
9Eh	CSGSCON	—	—	—	CSGS4	CSGS3	CSGS2	CSGS1	CSGS0	---x xxxx	---u uuuu
9Fh	RELEFF	MSDONE	RE6	RE5	RE4	RE3	RE2	RE1	RE0	0xxx xxxx	0uuu uuuu

Legend: — = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition, shaded = unimplemented

Note 1: Other (non power-up) resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.

2: GPA5 pull-up is enabled when pin is configured as $\overline{MCLR}$ in Configuration Word.

3: $\overline{MCLR}$ and WDT Reset does not affect the previous value data latch. The IOCF bit will be cleared upon reset but will set again if the mismatch exists.

TABLE 10-5: MCP19122/3 SPECIAL REGISTERS SUMMARY BANK 2

Addr.	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR Reset	Value on all other resets(1)
Bank 2											
100h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
101h	TMR0	Timer0 Module's Register								xxxx xxxx	uuuu uuuu
102h	PCL	Program Counter's (PC) Least Significant byte								0000 0000	0000 0000
103h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
104h	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
105h	WPUGPA	—	—	WPUA5	—	WCS1	WCS0	WPUA1	WPUA0	--1- 0011	--u- 00uu
106h	WPUGPB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	—	1111 111-	uuuu uuu-
107h	PE1	DECON	TOPO	HIDIS	LODIS	MEASEN	SPAN	UVTEE	OVTEE	0011 0000	0011 0000
108h	MODECON	CLMPSEL	VGNDEN	VDDEN	CNSG	EACLMP	MSC2	MSC1	MSC0	0000 0000	0000 0000
109h	—	Unimplemented								_	_
10Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of program counter					---0 0000	---0 0000
10Bh	INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF(2)	0000 000x	0000 000u
10Ch	—	Unimplemented								_	_
10Dh	—	Unimplemented								_	_
10Eh	—	Unimplemented								_	_
10Fh	—	Unimplemented								_	_
110h	SSPADD	ADD<7:0>								0000 0000	0000 0000
111h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
112h	SSPCON1	WCOL	SSPOV	SSPEN	CKP	SSPM>3:0>				0000 0000	0000 0000
113h	SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
114h	SSPCON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000
115h	SSPMSK	MSK<7:0>								1111 1111	1111 1111
116h	SSPSTAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	0000 0000
117h	SSPADD2	ADD2<7:0>								0000 0000	0000 0000
118h	SSPMSK2	MSK2<7:0>								1111 1111	1111 1111
119h	VREFCAL	—	—	—	VREF4	VREF3	VREF2	VREF1	VREF0	---0 0000	---0 0000
11Ah	VRFSCAL	—	—	—	VRFS4	VRFS3	VRFS2	VRFS1	VRFS0	---0 0000	---0 0000
11Bh	RAMPCAL	—	—	—	RAMP4	RAMP3	RAMP2	RAMP1	RAMP0	---x xxxx	---u uuuu
11Ch	CSRCAL	—	—	—	CSR4	CSR3	CSR2	CSR1	CSR0	---x xxxx	---u uuuu
11Dh	OVUVCAL	UVCO3	UVCO2	UVCO1	UVCO0	OVCO3	OVCO2	OVCO1	OVCO0	xxxx xxxx	uuuu uuuu
11Eh	DEMCAL	—	—	—	DEMOV4	DEMOV3	DEMOV2	DEMOV1	DEMOV0	---x xxxx	---u uuuu
11Fh	HCSOVCAL	—	HCSOV6	HCSOV5	HCSOV4	HCSOV3	HCSOV2	HCSOV1	HCSOV0	-xxx xxxx	-uuu uuuu

Legend: — = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition shaded = unimplemented

Note 1: Other (non power-up) resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.

2: $\overline{MCLR}$ and WDT reset does not affect the previous value data latch. The IOCF bit will be cleared upon reset but will set again if the mismatch exists.

TABLE 10-6: PIC18FXXXX SPECIAL REGISTERS SUMMARY BANK 3

Addr.	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR Reset	Values on all other resets(1)
Bank 3											
180h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	uuuu uuuu
181h	OPTION_REG	$\overline{RAPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
182h	PCL	Program Counter's (PC) Least Significant byte								0000 0000	0000 0000
183h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
184h	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
185h	IOCA	IOCA7	IOCA6	IOCA5	IOCA4	IOCA3	IOCA2	IOCA1	IOCA0	0000 0000	0000 0000
186h	IOCB	IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	0000 0000	0000 0000
187h	ANSELA	—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0	---- 1111	---- 1111
188h	ANSELB	—	—	ANSB5	ANSB4	—	ANSB2	ANSB1	—	--11 -11-	--11 -11-
189h	—	Unimplemented								—	—
18Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of program counter					---0 0000	---0 0000
18Bh	INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF(3)	0000 000x	0000 000u
18Ch	PORTICD(4)	In-Circuit Debug Port Register								—	—
18Dh	TRISICD(4)	In-Circuit Debug TRIS Register								—	—
18Eh	ICKBUG(4)	In-Circuit Debug Register								0--- ----	0--- ----
18Fh	BIGBUG(4)	In-Circuit Debug Breakpoint Register								---- ----	---- ----
190h	PMCON1	—	CALSEL	—	—	—	WREN	WR	RD	-0-- -000	-0-- -000
191h	PMCON2	Program Memory Control Register 2 (not a physical register)								---- ----	---- ----
192h	PMADRL	PMADRL7	PMADRL6	PMADRL5	PMADRL4	PMADRL3	PMADRL2	PMADRL1	PMADRL0	0000 0000	0000 0000
193h	PMADRH	—	—	—	—	PMADRH3	PMADRH2	PMADRH1	PMADRH0	---- 0000	---- 0000
194h	PMDATL	PMDATL7	PMDATL6	PMDATL5	PMDATL4	PMDATL3	PMDATL2	PMDATL1	PMDATL0	0000 0000	0000 0000
195h	PMDATH	—	—	PMDATH5	PMDATH4	PMDATH3	PMDATH2	PMDATH1	PMDATH0	--00 0000	--00 0000
196h	TTACAL	—	—	—	—	TTA3	TTA2	TTA1	TTA0	---- xxxx	---- uuuu
197h	OSCCAL	—	FACL6	FCAL5	FCAL4	FCAL3	FCAL2	FCAL1	FCAL0	-xxx xxxx	-uuu uuuu
198h	BGTCAL	—	—	—	—	BGT3	BGT2	BGT1	BGT0	---- xxxx	---- uuuu
199h	BGRCAL	—	—	BGR5	BGR4	BGR3	BGR2	BGR1	BGR0	--00 0000	--00 0000
19Ah	AVDDCAL	—	—	—	—	AVDD3	AVDD2	AVDD1	AVDD0	---- xxxx	---- uuuu
19Bh	VOURCAL	—	—	—	VOUR4	VOUR3	VOUR2	VOUR1	VOUR0	---x xxxx	---u uuuu
19Ch	DOVCAL	—	—	—	DOV4	DOV3	DOV2	DOV1	DOV0	---x xxxx	---u uuuu
19Dh	VEAOCAL	—	—	—	VEAO4	VEAO3	VEAO2	VEAO1	VEAO0	---x xxxx	---u uuuu
19Eh	BUFFCON	BNCHEN	DIGOEN	—	DSEL4	DSEL3	DSEL2	DSEL1	DSEL0	00-0 0000	00-0 0000
19Fh	—	Reserved								—	—

Legend: — = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition, shaded = unimplemented

Note 1: Other (non power-up) resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.

2: GPA5 pull-up is enabled when pin is configured as $\overline{MCLR}$ in Configuration Word.

3: $\overline{MCLR}$ and WDT Reset does not affect the previous value data latch. The IOCF bit will be cleared upon reset but will set again if the mismatch exists.

4: Only accessible when $\overline{DBGEN}$ = 0 and ICKBUG<INBUG> = 1.

10.3.0.1 OPTION Register

The OPTION register is a readable and writable register, which contains various control bits to configure:

- Timer0/WDT prescaler
- External GPA2/INT interrupt
- Timer0
- Weak pull-ups on PORTGPA and PORTGPB

Note 1: To achieve a 1:1 prescaler assignment for Timer0, assign the prescaler to the WDT by setting PSA bit to '1' of the OPTION register. See **Section 21.1.3 "Software Programmable Prescaler"**

REGISTER 10-2: OPTION_REG: OPTION REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RAPU}}$(1)	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
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 7 **$\overline{\text{RAPU}}$:** Port GPx Pull-up Enable bit
1 = Port GPx pull-ups are disabled
0 = Port GPx pull-ups are enabled

bit 6 **INTEDG:** Interrupt Edge Select bit
0 = Interrupt on rising edge of INT pin
1 = Interrupt on falling edge of INT pin

bit 5 **T0CS:** TMR0 Clock Source Select bit
1 = Transition on T0CKI pin
0 = Internal instruction cycle clock

bit 4 **T0SE:** TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on T0CKI pin
0 = Increment on low-to-high transition on T0CKI pin

bit 3 **PSA:** Prescaler Assignment bit
1 = Prescaler is assigned to WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0 **PS<2:0>:** Prescaler Rate Select bits

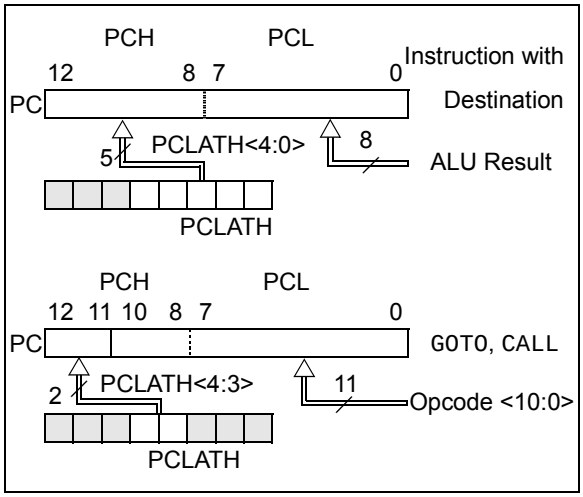
Bit Value	TMR0 Rate	WDT Rate
000	1: 2	1: 1
001	1: 4	1: 2
010	1: 8	1: 4
011	1: 16	1: 8
100	1: 32	1: 16
101	1: 64	1: 32
110	1: 128	1: 64
111	1: 256	1: 128

Note 1: Individual WPUx bit must also be enabled.

10.4 PCL and PCLATH

The Program Counter (PC) is 13 bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<12:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 10-2 shows the two situations for loading the PC. The upper example in Figure 10-2 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 10-2 shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 10-2: LOADING OF PC IN DIFFERENT SITUATIONS



10.4.1 MODIFYING PCL

Executing any instruction with the PCL register as the destination simultaneously causes the Program Counter PC<12:8> bits (PCH) to be replaced by the contents of the PCLATH register. This allows the entire content of the program counter to be changed by writing the desired upper 5 bits to the PCLATH register. When the lower 8 bits are written to the PCL register, all 13 bits of the program counter will change to the values contained in the PCLATH register and those being written to the PCL register.

10.4.2 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). Care should be exercised when jumping into a look-up table or program branch table (computed GOTO) by modifying the PCL register. Assuming that PCLATH is set to the table start address, if the table length is greater than 255 instructions or if the lower 8 bits of the memory address rolls over from 0xFFh to 0X00h in the middle of the table, then PCLATH must be incremented for each address rollover that occurs between the table beginning and the table location within the table.

For more information, refer to Application Note AN556 – *"Implementing a Table Read"* (DS00556).

10.4.3 COMPUTED FUNCTION CALLS

A computed function CALL allows programs to maintain tables of functions and provide another way to execute state machines or look-up tables. When performing a table read using a computed function CALL, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block).

If using the CALL instruction, the PCH<2:0> and PCL registers are loaded with the operand of the CALL instruction. PCH<4:3> is loaded with PCLATH<4:3>.

10.4.4 STACK

The MCP19122/3 has an 8-level x 13-bit wide hardware stack (refer to Figure 10-1). The stack space is not part of either program or data space and the Stack Pointer is not readable or writable. The PC is PUSHed onto the stack when CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

Note 1:	There are no Status bits to indicate Stack Overflow or Stack Underflow conditions.
2:	There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

10.5 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

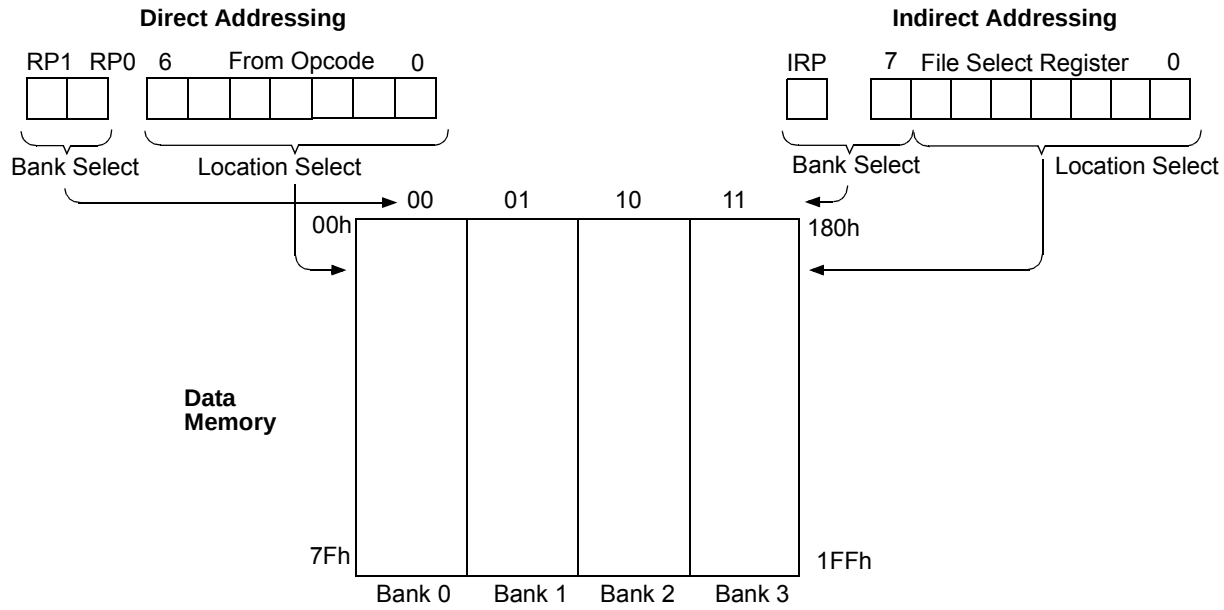
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register directly results in a no operation (although Status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR and the IRP bit of the STATUS register, as shown in Figure 10-3.

A simple program to clear RAM location 40h-7Fh using indirect addressing is shown in Example 10-2.

EXAMPLE 10-2: INDIRECT ADDRESSING

```
        MOVLW   0x40    ;initialize pointer
        MOVWF   FSR     ;to RAM
NEXT    CLRF    INDF    ;clear INDF register
        INCF    FSR     ;inc pointer
        BTFSS   FSR,7   ;all done?
        GOTO    NEXT    ;no clear next
CONTINUE                ;yes continue
```

FIGURE 10-3: DIRECT/INDIRECT ADDRESSING



For memory map detail, see Figure 10-2.

NOTES:

11.0 SPECIAL FEATURES OF THE CPU

The MCP19122/3 has a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power-saving features and offer code protection.

These features are:

- Reset
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- Oscillator selection
- Sleep
- Code protection
- ID Locations
- In-Circuit Serial Programming

The Power-up Timer (PWRT), which provides a fixed delay of 72 ms (nominal) on power-up only, is designed to keep the part in Reset while the power supply stabilizes. There is also circuitry to reset the device if a brown-out occurs, which can use the Power-up Timer to provide at least a 72 ms Reset. With these functions-on-chip, most applications need no external Reset circuitry.

The Sleep mode is designed to offer a very low-current Power-Down mode. The user can wake-up from Sleep through:

- External Reset
- Watchdog Timer Wake-up
- An interrupt

11.1 Configuration Bits

The Configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1') to select various device configurations as shown in Register 11-1. These bits are mapped in program memory location 2007h.

Note:	Address 2007h is beyond the user program memory space. It belongs to the special configuration memory space (2000h-3FFFh), which can be accessed only during programming. See "*Memory Programming Specification*" (DS41561) for more information.

REGISTER 11-1: CONFIGURATION WORD REGISTER

R/P-1	U-1	R/P-1	R/P-1	U-1	R/P-1
$\overline{\text{DEBUG}}$	—	WRT<1:0>		—	BOREN
bit 13					bit 8

U-1	R/P-1	R/P-1	R/P-1	R/P-1	U-1	U-1	U-1
—	$\overline{\text{CP}}$	MCLRE	$\overline{\text{PWRTE}}$	WDTE	—	—	—
bit 7							bit 0

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as ‘1’
‘0’ = Bit is cleared	‘1’ = Bit is set	-n = Value when blank or after Bulk Erase

bit 13 **$\overline{\text{DEBUG}}$:** Debug Mode Enable bit(2)
1 = Background debugger is disabled
0 = Background debugger is enabled

bit 12 Unimplemented: Read as ‘1’.

bit 11-10 **WRT<1:0>:** Flash Program Memory Self Write Enable bits
11 = Write protection off
10 = 000h to FFh write-protected, 100h to 3FFh may be modified by PMCON1 control
01 = 000h to 1FFh write-protected, 200h to 3FFh may be modified by PMCON1 control
00 = 000h to 3FFh write-protected, entire program is write-protected

bit 9 Unimplemented: Read as ‘1’.

bit 8 **BOREN:** Brown-out Reset Enable bit
1 = BOR enabled during operation and disabled in Sleep
0 = BOR disabled

bit 7 Unimplemented: Read as ‘1’.

bit 6 **$\overline{\text{CP}}$:** Code Protection bit
1 = Program memory code protection is disabled
0 = Program memory code protection is enabled

bit 5 **MCLRE:** $\overline{\text{MCLR}}/V_{\overline{PP}}$ Pin Function Select bit
1 = $\overline{\text{MCLR}}$ pin is $\overline{\text{MCLR}}$ function and weak internal pull-up is enabled
0 = $\overline{\text{MCLR}}$ pin is input function, $\overline{\text{MCLR}}$ function is internally disabled

bit 4 **$\overline{\text{PWRTE}}$:** Power-up Timer Enable bit(1)
1 = PWRT disabled
0 = PWRT enabled

bit 3 **WDTE:** Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled

bit 2-0 Unimplemented: Read as ‘1’.

Note 1: Enabling Brown-out Reset does not automatically enable Power-up Timer.

2: The Configuration bit is managed automatically by the device development tools. The user should not attempt to manually write this bit location. However, the user should ensure that this location has been programmed to a ‘1’ and the device checksum is correct for proper operation of production software.

11.2 Code Protection

Code protection allows the device to be protected from unauthorized access. Internal access to the program memory is unaffected by any code protection setting.

11.2.1 PROGRAM MEMORY PROTECTION

The entire program memory space is protected from external reads and writes by the $\overline{CP}$ bit in the Configuration Word. When $\overline{CP}$ = 0, external reads and writes of program memory are inhibited and a read will return all '0's. The CPU can continue to read program memory, regardless of the protection bit settings. Writing the program memory is dependent upon the write protection setting.

11.3 Write Protection

Write protection allows the device to be protected from unintended self-writes. Applications, such as boot loader software, can be protected while allowing other regions of the program memory to be modified.

The WRT<1:0> bits in the Configuration Word define the size of the program memory block that is protected.

11.4 ID Locations

Four memory locations (2000h-2003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are not accessible during normal execution but are readable and writable during Program/Verify mode. Only the Least Significant 7 bits of the ID locations are reported when using MPLAB Integrated Development Environment (IDE).

NOTES:

12.0 RESETS

The reset logic is used to place the MCP19122/3 into a known state. The source of the reset can be determined by using the device status bits.

There are multiple ways to reset this device:

- Power-on Reset (POR)
- WDT Reset during normal operation
- WDT Reset during Sleep
- $\overline{\text{MCLR}}$ Reset during normal operation
- $\overline{\text{MCLR}}$ Reset during Sleep
- Brown-out Reset (BOR)

To allow V_{DD} to stabilize, an optional power-up timer can be enabled to extend the Reset time after a POR event.

Some registers are not affected in any Reset condition; their status is unknown on POR and unchanged in any other Reset. Most other registers are reset to a "Reset state" on:

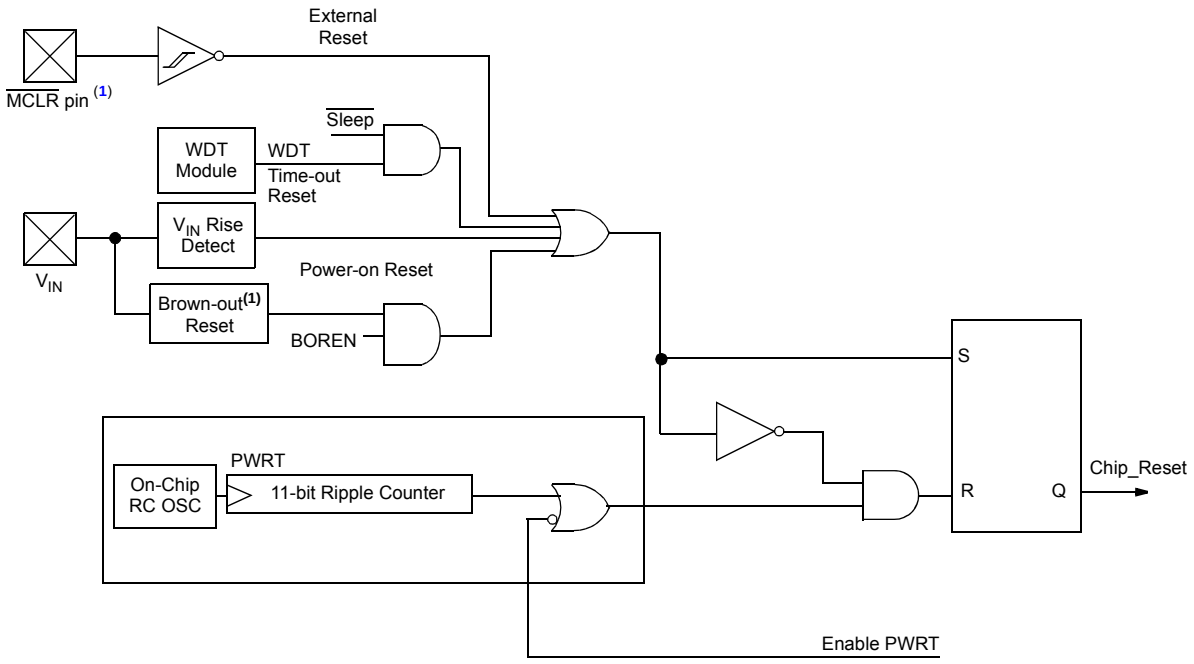
- Power-on Reset
- $\overline{\text{MCLR}}$ Reset
- $\overline{\text{MCLR}}$ Reset during Sleep
- WDT Reset
- Brown-out Reset (BOR)

WDT wake-up does not cause register resets in the same manner as a WDT Reset since wake-up is viewed as the resumption of normal operation. $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are set or cleared differently in different Reset situations, as indicated in Table 12-1. Software can use these bits to determine the nature of the Reset. See Table 12-2 for a full description of Reset states of all registers.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 12-1.

The $\overline{\text{MCLR}}$ Reset path has a noise filter to detect and ignore small pulses. See Section 5.0, Digital Electrical Characteristics for pulse-width specifications.

FIGURE 12-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



Note 1: Refer to the Configuration Word register (Register 11-1).

TABLE 12-1: TIME-OUT IN VARIOUS SITUATIONS

Oscillator Configuration	Power-up		Brown-out Reset		Wake-up from Sleep
	$\overline{PWRTE}$ = 0	$\overline{PWRTE}$ = 1	$\overline{PWRTE}$ = 0	$\overline{PWRTE}$ = 1	
INTOSC	TPWRT	—	TPWRT	—	—

TABLE 12-2: STATUS/PCON BITS AND THEIR SIGNIFICANCE

$\overline{POR}$	$\overline{BOR}$	$\overline{TO}$	$\overline{PD}$	Condition
0	x	1	1	Power-on Reset
u	0	1	1	Brown-out Reset
u	u	0	u	WDT Reset
u	u	0	0	WDT Wake-up
u	u	u	u	$\overline{MCLR}$ Reset during normal operation
u	u	1	0	$\overline{MCLR}$ Reset during Sleep

Legend: u = unchanged, x = unknown

12.1 Power-on Reset (POR)

The on-chip POR circuit holds the chip in Reset until V_{DD} has reached a high enough level for proper operation. To take advantage of the POR, simply connect the $\overline{MCLR}$ pin through a resistor to V_{DD}. This will eliminate external RC components usually needed to create Power-on Reset. If the BOR is enabled, the maximum rise time specification does not apply. The BOR circuitry will keep the device in Reset until V_{DD} reaches V_{BOR} (see **Section 12.3 "Brown-out Reset (BOR)"**).

When the device starts normal operation (exits the Reset condition), device operating parameters (i.e., voltage, frequency, temperature, etc.) must be met to ensure proper operation. If these conditions are not met, the device must be held in Reset until the operating conditions are met.

12.2 $\overline{MCLR}$

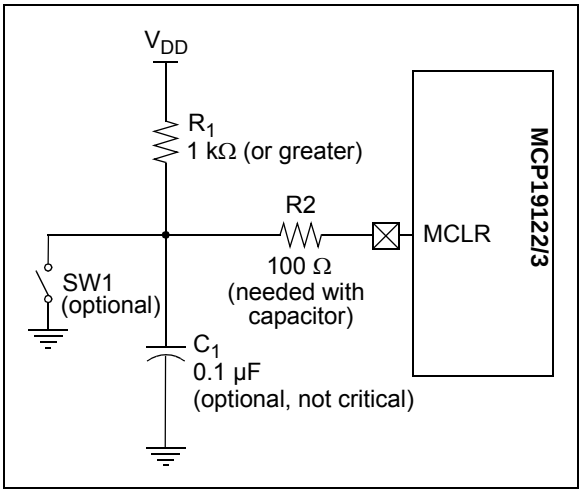
MCP19122/3 has a noise filter in the $\overline{MCLR}$ Reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive $\overline{MCLR}$ pin low.

Voltages applied to the $\overline{MCLR}$ pin that exceed its specification can result in both $\overline{MCLR}$ Resets and excessive current beyond the device specification during the ESD event. For this reason, Microchip recommends that the $\overline{MCLR}$ pin no longer be tied directly to V_{DD}. The use of an RC network, as shown in Figure 12-2, is suggested.

An internal $\overline{MCLR}$ option is enabled by clearing the MCLRE bit in the Configuration Word register. When MCLRE = 0, the Reset signal to the chip is generated internally. When the MCLRE = 1, the $\overline{MCLR}$ pin becomes an external Reset input. In this mode, the $\overline{MCLR}$ pin has a weak pull-up to V_{DD}.

FIGURE 12-2: RECOMMENDED $\overline{MCLR}$ CIRCUIT



12.3 Brown-out Reset (BOR)

The BOREN bit in the Configuration Word register enables or disables the BOR mode. See Register 11-1 for the Configuration Word definition.

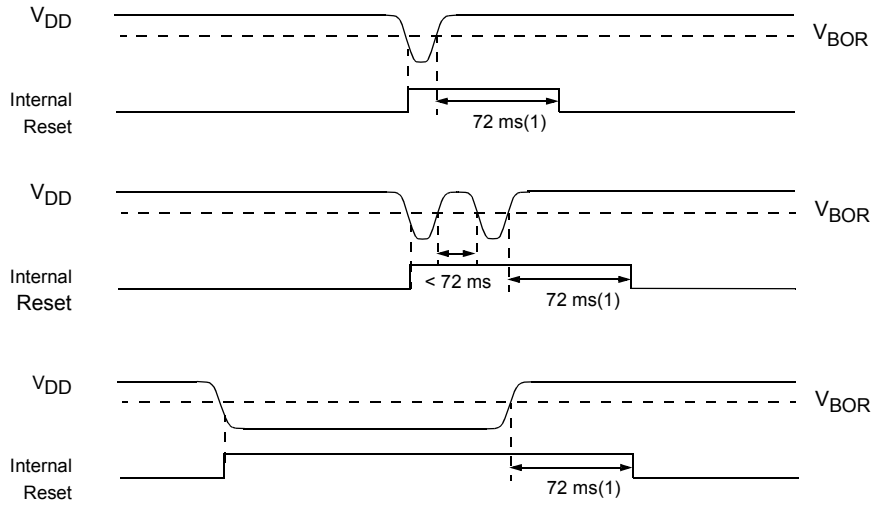
A brown-out occurs when V_{DD} falls below V_{BOR} for greater than 100 µs minimum. On a Reset (Power-On, Brown-Out, Watchdog Timer, etc.), the chip will remain in Reset until V_{DD} rises above V_{BOR} (refer to Figure 12-3). If enabled, the Power-up Timer will be invoked by the Reset and will keep the chip in Reset an additional 72 ms. During power-up, it is recommended that the BOR configuration bit is enabled, holding the MCU in Reset (OSC turned off and no code execution) until V_{DD} exceeds the V_{BOR} threshold. Users have the option of adding an additional 72 ms delay be clearing the PWRTE bit. At this time, the V_{DD} voltage level is high enough to operate the MCU functions only; all other device functionality is not operational. This is independent of the value of V_{IN}, which is typically $V_{DD} + V_{DROPOUT}$. During power-down with BOR enabled, the MCU operation will be held in Reset when V_{DD} falls below the V_{BOR} threshold. With BOR disabled or while operating in Sleep mode, the POR will hold the part in Reset when V_{DD} falls below the V_{POR} threshold.

A brown-out occurs when V_{IN} falls below V_{BOR} for greater than parameter T_{BOR}. The brown-out condition will reset the device. This will occur regardless of V_{IN} slew rate. A Brown-out Reset may not occur if V_{IN} falls below V_{BOR} for less than parameter T_{BOR}.

Note:	The Power-up Timer is enabled by the $\overline{\text{PWRTE}}$ bit in the Configuration Word register.

If V_{DD} drops below V_{BOR} while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be re-initialized. Once V_{DD} rises above V_{BOR}, the Power-up Timer will execute a 72 ms Reset.

FIGURE 12-3: BROWN-OUT SITUATIONS



Note 1: 72 ms delay only if $\overline{\text{PWRTE}}$ bit is programmed to '0'.

12.4 Power-up Timer (PWRT)

The Power-up Timer provides a fixed 64 ms (nominal) time-out on power-up only, from POR Reset. The Power-up Timer operates from an internal RC oscillator. The chip is kept in Reset as long as PWRT is active. The PWRT delay allows the V_{DD} to rise to an acceptable level. A Configuration bit ($\overline{PWRTE}$), can disable (if set) or enable (if cleared or programmed) the Power-up Timer.

The Power-up Timer delay will vary from chip to chip due to:

- V_{DD} variation
- Temperature variation
- Process variation

The Power-Up Timer optionally delays device execution after a POR event. This timer is typically used to allow V_{DD} to stabilize before allowing the device to start running.

The Power-up Timer is controlled by the $\overline{PWRTE}$ bit of the Configuration Word.

Note:	Voltage spikes below V_{SS} at the $\overline{MCLR}$ pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a "low" level to the $\overline{MCLR}$ pin, rather than pulling this pin directly to V_{SS}.

12.5 Watchdog Timer (WDT) Reset

The Watchdog Timer generates a Reset if the firmware does not issue a `CLRWDT` instruction within the time-out period. The $\overline{TO}$ and $\overline{PD}$ bits in the STATUS register are changed to indicate the WDT Reset. See Section 15.0, Watchdog Timer (WDT) for more information.

12.6 Start-up Sequence

Upon the release of a POR, the following must occur before the device will begin executing:

- Power-up Timer runs to completion (if enabled)
- Oscillator start-up timer runs to completion
- $\overline{MCLR}$ must be released (if enabled)

The total time-out will vary based on $\overline{PWRTE}$ bit status. For example, with $\overline{PWRTE}$ bit erased (PWRT disabled), there will be no time-out at all. Figures 12-4, 12-5 and 12-6 depict time-out sequences.

Since the time-outs occur from the POR pulse, if $\overline{MCLR}$ is kept low long enough, the time-outs will expire. Then, bringing $\overline{MCLR}$ high will begin execution immediately (see Figure 12-5). This is useful for testing purposes or to synchronize more than one MCP19122/3 device operating in parallel.

12.6.1 POWER CONTROL (PCON) REGISTER

The Power Control register PCON (address 8Eh) has two Status bits to indicate what type of Reset occurred last.

FIGURE 12-4: TIME-OUT SEQUENCE ON POWER-UP (DELAYED $\overline{MCLR}$): CASE 1

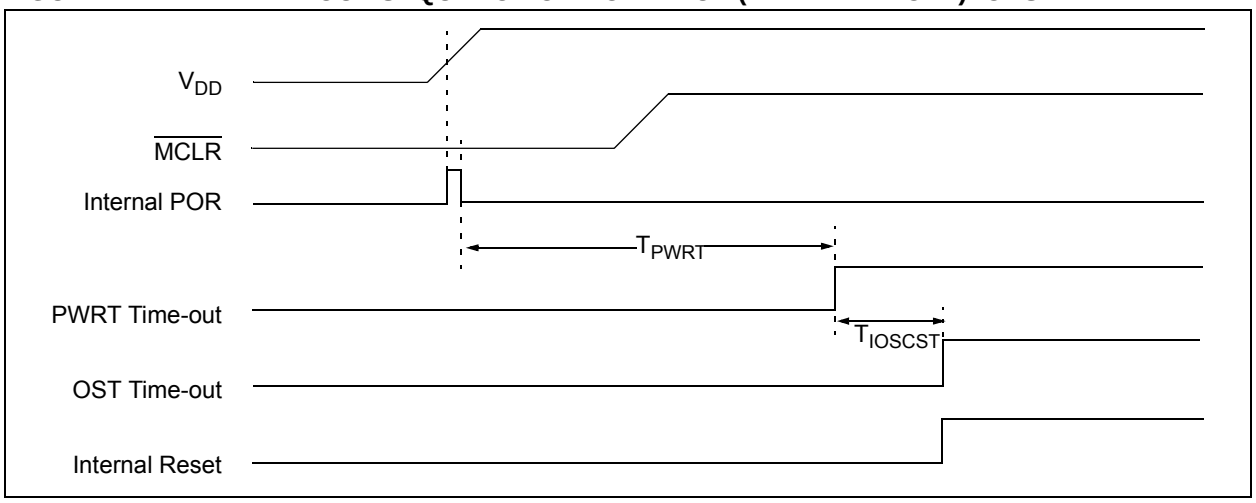


FIGURE 12-5: TIME-OUT SEQUENCE ON POWER-UP (DELAYED $\overline{\text{MCLR}}$): CASE 2

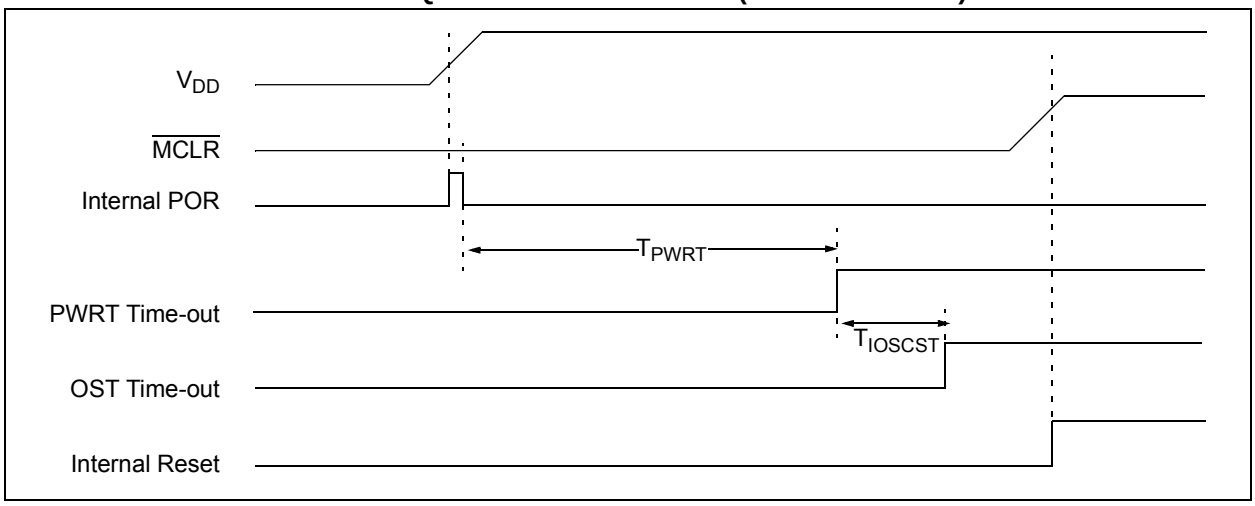
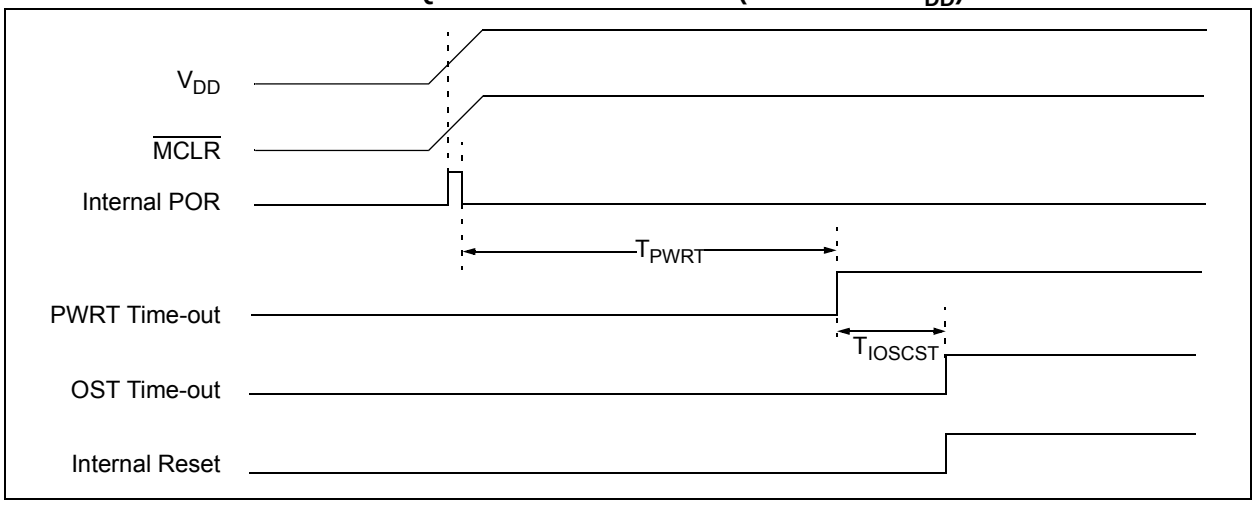


FIGURE 12-6: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ WITH V_{DD})



12.7 Determining the Cause of a Reset

Upon any Reset, multiple bits in the STATUS and PCON register are updated to indicate the cause of the Reset. Table 12-3 and Table 12-4 show the Reset conditions of these registers.

TABLE 12-3: RESET STATUS BITS AND THEIR SIGNIFICANCE

$\overline{POR}$	$\overline{BOR}$	$\overline{TO}$	$\overline{PD}$	Condition
0	x	1	1	Power-on Reset
u	0	1	1	Brown-out Reset
u	u	0	u	WDT Reset
u	u	0	0	WDT Wake-up from Sleep
u	u	1	0	Interrupt Wake-up from Sleep
u	u	u	u	$\overline{MCLR}$ Reset during normal operation
u	u	1	0	$\overline{MCLR}$ Reset during Sleep
0	u	0	x	Not allowed. $\overline{TO}$ is set on POR
0	u	x	0	Not allowed. $\overline{PD}$ is set on POR

TABLE 12-4: RESET CONDITION FOR SPECIAL REGISTERS (Note 2)

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	0000h	0001 1xxx	---- --0u
Brown-out Reset	000	0001 1xxx	---- --u0
$\overline{MCLR}$ Reset during normal operation	0000h	000u uuuu	---- --uu
$\overline{MCLR}$ Reset during Sleep	0000h	0001 0uuu	---- --uu
WDT Reset	0000h	0000 uuuu	---- --uu
WDT Wake-up from Sleep	PC + 1	uuu0 0uuu	---- --uu
Interrupt Wake-up from Sleep	PC + 1(1)	uuu1 0uuu	---- --uu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0'.

Note 1: When the wake-up is due to an interrupt and Global Enable bit (GIE) is set, the return address is pushed on the stack and PC is loaded with the interrupt vector (0004h) after execution of PC + 1.

2: If a Status bit is not implemented, that bit will be read as '0'.

12.8 Power Control (PCON) Register

The PCON register bits are shown in Register 12-1.

The Power Control (PCON) register contains flag bits to differentiate between a:

- Power-on Reset ($\overline{POR}$)
- Over Temperature ($\overline{OT}$)
- Brown-out Reset ($\overline{BOR}$)

REGISTER 12-1: PCON: POWER CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$
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 7-2 **Unimplemented:** Read as '0'

bit 1 **$\overline{POR}$:** Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

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

TABLE 12-5: SUMMARY OF REGISTERS ASSOCIATED WITH RESETS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
PCON	—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$	97
STATUS	IPR	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	77

Legend: — = unimplemented bit, reads as ‘0’. Shaded cells are not used by Resets.

Note 1: Other (non Power-up) Resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.

NOTES:

13.0 INTERRUPTS

The MCP19122/3 has multiple sources of interrupt:

- External Interrupt (INT pin)
- Interrupt-on-Change (IOC) Interrupts
- Timer0 Overflow Interrupt
- Timer1 Overflow Interrupt
- Timer1 Gate Interrupt
- Timer2 Match Interrupt
- ADC Interrupt
- SSP
- BCL
- Input Undervoltage Interrupt
- Input Overvoltage Interrupt
- System Output Overvoltage Interrupt
- System Output Undervoltage Interrupt
- System Output Overcurrent Interrupt
- Overtemperature
- CC1
- CC2

The Interrupt Control register (INTCON) and Peripheral Interrupt Request Registers (PIRx) record individual interrupt requests in flag bits. The INTCON register also has individual and global interrupt enable bits.

The Global Interrupt Enable bit, GIE of the INTCON register, enables (if set) all unmasked interrupts, or disables (if cleared) all interrupts. Individual interrupts can be disabled through their corresponding enable bits in the INTCON register and PIEx registers. GIE is cleared on Reset.

When an interrupt is serviced, the following actions occur automatically:

- The GIE is cleared to disable any further interrupt.
- The return address is pushed onto the stack.
- The PC is loaded with 0004h.

The Return from Interrupt instruction, `RETFIE`, exits the interrupt routine, as well as sets the GIE bit, which re-enables unmasked interrupts.

The following interrupt flags are contained in the INTCON register:

- INT Pin Interrupt
- Interrupt-on-Change (IOC) Interrupts
- Timer0 Overflow Interrupt

The peripheral interrupt flags are contained in the PIR1 and PIR2 registers. The corresponding interrupt enable bit is contained in the PIE1 and PIE2 registers.

For external interrupt events, such as the INT pin or PORTGPx change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends upon when the interrupt event occurs (see Figure 13-2). The latency is the same for one or two-cycle instructions. Once in the Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid multiple interrupt requests.

Note 1:	Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.
2:	When an instruction that clears the GIE bit is executed, any interrupts that were pending for execution in the next cycle are ignored. The interrupts, which were ignored, are still pending to be serviced when the GIE bit is set again.

For additional information on Timer1, Timer2, comparators, ADC, CCD modules, refer to the respective peripheral section.

13.0.1 GPA2/INT INTERRUPT

The external interrupt on the GPA2/INT pin is edge-triggered; either on the rising edge if the INTEDG bit of the OPTION register is set, or the falling edge, if the INTEDG bit is clear. When a valid edge appears on the GPA2/INT pin, the INTF bit of the INTCON register is set. This interrupt can be disabled by clearing the INTE control bit of the INTCON register. The INTF bit must be cleared by software in the Interrupt Service Routine before re-enabling this interrupt. The GPA2/INT interrupt can wake-up the processor from Sleep, if the INTE bit was set prior to going into Sleep. See **Section 14.0 "Power-Down Mode (Sleep)"** for details on Sleep.

Note:	The ANSELx register must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0' and cannot generate an interrupt.

13.0.2 TIMER0 INTERRUPT

An overflow (FFh → 00h) in the TMR0 register will set the T0IF bit of the INTCON register. The interrupt can be enabled/disabled by setting/clearing T0IE bit of the INTCON register. See **Section 21.0 "Timer0 Module"** for operation of the Timer0 module.

13.0.3 PORTGPX INTERRUPT-ON-CHANGE

An input change on PORTGPx sets the IOCIF bit of the INTCON register. The interrupt can be enabled/disabled by setting/clearing the IOCIE bit of the INTCON register. Plus, individual pins can be configured through the IOC register.

Note:	If a change on the I/O pin should occur when any PORTGPx operation is being executed, then the IOCIF interrupt flag may not get set.

FIGURE 13-1: INTERRUPT LOGIC

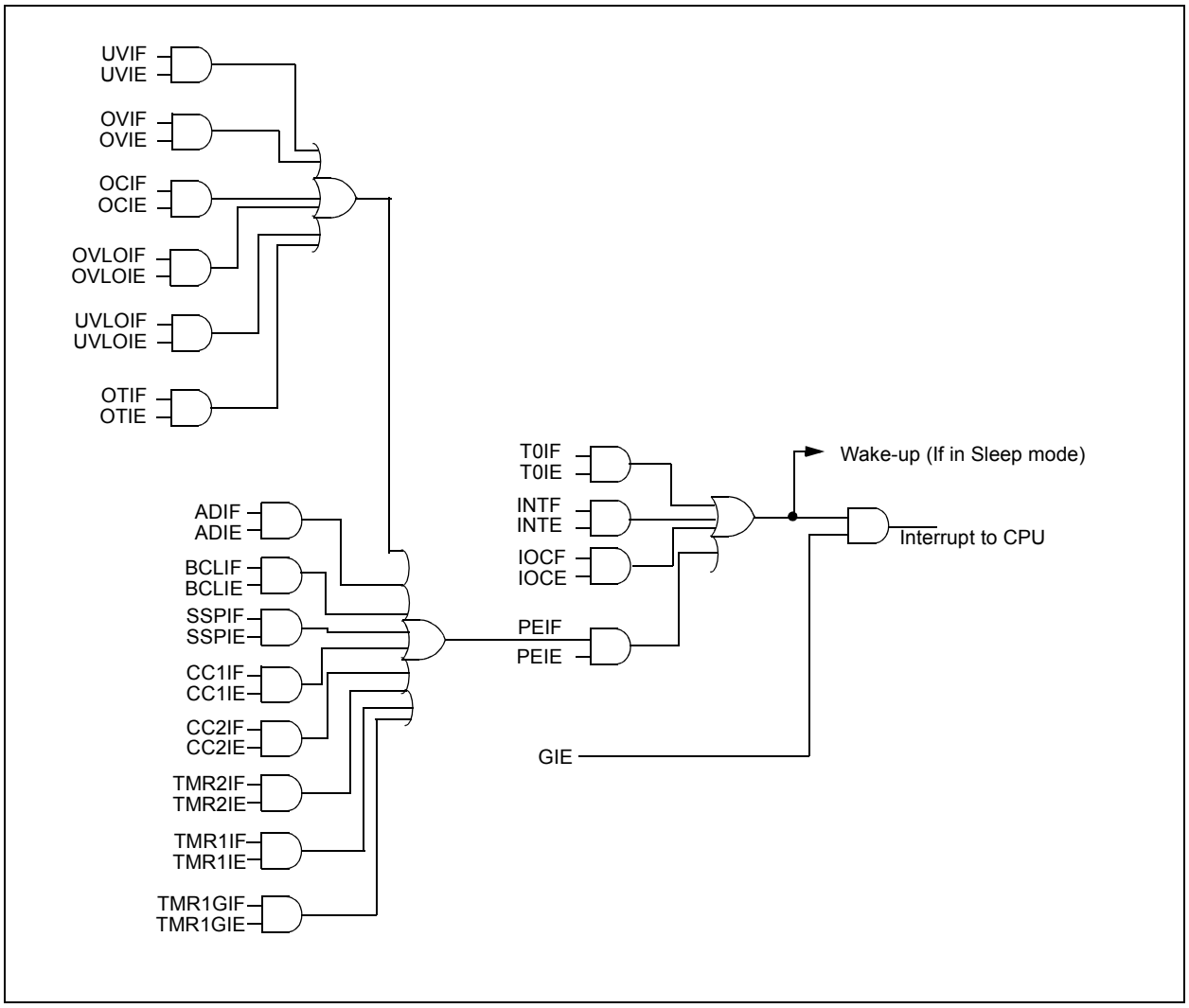
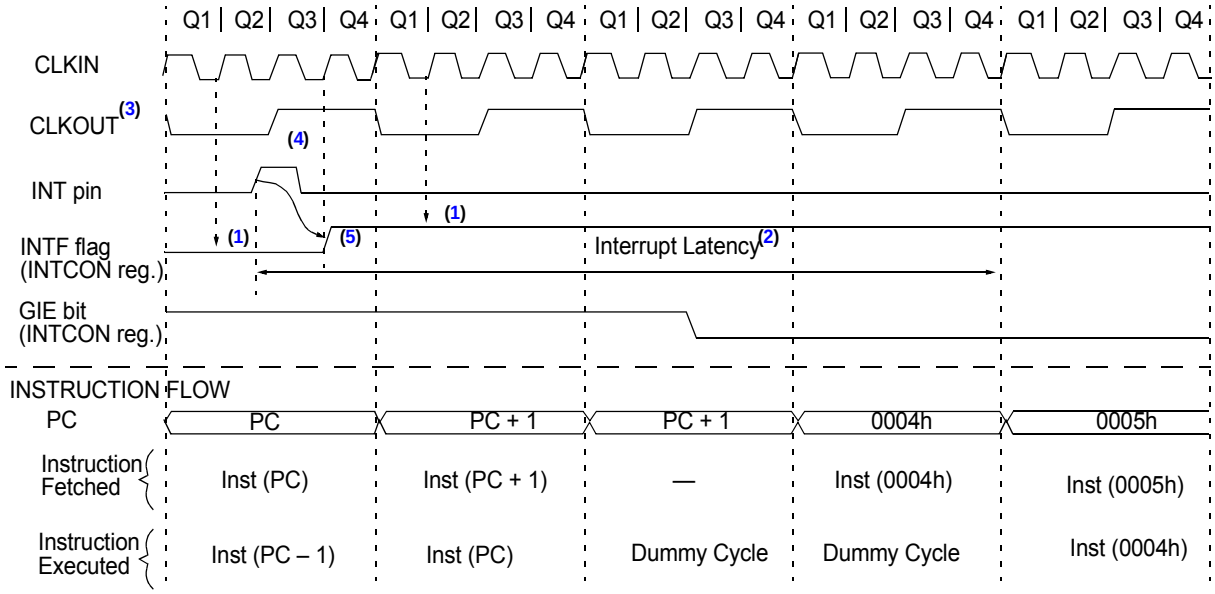


FIGURE 13-2: INT PIN INTERRUPT TIMING



Note 1: INTF flag is sampled here (every Q1).

2: Asynchronous interrupt latency = 3-4 T_{CY}. Synchronous latency = 3 T_{CY}, where T_{CY} = instruction cycle time. Latency is the same whether Inst (PC) is a single cycle or a 2-cycle instruction.

3: CLKOUT is available only in INTOSC and RC Oscillator modes.

4: For minimum width of INT pulse, refer to AC specifications in Section 5.0, Digital Electrical Characteristics.

5: INTF is enabled to be set any time during the Q4-Q1 cycles.

13.1 Interrupt Control Registers

13.1.1 INTCON REGISTER

The INTCON register is a readable and writable register, that contains the various enable and flag bits for the TMR0 register overflow, interrupt-on-change and external INT pin interrupts.

Note:	Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 13-1: INTCON: INTERRUPT CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF
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 7 **GIE:** Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts

bit 6 **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5 **T0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4 **INTE:** INT External Interrupt Enable bit
1 = Enables the INT external interrupt
0 = Disables the INT external interrupt

bit 3 **IOCE:** Interrupt-on-Change Enable bit(1)
1 = Enables the interrupt-on-change
0 = Disables the interrupt-on-change

bit 2 **T0IF:** TMR0 Overflow Interrupt Flag bit(2)
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1 **INTF:** External Interrupt Flag bit
1 = The external interrupt occurred (must be cleared in software)
0 = The external interrupt did not occur

bit 0 **IOCF:** Interrupt-on-Change Interrupt Flag bit
1 = When at least one of the interrupt-on-change pins changed state
0 = None of the interrupt-on-change pins have changed state

Note 1: IOC register must also be enabled.
2: T0IF bit is set when TMR0 rolls over. TMR0 is unchanged on Reset and should be initialized before clearing T0IF bit.

13.1.1.1 PIE1 Register

The PIE1 register contains the Peripheral Interrupt Enable bits, as shown in Register 13-1.

Note 1: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 13-1: PIE1: PERIPHERAL INTERRUPT ENABLE REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TMR1GIE	ADIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE
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 7 **TMR1GIE:** Timer1 Gate Interrupt Enable bit
1 = Enables the Timer1 gate interrupt
1 = Disables the Timer1 gate interrupt

bit 6 **ADIE:** ADC Interrupt Enable bit
1 = Enables the ADC interrupt
0 = Disables the ADC interrupt

bit 5 **BCLIE:** MSSP Bus Collision Interrupt Enable bit
1 = Enables the MSSP Bus Collision Interrupt
0 = Disables the MSSP Bus Collision Interrupt

bit 4 **SSPIE:** Synchronous Serial Port (MSSP) Interrupt Enable bit
1 = Enables the MSSP interrupt
0 = Disables the MSSP interrupt

bit 3 **CC2IE:** Capture2/Compare2 Interrupt Enable bit
1 = Enables the Capture2/Compare2 interrupt
0 = Disables the Capture2/Compare2 interrupt

bit 2 **CC1IE:** Capture1/Compare1 Interrupt Enable bit
1 = Enables the Capture1/Compare1 interrupt
0 = Disables the Capture1/Compare1 interrupt

bit 1 **TMR2IE:** Timer2 Interrupt Enable
1 = Enables the Timer2 interrupt
0 = Disables the Timer2 interrupt

bit 2 **TMR1IE:** Timer1 Interrupt Enable
1 = Enables the Timer1 interrupt
0 = Disables the Timer1 interrupt

13.1.1.2 PIE2 Register

The PIE2 register contains the Peripheral Interrupt Enable bits, as shown in Register 13-2.

Note 1: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 13-2: PIE2: PERIPHERAL INTERRUPT ENABLE REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
UVIE	OTIE	OCIE	OVIE	—	—	OVLOIE	UVLOIE
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 7 **UVIE:** Output Undervoltage Interrupt enable bit
1 = Enables the UV interrupt
0 = Disables the UV interrupt

bit 6 **OTIE:** Overtemperature Interrupt enable bit
1 = Enables over temperature interrupt
0 = Disables over temperature interrupt

bit 5 **OCIE:** Output Overcurrent Interrupt enable bit
1 = Enables the OC interrupt
0 = Disables the OC interrupt

bit 4 **OVIE:** Output Overvoltage Interrupt enable bit
1 = Enables the OV interrupt
0 = Disables the OV interrupt

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **OVLOIE:** V_{IN} Overvoltage Lock Out Interrupt Enable bit
1 = Enables the V_{IN} OVLO interrupt
0 = Disables the V_{IN} OVLO interrupt

bit 0 **UVLOIE:** V_{IN} Undervoltage Lock Out Interrupt Enable bit
1 = Enables the V_{IN} UVLO interrupt
0 = Disables the V_{IN} UVLO interrupt

13.1.1.3 PIR1 Register

The PIR1 register contains the Peripheral Interrupt Flag bits, as shown in Register 13-3.

Note 1:	Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 13-3: PIR1: PERIPHERAL INTERRUPT FLAG REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TMR1GIF	ADIF	BCLIF	SSPIF	CC2IF	CC1IE	TMR2IF	TMR1IF
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 7 **TMR1GIF:** Timer1 Gate Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 6 **ADIF:** ADC Interrupt Flag bit
1 = ADC conversion complete
0 = ADC conversion has not completed or has not been started

bit 5 **BCLIF:** MSSP Bus Collision Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 4 **SSPIF:** Synchronous Serial Port (MSSP) Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 3 **CC2IF:** Capture2/Compare2 Interrupt Flag bit
1 = Capture or Compare has occurred
0 = Capture or Compare has not occurred

bit 2 **CC1IF:** Capture2/Compare2 Interrupt Flag bit
1 = Capture or Compare has occurred
0 = Capture or Compare has not occurred

bit 1 **TMR2IF:** Timer2 to PR2 Match Interrupt Flag
1 = Timer2 to PR2 match occurred (must be cleared in software)
0 = Timer2 to PR2 match did not occur

bit 0 **TMR1IF:** Timer1 Interrupt Flag
1 = Timer1 rolled over (must be cleared in software)
0 = Timer1 has not rolled over

13.1.1.4 PIR2 Register

The PIR2 register contains the Peripheral Interrupt Flag bits, as shown in Register 13-4.

Note 1:	Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 13-4: PIR2: PERIPHERAL INTERRUPT FLAG REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
UVIF	OTIF	OCIF	OVIF	—	—	OVLOIF	UVLOIF
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 7 **UVIF:** Output Undervoltage error interrupt flag bit
1 = Output undervoltage error has occurred
0 = Output undervoltage error has not occurred

bit 6 **OTIF:** Overtemperature interrupt flag bit
1 = Overtemperature error has occurred
0 = Overtemperature error has not occurred

bit 5 **OCIF:** Output over current error interrupt flag bit
1 = Output overcurrent error has occurred
0 = Output overcurrent error has not occurred

bit 4 **OVIF:** Output overvoltage error interrupt flag bit
1 = Output overvoltage error has occurred
0 = Output overvoltage error has not occurred

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **OVLOIF:** V_{IN} Overvoltage Lock Interrupt Flag bit
With OVLOINTP bit set
1 = A V_{IN} NOT overvoltage to V_{IN} overvoltage edge has been detected
0 = A V_{IN} NOT overvoltage to V_{IN} overvoltage edge has NOT been detected
With OVLOINTN bit set
1 = A V_{IN} overvoltage to V_{IN} NOT overvoltage edge has been detected
0 = A V_{IN} overvoltage to V_{IN} NOT overvoltage edge has NOT been detected

bit 0 **UVLOIF:** V_{IN} Undervoltage Lock Out Interrupt Flag bit
With UVLOINTP bit set
1 = A V_{IN} NOT undervoltage to V_{IN} undervoltage edge has been detected
0 = A V_{IN} NOT undervoltage to V_{IN} undervoltage edge has NOT been detected
With UVLOINTN bit set
1 = A V_{IN} undervoltage to V_{IN} NOT undervoltage edge has been detected
0 = A V_{IN} undervoltage to V_{IN} NOT undervoltage edge has NOT been detected

TABLE 13-5: SUMMARY OF REGISTERS ASSOCIATED WITH INTERRUPTS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	102
OPTION_REG	$\overline{\text{RAPU}}$	INTEDG	T0CE	T0SE	PSA	PS2	PS1	PS0	83
PIE1	—	ADIE	BCLIE	SSPIE	—	—	TMR2IE	TMR1IE	103
PIE2	UVIE	—	OCIE	OVIE	—	—	VINIE	—	104
PIR1	—	ADIF	BCLIF	SSPIF	—	—	TMR2IF	TMR1IF	105
PIR2	UVIF	—	OCIF	OVIF	—	—	VINIF	—	106

Legend: — = unimplemented locations read as '0'. Shaded cells are not used by Interrupts.

13.2 Context Saving During Interrupts

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt (e.g., W and STATUS registers). This must be implemented in software.

Temporary holding registers W_TEMP and STATUS_TEMP should be placed in the last 16 bytes of GPR. These 16 locations are common to all banks and do not require banking. This makes context save and restore operations simpler. The code shown in Example 13-1 can be used to:

- Store the W register
- Store the STATUS register
- Execute the ISR code
- Restore the Status (and Bank Select Bit register)
- Restore the W register

Note: The MCP19122/3 device does not require saving the PCLATH. However, if computed GOTOs are used in both the ISR and the main code, the PCLATH must be saved and restored in the ISR.

EXAMPLE 13-1: SAVING STATUS AND W REGISTERS IN RAM

```
    MOVWF   W_TEMP              ;Copy W to TEMP register
    SWAPF   STATUS,W            ;Swap status to be saved into W
                                ;Swaps are used because they do not affect the status bits
    MOVWF   STATUS_TEMP         ;Save status to bank zero STATUS_TEMP register
    :
    :(ISR)                      ;Insert user code here
    :
    SWAPF   STATUS_TEMP,W       ;Swap STATUS_TEMP register into W
                                ;(sets bank to original state)
    MOVWF   STATUS              ;Move W into STATUS register
    SWAPF   W_TEMP,F            ;Swap W_TEMP
    SWAPF   W_TEMP,W            ;Swap W_TEMP into W
```

NOTES:

14.0 POWER-DOWN MODE (SLEEP)

The Power-down mode is entered by executing a SLEEP instruction.

Upon entering SLEEP, the following conditions exist:

- WDT will be cleared but keeps running, if enabled for operation during SLEEP.
- $\overline{PD}$ bit of the STATUS register is cleared.
- $\overline{TO}$ bit of the STATUS register is set.
- CPU clock is disabled.
- The ADC is inoperable due to the absence of the 4V LDO power (AV_{DD}) while the ADC reference is set to AV_{DD}. To minimize sleep current the ADC reference must be set to the default AV_{DD}.
- I/O Ports maintain the status they had before SLEEP was executed (driving high, low or high-impedance).
- Resets other than WDT and BOR are not affected by SLEEP mode.
- Analog circuitry power (AV_{DD}) is removed during SLEEP.

Refer to individual chapters for more details on peripheral operation during SLEEP.

To minimize current consumption, the following conditions should be considered:

- I/O pins should not be floating.
- External circuitry sinking current from I/O pins.
- Internal circuitry sourcing current from I/O pins.
- Current draw from pins with internal weak pull-ups.
- Modules using Timer1 oscillator.
- ADC reference must be set to default condition (AV_{DD}).
- I/O pins that are high-impedance inputs should be pulled to V_{DD} or GND externally to avoid switching currents caused by floating inputs.

If the VDDEN bit is set, the SLEEP instruction removes power from the analog circuitry. AV_{DD} is shut down to minimize current draw in SLEEP Mode and to achieve the 50µA typical shutdown current. Shutdown current specifications can only be met with no current draw from external loads. The 5V V_{DD} voltage drops to 2.5V-3.0V and is only capable of supplying >1mA in SLEEP Mode. If more than 1mA of current are drawn form V_{DD} while in the low current SLEEP Mode, V_{DD} will collapse causing a reset of the device and the device coming back out of SLEEP.

A POR event during SLEEP will wake the device from SLEEP. The enable state of the analog circuitry does not change with the execution of the SLEEP command.

14.0.1 WAKE-UP FROM SLEEP

The device can wake-up from SLEEP through one of the following events:

1. External Reset input on $\overline{MCLR}$ pin, if enabled.
2. POR Reset.
3. Watchdog Timer, if enabled.
4. Any external interrupt.
5. Interrupts by peripherals capable of running during SLEEP (see individual peripheral for more information).

The first two events will cause a device reset. The last three events are considered a continuation of program execution. To determine whether a device reset or wake-up event occurred, refer to **Section 12.7 "Determining the Cause of a Reset"**

The following peripheral interrupts can wake the device from SLEEP:

- Interrupt-on-change
- External interrupt from INT pin

When the SLEEP instruction is being executed, the next instruction (PC + 1) is prefetched. For the device to wake up through an interrupt event, the corresponding interrupt enable bit must be enabled. Wake-up will occur regardless of the state of the GIE bit. If the GIE bit is disabled, the device continues execution at the instruction after the SLEEP instruction, the device will then call the Interrupt Service Routine. In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

The WDT is cleared when the device wakes up from SLEEP, regardless of the source of the wake-up.

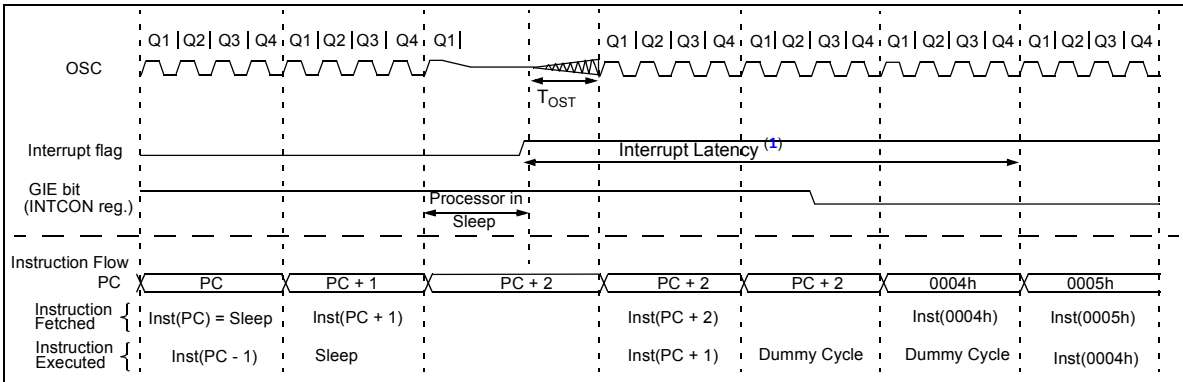
14.0.2 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of the SLEEP instruction
  - SLEEP instruction will execute as a NOP.
  - WDT and WDT prescaler will not be cleared.
  - $\overline{TO}$ bit in the STATUS register will not be set.
  - $\overline{PD}$ bit in the STATUS register will not be cleared.
- If the interrupt occurs **during** or **after** the execution of a SLEEP instruction
  - SLEEP instruction will be completely executed
  - Device will immediately wake up from SLEEP
  - WDT and WDT prescaler will be cleared
  - $\overline{TO}$ bit in STATUS register will be set
  - $\overline{PD}$ bit in the STATUS register will be cleared

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flags bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the $\overline{PD}$ bit. If the $\overline{PD}$ bit is set, the SLEEP instruction was executed as a NOP.

FIGURE 14-1: WAKE-UP FROM SLEEP THROUGH INTERRUPT



Note 1: GIE = 1 assumed. In this case after wake-up, the processor calls the ISR at 0004h. If GIE = 0, execution will continue in-line.

TABLE 14-1: SUMMARY OF REGISTERS ASSOCIATED WITH POWER-DOWN MODE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	102
IOCA	IOCA7	IOCA6	IOCA5	—	IOCA3	IOCA2	IOCA1	IOCA0	130
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	—	—	IOCB1	IOCB0	130
PIE1	TXIE	RCIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE	103
PIE2	CDSIE	ADIE	—	OTIE	OVIE	DRUVIE	OVLOIE	UVLOIE	104
PIR1	TXIF	RCIF	BCLIF	SSPIF	CC2IF	CC1IF	TMR2IF	TMR1IF	105
PIR2	CDSIF	ADIF	—	OTIF	OVIF	DRUVIF	OVLOIF	UVLOIF	106
STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	77

Legend: — = unimplemented, read as '0'. Shaded cells are not used in Power-down mode.

15.0 WATCHDOG TIMER (WDT)

The Watchdog Timer is a free-running timer, using the on chip RC oscillator as its clock source. The WDT is enabled by setting the WDTE bit of the Configuration Word (default setting). When WDTE is set, the on-chip RC oscillator will always be enabled to provide a clock source to the WDT module.

15.1 Watchdog Timer (WDT) Operation

During normal operation, a WDT time-out generates a device Reset. If the device is in Sleep mode, a WDT time-out causes the device to wake-up and continue with normal operation; this is known as a WDT wake-up. The WDT can be permanently disabled by clearing the WDTE configuration bit.

The postscaler assignment is fully under software control and can be changed during program execution.

15.2 WDT Period

The WDT has a nominal time-out period of 18 ms (with no prescaler). The time-out periods vary with temperature, V_{DD} and process variations from part to part (see DC specs). If longer time-out periods are desired, a prescaler with a division ratio of up to 1:128 can be assigned to the WDT under software control by writing to the OPTION register. Thus, time-out periods up to 2.3 seconds can be realized.

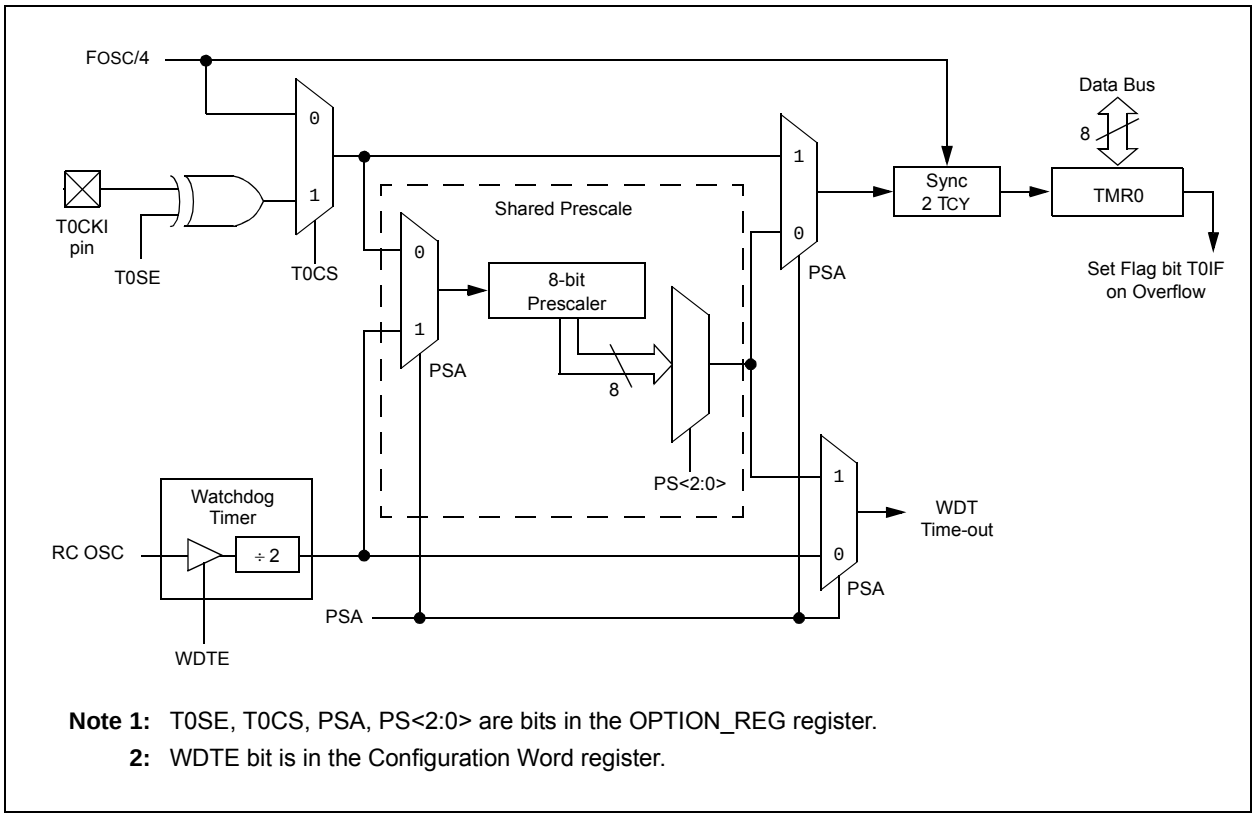
The `CLRWDT` and `SLEEP` instructions clear the WDT and the prescaler, if assigned to the WDT, and prevent it from timing out and generating a device Reset.

The $\overline{TO}$ bit in the STATUS register will be cleared upon a Watchdog Timer time-out.

15.3 WDT Programming Considerations

It should also be taken in account that under worst-case conditions (i.e., V_{DD} = Min., Temperature = Max., Max. WDT prescaler) it may take several seconds before a WDT time-out occurs.

FIGURE 15-1: WATCHDOG TIMER WITH SHARED PRESCALE BLOCK DIAGRAM



Note 1: T0SE, T0CS, PSA, PS<2:0> are bits in the OPTION_REG register.

2: WDTE bit is in the Configuration Word register.

TABLE 15-1: WDT STATUS

Conditions	WDT
WDTE = 0	Cleared
CLRWDT Command	
Exit Sleep	
SLEEP Command	

TABLE 15-2: SUMMARY OF REGISTERS ASSOCIATED WITH WATCHDOG TIMER

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
OPTION_REG	$\overline{RAPU}$	INTEDG	T0CE	T0SE	PSA	PS2	PS1	PS0	83

Legend: — = unimplemented locations read as '0'. Shaded cells are not used by Watchdog Timer.

TABLE 15-3: SUMMARY OF CONFIGURATION WORD ASSOCIATED WITH WATCHDOG TIMER

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONFIG	13:8	—	—	$\overline{DBGEN}$	—	WRT1	WRT0	—	BOREN	88
	7:0	—	$\overline{CP}$	MCLRE	$\overline{PWRTE}$	WDTE	—	—	—	

Legend: — = unimplemented locations read as '0'. Shaded cells are not used by Watchdog Timer.

16.0 OSCILLATOR MODES

The MCP19122/3 has one oscillator configuration, which is an 8 MHz internal oscillator.

16.1 Internal Oscillator (INTOSC)

The Internal Oscillator module provides a system clock source of 8 MHz. The frequency of the internal oscillator can be trimmed with a calibration value in the OSCTUNE register.

16.2 Oscillator Calibration

The 8 MHz internal oscillator is factory calibrated. The factory calibration values reside in the read-only Calibration Word 1 register. These values must be read from the Calibration Word 1 register and stored in the OSCCAL register. Refer to **Section 20.0 "Flash Program Memory Control"** for the procedure on reading from program memory.

Note 1:	The FCAL<6:0> bits from the Calibration Word 1 register must be written into the OSCCAL register to calibrate the internal oscillator.

16.3 Frequency Tuning in User Mode

In addition to the factory calibration, the base frequency can be tuned in the user's application. This frequency tuning capability allows the user to deviate from the factory calibrated frequency. The user can tune the frequency by writing to the OSCTUNE register (see Register 16-1).

REGISTER 16-1: OSCTUNE: – OSCILLATOR TUNING REGISTER

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0
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 7-5 **Unimplemented:** Read as '0'

bit 4-0 **TUN<4:0>:** Frequency Tuning bits

01111 = Maximum frequency
01110 =
•
•
•
00001 =
00000 = Center frequency. Oscillator Module is running at the calibrated frequency.
11111 =
•
•
•
10000 = Minimum frequency

16.3.1 OSCILLATOR DELAY UPON POWER-UP, WAKE-UP AND BASE FREQUENCY CHANGE

In applications where the OSCTUNE register is used to shift the frequency of the internal oscillator, the user should not expect the frequency of the internal oscillator to stabilize immediately. In this case, the frequency may shift gradually toward the new value. The time for this frequency shift is less than eight cycles of the base frequency.

On power-up, the device is held in reset by the power-up time, if the power-up timer is enabled.

Following a wake-up from Sleep mode or POR, an internal delay of ~10 µs is invoked to allow the memory bias to stabilize before program execution can begin.

TABLE 16-1: SUMMARY OF REGISTERS ASSOCIATED WITH CLOCK SOURCES

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
OSCTUNE	—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0	113

Legend: — = unimplemented locations read as '0'. Shaded cells are not used by clock sources.

TABLE 16-2: SUMMARY OF CALIBRATION WORD ASSOCIATED WITH CLOCK SOURCES

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CALWD1	13:8	—	—	BGR5	BGR4	BGR3	BGR2	BGR1	BGR0	61
	7:0	—	FCAL6	FCAL5	FCAL4	FCAL3	FCAL2	FCAL1	FCAL0	

Legend: — = unimplemented locations read as '0'. Shaded cells are not used by clock sources.

17.0 I/O PORTS

In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Each port has two registers for its operation. These registers are:

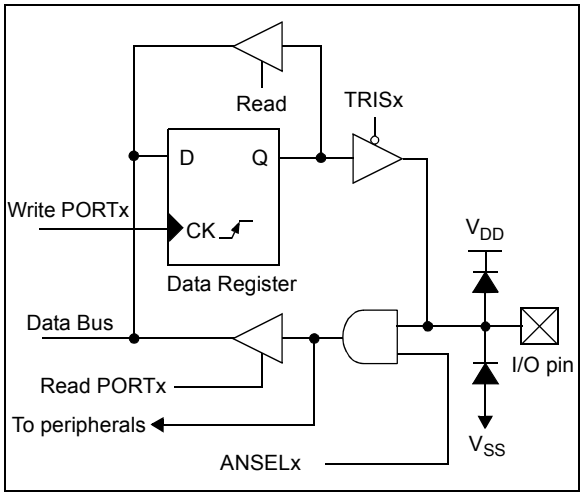
- TRISGPx registers (data direction register)
- PORTGPx registers (reads the levels on the pins of the device)

Some ports may have one or more of the following additional registers. These registers are:

- ANSELx (analog select)
- WPUx (weak pull-up)

Ports with analog functions also have an ANSELx register, which can disable the digital input and save power. A simplified model of a generic I/O port, without the interfaces to other peripherals, is shown in Figure 17-1.

FIGURE 17-1: GENERIC I/O PORTGPX OPERATION



EXAMPLE 17-1: INITIALIZING PORTA

```
; This code example illustrates
; initializing the PORTGPA register. The
; other ports are initialized in the same
; manner.

BANKSEL PORTGPA;
CLRF    PORTGPA;Init PORTA
BANKSEL ANSELA;
CLRF    ANSELA;digital I/O
BANKSEL TRISGPA;
MOVLW   B'00011111';Set GPA<4:0> as
            ;inputs
MOVWF   TRISGPA;and set GPA<7:6> as
            ;outputs
```

17.1 PORTGPA and TRISGPA Registers

PORTGPA is an 8-bit wide, bidirectional port consisting of five CMOS I/O, two open drain I/O, and one open drain input-only pin. The corresponding data direction register is TRISGPA (Register 17-2). Setting a TRISGPA bit (= 1) will make the corresponding PORTGPA pin an input (i.e., disable the output driver). Clearing a TRISGPA bit (= 0) will make the corresponding PORTGPA pin an output (i.e., enables output driver). The exception is GPA5, which is input only and its TRISGPA bit will always read as '1'. Example 17-1 shows how to initialize an I/O port.

Reading the PORTGPA register (Register 17-1) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations.

The TRISGPA register (Register 17-2) controls the PORTGPA pin output drivers, even when they are being used as analog inputs. The user must ensure the bits in the TRISGPA register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'. If the pin is configured for a digital output (either port or alternate function), the TRISGPA bit must be cleared in order for the pin to drive the signal, and a read will reflect the state of the pin.

17.1.1 INTERRUPT-ON-CHANGE

Each PORTGPA pin is individually configurable as an interrupt-on-change pin. Control bits IOCA<7:0> enable or disable the interrupt function for each pin. The interrupt-on-change feature is disabled on a Power-on Reset. Reference **Section 18.0 "Interrupt-On-Change"** for more information.

17.1.2 WEAK PULL-UPS

PORTGPA <1:0> and PORTGPA5 have an internal weak pull-up. PORTGPA<7:6> are special ports for the SSP module and do not have weak pull-ups. PORTGPA<3:2> are special current source ports and do not have weak pull-ups. PORTGPA4 is a true open drain pin and does not have a weak pull-up. Individual control bits can enable or disable the internal weak pull-ups (see Register 17-3). The weak pull-up is automatically turned off when the port pin is configured as an output or on a Power-on Reset setting the $\overline{\text{RAPU}}$ bit of the OPTION register. The weak pull-up on GPA5 is enabled when configured as $\overline{\text{MCLR}}$ pin by setting bit 5 of the Configuration Word or when controlled by software.

17.1.3 WEAK CURRENT SOURCE

PORTGPA<3:2> are capable of being configured as weak current sources. Setting WPUGPA<3:2> allow each pin to source 50 µA (typical). By connecting GPA2 or GPA3 to ground with a resistor The voltage on the pin can be read with the A/D to determine device I2C/PMBus address. The value of the resistor to ground shall be 3 kΩ to 50 kΩ.

The current source on GPA3 also functions as the Error Amplifier clamp control source.

17.1.4 ANSELA REGISTER

The ANSELA register (Register 17-4) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELA bit high will cause all digital reads on the pin to be read as '0' and allows analog functions on the pin to operate correctly.

The state of the ANSELA bits has no effect on the digital output functions. A pin with TRISGPA clear and ANSELA set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

Note:	The ANSELA bits default to the Analog mode after Reset. To use any pins as digital general purpose or peripheral inputs, the corresponding ANSELA bits must be initialized to '0' by user software.

17.1.5 PORTGPA FUNCTIONS AND OUTPUT PRIORITIES

Each PORTGPA pin is multiplexed with other functions. The pins, their combined functions and their output priorities are shown in Table 17-1. For additional information, refer to the appropriate section in this data sheet.

PORTGPA pins GPA7 and GPA4 are true open-drain pins with no connection back to V_{DD}.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the highest priority.

Analog input functions, such as ADC, are not shown in the priority lists. These inputs are active when the I/O pin is set for Analog mode using the ANSELA registers. Digital output functions may control the pin when it is in Analog mode with the priority shown in Table 17-1.

TABLE 17-1: PORTGPA FUNCTIONS[2]

Pad Name	Function	I/O	Type & Priority[(1)]	Description
GPA0	The GPA0 pad has basic port, peripheral and test mode features. The test mode outputs of the ALT_ICSPDAT and ANALOG_TEST take priority over the port data. The TRISGPA bit is overridden when configured as ALT_ICSPDAT or analog test output. The pad is an output only when configured so by the user. Enabling the AN0 input disables the input buffers and forces PORTGPA<0> to read '0'.			
	GPA0	OUT	CMOS-4	PORTGPA<0> data output
		IN	TTL	PORTGPA<0> data input
	AN0	IN	ANA	Channel 0 A/D input
	ANALOG_TEST	OUT	ANA-3	Analog test mode output
GPA1	The GPA1 pad has basic port, peripheral and test mode features. The pad is an output only when configured so by the user. Enabling the AN1 input disables the input buffers and forces PORTGPA<1> to read '0'.			
	GPA1	OUT	CMOS-2	PORTGPA<1> data output
		IN	TTL	PORTGPA<1> data input
	AN1	IN	ANA	Channel 1 A/D input
	SYC_SIGNAL	IN	ST	Synchronization signal input
		OUT	CMOS-1	Synchronization signal output
GPA2	The GPA2 pad has basic port, peripheral and test mode features. The pad is an output only when configured so by the user. Enabling the AN2 input disables the input buffers and forces PORTGPA<2> to read '0'.			
	GPA2	OUT	CMOS-1	PORTGPA<2> data output
		IN	ST	PORTGPA<2> data input
	AN2	IN	ANA	Channel 2 A/D input
	T0CKI	IN	ST	Timer 0 input
	INT	IN	ST	External interrupt input
GPA3	The GPA3 pad has basic port, peripheral and test mode features. The pad is an output only when configured so by the user. Enabling the AN3 input disables the input buffers and forces PORTGPA<3> to read '0'.			
	GPA3	OUT	CMOS-1	PORTGPA<3> data output
		IN	ST	PORTGPA<3> data input
	AN3	IN	ANA	Channel 3 A/D input
	T1G1	IN	ST	Input 1 to Timer1 gate
GPA4	The GPA4 pad is a high voltage port. The TRISGPA bit is always set.			
	GPA4	OUT	OD	PORTGPA<4> open drain data output
		IN	TTL	PORTGPA<4> open drain date input
GPA5	The GPA5 pad is an input-only high voltage port. The TRISGPA bit is always set.			
	GPA5	IN	TTL	PORTGPA<5> open drain data input
	$\overline{\text{MCLR}}$	IN	ST	$\overline{\text{MCLR}}$ input
	TEST	IN	HV	ICSP and test mode entry high voltage pin

Legend: OUT - Output, IN - Input, ANA - Analog Signal, DIG - Digital Output, OD - Open Drain Output, ST - Schmitt Buffer Input, TTL - TTL Buffer Input, XTAL - Crystal connection, HV - High Voltage

Note 1: Output priority number determines the precedence of data into the MUX when multiple outputs are available at the same time (1 - highest priority). This number affects drive data, but not drive enable. Items with same priority number are mutually exclusive.

2: Pad module signal connections reflect only the module input signals. Output connections are addressed in the corresponding consumer module.

TABLE 17-1: PORTGPA FUNCTIONS[2] (CONTINUED)

Pad Name	Function	I/O	Type & Priority[(1)]	Description
GPA6	The GPA6 pad has basic port, peripheral and test mode features. The ICSPDAT output signal takes priority over the port data. The TRISGPA bit is overridden to '0' when configured to output ICSPDAT, otherwise, the pad is an output only when configured so by the user.			
	GPA6	OUT	CMOS-3	PORTGPA<6> data output
		IN	ST	PORTGPA<6> data input
	ICSPDAT	OUT	CMOS-1	ICSP Data output (MCP19122 Only)
		IN	ST	ICSP Data input (MCP19122 Only)
	CCD1	OUT	CMOS-2	CCD1 output
		IN	ST	CCD1 input
GPA7	The GPA7 pad has basic port, peripheral and test mode features.			
	GPA7	OUT	OD-2	PORTGPA<7> open drain
		IN	ST	PORTGPA<7> data input
	SCL	IN	I2C	I2C slave mode clock input with selectable I2C or SMBus input levels
		OUT	OD-1	I2C master mode clock output
	ICSPCLK	IN	ST	ICSPCK input (MCP19122 Only)

Legend: OUT - Output, IN - Input, ANA - Analog Signal, DIG - Digital Output, OD - Open Drain Output, ST - Schmitt Buffer Input, TTL - TTL Buffer Input, XTAL - Crystal connection, HV - High Voltage

Note 1: Output priority number determines the precedence of data into the MUX when multiple outputs are available at the same time (1 - highest priority). This number affects drive data, but not drive enable. Items with same priority number are mutually exclusive.

2: Pad module signal connections reflect only the module input signals. Output connections are addressed in the corresponding consumer module.

REGISTER 17-1: PORTGPA: PORTGPA REGISTER

R/W-x	R/W-x	R-x	R-x	R/W-x	R/W-x	R/W-x	R/W-x
GPA7	GPA6	GPA5	GPA4	GPA3	GPA2	GPA1	GPA0
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 7 **GPA7**: General Purpose Open Drain I/O pin

bit 6 **GPA6**: General Purpose I/O pin
1 = Port pin is > V_{IH}
0 = Port pin is < V_{IL}

bit 5 **GPA5/$\overline{\text{MCLR}}$**: General Purpose Open Drain Input pin

bit 4 **GPA4**: General Purpose Open Drain I/O pin

bit 3-0 **GPA<3:0>**: General Purpose I/O pin
1 = Port pin is > V_{IH}
0 = Port pin is < V_{IL}

REGISTER 17-2: TRISGPA: PORTGPA TRI-STATE REGISTER

R/W-1	R/W-1	R-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0
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 7-6 **TRISA<7:6>:** PORTGPA Tri-State Control bit
1 = PORTGPA pin configured as an input (tri-stated)
0 = PORTGPA pin configured as an output

bit 5 **TRISA5:** GPA5 Port Tri-State Control bit
This bit is always '1' as GPA5 is an input only

bit 4-0 **TRISA<4:0>:** PORTGPA Tri-State Control bit
1 = PORTGPA pin configured as an input (tri-stated)
0 = PORTGPA pin configured as an output

REGISTER 17-3: WPUGPA: WEAK PULL-UP PORTGPA REGISTER

U-0	U-0	R/W-1	U-0	R/W-1	R/W-1	R/W-1	R/W-1
—	—	WPUA5	—	WCS1	WCS0	WPUA1	WPUA0
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 7-6 **Unimplemented:** Read as '0'

bit 5 **WPUA5:** Weak Pull-up Register bit
1 = Pull-up enabled
0 = Pull-up disabled

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

bit 3-2 **WCS<1:0>:** Weak Current Source bit
1 = Pull-up enabled
0 = Pull-up disabled

bit 1-0 **WPUA<1:0>:** Weak Pull-up Register bit
1 = Pull-up enabled
0 = Pull-up disabled

Note 1: The weak pull-up device is enabled only when the global $\overline{\text{RAPU}}$ bit is enabled, the pin is in input mode (TRISGPA = 1), and the individual WPUA bit is enabled (WPUA = 1), and the pin is not configured as an analog input.

2: GPA5 weak pull-up is also enabled when the pin is configured as $\overline{\text{MCLR}}$ in Configuration word.

3: GPA2 and GPA3 weak current sources are not dependant on the global $\overline{\text{RAPU}}$.

REGISTER 17-4: ANSELA: ANALOG SELECT PORTGPA REGISTER

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0
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 7-4 **Unimplemented:** Read as '0'

bit 3-0 **ANSA<3:0>**: Analog Select PORTGPA Register bit
1 = Analog input. Pin is assigned as analog input.(1)
0 = Digital I/O. Pin is assigned to port or special function.

Note 1: Setting a pin to an analog input automatically disables the digital input circuitry, weak pull-ups, and interrupt-on-change if available. The corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

TABLE 17-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTGPA

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0	120
OPTION_REG	RAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	83
IOCA	IOCA7	IOCA6	IOCA5	IOCA4	IOCA3	IOCA2	IOCA1	IOCA0	130
PORTGPA	GPA7	GPA6	GPA5	GPA4	GPA3	GPA2	GPA1	GPA0	119
TRISGPA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	119
WPUGPA	—	—	WPUA5	—	WCS1	WCS0	WPUA1	WPUA0	120

Legend: — = unimplemented locations read as '0'. Shaded cells are not used by PORTGPA.

17.2 PORTGPB and TRISGPB Registers

PORTGPB is an 8-bit wide, bidirectional port consisting of seven general purpose I/O ports. The corresponding data direction register is TRISGPB (Register 17-6). Setting a TRISGPB bit (= 1) will make the corresponding PORTGPB pin an input (i.e., disable the output driver). Clearing a TRISGPB bit (= 0) will make the corresponding PORTGPB pin an output (i.e., enable the output driver). Example 17-1 shows how to initialize an I/O port.

Some pins for PORTGPB are multiplexed with an alternate function for the peripheral, or a clock function. In general, when a peripheral or clock function is enabled, that pin may not be used as a general purpose I/O pin.

Reading the PORTGPB register (Register 17-5) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations.

The TRISGPB register (Register 17-6) controls the PORTGPB pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISGPB register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'. If the pin is configured for a digital output (either port or alternate function), the TRISGPB bit must be cleared in order for the pin to drive the signal and a read will reflect the state of the pin.

17.2.1 INTERRUPT-ON-CHANGE

Each PORTGPB pin is individually configurable as an interrupt-on-change pin. Control bits IOCB<7:4> and IOCB<2:0> enable or disable the interrupt function for each pin. The interrupt-on-change feature is disabled on a Power-on Reset. Reference **Section 18.0 "Interrupt-On-Change"** for more information.

17.2.2 WEAK PULL-UPS

Each of the PORTGPB pins, except GPB0, has an individually configurable internal weak pull-up. Control bits WPUB<7:4> and WPUB<2:1> enable or disable each pull-up (see Register 17-7). Each weak pull-up is automatically turned off when the port pin is configured as an output. All pull-ups are disabled on a Power-on Reset by the $\overline{RAPU}$ bit of the OPTION register.

17.2.3 ANSELB REGISTER

The ANSELB register (Register 17-8) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELB bit high will cause all digital reads on the pin to be read as '0' and allows analog functions on the pin to operate correctly.

The state of the ANSELB bits has no effect on the digital output functions. A pin with TRISGPB clear and ANSELB set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

Note:	The ANSELB bits default to the Analog mode after Reset. To use any pins as digital general purpose or peripheral inputs, the corresponding ANSELB bits must be initialized to '0' by the user's software.

17.2.4 PORTGPB FUNCTIONS AND OUTPUT PRIORITIES

Each PORTGPB pin is multiplexed with other functions. The pins, their combined functions and their output priorities are shown in Table 17-3. For additional information, refer to the appropriate section in this data sheet.

PORTGPB pin GPB0 is a true open drain pin with no connection back to V_{DD}.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the highest priority.

Analog input functions, such as ADC, and some digital input functions are not included in the list below. These inputs are active when the I/O pin is set for Analog mode using the ANSELB registers. Digital output functions may control the pin when it is in Analog mode, with the priority shown in Table 17-3.

TABLE 17-3: PORTGPB FUNCTIONS(2)

Pad Name	Function	I/O	Type & Priority (1)	Description
GPB0	The GPB0 pad has basic port, peripheral and test mode features. The pad is an output only when configured so by the user.			
	GPB0	OUT	OD-2	PORTGPB<0> data output, open drain
		IN	TTL	PORTGPB<0> data input, open drain
	SDA	IN	I2C	I2C slave mode data input with selectable I2C or SMBus input levels
		OUT	OD-1	I2C master mode data open drain output
GPB1	The GPB1 pad has basic port, peripheral and test mode features. The pad is an output only when sp configured by the user. Enabling the AN4 input disables the input buffers and forces PORTGPB<1> to read '0'.			
	GPB1	OUT	CMOS-2	PORTGPB<1> data output
		IN	TTL	PORTGPB<1> data input
	AN4	IN	ANA	Channel 4 A/D input
	CON_SIGNAL	IN	ANA	Slave mode Current Reference input
		OUT	ANA-1	Master mode Current Sense output
GPB2	The GPB2 pad has basic port and peripheral features. The pad is an output only when so configured by the user. Enabling the AN5 input disables the input buffers and forces PORTGPB<2> to read '0'.			
	GPB2	OUT	CMOS	PORTGPB<2> data output
		IN	TTL	PORTGPB<2> data input
	AN5	IN	ANA	Channel 5 A/D input
	T1G2	IN	ST	Input 2 to TIMER1 gate
GPB3	The GPB3 pad has basic port and peripheral features. The pad is an output only when so configured by the user.			
	GBP3	OUT	CMOS	PORTGPB<3> data output
		IN	TTL	PORTGPA<3> data input
	CLOCK	OUT	CMOS	Oscillator output
		IN	ST	Oscillator input
GPB4	The GPB4 pad has basic port, peripheral and test mode features. The test mode outputs of ICSPDAT take priority over the port data. The TRISGPB bit is overridden to '0' when configured as ICSPDAT. The pad is an output only when configured so by the user. Enabling the AN6 input disables the input buffers and forces PORTGPB<4> to read '0'. This pin is only available on the MCP19123.			
	GPB4	OUT	CMOS-3	PORTGPB<4> data output
		IN	TTL	PORTGPB<4> data input
	AN6	IN	ANA	Channel 6 A/D input
	ICSPDAT	OUT	CMOS-2	Serial programming data output
		IN	ST	Serial programming data input
	ICDDAT	OUT	CMOS-1	In-Circuit debug data output
		IN	ST	In-Circuit debug data input

Legend: OUT - Output, IN - Input, ANA - Analog Signal, DIG - Digital Output, OD - Open Drain Output, ST - Schmitt Buffer Input, TTL - TTL Buffer Input, XTAL - Crystal connection, HV - High Voltage

Note 1: Output priority number determines the precedence of data into the MUX when multiple outputs are available at the same time (1 - highest priority). This number affects drive data, but not drive enable. Items with same priority number are mutually exclusive.

2: Pad module signal connections reflect only the module input signals. Output connections are addressed in the corresponding consumer module.

Pad Name	Function	I/O	Type & Priority (1)	Description
GPB5	The GPB5 pad has basic port and peripheral features. The pad is an output only when configured so by the user. This pin is only available on the MCP19123.			
	GPB5	OUT	CMOS	PORTGPB<5> data output
		IN	TTL	PORTGPB<5> data input
	AN7	IN	ANA	Channel 7 A/D input
	ICSPCLK	IN	ST	Serial programming clock input
	ICDCLK	IN	ST	In-Circuit debugger clock input
GPB6	The GPB6 pad has basic port and peripheral features. The pad is an output only when so configured by the user. This pin is only available on the MCP19123.			
	GPB6	OUT	CMOS-2	PORTGPB<6> data output
		IN	TTL	PORTGPB<6> data input
	CCD2	OUT	CMOS-1	CCD2 output
		IN	ST	CCD2 input
GPB7	The GPB7 pad has basic port and peripheral features. The pad is an output only when so configured by the user. This pin is only available on the MCP19123.			
	GPB7	OUT	CMOS-1	PORTGPB<7> data output
		IN	TTL	PORTGPB<7> data input
	VDAC	IN	ANA	External ADC reference

Legend: OUT - Output, IN - Input, ANA - Analog Signal, DIG - Digital Output, OD - Open Drain Output, ST - Schmitt Buffer Input, TTL - TTL Buffer Input, XTAL - Crystal connection, HV - High Voltage

Note 1: Output priority number determines the precedence of data into the MUX when multiple outputs are available at the same time (1 - highest priority). This number affects drive data, but not drive enable. Items with same priority number are mutually exclusive.

2: Pad module signal connections reflect only the module input signals. Output connections are addressed in the corresponding consumer module.

REGISTER 17-5: PORTGPB: PORTGPB REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
GPB7(1)	GPB6(1)	GPB5(1)	GPB4(1)	GPB3	GPB2	GPB1	GPB0
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 7-0 **GPB<7:0>**: General Purpose I/O Pin bit
1 = Port pin is > V_{IH}
0 = Port pin is < V_{IL}

Note 1: Not implemented on MCP19122.

REGISTER 17-6: TRISGPB: PORTGPB TRI-STATE REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISB7(1)	TRISB6(1)	TRISB5(1)	TRISB4(1)	TRISB3	TRISB2	TRISB1	TRISB0
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 7-0 **TRISB<7:0>:** PORTGPB Tri-State Control bit
1 = PORTGPB pin configured as an input (tri-stated)
0 = PORTGPB pin configured as an output

Note 1: Not implemented on MCP19122.

REGISTER 17-7: WPUGPB: WEAK PULL-UP PORTGPB REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-0	R/W-1	R/W-1	U-0
WPUB7(2)	WPUB6(2)	WPUB5(2)	WPUB4(2)	WPUB3	WPUB2	WPUB1	—
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 7-3 **WPUB<7:3>**: Weak Pull-up Register bit
1 = Pull-up enabled
0 = Pull-up disabled

bit 2-1 **WPUB<2:1>**: Weak Pull-up Register bit
1 = Pull-up enabled
0 = Pull-up disabled

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

Note 1: The weak pull-up device is enabled only when the global $\overline{\text{RAPU}}$ bit is enabled, the pin is in Input mode (TRISGPA = 1), the individual WPUB bit is enabled (WPUB = 1), and the pin is not configured as an analog input.

2: Not implemented on MCP19122.

REGISTER 17-8: ANSELB: ANALOG SELECT PORTGPB REGISTER

U-0	U-0	R/W-1	R/W-1	U-0	R/W-1	R/W-1	U-0
—	—	ANSB5(2)	ANSB4(2)	—	ANSB2	ANSB1	—
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 7-6 **Unimplemented:** Read as '0'

bit 5-4 **ANSB<5:4>**: Analog Select PORTGPB Register bit
1 = Analog input. Pin is assigned as analog input(1).
0 = Digital I/O. Pin is assigned to port or special function.

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

bit 2-1 **ANSB<2:1>**: Analog Select PORTGPB Register bit
1 = Analog input. Pin is assigned as analog input(1).
0 = Digital I/O. Pin is assigned to port or special function.

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

Note 1: Setting a pin to an analog input automatically disables the digital input circuitry, weak pull-ups and interrupt-on-change if available. The corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

2: Not implemented on MCP19122.

TABLE 17-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTGPB

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELB	—	—	ANSB5	ANSB4	—	ANSB2	ANSB1	—	126
OPTION_REG	RAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	83
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	130
PORTGPB	GPB7	GPB6	GPB5	GPB4	GPB3	GPB2	GPB1	GPB0	125
TRISGPB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	125
WPUGPB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	—	126

Legend: — = unimplemented locations read as '0'. Shaded cells are not used by PORTGPB.

NOTES:

18.0 INTERRUPT-ON-CHANGE

Each PORTGPA and PORTGPB pin is individually configurable as an interrupt-on-change pin. Control bits IOCA and IOCB enable or disable the interrupt function for each pin. Refer to Register 18-1 and Register 18-2. The interrupt-on-change is disabled on a Power-on Reset.

The interrupt-on-change on GPA5 is disabled when configured as $\overline{\text{MCLR}}$ pin in the Configuration Word.

For enabled interrupt-on-change pins, the values are compared with the old value latched on the last read of PORTGPA or PORTGPB. The mismatched outputs of the last read of all the PORTGPA and PORTGPB pins are OR'ed together to set the Interrupt-on-Change Interrupt Flag bit (IOCF) in the INTCON register.

18.1 Enabling the Module

To allow individual port pins to generate an interrupt, the IOCIE bit of the INTCON register must be set. If the IOCIE bit is disabled, the edge detection on the pin will still occur, but an interrupt will not be generated.

18.2 Individual Pin Configuration

To enable a pin to detect an interrupt-on-change, the associated IOCAx or IOCBx bit of the IOCA or IOCB register is set.

18.3 Clearing Interrupt Flags

The user, in the Interrupt Service Routine, clears the interrupt by:

a) Any read of PORTGPA or PORTGPB AND Clear flag bit IOCF. This will end the mismatch condition;

   OR

b) Any write of PORTGPA or PORTGPB AND Clear flag bit IOCF will end the mismatch condition.

A mismatch condition will continue to set flag bit IOCF. Reading PORTGPA or PORTGPB will end the mismatch condition and allow flag bit IOCF to be cleared. The latch holding the last read value is not affected by a $\overline{\text{MCLR}}$ Reset. After this Reset, the IOCF flag will continue to be set if a mismatch is present.

Note:	If a change on the I/O pin should occur when any PORTGPA or PORTGPB operation is being executed, then the IOCF interrupt flag may not get set.

18.4 Operation in Sleep

The interrupt-on-change interrupt sequence will wake the device from Sleep mode, if the IOCE bit is set.

18.5 Interrupt-On-Change Registers

REGISTER 18-1: IOCA: INTERRUPT-ON-CHANGE PORTGPA REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IOCA7	IOCA6	IOCA5	IOCA4	IOCA3	IOCA2	IOCA1	IOCA0
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 7-6 **IOCA<7:6>**: Interrupt-on-Change PORTGPA Register bits.
1 = Interrupt-on-change enabled on the pin
0 = Interrupt-on-change disabled on the pin

bit 5 **IOCA<5>**: Interrupt-on-Change PORTGPA Register bits(1).
1 = Interrupt-on-change enabled on the pin
0 = Interrupt-on-change disabled on the pin

bit 4-0 **IOCA<4:0>**: Interrupt-on-Change PORTGPA Register bits.
1 = Interrupt-on-change enabled on the pin
0 = Interrupt-on-change disabled on the pin

Note 1: The Interrupt-on-change on GPA5 is disabled if GPA5 is configured as $\overline{\text{MCLR}}$.

REGISTER 18-2: IOCB: INTERRUPT-ON-CHANGE PORTGPB REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IOCB7(1)	IOCB6(1)	IOCB5(1)	IOCB4(1)	IOCB3	IOCB2	IOCB1	IOCB0
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 7-0 **IOCB<7:0>**: Interrupt-on-Change PORTGPB Register bits.
1 = Interrupt-on-change enabled on the pin
0 = Interrupt-on-change disabled on the pin

Note 1: Not implemented on MCP19122.

TABLE 18-1: SUMMARY OF REGISTERS ASSOCIATED WITH INTERRUPT-ON-CHANGE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0	120
ANSELB	—	—	ANSB5	ANSB4	—	ANSB2	ANSB1	—	126
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	102
IOCA	IOCA7	IOCA6	IOCA5	IOCA4	IOCA3	IOCA2	IOCA1	IOCA0	130
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	130
TRISGPA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	119
TRISGPB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	125

Legend: — = unimplemented locations read as '0'. Shaded cells are not used by interrupt-on-change.

NOTES:

19.0 ANALOG-TO-DIGITAL CONVERTER (ADC) MODULE

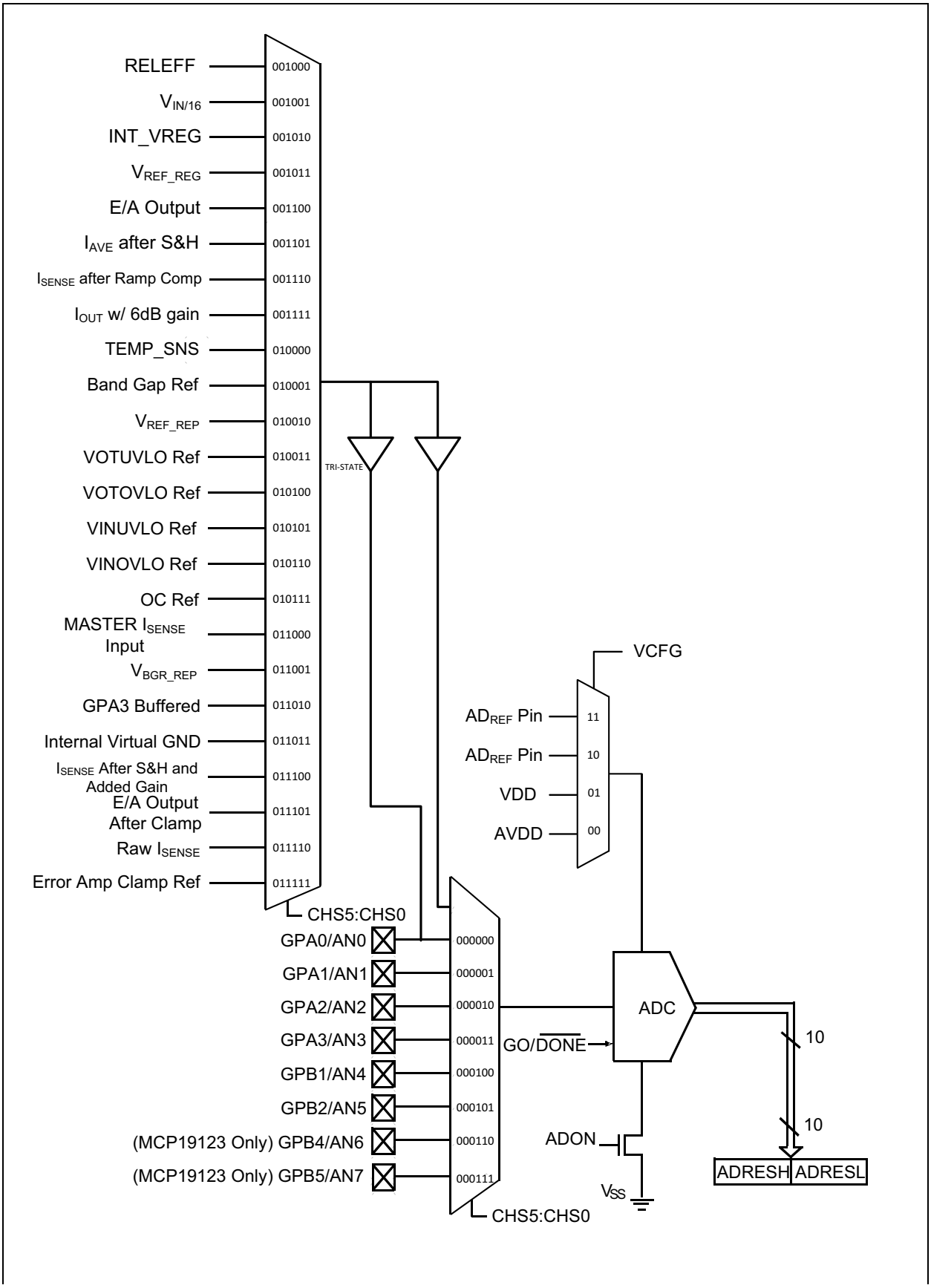
The Analog-to-Digital Converter (ADC) allows conversion of an analog input signal to a 10-bit binary representation of that signal. This device uses analog inputs, which are multiplexed into a single sample and hold circuit. The output of the sample and hold is connected to the input of the converter. The converter generates a 10-bit binary result via successive approximation and stores the conversion result into the ADC result registers (ADRESH:ADRESL register pair). Figure 19-1 shows the block diagram of the ADC.

On the MCP19122/3 devices the 4.096V AV_{DD} or 5V V_{DD} can be used for the ADC reference. On the MCP19123 device an external reference can be used by selecting the AD_{REF} pin as the ADC voltage reference source. The ADCON1<ACFG> bit controls the ADC reference source.

The ADC can generate an interrupt upon completion of a conversion. This interrupt can be used to wake-up the device from Sleep.

Note:	Once V_{IN} is greater than $AV_{DD} + V_{DROPOUT}$, AV_{DD} is in regulation, allowing A/D readings to be accurate. Once V_{IN} is greater than $V_{DD} + V_{DROPOUT}$, V_{DD} is in regulation. Setting the ADC reference to V_{DD} allows accurate ratiometric measurements.

FIGURE 19-1: ADC BLOCK DIAGRAM



19.1 ADC Configuration

When configuring and using the ADC, the following functions must be considered:

- Port configuration
- Channel selection
- ADC conversion clock source
- Interrupt control
- Result formatting

19.1.1 PORT CONFIGURATION

The ADC can be used to convert both analog and digital signals. When converting analog signals, the I/O pin should be configured for analog by setting the associated TRIS and ANSEL bits. Refer to **Section 17.0 "I/O Ports"** for more information.

Note:	Analog voltages on any pin that is defined as a digital input may cause the input buffer to conduct excess current.

19.1.2 CHANNEL SELECTION

There are up to 30 channel selections available on the MCP19122 and 32 channel selections available on the MCP19123. See Figure 19-1 and Register 19-1 for channel information.

The CHS<4:0> bits of the ADCON0 register determine which channel is connected to the sample and hold circuit.

When changing channels, a delay is required before starting the next conversion. Refer to **Section 19.2 "ADC Operation"** for more information.

19.1.3 ADC CONVERSION CLOCK

The source of the conversion clock is software selectable via the ADCS bits of the ADCON1 register. There are five possible clock options:

- $F_{OSC}/8$
- $F_{OSC}/16$
- $F_{OSC}/32$
- $F_{OSC}/64$
- F_{RC} (clock derived from internal oscillator with a divisor of 16)

The time to complete one bit conversion is defined as T_{AD}. One full 10-bit conversion requires 11 T_{AD} periods as shown in Figure 19-2.

For a correct conversion, the appropriate T_{AD} specification must be met. Refer to the A/D conversion requirements in **Section 5.0 "Digital Electrical Characteristics"** for more information. Table 19-1 gives examples of appropriate ADC clock selections.

Note:	Unless using the F_{RC}, any changes in the system clock frequency will change the ADC clock frequency, which may adversely affect the ADC result.

TABLE 19-1: ADC CLOCK PERIOD (T_{AD}) VS. DEVICE OPERATING FREQUENCIES

ADC Clock Period (T_{AD})		Device Frequency (F_{OSC})
ADC Clock Source	**ADCS<2:0>**	**8 MHz**
$F_{OSC}/8$	001	1.0 µs(2)
$F_{OSC}/16$	101	2.0 µs
$F_{OSC}/32$	010	4.0 µs
$F_{OSC}/64$	110	8.0 µs(3)
F_{RC}	x11	2.0 – 6.0 µs(1,4)

Legend: Shaded cells are outside of recommended range.

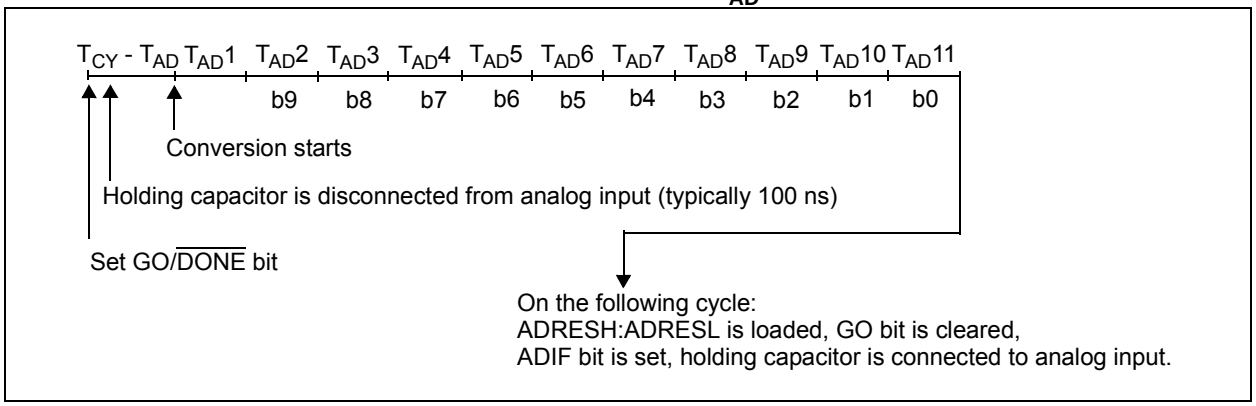
Note 1: The F_{RC} source has a typical T_{AD} time of 4 µs for V_{DD} > 3.0V.

2: These values violate the minimum required T_{AD} time.

3: For faster conversion times, the selection of another clock source is recommended.

4: The F_{RC} clock source is only recommended if the conversion will be preformed during Sleep.

FIGURE 19-2: ANALOG-TO-DIGITAL CONVERSION T_{AD} CYCLES



19.1.4 INTERRUPTS

The ADC module allows for the ability to generate an interrupt upon completion of an Analog-to-Digital conversion. The ADC Interrupt Flag is the ADIF bit in the PIR1 register. The ADC Interrupt Enable is the ADIE bit in the PIE1 register. The ADIF bit must be cleared in software.

Note 1:	The ADIF bit is set at the completion of every conversion, regardless of whether or not the ADC interrupt is enabled.

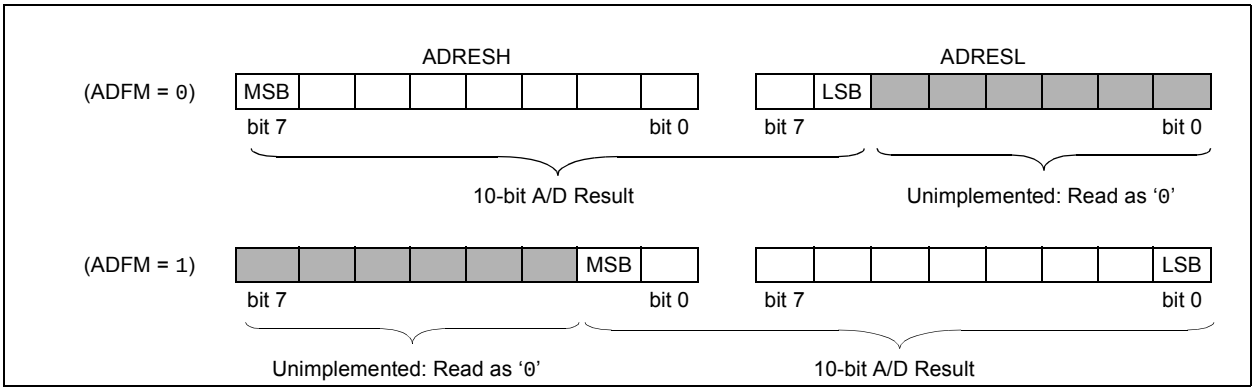
This interrupt can be generated while the device is operating, or while in Sleep. If the device is in Sleep, the interrupt will wake-up the device. Upon waking from Sleep, the next instruction following the SLEEP instruction is always executed. If the user is attempting to wake-up from Sleep and resume in-line code execution, the GIE and PEIE bits of the INTCON register must be disabled. If the GIE and PEIE bits of the INTCON register are enabled, execution will switch to the Interrupt Service Routine.

19.1.5 RESULT FORMATTING

The 10-bit A/D conversion result can be supplied in two formats, left justified or right justified. The ADFM bit of the ADCON0 register controls the output format.

Figure 19-3 shows the output format.

FIGURE 19-3: 10-BIT A/D CONVERSION RESULT FORMAT



19.2 ADC Operation

19.2.1 STARTING A CONVERSION

To enable the ADC module, the ADON bit of the ADCON0 register must be set to a '1'. Setting the GO/$\overline{\text{DONE}}$ bit of the ADCON0 register to a '1' will start the Analog-to-Digital conversion.

Note:	The GO/$\overline{\text{DONE}}$ bit should not be set in the same instruction that turns on the ADC. Refer to **Section 19.2.4 "A/D Conversion Procedure"**.

19.2.2 COMPLETION OF A CONVERSION

When the conversion is complete, the ADC module will:

- Clear the GO/$\overline{\text{DONE}}$ bit
- Set the ADIF Interrupt Flag bit
- Update the ADRESH:ADRESL registers with new conversion result

19.2.3 TERMINATING A CONVERSION

If a conversion must be terminated before completion, the GO/$\overline{\text{DONE}}$ bit can be cleared in software. The ADRESH:ADRESL registers will not be updated with the partially complete Analog-to-Digital conversion sample. Instead, the ADRESH:ADRESL register pair will retain the value of the previous conversion. Additionally, a two T_{AD} delay is required before another acquisition can be initiated. Following the delay, an input acquisition is automatically started on the selected channel.

Note:	A device Reset forces all registers to their Reset state. Thus, the ADC module is turned off and any pending conversion is terminated.

19.2.4 A/D CONVERSION PROCEDURE

This is an example procedure for using the ADC to perform an Analog-to-Digital conversion:

1. Configure Port:
   - Disable pin output driver (Refer to the TRIS register)
   - Configure pin as analog (Refer to the ANSEL register)
2. Configure the ADC module:
   - Select ADC conversion clock
   - Select ADC input channel
   - Turn on ADC module
3. Configure ADC interrupt (optional):
   - Clear ADC interrupt flag
   - Enable ADC interrupt
   - Enable peripheral interrupt
   - Enable global interrupt(1)
4. Wait the required acquisition time(2).
5. Start conversion by setting the GO/$\overline{\text{DONE}}$ bit.
6. Wait for ADC conversion to complete by one of the following:
   - Polling the GO/$\overline{\text{DONE}}$ bit
   - Waiting for the ADC interrupt (interrupts enabled)
7. Read ADC Result.
8. Clear the ADC interrupt flag (required if interrupt is enabled).

Note 1:	The global interrupt can be disabled if the user is attempting to wake-up from Sleep and resume in-line code execution.
2:	Refer to **Section 19.4 "A/D Acquisition Requirements"**.

EXAMPLE 19-1: A/D CONVERSION

```
;This code block configures the ADC
;for polling, Frc clock and AN0 input.
;
;Conversion start & polling for completion ;
are included.
;
BANKSEL   ADCON1         ;
MOVLW     B'01110000'    ;Frc clock
MOVWF     ADCON1         ;
BANKSEL   TRISGPA        ;
BSF       TRISGPA,0      ;Set GPA0 to input
BANKSEL   ANSELA         ;
BSF       ANSELA,0       ;Set GPA0 to analog
BANKSEL   ADCON0         ;
MOVLW     B'01000001'    ;Select channel AN0
MOVWF     ADCON0         ;Turn ADC On
CALL      SampleTime     ;Acquisiton delay
BSF       ADCON0,1       ;Start conversion
BTFSC     ADCON0,1       ;Is conversion done?
GOTO      $-1            ;No, test again
BANKSEL   ADRESH         ;
MOVF      ADRESH,W       ;Read upper 2 bits
MOVWF     RESULTHI       ;store in GPR space
BANKSEL   ADRESL         ;
MOVF      ADRESL,W       ;Read lower 8 bits
MOVWF     RESULTLO       ;Store in GPR space
```

19.3 ADC Register Definitions

REGISTER 19-1: ADCON0: A/D CONTROL REGISTER 0

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CHS5	CHS4	CHS3	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON
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 7-2 **CHS<5:0>:** Analog Channel Select bits

000000 = GPA0
000001 = GPA1
000010 = GPA2
000011 = GPA3
000100 = GPB1
000101 = GPB2
000110 = GPB4(1)
000111 = GPB5(1)
001000 = RELEFF (see **Section 3.11.6 "Relative Efficiency Ramp Measurement Control"**)
001001 = Ratio of input voltage (V_{IN}/16)
001010 = Output voltage measured after differential amplifier
001011 = V_{REF_REG}
001100 = Error amplifier output
001101 = Average current after Sample & Hold and gain trim
001110 = I_{SENSE} signal after gain and slope compensation signal
001111 = Average output current with +6dB gain added
010000 = Internal temperature sensor
010001 = Band gap voltage reference
010010 = V_{REF_REP}: Center of the two V_{OUT} floating references
010011 = Output under voltage comparator reference
010100 = Output over voltage comparator reference
010101 = Input under voltage comparator reference
010110 = Input over voltage comparator reference
010111 = Over current reference
011000 = Master's current sense signal input (measured on Slave unit)
011001 = V_{BGR_REP}: DAC reference voltage amplifier output
011010 = GPA3 Buffered
011011 = Internal virtual ground (~500mV)
011100 = RAWI after Sample & Hold and added gain, but before slope is added
011101 = E/A output after the clamp, input to the PWM comparator
011110 = Raw I_{SENSE} (input to Sample & Hold)
011111 = Error Amplifier clamp reference level shifted by 500mV
•
111111 = Unimplemented

bit 1 **GO/$\overline{\text{DONE}}$:** A/D Conversion Status bit

1 = A/D conversion cycle in progress. Setting this bit starts an A/D conversion cycle.
This bit is automatically cleared by hardware when the A/D conversion has completed.
0 = A/D conversion completed/not in progress

bit 0 **ADON:** ADC Enable bit

1 = ADC is enabled
0 = ADC is disabled and consumes no operating current

Note 1: Not implemented on MCP19122.

REGISTER 19-2: ADCON1: A/D CONTROL REGISTER 1

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
—	ADCS2	ADCS1	ADCS0	—	ADFM	VCFG1	VCFG0
bit 7							bit 0

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

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

bit 6-4 **ADCS<2:0>:** A/D Conversion Clock Select bits
000 = Reserved
001 = $F_{OSC}/8$
010 = $F_{OSC}/32$
x11 = F_{RC} (clock derived from internal oscillator with a divisor of 16)
100 = Reserved
101 = $F_{OSC}/16$
110 = $F_{OSC}/64$

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

bit 2 **ADFM:** A/D Result Format Select
1 = Right justified
0 = Left justified

bit 1-0 **VCFG<1:0>:** A/D Voltage Reference bit
11 = AD_{REF} pin
10 = AD_{REF} pin
01 = Internal Vdd Reference
00 = Internal A/D Reference

REGISTER 19-3: ADRESH: ADC RESULT REGISTER HIGH (ADRESH)

U-0	U-0	U-0	U-0	U-0	U-0	R-x	R-x
—	—	—	—	—	—	ADRES9	ADRES8
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 7-2 **Unimplemented:** Read as ‘0’

bit 1-0 **ADRES<9:8>**: Most Significant A/D Results

Note 1: Only for ADFM = 1.

REGISTER 19-4: ADRESL: ADC RESULT REGISTER LOW (ADRESL)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
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 7-0 **ADRES<7:0>**: Least Significant A/D results

Note 1: Only for ADFM = 1.

19.4 A/D Acquisition Requirements

For the ADC to meet its specified accuracy, the charge holding capacitor (C_{HOLD}) must be allowed to fully charge to the input channel voltage level. The Analog Input model is shown in Figure 19-4. The source impedance (R_S) and the internal sampling switch (R_{SS}) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (R_{SS}) impedance varies over the device voltage (V_{DD}), refer to Figure 19-4.

The maximum recommended impedance for analog sources is 10 kΩ. As the source impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (or changed), an A/D acquisition must be done before the conversion can be started. To calculate the minimum acquisition time, Equation 19-1 may be used. This equation assumes that 1/2 LSb error is used (1,024 steps for the ADC). The 1/2 LSb error is the maximum error allowed for the ADC to meet its specified resolution.

EQUATION 19-1: ACQUISITION TIME EXAMPLE

Assumptions: *Temperature = +50°C and external impedance of 10 kΩ 5.0V* V_{DD}

$$T_{ACQ} = \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient}$$
$$= T_{AMP} + T_C + T_{COFF}$$
$$= 2\,\mu s + T_C + [(\text{Temperature} - 25°C)(0.05\,\mu s/°C)]$$

The value for T_C *can be approximated with the following equations:*

$$V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) = V_{CHOLD} \quad ;[1]\ V_{CHOLD} \text{ charged to within 1/2 lsb}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{CHOLD} \quad ;[2]\ V_{CHOLD} \text{ charge response to } V_{APPLIED}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) \quad ;\text{combining [1] and [2]}$$

Note: *Where n = number of bits of the ADC.*

Solving for T_C*:*

$$T_C = -C_{HOLD}(R_{IC} + R_{SS} + R_S)\ ln(1/2047)$$
$$= -10\,pF(1\,k\Omega + 7\,k\Omega + 10\,k\Omega)\ ln(0.0004885)$$
$$= 1.37\mu s$$

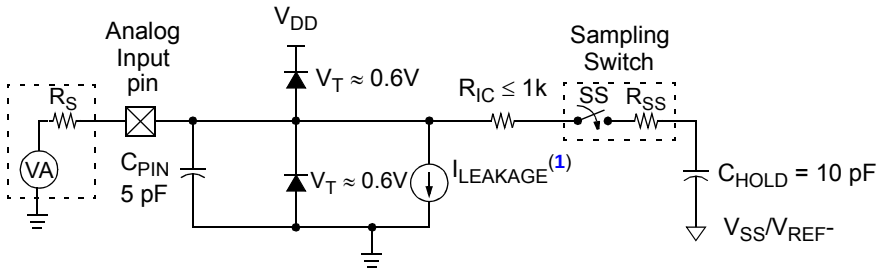
Therefore:

$$T_{ACQ} = 2\,\mu s + 1.37\mu s + [(50°C - 25°C)(0.05\mu s/°C)]$$
$$= 4.67\,\mu s$$

Note 1: The charge holding capacitor (C_{HOLD}) is not discharged after each conversion.

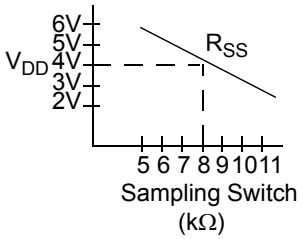
2: The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

FIGURE 19-4: ANALOG INPUT MODEL



Legend:

C_{HOLD} = Sample/Hold Capacitance
C_{PIN} = Input Capacitance
$I_{LEAKAGE}$ = Leakage current at the pin due to various junctions
R_{IC} = Interconnect Resistance
R_{SS} = Resistance of Sampling Switch
SS = Sampling Switch
V_T = Threshold Voltage



Note 1: Refer to **Section 5.0 "Digital Electrical Characteristics"**.

FIGURE 19-5: ADC TRANSFER FUNCTION

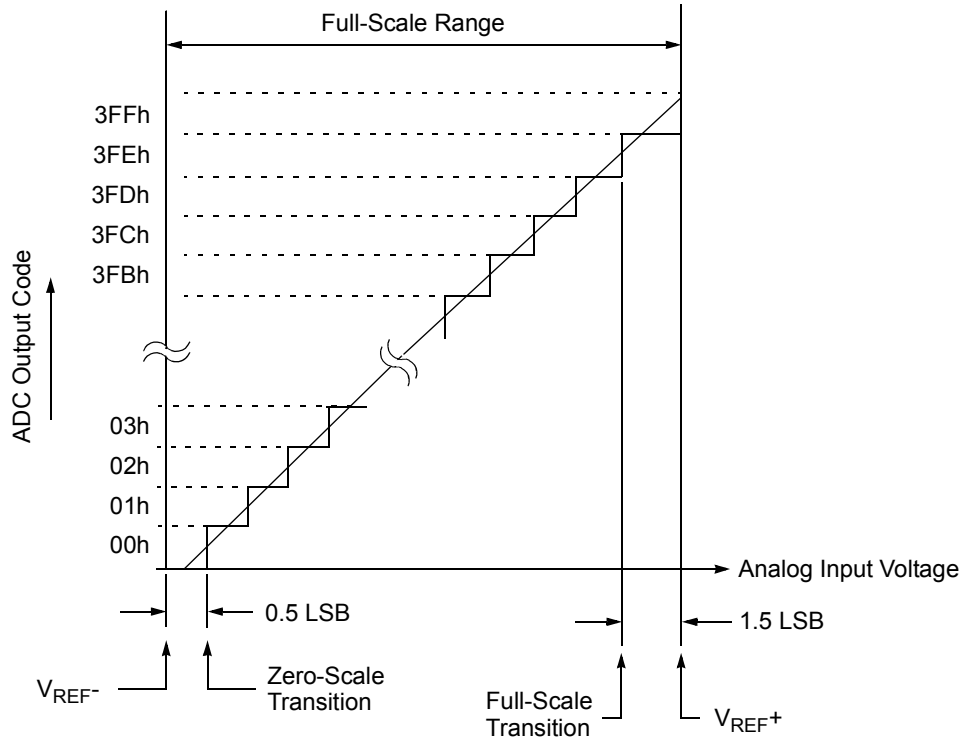


TABLE 19-2: SUMMARY OF REGISTERS ASSOCIATED WITH ADC

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ADCON0	—	CHS4	CHS3	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON	138
ADCON1	—	ADCS2	ADCS1	ADCS0	—	ADFM	VCFG1	VCFG0	139
ADRESH	—	—	—	—	—	—	ADRES9	ADRES8	140
ADRESL	ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0	140
ANSELA	—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0	120
ANSELB	—	—	ANSB5	ANSB4	—	ANSB2	ANSB1	—	126
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	102
PIE1	—	ADIE	BCLIE	SSPIE	—	—	TMR2IE	TMR1IE	103
PIR1	—	ADIF	BCLIF	SSPIF	—	—	TMR2IF	TMR1IF	105
TRISGPA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	119
TRISGPB	TRISB7	TRISB6	TRISB5	TRISB4	—	TRISB2	TRISB1	TRISB0	125

Legend: — = unimplemented read as '0'. Shaded cells are not used for ADC module.

NOTES:

20.0 FLASH PROGRAM MEMORY CONTROL

The Flash program memory is readable and writable during normal operation (full V_{IN} range). This memory is not directly mapped in the register file space. Instead, it is indirectly addressed through the Special Function Registers (see Registers 20-1 to 20-5). There are six SFRs used to read and write this memory:

- PMCON1
- PMCON2
- PMDATL
- PMDATH
- PMADRL
- PMADRH

When interfacing the program memory block, the PMDATL and PMDATH registers form a two-byte word, which holds the 14-bit data for read/write, and the PMADRL and PMADRH registers form a two-byte word, which holds the 13-bit address of the FLASH location being accessed. These devices have 4K words of program Flash with an address range from 0000h to 0FFFh.

The program memory allows single-word read and a by four word write. A four-word write automatically erases the row of the location and writes the new data (erase before write).

The write time is controlled by an on-chip timer. The write/erase voltages are generated by an on-chip charge pump rated to operate over the voltage range of the device for byte or word operations.

When the device is code protected, the CPU may continue to read and write the Flash program memory.

Depending on the settings of the Flash Program Memory Enable (WRT<1:0>) bits, the device may or may not be able to write certain blocks of the program memory, however, reads of the program memory are allowed.

When the Flash Program Memory Code Protection ($\overline{CP}$) bit is enabled, the program memory is code protected, and the device programmer (ICSP) cannot access data or program memory.

20.1 PMADRH and PMADRL Registers

The PMADRH and PMADRL registers can address up to a maximum of 4K words of program memory.

When selecting a program address value, the Most Significant Byte (MSB) of the address is written to the PMADRH register and the Least Significant Byte (LSB) is written to the PMADRL register.

20.2 PMCON1 and PMCON2 Registers

PMCON1 is the control register for the data program memory accesses.

Control bits RD and WR initiate read and write, respectively. These bits cannot be cleared, only set in software. They are cleared in hardware at completion of the read or write operation. The inability to clear the WR bit in software prevents the accidental premature termination of a write operation.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear.

The CALSEL bit allows the user to read locations in test memory in case there are calibration bits stored in the calibration word locations that need to be transferred to SFR trim registers. The CALSEL bit is only for reads, and if a write operation is attempted with CALSEL = 1, no write will occur.

PMCON2 is not a physical register. Reading PMCON2 will read all '0's. The PMCON2 register is used exclusively in the flash memory write sequence.

20.3 Flash Program Memory Control Registers

REGISTER 20-1: PMDATL: PROGRAM MEMORY DATA LOW BYTE REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PMDATL<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 7-0 **PMDATL<7:0>**: 8 Least Significant Data bits Read from Program Memory

REGISTER 20-2: PMADRL: PROGRAM MEMORY ADDRESS LOW BYTE REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PMADRL<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 7-0 **PMADRL<7:0>**: 8 Least Significant Address bits for Program Memory Read/Write Operation

REGISTER 20-3: PMDATH: PROGRAM MEMORY DATA HIGH BYTE REGISTER

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	PMDATH<5:0>					
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 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **PMDATH<5:0>**: 6 Most Significant Data bits Read from Program Memory

REGISTER 20-4: PMADRH: PROGRAM MEMORY ADDRESS HIGH BYTE REGISTER

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	PMADRH<3:0>			
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 7-4 **Unimplemented:** Read as '0'

bit 3-0 **PMADRH<3:0>**: Specifies the 4 Most Significant Address bits or High bits for Program Memory Reads.

REGISTER 20-5: PMCON1: PROGRAM MEMORY CONTROL REGISTER 1

U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/S-0	R/S-0
—	CALSEL	—	—	—	WREN	WR	RD
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 7 **Unimplemented:** Read as '0'

bit 6 **CALSEL:** Program Memory calibration space select bit
- 1 = Select test memory area for reads only, for loading calibration (excluding Configuration Word and Device ID)
- 0 = Select user area for reads

bit 5-3 **Unimplemented:** Read as '0'

bit 2 **WREN:** Program Memory Write Enable bit
- 1 = Allows write cycles
- 0 = Inhibits write to the Flash Program Memory

bit 1 **WR:** Write Control bit
- 1 = Initiates a write cycle to program memory. (The bit is cleared by hardware when write is complete. The WR bit can only be set (not cleared) in software.)
- 0 = Write cycle to the Flash memory is complete

bit 0 **RD:** Read Control bit
- 1 = Initiates a program memory read. (The read takes one cycle. The RD is cleared in hardware; the RD bit can only be set (not cleared) in software).
- 0 = Does not initiate a Flash memory read

20.3.1 READING THE FLASH PROGRAM MEMORY

To read a program memory location, the user must write two bytes of the address to the PMADRL and PMADRH registers, and then set control bit RD (PMCON1<0>). Once the read control bit is set, the program memory Flash controller will use the second instruction cycle after to read the data. This causes the second instruction immediately following the "BSF PMCON1,RD" instruction to be ignored. The data is available, in the very next cycle, in the PMDATL and PMDATH registers; it can be read as two bytes in the following instructions. PMDATL and PMDATH registers will hold this value until another read or until it is written to by the user (during a write operation).

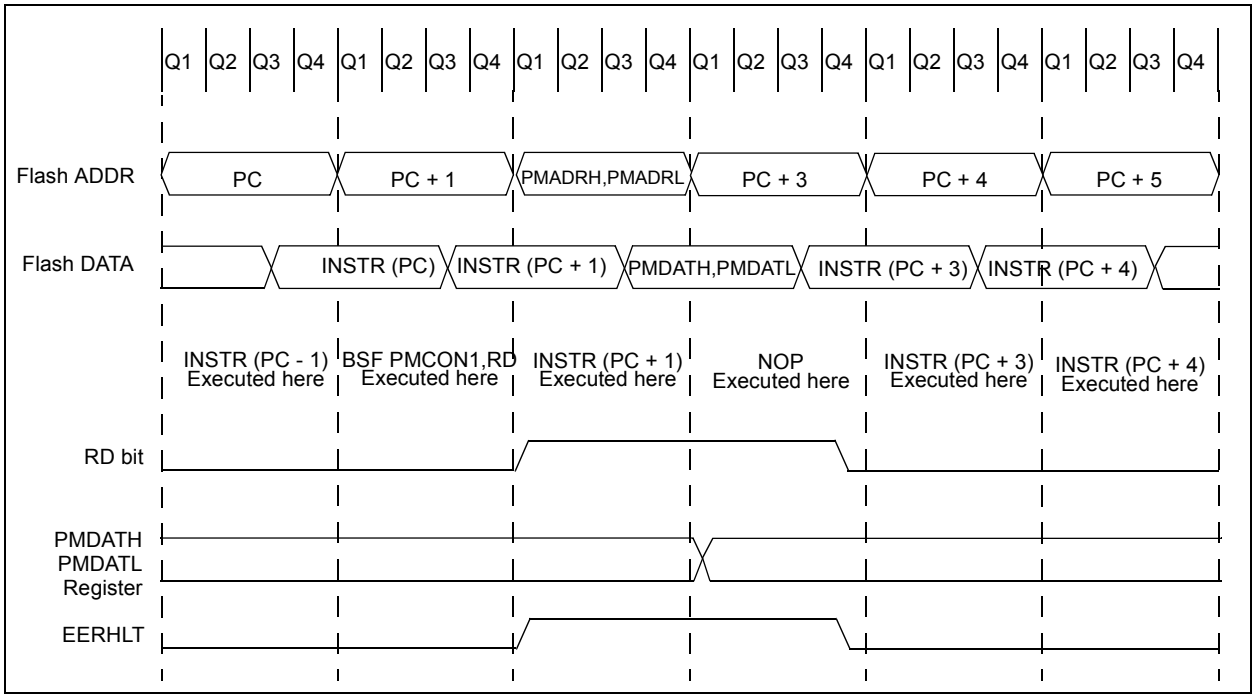
EXAMPLE 20-1: FLASH PROGRAM READ

```
    BANKSELPM_ADR; Change STATUS bits RP1:0 to select bank with PMADR
    MOVLWMS_PROG_PM_ADDR;
    MOVWFPMADRH; MS Byte of Program Address to read
    MOVLWLS_PROG_PM_ADDR;
    MOVWFPMADRL; LS Byte of Program Address to read
    BANKSELPMCON1; Bank to containing PMCON1
    BSF PMCON1, RD; EE Read

    NOP     ; First instruction after BSF PMCON1,RD executes normally

    NOP     ; Any instructions here are ignored as program
            ; memory is read in second cycle after BSF PMCON1,RD
            ;
    BANKSELPMDATL; Bank to containing PMADRL
    MOVFPMDATL, W; W = LS Byte of Program PMDATL
    MOVFPMDATH, W; W = MS Byte of Program PMDATL
```

FIGURE 20-1: FLASH PROGRAM MEMORY READ CYCLE EXECUTION – NORMAL MODE



20.3.2 WRITING TO THE FLASH PROGRAM MEMORY

A word of the Flash program memory may only be written to if the word is in an unprotected segment of memory, as defined in **Section 11.1 "Configuration Bits"** (bits WRT1:WRT0).

Note:	The write protect bits are used to protect the users' program from modification by the user's code. They have no effect when programming is performed by ICSP. The code-protect bits, when programmed for code protection, will prevent the program memory from being written via the ICSP interface.

Flash program memory must be written in four-word blocks. See Figures 20-2 and 20-3 for more details. A block consists of four words with sequential addresses, with a lower boundary defined by an address, where PMADRL<1:0> = 00. All block writes to program memory are done as 16-word erase by four-word write operations. The write operation is edge-aligned and cannot occur across boundaries.

To write program data, the WREN bit must be set and the data must first be loaded into the buffer registers (see Figure 20-2). This is accomplished by first writing the destination address to PMADRL and PMADRH, and then writing the data to PMDATL and PMDATH. After the address and data have been set, then the following sequence of events must be executed:

1. Write 55h, then AAh, to PMCON2 (Flash programming sequence).
2. Set the WR control bit of the PMCON1 register.

All four buffer register locations should be written to with correct data. If less than four words are being written to in the block of four words, then a read from the program memory location(s) not being written to must be performed. This takes the data from the program memory location(s) not being written and loads it into the PMDATL and PMDATH registers. Then the sequence of events to transfer data to the buffer registers must be executed.

To transfer data from the buffer registers to the program memory, the PMADRL and PMADRH must point to the last location in the four-word block (PMADRL<1:0> = 11). Then the following sequence of events must be executed:

1. Write 55h, then AAh, to PMCON2 (Flash programming sequence).
2. Set control bit WR of the PMCON1 register to begin the write operation.

The user must follow the same specific sequence to initiate the write for each word in the program block, writing each program word in sequence (000, 001, 010, 011). When the write is performed on the last word (PMADRL<1:0> = 11), a block of 16 words is automatically erased and the content of the four-word buffer registers are written into the program memory.

After the "BSF PMCON1,WR" instruction, the processor requires two cycles to set up the erase/write operation. The user must place two NOP instructions after the WR bit is set. Since data is being written to buffer registers, the writing of the first three words of the block appears to occur immediately. The processor will halt internal operation for the typical 2ms to write to the memory only when the PMADRL<3:0> = xx11. The halt time will be 4ms typical if the part is also erasing which only occurs if the PMADRL<3:0> = 0011.

Refer to Figure 20-2 for a block diagram of the buffer registers and the control signals for test mode.

20.3.3 PROTECTION AGAINST SPURIOUS WRITE

There are conditions when the device should not write to the program memory. To protect against spurious writes, various mechanisms have been built in. On power-up, WREN is cleared. Also, the Power-up Timer (72 ms duration) prevents program memory writes.

The write initiate sequence, and the WREN bit, help prevent an accidental write during a power glitch or software malfunction.

20.3.4 OPERATION DURING CODE PROTECT

When the device is code protected, the CPU is able to read and write unscrambled data to the program memory. The test mode access is disabled.

20.3.5 OPERATION DURING WRITE PROTECT

When the program memory is write protected, the CPU can read and execute from the program memory. The portions of program memory that are write protected can not be modified by the CPU using the PMCON registers. The write protection has no effect in ICSP mode.

FIGURE 20-2: BLOCK WRITES TO 4K FLASH PROGRAM MEMORY

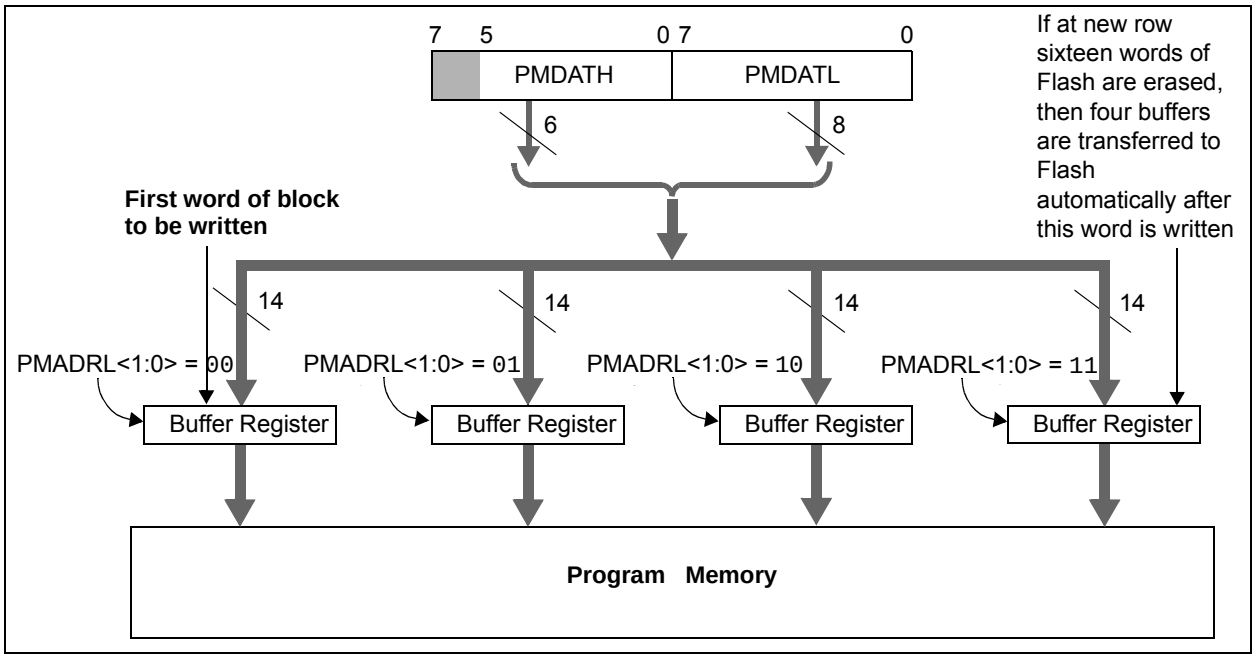
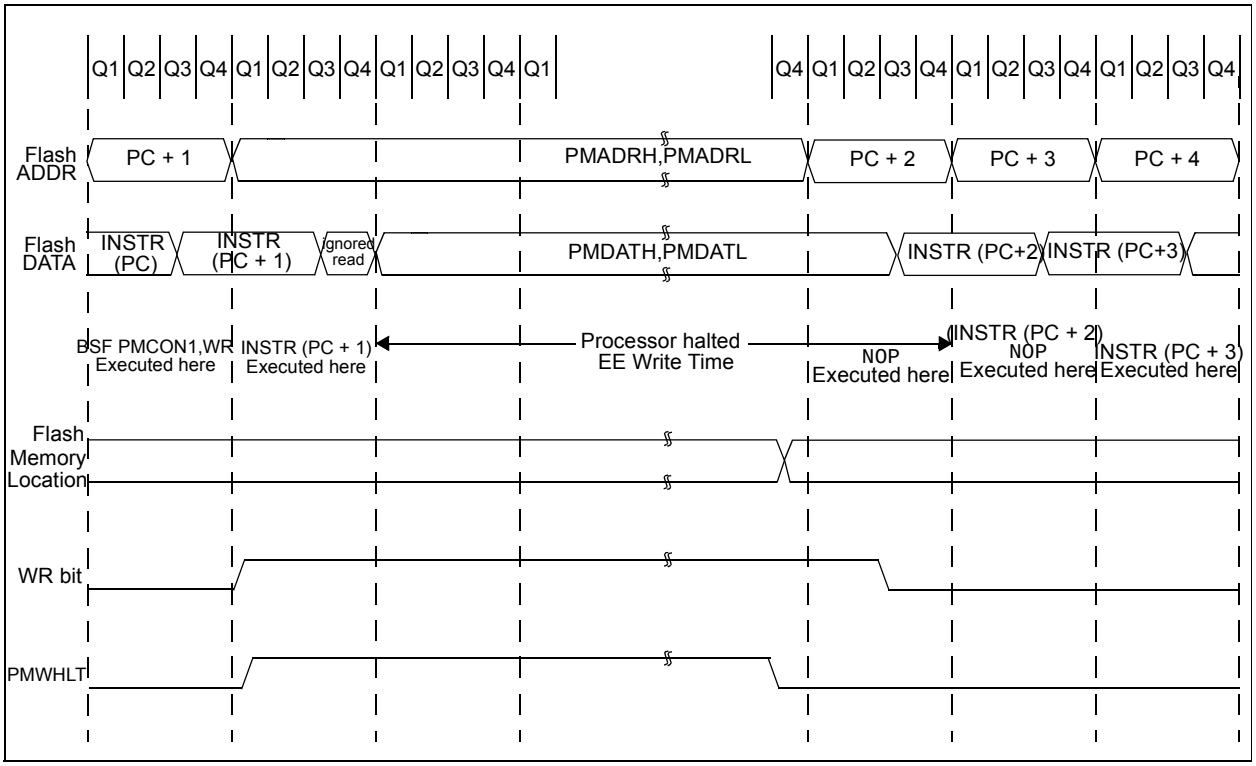


FIGURE 20-3: FLASH PROGRAM MEMORY LONG WRITE CYCLE EXECUTION



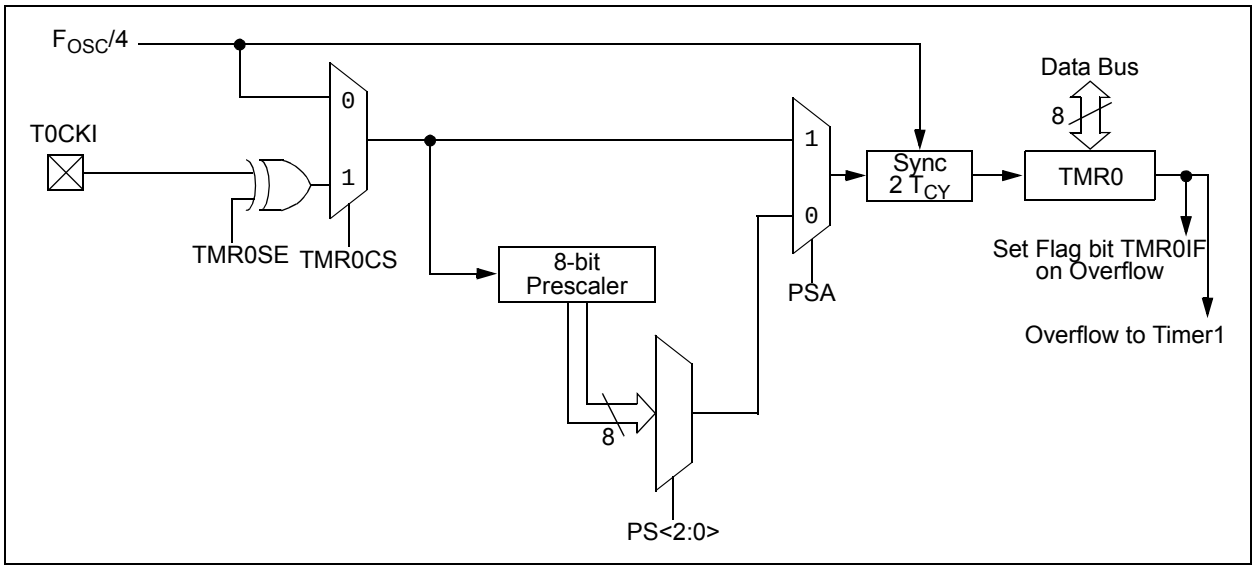
21.0 TIMER0 MODULE

The Timer0 module is an 8-bit timer/counter with the following features:

- 8-bit timer/counter register (TMR0)
- 8-bit prescaler (shared with Watchdog Timer)
- Programmable internal or external clock source
- Programmable external clock edge selection
- Interrupt on overflow

Figure 21-1 is a block diagram of the Timer0 module.

FIGURE 21-1: BLOCK DIAGRAM OF THE TIMER0



21.1 Timer0 Operation

The Timer0 module can be used as either an 8-bit timer or an 8-bit counter.

21.1.1 8-BIT TIMER MODE

The Timer0 module will increment every instruction cycle, if used without a prescaler. 8-Bit Timer mode is selected by clearing the T0CS bit of the OPTION_REG register.

When TMR0 is written, the increment is inhibited for two instruction cycles immediately following the write.

Note:	The value written to the TMR0 register can be adjusted, in order to account for the two instruction cycle delay when TMR0 is written.

21.1.2 8-BIT COUNTER MODE

In 8-Bit Counter mode, the Timer0 module will increment on every rising or falling edge of the T0CKI pin. The incrementing edge is determined by the T0SE bit of the OPTION_REG register.

8-Bit Counter mode using the T0CKI pin is selected by setting the T0CS bit in the OPTION_REG register to '1'.

21.1.3 SOFTWARE PROGRAMMABLE PRESCALER

A single software programmable prescaler is available for use with either Timer0 or the Watchdog Timer (WDT), but not both simultaneously. The prescaler assignment is controlled by the PSA bit of the OPTION_REG register. To assign the prescaler to Timer0, the PSA bit must be cleared to '0'.

There are 8 prescaler options for the Timer0 module ranging from 1:2 to 1:256. The prescale values are selectable via the PS<2:0> bits of the OPTION_REG register. In order to have a 1:1 prescaler value for the Timer0 module, the prescaler must be disabled by setting the PSA bit of the OPTION_REG register.

The prescaler is not readable or writable. When assigned to the Timer0 module, all instructions writing to the TMR0 register will clear the prescaler.

21.1.4 SWITCHING PRESCALER BETWEEN TIMER0 AND WDT MODULES

The prescaler is shared between the Timer0 and the WDT. As a result of having the prescaler assigned to either Timer0 or the WDT, it is possible to generate an unintended device Reset when switching prescaler values. When changing the prescaler assignment from Timer0 to the WDT module, the instruction sequence shown in Example 21-1 must be executed.

EXAMPLE 21-1: CHANGING PRESCALER (TIMER0 → WDT)

```
BANKSELTMR0     ;
CLRWDT          ;Clear WDT
CLRFTMR0       ;Clear TMR0 and
                ;prescaler
BANKSELOPTION_REG;
BSF OPTION_REG,PSA;Select WDT
CLRWDT          ;
                ;
MOVLWb'11111000';Mask prescaler
ANDWFOPTION_REG,W;bits
IORLWb'00000101';Set WDT prescaler
MOVWFOPTION_REG;to 1:32
```

When changing the prescaler assignment from the WDT to the Timer0 module, the following instruction sequence must be executed (see Example 21-2).

EXAMPLE 21-2: CHANGING PRESCALER (WDT → TIMER0)

```
CLRWDT              ;Clear WDT and
                    ;prescaler
BANKSEL OPTION_REG  ;
MOVLW   b'11110000' ;Mask TMR0 select and
ANDWF   OPTION_REG,W ;prescaler bits
IORLW   b'00000011' ;Set prescale to 1:16
MOVWF   OPTION_REG  ;
```

21.1.5 TIMER0 INTERRUPT

Timer0 will generate an interrupt when the TMR0 register overflows from FFh to 00h. The T0IF interrupt flag bit of the INTCON register is set every time the TMR0 register overflows, regardless of whether or not the Timer0 interrupt is enabled. The T0IF bit can only be cleared in software. The Timer0 interrupt enable is the T0IE bit of the INTCON register.

Note:	The Timer0 interrupt cannot wake the processor from Sleep since the timer is frozen during Sleep.

21.1.6 USING TIMER0 WITH AN EXTERNAL CLOCK

When Timer0 is in Counter mode, the synchronization of the T0CKI input and the Timer0 register is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks. Therefore, the high and low periods of the external clock source must meet the timing requirements as shown in **Section 5.0 "Digital Electrical Characteristics"**.

21.1.7 OPERATION DURING SLEEP

Timer0 cannot operate while the processor is in Sleep mode. The contents of the TMR0 register will remain unchanged while the processor is in Sleep mode.

TABLE 21-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER0

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	T0IE	INTE	IOCIE	T0IF	INTF	IOCIF	102
OPTION_REG	RAUP	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	83
TMR0	Timer0 Module Register								151*
TRISGPA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	119

Legend: — = Unimplemented locations, read as '0'. Shaded cells are not used by the Timer0 module.

* Page provides register information.

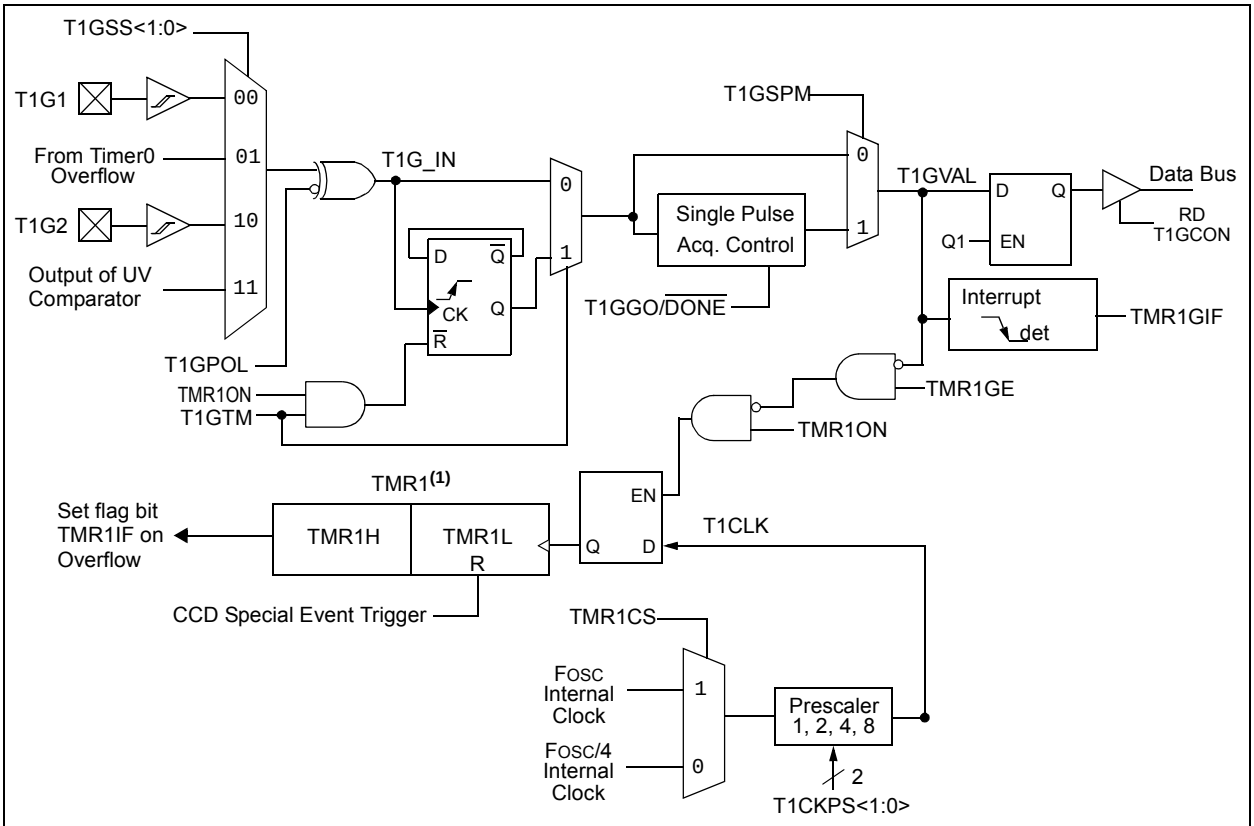
22.0 TIMER1 MODULE WITH GATE CONTROL

The Timer1 module is a 16-bit timer with the following features:

- 16-bit timer/counter register pair (TMR1H:TMR1L)
- Two selectable internal clock sources
- 2-bit prescaler
- Synchronous or asynchronous operation
- Multiple Timer1 gate (count enable) sources
- Interrupt on overflow
- Time base for the Capture/Compare function
- Special Event Trigger (with CCD)
- Selectable Gate Source Polarity
- Gate Toggle mode
- Gate Single-pulse mode
- Gate Value Status
- Gate Event Interrupt

Figure 22-1 is a block diagram of the Timer1 module.

FIGURE 22-1: TIMER1 BLOCK DIAGRAM



Note 1: Timer1 register increments on rising edge.

22.1 Timer1 Operation

The Timer1 module is a 16-bit incrementing timer, which is accessed through the TMR1H:TMR1L register pair. Writes to TMR1H or TMR1L directly update the timer. The module is a timer and increments on every instruction cycle.

Timer1 is enabled by configuring the TMR1ON and TMR1GE bits in the T1CON and T1GCON registers, respectively. Table 22-1 displays the Timer1 enable selections.

TABLE 22-1: TIMER1 ENABLE SELECTIONS

TMR1ON	TMR1GE	Timer1 Operation
0	0	Off
0	1	Off
1	0	Always On
1	1	Externally Enabled

22.2 Clock Source Selection

The TMR1CS bit of the T1CON register is used to select the clock source for Timer1. Table 22-2 displays the clock source selections.

TABLE 22-2: CLOCK SOURCE SELECTIONS

TMR1CS	Clock Source
1	8 MHz System Clock (F_{OSC})
0	2 MHz Instruction Clock ($F_{OSC}/4$)

22.2.1 INTERNAL CLOCK SOURCE

When the internal clock source is selected, the TMR1H:TMR1L register pair will increment on multiples of FOSC or FOSC/4 as determined by the Timer1 prescaler.

As an example, when the FOSC internal claok source is selected, the TIMER1 register value will increment by four counts every instruction clock cycle.

Note: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge (see Figure 22-2) after any one or more of the following conditions:

- Timer1 enabled after POR Reset
- Write to TMR1H or TMR1L
- Timer1 is disabled
- Timer1 is disabled (TMR1ON = 0) when T1CKI is high then Timer1 is enabled (TMR1ON=1) when T1CKI is low.

22.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The T1CKPS bits of the T1CON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

22.4 Timer1 Gate

Timer1 can be configured to increment freely or the incrementing can be enabled and disabled using Timer1 gate circuitry. This is also referred to as Timer1 gate increment enable.

Timer1 gate can also be driven by multiple selectable sources.

22.4.1 TIMER1 GATE INCREMENT ENABLE

The Timer1 gate is enabled by setting the TMR1GE bit of the T1GCON register. The polarity of the Timer1 gate is configured using the T1GPOL bit of the T1GCON register.

When Timer1 Gate (T1Gx) input is active, Timer1 will increment on the rising edge of the Timer1 clock source. When Timer1 gate input is inactive, no incrementing will occur and Timer1 will hold the current count. See Figure 22-3 for timing details.

TABLE 22-3: TIMER1 GATE ENABLE SELECTIONS

T1CLK	T1GPOL	T1Gx	Timer1 Operation
↑	0	0	Increments
↑	0	1	Holds Count
↑	1	0	Holds Count
↑	1	1	Increments

22.4.2 TIMER1 GATE SOURCE SELECTION

The Timer1 gate source can be selected from one of three different sources. Source selection is controlled by the T1GSS bits of the T1GCON register. The polarity for each available source is also selectable. Polarity selection is controlled by the T1GPOL bit of the T1GCON register.

TABLE 22-4: TIMER1 GATE SOURCES

T1GSS	Timer1 Gate Source
11	Output of UV Comparator
10	Timer1 Gate Pin T1G2
01	Overflow of Timer0 (TMR0 increments from FFh to 00h)
00	Timer1 Gate Pin T1G1

22.4.2.1 T1G1 Pin Gate Operation

The GPBA3/T1G1 pin is one source for Timer1 gate control. It can be used to supply an external source to the Timer1 gate circuitry.

22.4.2.2 Timer0 Overflow Gate Operation

When Timer0 increments from FFh to 00h, a low-to-high pulse will automatically be generated and internally supplied to the Timer1 gate circuitry.

22.4.2.3 T1G2 Pin Gate Operation

The GPB2/T1G2 pin is one source for the Timer1 gate control. It can be used to supply an external source to the Timer1 gate circuitry.

22.4.2.4 UV Comparator Output

The output of the output under voltage comparator is one source for the Timer1 gate control. A low-to-high transition of the comparator output shall stop TIMER1.

22.4.3 TIMER1 GATE TOGGLE MODE

When Timer1 Gate Toggle mode is enabled, it is possible to measure the full-cycle length of a Timer1 gate signal, as opposed to the duration of a single level pulse.

The Timer1 gate source is routed through a flip-flop that changes state on every incrementing edge of the signal. See Figure 22-4 for timing details.

Timer1 Gate Toggle mode is enabled by setting the T1GTM bit of the T1GCON register. When the T1GTM bit is cleared, the flip-flop is cleared and held clear. This is necessary in order to control which edge is measured.

Note:	Enabling Toggle mode at the same time as changing the gate polarity may result in indeterminate operation.

22.4.4 TIMER1 GATE SINGLE-PULSE MODE

When Timer1 Gate Single-Pulse mode is enabled, it is possible to capture a single pulse gate event. Timer1 Gate Single-Pulse mode is first enabled by setting the T1GSPM bit in the T1GCON register. Next, the T1GGO/$\overline{\text{DONE}}$ bit in the T1GCON register must be set. The Timer1 will be fully enabled on the next incrementing edge. On the next trailing edge of the

pulse, the T1GGO/$\overline{\text{DONE}}$ bit will automatically be cleared. No other gate events will be allowed to increment Timer1 until the T1GGO/$\overline{\text{DONE}}$ bit is once again set in software.

Clearing the T1GSPM bit of the T1GCON register will also clear the T1GGO/$\overline{\text{DONE}}$ bit. See Figure 22-5 for timing details.

Enabling the Toggle mode and the Single-Pulse mode simultaneously will permit both sections to work together. This allows the cycle times on the Timer1 gate source to be measured. See Figure 22-6 for timing details.

22.4.5 TIMER1 GATE VALUE STATUS

When Timer1 gate value status is utilized, it is possible to read the most current level of the gate control value. The value is stored in the T1GVAL bit in the T1GCON register. The T1GVAL bit is valid even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

22.4.6 TIMER1 GATE EVENT INTERRUPT

When Timer1 gate event interrupt is enabled, it is possible to generate an interrupt upon the completion of a gate event. When the falling edge of T1GVAL occurs, the TMR1GIF flag bit in the PIR1 register will be set. If the TMR1GIE bit in the PIE1 register is set, then an interrupt will be recognized.

The TMR1GIF flag bit operates even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

22.5 Timer1 Interrupt

The Timer1 register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit of the PIR1 register is set. To enable the interrupt on rollover, you must set these bits:

- TMR1ON bit of the T1CON register
- TMR1IE bit of the PIE1 register
- PEIE bit of the INTCON register
- GIE bit of the INTCON register

The interrupt is cleared by clearing the TMR1IF bit in the Interrupt Service Routine.

Note:	The TMR1H:TMR1L register pair and the TMR1IF bit should be cleared before enabling interrupts.

22.6 CCD Capture/Compare Time Base

The CCD module uses the TMR1H:TMR1L register pair as the time base when operating in Capture or Compare mode.

In Capture mode, the value in the TMR1H:TMR1L register pair is copied into the CCxRH:CCxRL register pair on a configured event.

In Compare mode, an event is triggered when the value CCxRH:CCxRL register pair matches the value in the TMR1H:TMR1L register pair. This event can be a Special Event Trigger.

For more information, see Section 24.0, Dual Capture/Compare (CCD) Module.

22.7 CCD Special Event Trigger

When the CCD is configured to trigger a special event, the trigger will clear the TMR1H:TMR1L register pair. This special event does not cause a Timer1 interrupt. The CCD module may still be configured to generate a CCD interrupt.

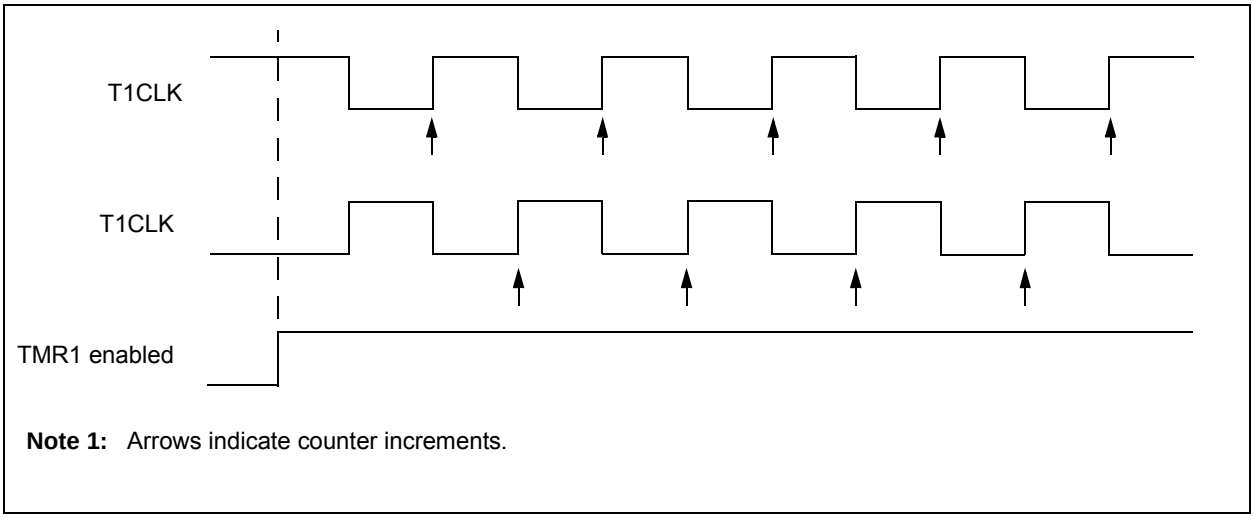
In this mode of operation, the CCxRH:CCxRL register pair becomes the period register for Timer1.

Timer1 should be clocked by FOSC/4 to utilize the Special Event Trigger.

In the event that a write to TMR1H or TMR1L coincides with a Special Event Trigger from the CCD, the write will take precedence.

For more information, see Section 24.2.3, Special Event Trigger.

FIGURE 22-2: TIMER1 INCREMENTING EDGE



Note 1: Arrows indicate counter increments.

FIGURE 22-3: TIMER1 GATE COUNT ENABLE MODE

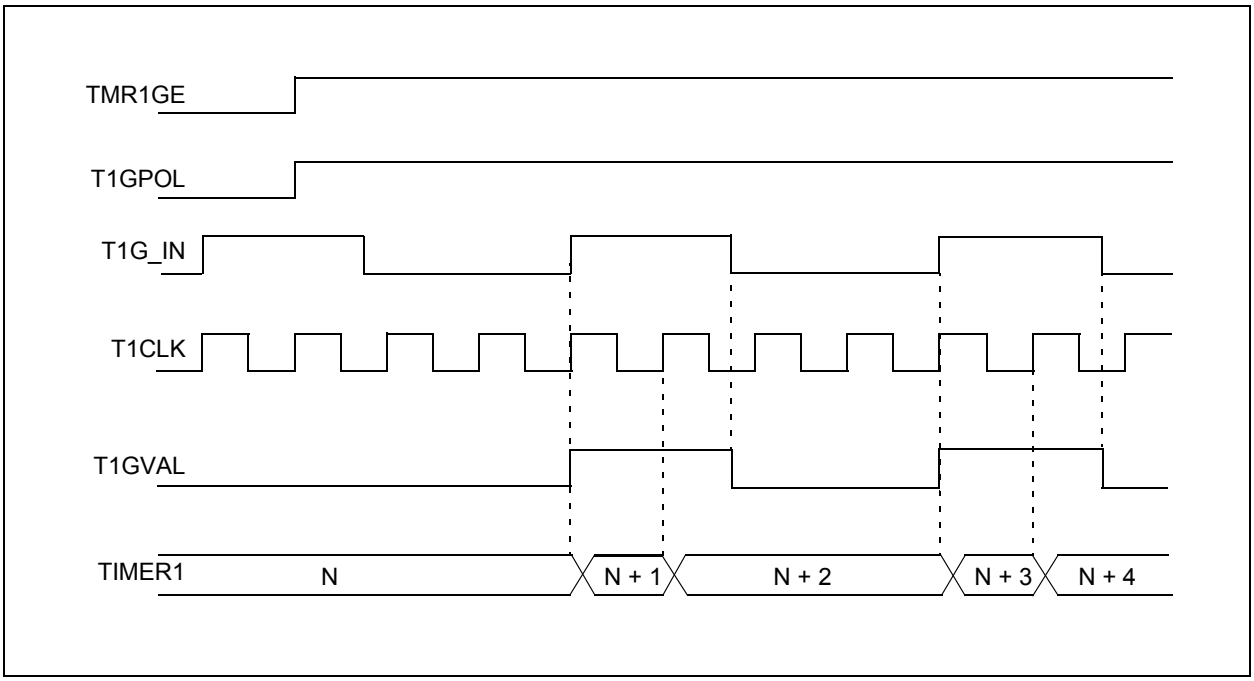


FIGURE 22-4: TIMER1 GATE TOGGLE MODE

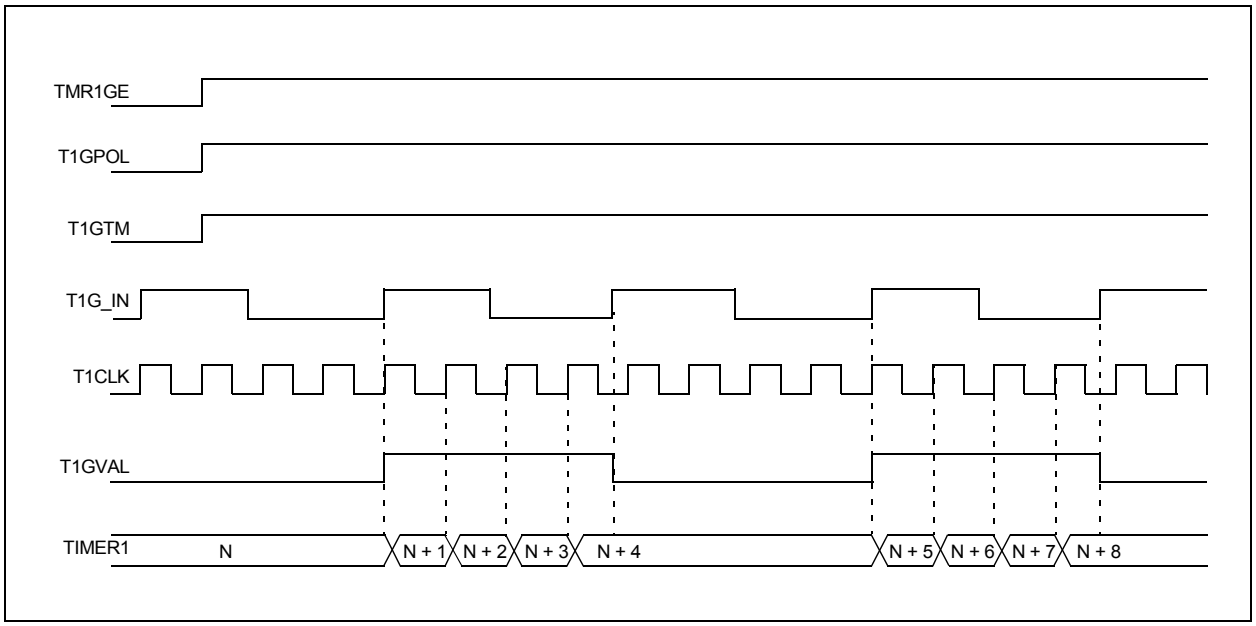


FIGURE 22-5: TIMER1 GATE SINGLE-PULSE MODE

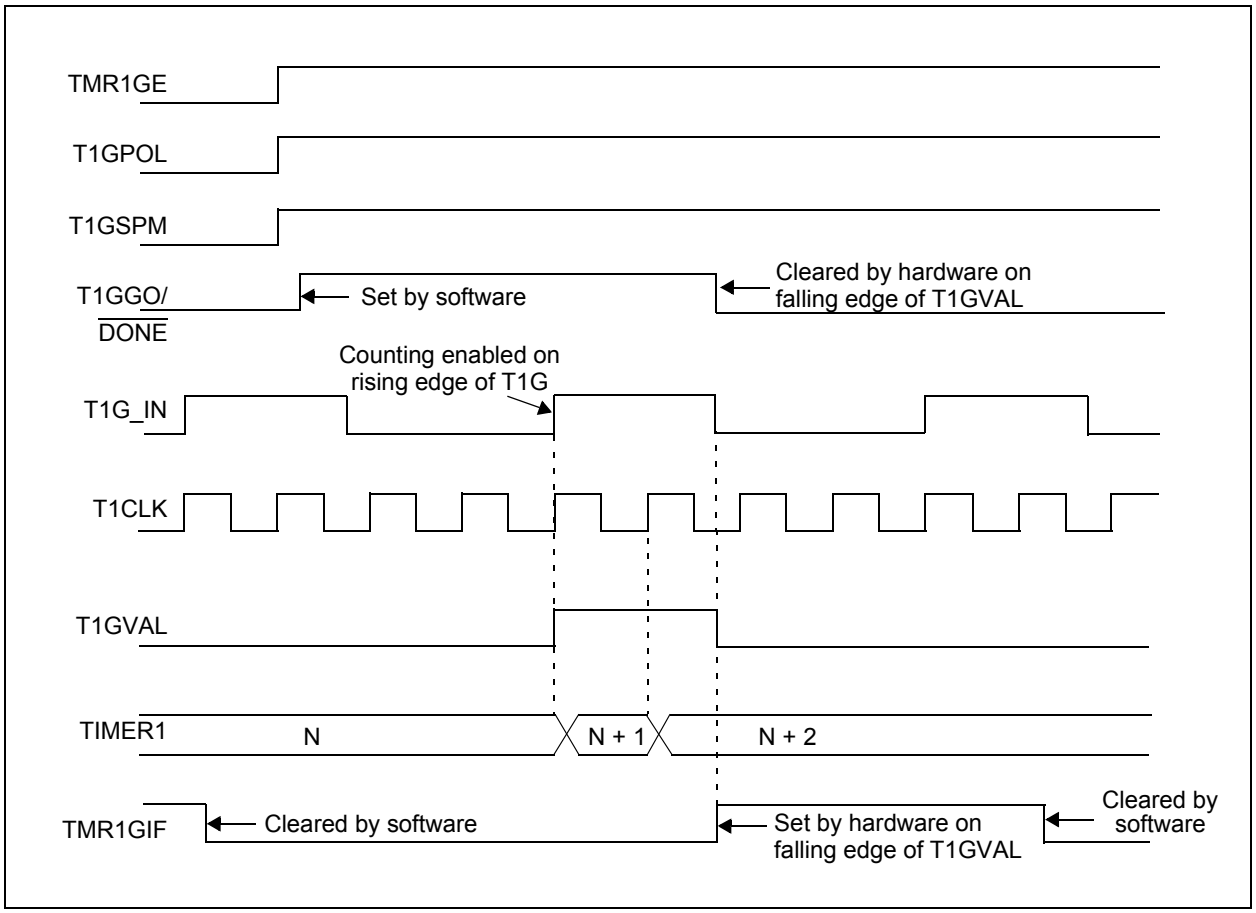
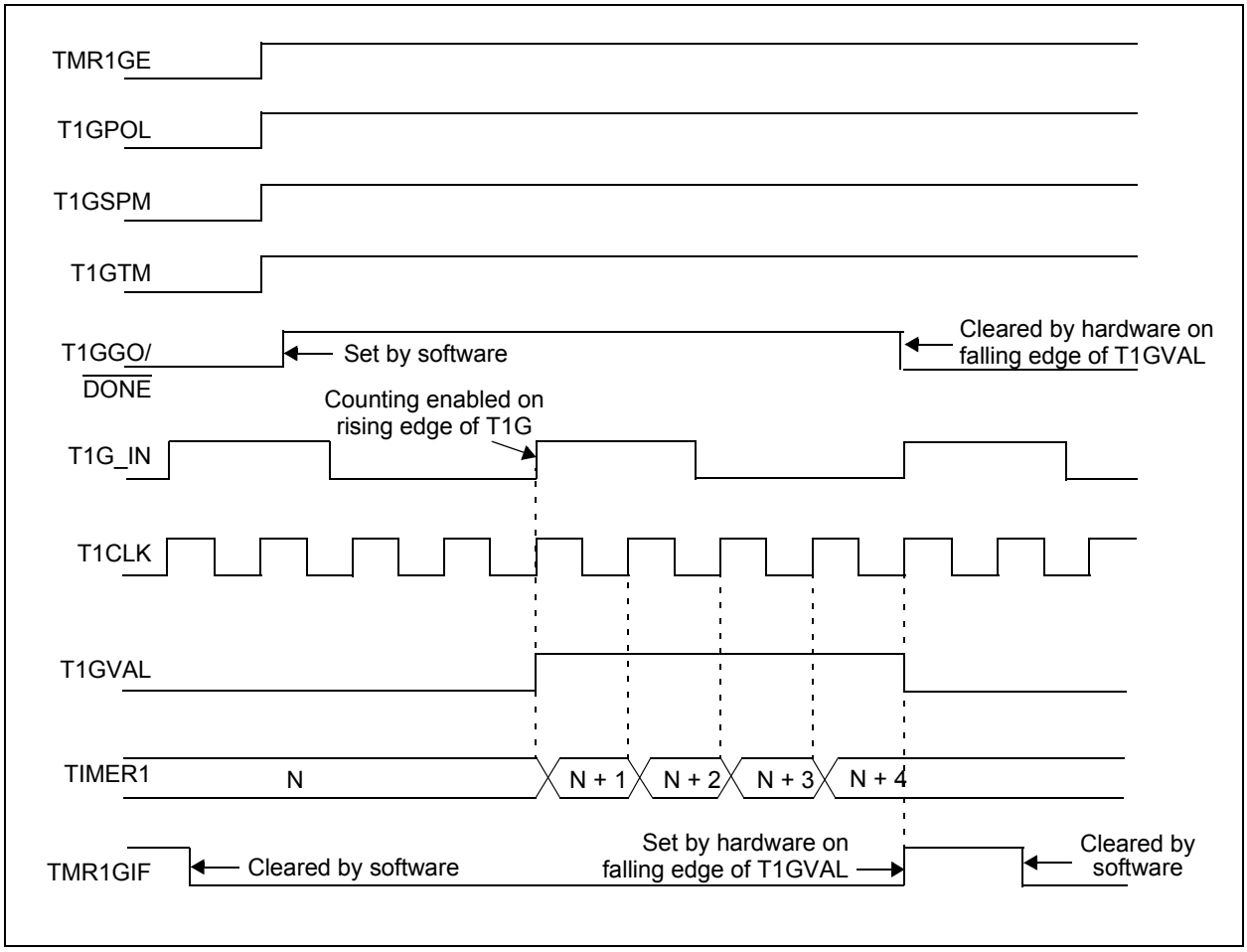


FIGURE 22-6: TIMER1 GATE SINGLE-PULSE AND TOGGLE COMBINED MODE



22.8 Timer1 Control Registers

REGISTER 22-1: T1CON: TIMER1 CONTROL REGISTER

U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	—	T1CKPS<1:0>		—	—	TMR1CS	TMR1ON
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 7-6 **Unimplemented:** Read as '0'

bit 5-4 **T1CKPS<1:0>:** Timer1 Input Clock Prescale Select bits
11 = 1:8 Prescale value
10 = 1:4 Prescale value
01 = 1:2 Prescale value
00 = 1:1 Prescale value

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **TMR1CS:** Timer1 Clock Source Select bits
1 = Timer1 clock source is 8 MHz system clock (FOSC)
0 = Timer1 clock source is 2 MHz instruction clock (FOSC/4)

bit 0 **TMR1ON:** Timer1 On bit
1 = Enables Timer1
0 = Stops Timer1
Clears Timer1 gate flip-flop

REGISTER 22-2: T1GCON: TIMER1 GATE CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-x	R/W-0	R/W-0
TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/$\overline{\text{DONE}}$	T1GVAL	T1GSS<1:0>	
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 7 **TMR1GE:** Timer1 Gate Enable bit

If TMR1ON = 0:
This bit is ignored
If TMR1ON = 1:
1 = Timer1 incrementing is controlled by the Timer1 gate function
0 = Timer1 increments regardless of Timer1 gate function

bit 6 **T1GPOL:** Timer1 Gate Polarity bit

1 = Timer1 gate is active-high (Timer1 counts when gate is high)
0 = Timer1 gate is active-low (Timer1 counts when gate is low)

bit 5 **T1GTM:** Timer1 Gate Toggle mode bit

1 = Timer1 Gate Toggle mode is enabled.
0 = Timer1 Gate Toggle mode is disabled and toggle flip-flop is cleared
Timer1 gate flip-flop toggles on every rising edge.

bit 4 **T1GSPM:** Timer1 Gate Single Pulse mode bit

1 = Timer1 Gate Single-Pulse mode is enabled and is controlling Timer1 gate
0 = Timer1 Gate Single-Pulse mode is disabled

bit 3 **T1GGO/$\overline{\text{DONE}}$:** Timer1 Gate Single-Pulse Acquisition Status bit

1 = Timer1 gate single-pulse acquisition is ready, waiting for an edge
0 = Timer1 gate single-pulse acquisition has completed or has not been started
This bit is automatically cleared when T1GSPM is cleared.

bit 2 **T1GVAL:** Timer1 Gate Current State bit

Indicates the current state of the Timer1 gate that could be provided to TMR1H:TMR1L.
Unaffected by Timer1 Gate Enable (TMR1GE).

bit 1-0 **T1GSS<1:0>:** Timer1 Gate Source Select bits

11 = UV Comparator Output
10 = Timer1 gate pin T1G2
01 = Timer0 overflow output
00 = Timer1 gate pin T1G1

NOTES:

23.0 TIMER2 MODULE

The Timer2 module is an 8-bit timer with the following features:

- 8-bit timer register (TMR2)
- 8-bit period register (PR2)
- Interrupt on TMR2 match with PR2
- Software programmable prescaler (1:1, 1:4, 1:16)

See Figure 23-1 for a block diagram of Timer2.

23.1 Timer2 Operation

The clock input to the Timer2 module is the system clock (F_{OSC}). The clock is fed into the Timer2 prescaler, which has prescale options of 1:1, 1:4 or 1:16. The output of the prescaler is then used to increment the TMR2 register.

The values of TMR2 and PR2 are constantly compared to determine when they match. TMR2 will increment from 00h until it matches the value in PR2. When a match occurs, TMR2 is reset to 00h on the next increment cycle.

The match output of the Timer2/PR2 comparator is used to set the TMR2IF interrupt flag bit in the PIR1 register.

The TMR2 and PR2 registers are both fully readable and writable. On any Reset, the TMR2 register is set to 00h and the PR2 register is set to FFh.

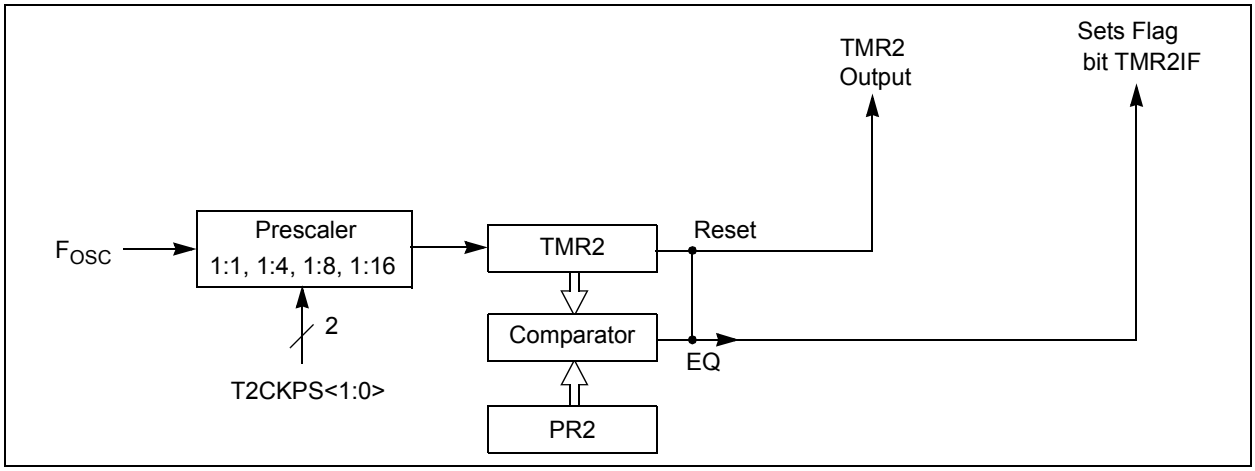
Timer2 is turned on by setting the TMR2ON bit in the T2CON register to a '1'. Timer2 is turned off by clearing the TMR2ON bit to a '0'.

The Timer2 prescaler is controlled by the T2CKPS bits in the T2CON register. The prescaler counter are cleared when:

- A write to TMR2 occurs.
- A write to T2CON occurs.
- Any device Reset occurs (Power-on Reset, $\overline{MCLR}$ Reset, Watchdog Timer Reset, or Brown-out Reset).

Note:	TMR2 is not cleared when T2CON is written.

FIGURE 23-1: TIMER2 BLOCK DIAGRAM



23.2 Timer2 Control Register

REGISTER 23-1: T2CON: TIMER2 CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	TMR2ON	T2CKPS1	T2CKPS0
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 7-3 **Unimplemented:** Read as '0'

bit 2 **TMR2ON:** Timer2 On bit
- 1 = Timer2 is on
- 0 = Timer2 is off

bit 1-0 **T2CKPS<1:0>:** Timer2 Clock Prescale Select bits
- 00 = Prescaler is 1
- 01 = Prescaler is 4
- 10 = Prescaler is 8
- 11 = Prescaler is 16

TABLE 23-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER2

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	102
PIE1	—	ADIE	BCLIE	SSPIE	—	—	TMR2IE	TMR1IE	103
PIR1	—	ADIF	BCLIF	SSPIF	—	—	TMR2IF	TMR1IF	105
PR2	Timer2 Module Period Register								163*
T2CON	—	—	—	—	—	TMR2ON	T2CKPS1	T2CKPS0	164
TMR2	Holding Register for the 8-bit TMR2 Time Base								163*

Legend: — = unimplemented read as '0'. Shaded cells are not used for Timer2 module.

* Page provides register information.

24.0 DUAL CAPTURE/COMPARE (CCD) MODULE

The Dual Capture/Compare module is a peripheral that allows the user to time and control different events. In Capture mode, the peripheral allows the timing of the duration of an event. The Compare mode allows the user to trigger an external event when a predetermined amount of time has expired. This module is only available in the MCP19123 device.

24.1 Capture Mode

Capture mode makes use of the 16-bit Timer1 resource. When an event occurs on the CCDx pin, the 16-bit CCxRH:CCxRL register pair captures and stores the 16-bit value of the TMR1H:TMR1L register pair, respectively. An event is defined as one of the following and is configured by the CCxM<3:0> bits of the CCDCON register:

- Every falling edge
- Every rising edge
- Every 4th rising edge
- Every 16th rising edge

When a capture is made, the Interrupt Request Flag bit CCDxIF of the PIR2 register is set. The interrupt flag must be cleared in software. If another capture occurs before the value in the CCxRH:CCxRL register pair is read, the old captured value is overwritten by the new captured value.

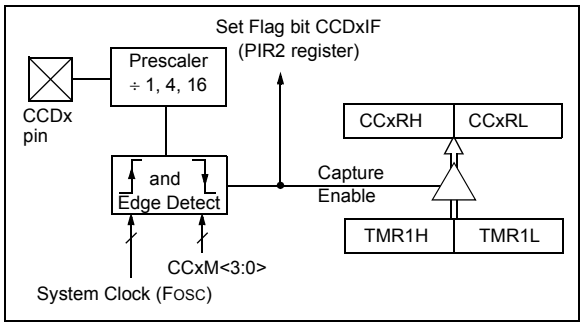
Figure 24-1 shows a simplified diagram of the Capture operation.

24.1.1 CCDX PIN CONFIGURATION

In Capture mode, the CCDx pin should be configured as an input by setting the associated TRIS control bit.

Note:	If the CCDx pin is configured as an output, a write to the port can cause a capture condition.

FIGURE 24-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



24.1.2 SOFTWARE INTERRUPT MODE

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCDxIE interrupt enable bit of the PIE2 register clear to avoid false interrupts. Additionally, the user should clear the CCDxIF interrupt flag bit of the PIR2 register following any change in Operating mode.

Note:	Clocking Timer1 from the system clock (FOSC) should not be used in Capture mode. In order for Capture mode to recognize the trigger event on the CCDx pin, Timer1 must be clocked from the instruction clock (FOSC/4).

24.1.3 CCP1 PRESCALER

There are four prescaler settings specified by the CCxM<3:0> bits of the CCDCON register. Whenever the CCDx module is turned off, or the CCDx module is not in Capture mode, the prescaler counter is cleared. Any Reset will clear the prescaler counter.

Switching from one capture prescaler to another does not clear the prescaler and may generate a false interrupt. To avoid this unexpected operation, turn the module off by clearing the CCDCON register before changing the prescaler. Example 24-1 demonstrates the code to perform this function.

EXAMPLE 24-1: CHANGING BETWEEN CAPTURE PRESCALERS

```
BANKSEL CCDCON       ;Set Bank bits to point
                     ;to CCDCON
CLRF    CCDCON       ;Turn CCDx module off
MOVLW   NEW_CAPT_PS  ;Load the W reg with
                     ;the new prescaler
                     ;move value and CCDx ON
MOVWF   CCDCON       ;Load CCDCON with this
                     ;value
```

24.2 Compare Mode

Compare mode makes use of the 16-bit Timer1 resource. The 16-bit value of the CCxRH:CCxRL register pair is constantly compared against the 16-bit value of the TMR1H:TMR1L register pair. When a match occurs, one of the following events can occur:

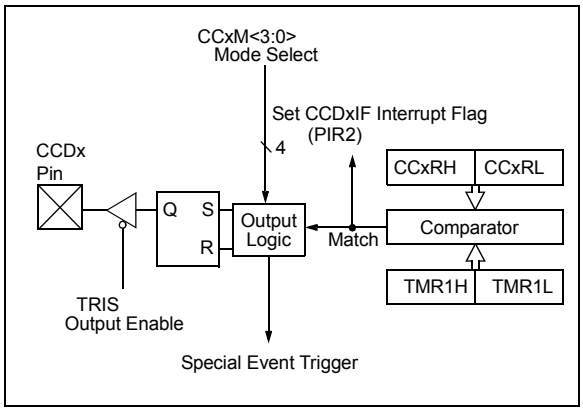
- Toggle the CCDx output
- Set the CCDx output
- Clear the CCDx output
- Generate a Special Event Trigger
- Generate a Software Interrupt

The action on the pin is based on the value of the CCXM<3:0> control bits of the CCDCON register. At the same time, the interrupt flag CCDxIF bit is set.

All Compare modes can generate an interrupt.

Figure 24-2 shows a simplified diagram of the Compare operation.

FIGURE 24-2: COMPARE MODE OPERATION BLOCK DIAGRAM



24.2.1 CCDX PIN CONFIGURATION

The user must configure the CCDx pin as an output by clearing the associated TRIS bit.

Note:	Clearing the CCDCON register will force the CCDx compare output latch to the default low level. This is not the PORT I/O data latch.

24.2.2 SOFTWARE INTERRUPT MODE

When Generate Software Interrupt mode is chosen (CCxM<3:0> = `1010`), the CCD module does not assert control of the CCDx pin (see the CCP1CON register).

24.2.3 SPECIAL EVENT TRIGGER

When Special Event Trigger mode is chosen (CCxM<3:0> = `1011`), the CCD module does the following:

- Starts an ADC conversion if ADC is enabled

The CCD module does not assert control of the CCDx pin in this mode.

The Special Event Trigger output of the CCDx occurs immediately upon a match between the TMR1H, TMR1L register pair and the CCxRH, CCxRL register pair. The TMR1H, TMR1L register pair is not reset until the next rising edge of the Timer1 clock. The Special Event Trigger output starts an A/D conversion (if the A/D module is enabled). This allows the CCxRH, CCxRL register pair to effectively provide a 16-bit programmable period register for Timer1.

TABLE 24-1: SPECIAL EVENT TRIGGER

Device	CCD1/CCD2
MCP19123	CCD1/CCD2

Refer to A/D Section for more information.

Note 1:	The Special Event Trigger from the CCD module does not set interrupt flag bit TMR1IF of the PIR1 register.

24.2.4 COMPARE DURING SLEEP

The Compare mode is dependent upon the system clock (FOSC) for proper operation. Since FOSC is shut down during Sleep mode, the Compare mode will not function properly during Sleep.

24.3 CCP Control Registers

REGISTER 24-1: CCDCON: DUAL CAPTURE/COMPARE CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CC2M3	CC2M2	CC2M1	CC2M0	CC1M3	CC1M2	CC1M1	CC1M0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Reset
'1' = Bit is set	'0' = Bit is cleared	

bit 7-4 **CC2M<3:0>:** Capture/Compare Register Set 2 Mode Select bits
- 00xx = Capture/Compare off (resets module)
- 0100 = Capture mode: every falling edge
- 0101 = Capture mode: every rising edge
- 0110 = Capture mode: every 4th rising edge
- 0111 = Capture mode: every 16th rising edge
- 1000 = Compare mode: set output on match (CC2IF bit is set)
- 1001 = Compare mode: clear output on match (CC2IF bit is set)
- 1010 = Compare mode: toggle output on match (CC2IF bit is set)
- 1011 = Reserved
- 11xx = Compare mode: generate software interrupt on match (CC2IF bit is set, CMP2 pin is unaffected and configured as an I/O port)
- 1111 = Compare mode: trigger special event (CC2IF bit is set; CCD2 does not rest TMR1 and starts an A/D conversion, if the A/D module is enabled. CMP2 pin is unaffected and configured as an I/O port).

bit 3-0 **CC1M<3:0>:** Capture/Compare Register Set 1 Mode Select bits
- 00xx = Capture/Compare off (resets module)
- 0100 = Capture mode: every falling edge
- 0101 = Capture mode: every rising edge
- 0110 = Capture mode: every 4th rising edge
- 0111 = Capture mode: every 16th rising edge
- 1000 = Compare mode: set output on match (CC1IF bit is set)
- 1001 = Compare mode: clear output on match (CC1IF bit is set)
- 1010 = Compare mode: toggle output on match (CC1IF bit is set)
- 1011 = Reserved
- 11xx = Compare mode: generate software interrupt on match (CC1IF bit is set, CMP1 pin is unaffected and configured as an I/O port)
- 1111 = Compare mode: trigger special event (CC1IF bit is set; CCD1 does not rest TMR1 and starts an A/D conversion, if the A/D module is enabled. CMP1 pin is unaffected and configured as an I/O port).

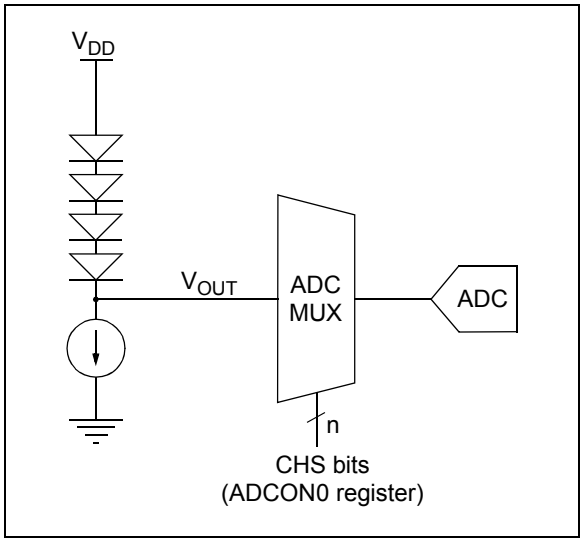
Note: When a Compare interrupt is set, the TMR1 is not rest. This is different than standard Microchip microcontroller operation.

NOTES:

25.0 INTERNAL TEMPERATURE INDICATOR MODULE

The MCP19122/3 is equipped with a temperature circuit designed to measure the operating temperature of the silicon die. The circuit's range of the operating temperature falls between –40°C and +125°C. The output is a voltage that is proportional to the device temperature. The output of the temperature indicator is internally connected to the device ADC.

FIGURE 25-1: TEMPERATURE CIRCUIT DIAGRAM



25.1 Temperature Output

The output of the circuit is measured using the internal analog-to-digital converter. Channel 10 is reserved for the temperature circuit output. Refer to **Section 19.0 "Analog-to-Digital Converter (ADC) Module"** for detailed information.

The temperature of the silicon die can be calculated by the ADC measurement by using Equation 25-1.

EQUATION 25-1: SILICON DIE TEMPERATURE

$$TEMP\_DIE = \frac{ADC\ READING - 305}{3.08mV/^{\circ}C}$$

The 10-bit ADC value located at memory address 2089h can be used to obtain a more accurate reading of the silicon die. This factory stored ADC value is obtained by measuring the silicon die temperature with an ambient temperature of 25C (+/-5C). Equation 25-2 shows how to use this stored value.

EQUATION 25-2: USING CALWD 10 TO OBTAIN SILICON DIE TEMPERATURE

$$TEMP\_DIE(^{\circ}C) = \frac{(ADC\_READING(counts) - ADC25\ ^{\circ}C\_READING\ (counts)}{3(counts/^{\circ}C)} + 25\ ^{\circ}C$$

26.0 ENHANCED PWM MODULE

The PWM module implemented on the MCP19122/3 is a modified version of the Capture/Compare/PWM (CCP) module found in standard mid-range microcontrollers. The module only features the PWM module, which is slightly modified from standard mid-range microcontrollers. In the MCP19122/3, the PWM module is used to generate the system clock or system oscillator. This system clock will control the MCP19122/3 switching frequency, as well as set the maximum allowable duty cycle. The PWM module does not continuously adjust the duty cycle to control the output voltage. This is accomplished by the analog control loop and associated circuitry.

26.1 Standard Pulse-Width Modulation (PWM) Mode

The PWM module output signal is used to set the operating switching frequency and maximum allowable duty cycle of the MCP19122/3. The actual duty cycle on the HDRV and LDRV is controlled by the analog PWM control loop. However, this duty cycle cannot be greater than the value in the PWMRL register.

There are two modes of operation that concern the system clock PWM signal. These modes are stand-alone (non-frequency synchronization) and frequency synchronization.

26.1.1 STAND-ALONE (NON-FREQUENCY SYNCHRONIZATION) MODE

When the MCP19122/3 is running stand-alone, the PWM signal functions as the system clock. It is operating at the programmed switching frequency with a programmed maximum duty cycle (D_{CLOCK}). The programmed maximum duty cycle is not adjusted on a cycle-by-cycle basis to control the MCP19122/3 system output. The required duty cycle (D_{BUCK}) to control the output is adjusted by the MCP19122/3 analog control loop and associated circuitry. D_{CLOCK} does, however, set the maximum allowable D_{BUCK}.

EQUATION 26-1:

$$D_{BUCK} < 1 - D_{CLOCK}$$

26.1.2 SWITCHING FREQUENCY SYNCHRONIZATION MODE

The MCP19122/3 can be programmed to be a switching frequency MASTER or SLAVE device. The MASTER device functions as described in **Section 26.1.1 "Stand-Alone (Non-Frequency Synchronization) Mode"** with the exception of the 8 MHz system clock also being applied to GPB3 and the synchronization signal applied to GPA1.

A SLAVE device will receive the MASTER 8 MHz system clock on GPB3 and synchronization signal on GPA1. The synchronization signal will be ORed with the output of the TIMER2 module. This ORed signal will latch PWMRL into PWMRH and PWMPHL into PWMPHH.

Figure 26-1 shows a simplified block diagram of the CCP module in PWM mode.

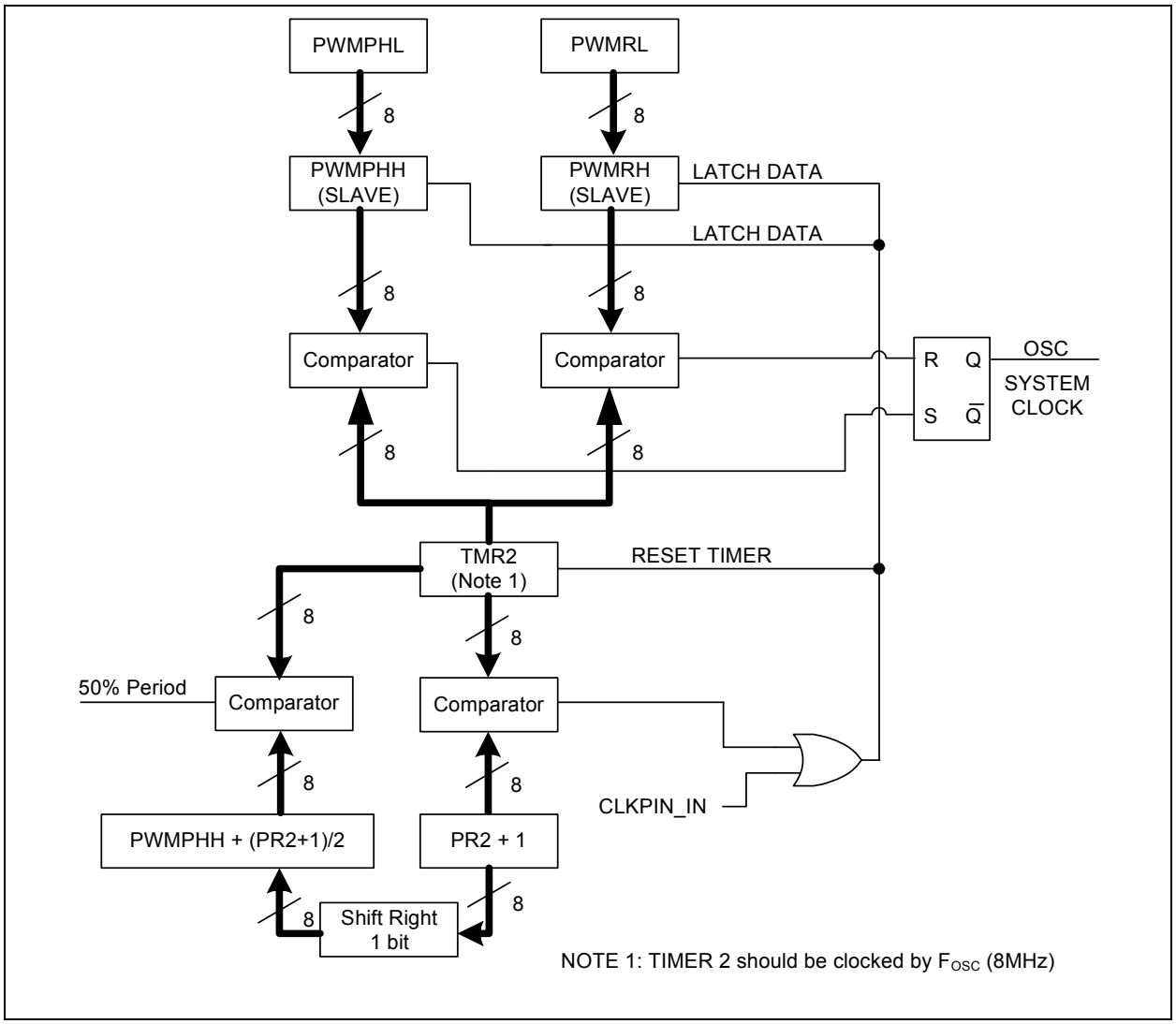
The PWMPHL register allows for a phase shift to be added to the SLAVE system clock.

It is desired to have the MCP19122/3 SLAVE devices system clock start point shifted by a programmed amount from the MASTER system clock. This SLAVE phase shift is specified by writing to the PWMPHL register. The SLAVE phase shift can be calculated by using the following equation.

EQUATION 26-2:

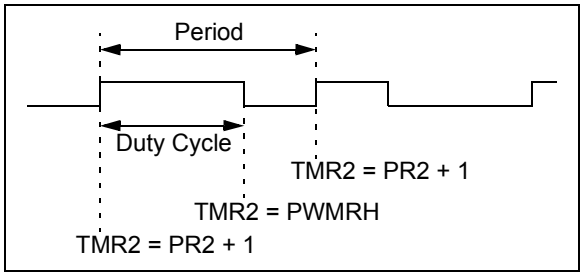
$$SLAVE\ PHASE\ SHIFT = PWMPHL \cdot T_{OSC} \cdot (T2\ PRESCALE\ VALUE)$$

FIGURE 26-1: SIMPLIFIED PWM BLOCK DIAGRAM



A PWM output (Figure 26-2) has a time base (period) and a time that the output stays high (duty cycle). The frequency of the PWM is the inverse of the period (1/period).

FIGURE 26-2: PWM OUTPUT



26.1.3 PWM PERIOD

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following equation:

EQUATION 26-3:

$$PWM\ PERIOD = [(PR2)+1] \times T_{OSC} \times (T2\ PRESCALE\ VALUE)$$

When TMR2 is equal to PR2, the following two events occur on the next increment cycle:

- TMR2 is cleared
- The PWM duty cycle is latched from PWMRL into PWMRH

26.1.4 PWM DUTY CYCLE (D_{CLOCK})

The PWM duty cycle (D_{CLOCK}) is specified by writing to the PWMRL register. Up to 8-bit resolution is available. The following equation is used to calculate the PWM duty cycle (D_{CLOCK}):

EQUATION 26-4:

$$PWM\ DUTY\ CYCLE = PWMRL \times T_{OSC} \times (T2\ PRESCALE\ VALUE)$$

The PWMRL bits can be written to at any time, but the duty cycle value is not latched into PWMRH until after a match between PR2 and TMR2 occurs.

26.2 Operation during Sleep

When the device is placed in Sleep, the allocated timer will not increment and the state of the module will not change. If the CLKPIN pin is driving a value, it will continue to drive that value. When the device wakes up, it will continue from this state.

TABLE 26-1: SUMMARY OF REGISTERS ASSOCIATED WITH PWM MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
T2CON	—	—	—	—	—	TMR2ON	T2CKPS1	T2CKPS0	164
PR2	Timer2 Module Period Register								172*
PWMRL	PWM Register Low Byte								171*
PWMPHL	SLAVE Phase Shift Byte								171*

Legend: — = Unimplemented locations, read as '0'. Shaded cells are not used by Capture mode.

* Page provides register information.

NOTES:

27.0 MASTER SYNCHRONOUS SERIAL PORT (MSSP) MODULE

27.1 MSSP Module Overview

The Master Synchronous Serial Port (MSSP) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be Serial EEPROMs, shift registers, display drivers, A/D converters, etc. The MSSP module in the MCP19122/3 only operates in Inter-Integrated Circuit (I^2C) mode.

The I^2C interface supports the following modes and features:

- Master mode
- Slave mode
- Byte NACKing (Slave mode)
- Limited Multi-Master support
- 7-bit and 10-bit addressing
- Start and Stop interrupts
- Interrupt masking
- Clock stretching
- Bus collision detection
- General call address matching
- Dual Address masking
- Address Hold and Data Hold modes
- Selectable SDA hold times

Figure 27-1 is a block diagram of the I^2C interface module in Master mode. Figure 27-2 is a diagram of the I^2C interface module in Slave mode.

FIGURE 27-1: MSSP BLOCK DIAGRAM (I^2C MASTER MODE)

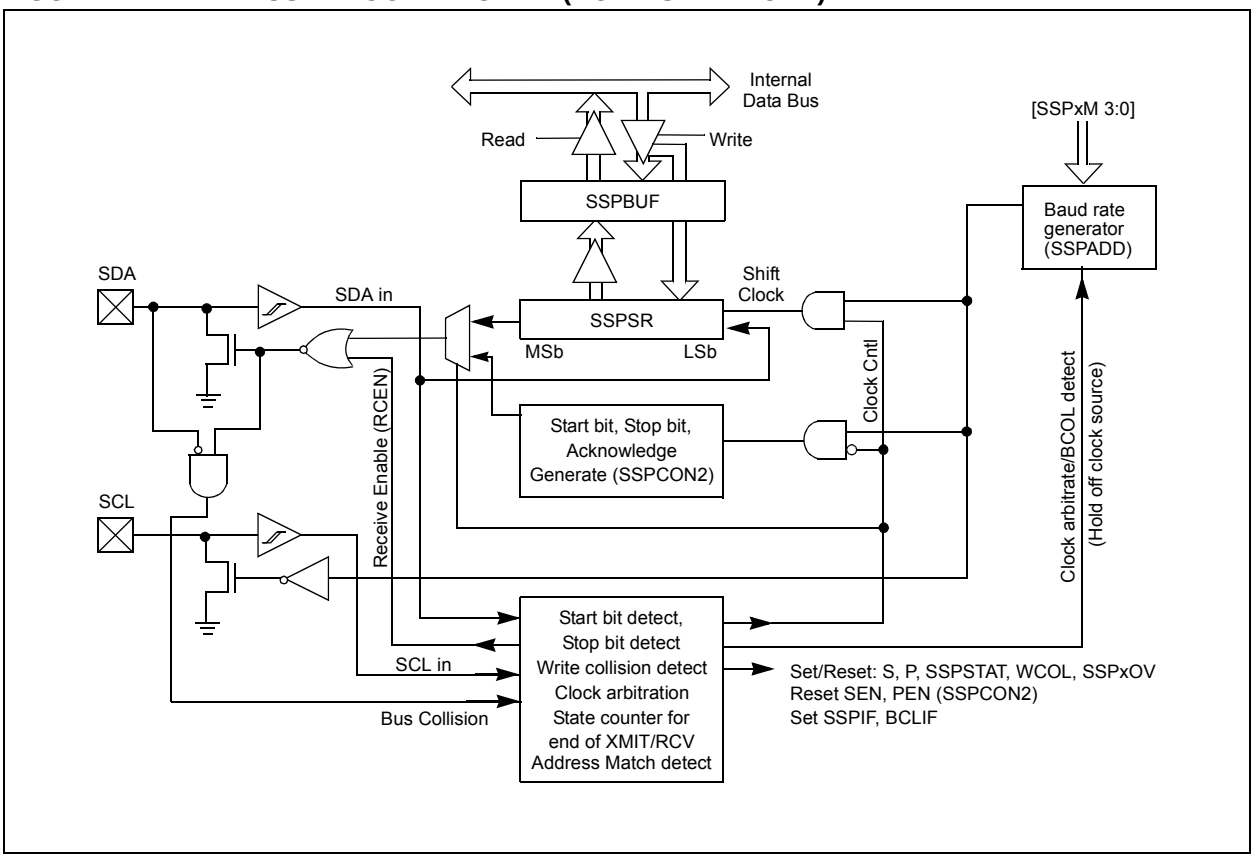
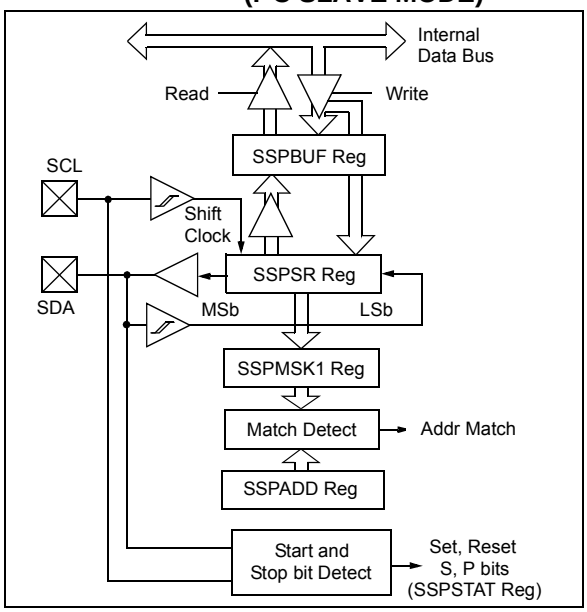


FIGURE 27-2: MSSP BLOCK DIAGRAM (I^2C SLAVE MODE)



27.2 I^2C MODE OVERVIEW

The Inter-Integrated Circuit Bus (I^2C) is a multi-master serial data communication bus. Devices communicate in a master/slave environment, where the master devices initiate the communication. A Slave device is controlled through addressing.

The I^2C bus specifies two signal connections:

- Serial Clock (SCL)
- Serial Data (SDA)

Both the SCL and SDA connections are bidirectional open-drain lines, each requiring pull-up resistors for the supply voltage. Pulling the line to ground is considered a logical zero; letting the line float is considered a logical one.

Figure 27-3 shows a typical connection between two devices configured as master and slave.

The I^2C bus can operate with one or more master devices and one or more slave devices.

There are four potential modes of operation for a given device:

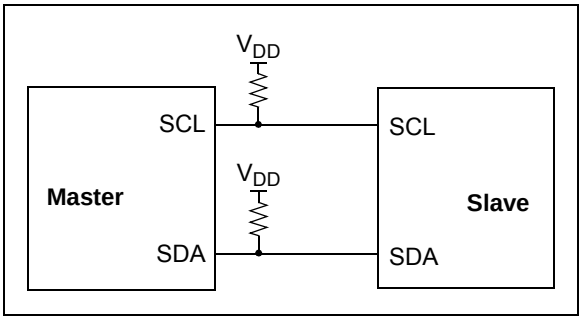
- Master Transmit mode
  (master is transmitting data to a slave)
- Master Receive mode
  (master is receiving data from a slave)
- Slave Transmit mode
  (slave is transmitting data to a master)
- Slave Receive mode
  (slave is receiving data from a master)

To begin communication, a master device starts out in Master Transmit mode. The master device sends out a Start bit followed by the address byte of the slave it intends to communicate with. This is followed by a single Read/Write bit, which determines whether the master intends to transmit to or receive data from the slave device.

If the requested slave exists on the bus, it will respond with an Acknowledge bit, otherwise known as an $\overline{ACK}$. The master then continues in either Transmit mode or Receive mode and the slave continues in the complement, either in Receive mode or Transmit mode, respectively.

A Start bit is indicated by a high-to-low transition of the SDA line while the SCL line is held high. Address and data bytes are sent out Most Significant bit (MSb) first. The Read/Write bit is sent out as a logical one when the master intends to read data from the slave, and is sent out as a logical zero when it intends to write data to the slave.

FIGURE 27-3: I^2C MASTER/ SLAVE CONNECTION



The Acknowledge bit ($\overline{ACK}$) is an active-low signal that holds the SDA line low to indicate to the transmitter that the slave device has received the transmitted data and is ready to receive more.

The transition of a data bit is always performed while the SCL line is held low. Transitions that occur while the SCL line is held high are used to indicate Start and Stop bits.

If the master intends to write to the slave, it repeatedly sends out a byte of data, with the slave responding after each byte with an $\overline{ACK}$ bit. In this example, the master device is in Master Transmit mode and the slave is in Slave Receive mode.

If the master intends to read from the slave, it repeatedly receives a byte of data from the slave and responds after each byte with an $\overline{ACK}$ bit. In this example, the master device is in Master Receive mode and the slave is Slave Transmit mode.

On the last byte of data communicated, the master device may end the transmission by sending a Stop bit. If the master device is in Receive mode, it sends the Stop bit in place of the last $\overline{ACK}$ bit. A Stop bit is indicated by a low-to-high transition of the SDA line while the SCL line is held high.

In some cases, the master may want to maintain control of the bus and re-initiate another transmission. If so, the master device may send another Start bit in place of the Stop bit or last $\overline{ACK}$ bit when it is in Receive mode.

The I^2C bus specifies three message protocols:

- Single message where a master writes data to a slave
- Single message where a master reads data from a slave
- Combined message where a master initiates a minimum of two writes, or two reads, or a combination of writes and reads, to one or more slaves

When one device is transmitting a logical one, or letting the line float, and a second device is transmitting a logical zero, or holding the line low, the first device can detect that the line is not a logical one. This detection, when used on the SCL line, is called clock stretching. Clock stretching gives slave devices a mechanism to control the flow of data. When this detection is used on the SDA line, it is called arbitration. Arbitration ensures that there is only one master device communicating at any single time.

27.2.1 CLOCK STRETCHING

When a slave device has not completed processing data, it can delay the transfer of more data through the process of Clock Stretching. An addressed slave device may hold the SCL clock line low after receiving or sending a bit, indicating that it is not yet ready to continue. The master that is communicating with the slave will attempt to raise the SCL line in order to transfer the next bit, but will detect that the clock line has not yet been released. Because the SCL connection is open-drain, the slave has the ability to hold that line low until it is ready to continue communicating.

Clock stretching allows receivers that cannot keep up with a transmitter to control the flow of incoming data.

27.2.2 ARBITRATION

Each master device must monitor the bus for Start and Stop bits. If the device detects that the bus is busy, it cannot begin a new message until the bus returns to an idle state.

However, two master devices may try to initiate a transmission at or about the same time. When this occurs, the process of arbitration begins. Each transmitter checks the level of the SDA data line and compares it to the level that it expects to find. The first transmitter to observe that the two levels don't match loses arbitration and must stop transmitting on the SDA line.

For example, if one transmitter holds the SDA line to a logical one (lets it float) and a second transmitter holds it to a logical zero (pulls it low), the result is that the SDA line will be low. The first transmitter then observes that the level of the line is different than expected and concludes that another transmitter is communicating.

The first transmitter to notice this difference is the one that loses arbitration and must stop driving the SDA line. If this transmitter is also a master device, it must also stop driving the SCL line. It then can monitor the lines for a Stop condition before trying to reissue its transmission. In the meantime, the other device that has not noticed any difference between the expected and actual levels on the SDA line continues with its original transmission. It can do so without any complications, because so far the transmission appears exactly as expected, with no other transmitter disturbing the message.

Slave Transmit mode can also be arbitrated, when a master addresses multiple slaves, but this is less common.

If two master devices are sending a message to two different slave devices at the address stage, the master sending the lower slave address always wins arbitration. When two master devices send messages to the same slave address, and addresses can sometimes refer to multiple slaves, the arbitration process must continue into the data stage.

Arbitration usually occurs very rarely, but it is a necessary process for proper multi-master support.

27.3 I²C MODE OPERATION

All MSSP I²C communication is byte-oriented and shifted out MSb first. Six SFR registers and two interrupt flags interface the module with the PIC microcontroller and with the user's software. Two pins, SDA and SCL, are exercised by the module to communicate with other external I²C devices.

27.3.1 BYTE FORMAT

All communication in I²C is done in 9-bit segments. A byte is sent from a Master to a Slave or vice versa, followed by an Acknowledge bit sent back. After the 8th falling edge of the SCL line, the device outputting data on the SDA changes that pin to an input and reads in an acknowledge value on the next clock pulse.

The clock signal, SCL, is provided by the master. Data is valid to change while the SCL signal is low, and sampled on the rising edge of the clock. Changes on the SDA line while the SCL line is high define special conditions on the bus, explained in the following sections.

27.3.2 DEFINITION OF I²C TERMINOLOGY

There is language and terminology in the description of I²C communication that have definitions specific to I²C. Such word usage is defined in Table 27-1 and may be used in the rest of this document without explanation. The information in this table was adapted from the Philips I²C specification.

27.3.3 SDA AND SCL PINS

Selecting any I²C mode with the SSPEN bit set forces the SCL and SDA pins to be open-drain. These pins should be set by the user to inputs by setting the appropriate TRIS bits.

Note: Data is tied to output zero when an I²C mode is enabled.

27.3.4 SDA HOLD TIME

The hold time of the SDA pin is selected by the SDAHT bit in the SSPCON3 register. Hold time is the time SDA is held valid after the falling edge of SCL. Setting the SDAHT bit selects a longer 300 ns minimum hold time and may help on buses with large capacitance.

TABLE 27-1: I²C BUS TERMS

Term	Description
Transmitter	The device that shifts data out onto the bus
Receiver	The device that shifts data in from the bus
Master	The device that initiates a transfer, generates clock signals and terminates a transfer
Slave	The device addressed by the master
Multi-Master	A bus with more than one device that can initiate data transfers
Arbitration	Procedure to ensure that only one master at a time controls the bus. Winning arbitration ensures that the message is not corrupted.
Synchronization	Procedure to synchronize the clocks of two or more devices on the bus
Idle	No master is controlling the bus and both SDA and SCL lines are high
Active	Any time one or more master devices are controlling the bus
Addressed Slave	Slave device that has received a matching address and is actively being clocked by a master
Matching Address	Address byte that is clocked into a slave that matches the value stored in SSPADDx
Write Request	Slave receives a matching address with R/$\overline{W}$ bit clear and is ready to clock in data
Read Request	Master sends an address byte with the R/$\overline{W}$ bit set, indicating that it wishes to clock data out of the Slave. This data is the next and all following bytes until a Restart or Stop.
Clock Stretching	When a device on the bus holds SCL low to stall communication
Bus Collision	Any time the SDA line is sampled low by the module while it is outputting and expected high state

27.3.5 START CONDITION

The I^2C specification defines a Start condition as a transition of SDA from a high state to a low state, while the SCL line is high. A Start condition is always generated by the master and signifies the transition of the bus from an Idle to an Active state. Figure 27-4 shows the wave forms for Start and Stop conditions.

A bus collision can occur on a Start condition if the module samples the SDA line low before asserting it low. This does not conform to the I^2C Specification that states no bus collision can occur on a Start.

27.3.6 STOP CONDITION

A Stop condition is a transition of the SDA line from low-to-high state while the SCL line is high.

Note: At least one SCL low time must appear before a Stop is valid. Therefore, if the SDA line goes low then high again while the SCL line stays high, only the Start condition is detected.

27.3.7 RESTART CONDITION

A Restart is valid any time that a Stop is valid. A master can issue a Restart if it wishes to hold the bus after terminating the current transfer. A Restart has the same effect on the slave that a Start would, resetting all slave logic and preparing it to clock in an address. The master may want to address the same or another slave.

In 10-bit Addressing Slave mode, a Restart is required for the master to clock data out of the addressed slave. Once a slave has been fully addressed, matching both high and low address bytes, the master can issue a Restart and the high address byte with the R/$\overline{W}$ bit set. The slave logic will then hold the clock and prepare to clock out data.

After a full match with R/$\overline{W}$ clear in 10-bit mode, a prior match flag is set and maintained. Until a Stop condition, a high address with R/$\overline{W}$ clear or a high address match fails.

27.3.8 START/STOP CONDITION INTERRUPT MASKING

The SCIE and PCIE bits in the SSPCON3 register can enable the generation of an interrupt in Slave modes that do not typically support this function. These bits will have no effect on slave modes where interrupt on Start and Stop detect are already enabled.

FIGURE 27-4: I^2C START AND STOP CONDITIONS

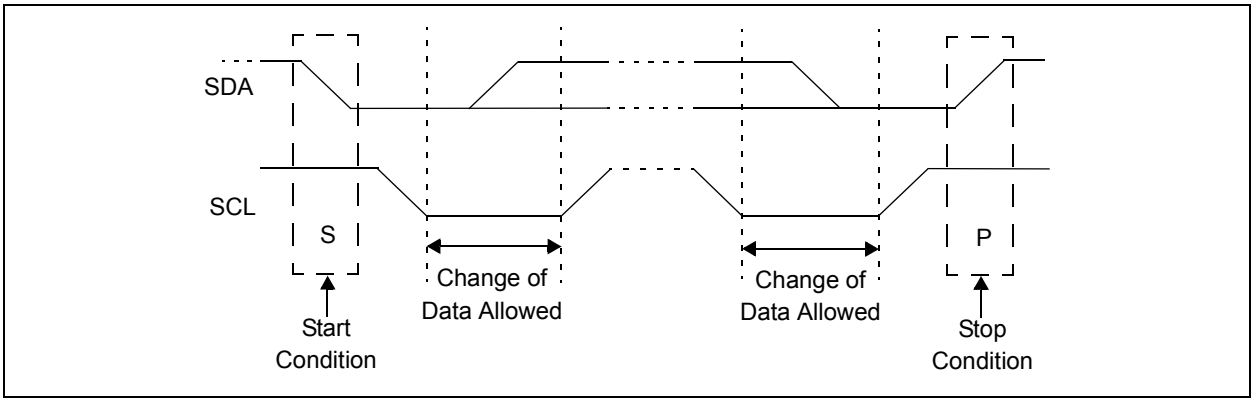
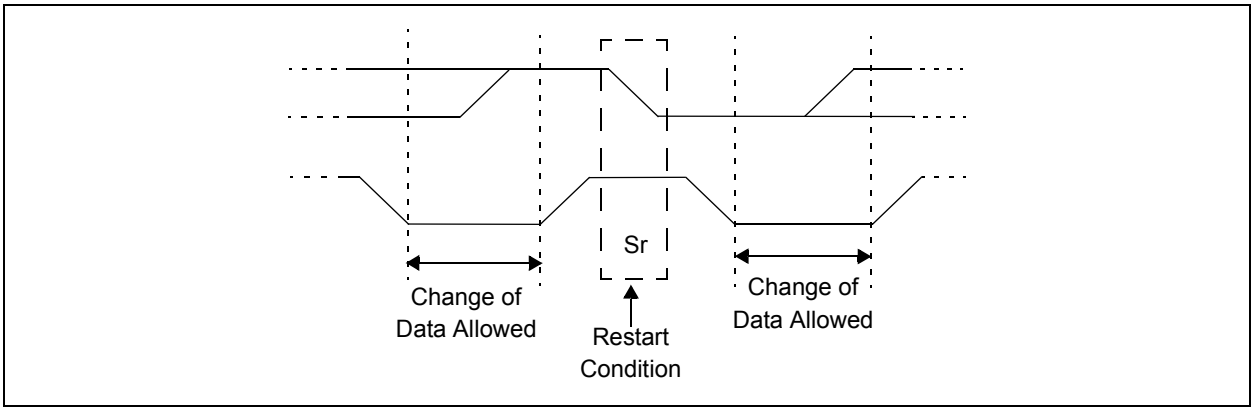


FIGURE 27-5: I^2C RESTART CONDITION



27.3.9 ACKNOWLEDGE SEQUENCE

The 9th SCL pulse for any transferred byte in I2C is dedicated as an Acknowledge. It allows receiving devices to respond back to the transmitter by pulling the SDA line low. The transmitter must release control of the line during this time to shift in the response. The Acknowledge ($\overline{ACK}$) is an active-low signal, pulling the SDA line low, indicating to the transmitter that the device has received the transmitted data and is ready to receive more.

The result of an $\overline{ACK}$ is placed in the ACKSTAT bit in the SSPCON2 register.

Slave software, when the AHEN and DHEN bits are set, allows the user to set the $\overline{ACK}$ value sent back to the transmitter. The ACKDT bit in the SSPCON2 register is set/cleared to determine the response.

Slave hardware will generate an $\overline{ACK}$ response if the AHEN and DHEN bits in the SSPCON3 register are clear.

There are certain conditions where an $\overline{ACK}$ will not be sent by the slave. If the BF bit in the SSPSTAT register or the SSPOV bit in the SSPCON1 register are set when a byte is received, the $\overline{ACK}$ will not be sent.

When the module is addressed, after the 8th falling edge of SCL on the bus, the ACKTIM bit in the SSPCON3 register is set. The ACKTIM bit indicates the acknowledge time of the active bus. The ACKTIM status bit is only active when the AHEN or DHEN bits are enabled.

27.4 I2C SLAVE MODE OPERATION

The MSSP Slave mode operates in one of the four modes selected in the SSPM bits in SSPCON1 register. The modes can be divided into 7-bit and 10-bit Addressing mode. 10-bit Addressing mode operates the same as 7-bit, with some additional overhead for handling the larger addresses.

Modes with Start and Stop bit interrupts operate the same as the other modes, with SSPIF additionally getting set upon detection of a Start, Restart or Stop condition.

27.4.1 SLAVE MODE ADDRESSES

The SSPADD register contains the Slave mode address. The first byte received after a Start or Restart condition is compared against the value stored in this register. If the byte matches, the value is loaded into the SSPBUF register and an interrupt is generated. If the value does not match, the module goes idle and no indication is given to the software that anything happened.

The SSPMSK1 register affects the address matching process. Refer to **Section 27.4.10 "SSPMSK1 Register"** for more information.

27.4.2 SECOND SLAVE MODE ADDRESS

The SSPADD2 register contains a second 7-bit Slave mode address. The first byte received after a Start or Restart condition is compared against the value stored in this register. If the byte matches, the value is loaded into the SSPBUF register and an interrupt is generated. If the value does not match, the module goes idle and no indication is given to the software that anything happened.

The SSPMSK2 register affects the address matching process. Refer to **Section 27.4.10 "SSPMSK1 Register"** for more information.

27.4.2.1 I2C Slave 7-Bit Addressing Mode

In 7-bit Addressing mode, the LSb of the received data byte is ignored when determining if there is an address match.

27.4.2.2 I2C Slave 10-Bit Addressing Mode

In 10-bit Addressing mode, the first received byte is compared to the binary value of '`1 1 1 1 0 A9 A8 0`'. A9 and A8 are the two MSb of the 10-bit address and are stored in bits 2 and 1 in the SSPADD register.

After the high byte has been acknowledged, the UA bit is set and SCL is held low until the user updates SSPADD with the low address. The low address byte is clocked in, and all 8 bits are compared to the low address value in SSPADD. Even if there is no address match, SSPIF and UA are set and SCL is held low until SSPADD is updated to receive a high byte again. When SSPADD is updated, the UA bit is cleared. This ensures the module is ready to receive the high address byte on the next communication.

A high and low address match as a write request is required at the start of all 10-bit addressing communication. A transmission can be initiated by issuing a Restart once the slave is addressed, and clocking in the high address with the R/$\overline{W}$ bit set. The slave hardware will then acknowledge the read request and prepare to clock out data. This is only valid for a slave after it has received a complete high and low address-byte match.

27.4.3 SLAVE RECEPTION

When the R/$\overline{W}$ bit of a matching received address byte is clear, the R/$\overline{W}$ bit in the SSPSTAT register is cleared. The received address is loaded into the SSPBUF register and acknowledged.

When an overflow condition exists for a received address, then Not Acknowledge is given. An overflow condition is defined as either bit BF in the SSPSTAT register is set, or bit SSPOV in the SSPCON1 register is set. The BOEN bit in the SSPCON3 register modifies this operation. For more information, refer to Register 27-4.

An MSSP interrupt is generated for each transferred data byte. Flag bit SSPIF must be cleared by software.

When the SEN bit in the SSPCON2 register is set, SCL will be held low (clock stretch) following each received byte. The clock must be released by setting the CKP bit in the SSPCON1 register, except sometimes in 10-bit mode.

27.4.3.1 7-Bit Addressing Reception

This section describes a standard sequence of events for the MSSP module configured as an I^2C Slave in 7-bit Addressing mode, including all decisions made by hardware or software and their effect on reception. Figures 27-5 and 27-6 are used as a visual reference for this description.

This is a step-by-step process of what typically must be done to accomplish I^2C communication.

1. Start bit detected.
2. S bit in the SSPSTAT register is set; SSPIF is set if interrupt on Start detect is enabled.
3. Matching address with R/$\overline{W}$ bit clear is received.
4. The slave pulls SDA low, sending an $\overline{ACK}$ to the master, and sets SSPIF bit.
5. Software clears the SSPIF bit.
6. Software reads received address from SSPBUF clearing the BF flag.
7. If SEN = 1, Slave software sets CKP bit to release the SCL line.
8. The master clocks out a data byte.
9. Slave drives SDA low sending an $\overline{ACK}$ to the master, and sets SSPIF bit.
10. Software clears SSPIF.
11. Software reads the received byte from SSPBUF clearing BF.
12. Steps 8–12 are repeated for all received bytes from the Master.
13. Master sends Stop condition, setting P bit in the SSPSTAT register, and the bus goes idle.

27.4.3.2 7-Bit Reception with AHEN and DHEN

Slave device reception with AHEN and DHEN set operates the same as without these options with extra interrupts and clock stretching added after the 8th falling edge of SCL. These additional interrupts allow the slave software to decide whether it wants the $\overline{ACK}$ to receive address or data byte, rather than the hardware.

This list describes the steps that need to be taken by slave software to use these options for I^2C communication. Figure 27-7 displays a module using both address and data holding. Figure 27-8 includes the operation with the SEN bit in the SSPCON2 register set.

1. S bit in the SSPSTAT register is set; SSPIF is set if interrupt on Start detect is enabled.
2. Matching address with R/$\overline{W}$ bit clear is clocked in. SSPIF is set and CKP cleared after the 8th falling edge of SCL.
3. Slave clears the SSPIF.
4. Slave can look at the ACKTIM bit in the SSPCON3 register to determine if the SSPIF was after or before the $\overline{ACK}$.
5. Slave reads the address value from SSPBUF, clearing the BF flag.
6. Slave sets $\overline{ACK}$ value clocked out to the master by setting ACKDT.
7. Slave releases the clock by setting CKP.
8. SSPxIF is set after an $\overline{ACK}$, not after a NACK.
9. If SEN = 1 the slave hardware will stretch the clock after the $\overline{ACK}$.
10. Slave clears SSPIF.

Note: SSPIF is still set after the 9th falling edge of SCL even if there is no clock stretching and BF has been cleared. Only if NACK is sent to Master is SSPIF not set.

11. SSPIF set and CKP cleared after 8th falling edge of SCL for a received data byte.
12. Slave looks at ACKTIM bit in the SSPCON3 register to determine the source of the interrupt.
13. Slave reads the received data from SSPBUF clearing BF.
14. Steps 7-14 are the same for each received data byte.
15. Communication is ended by either the slave sending an $\overline{ACK}$ = 1 or the master sending a Stop condition. If a Stop is sent and Interrupt on Stop Detect is disabled, the slave will only know by polling the P bit in the SSPSTAT register.

FIGURE 27-6: I²C SLAVE, 7-BIT ADDRESS, RECEPTION (SEN = 0, AHEN = 0, DHEN = 0)

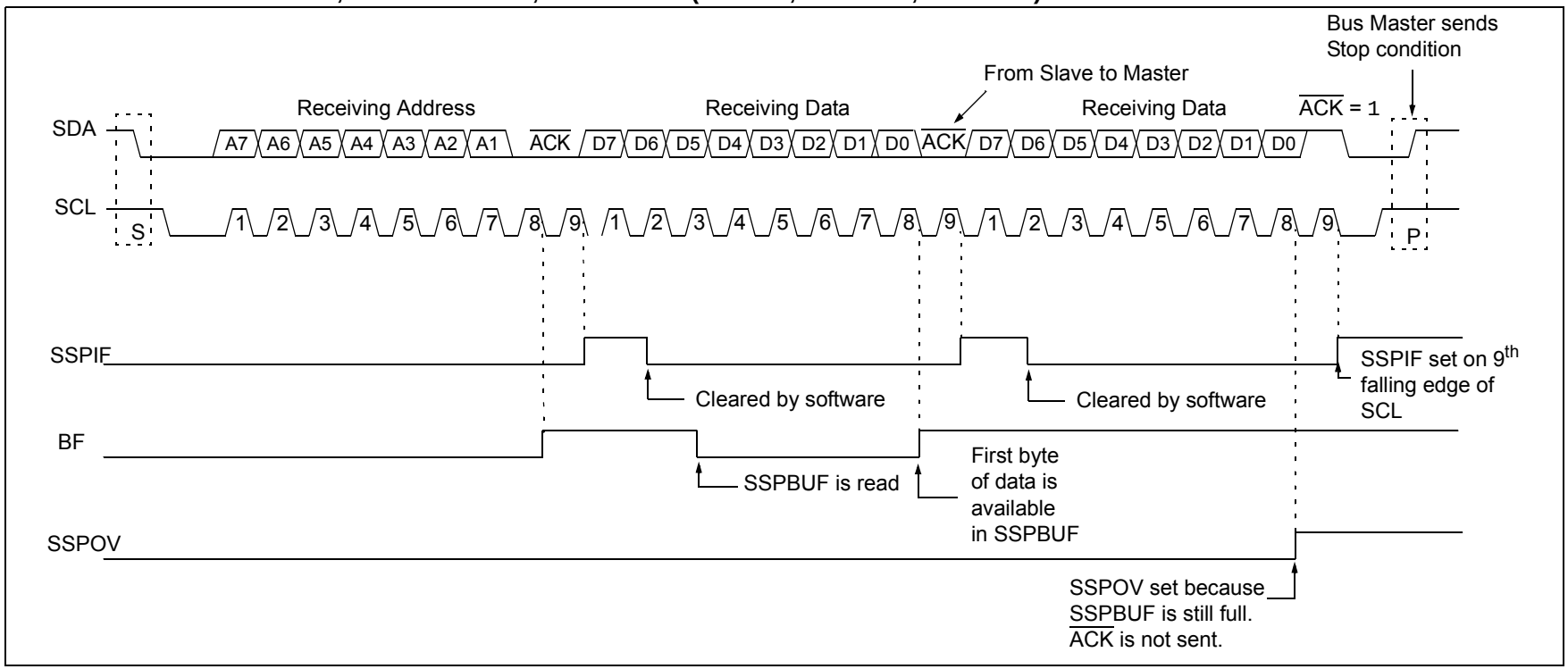


FIGURE 27-7: I²C SLAVE, 7-BIT ADDRESS, RECEPTION (SEN = 1, AHEN = 0, DHEN = 0)

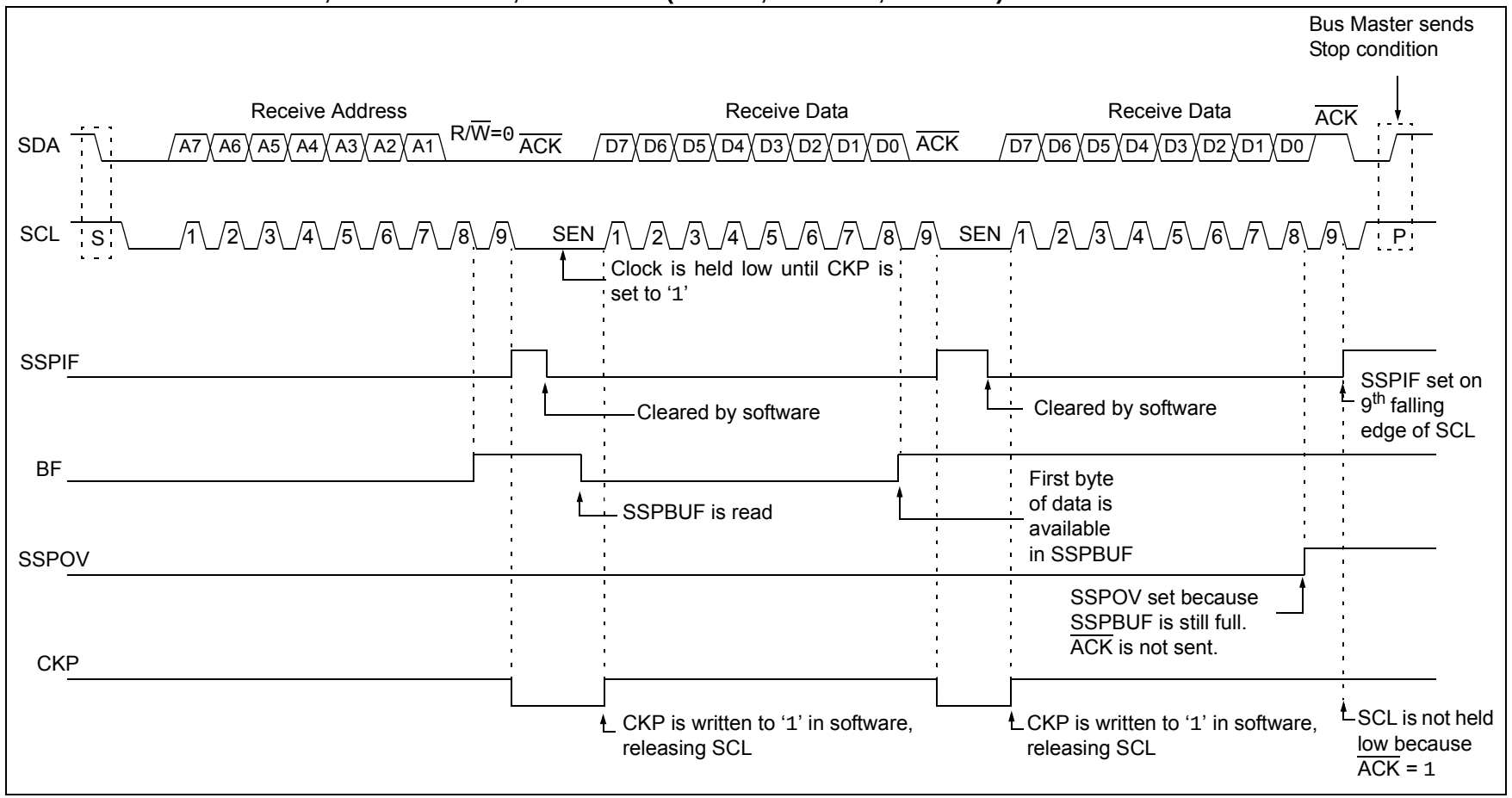


FIGURE 27-8: I²C SLAVE, 7-BIT ADDRESS, RECEPTION (SEN = 0, AHEN = 1, DHEN = 1)

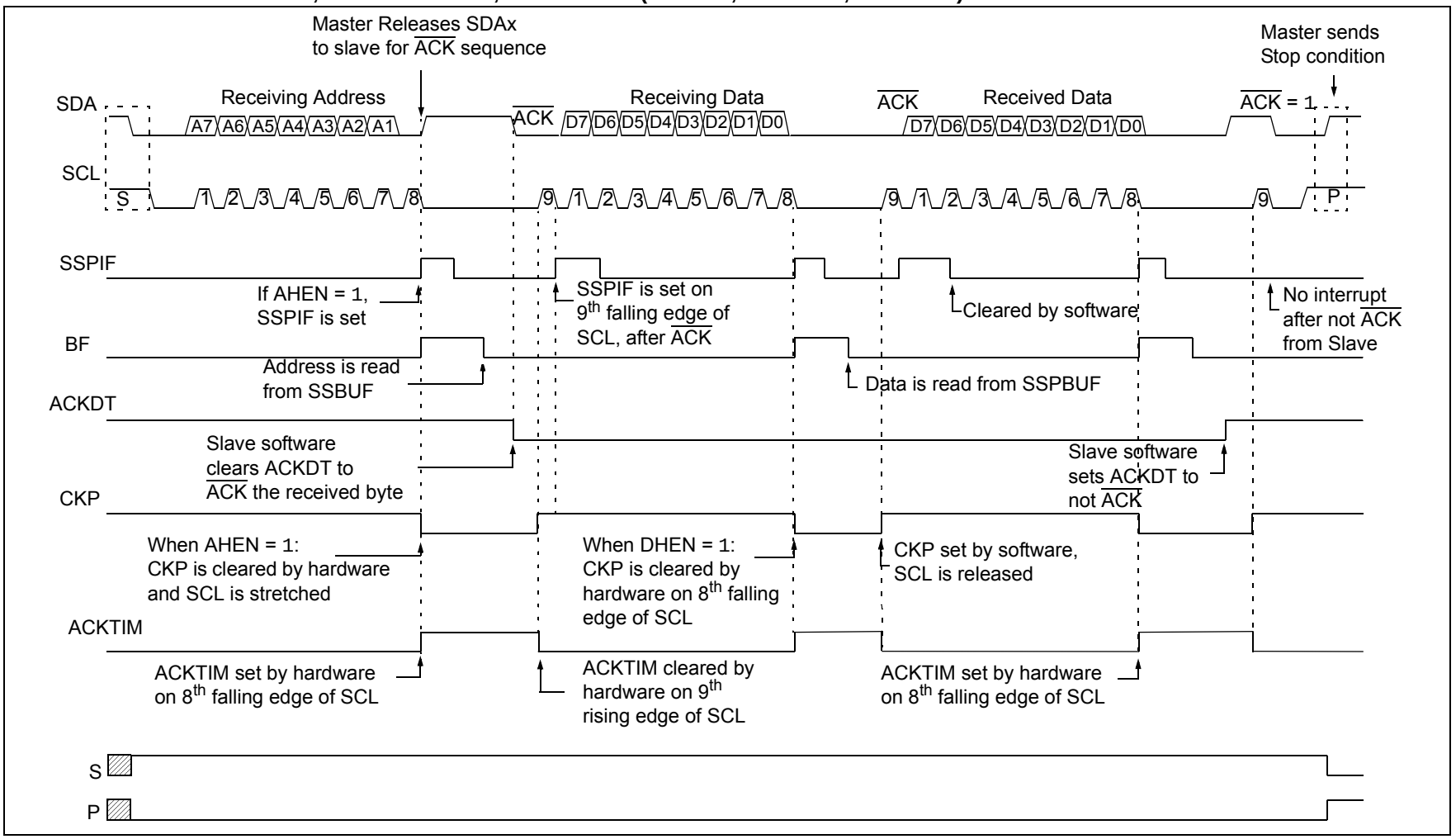
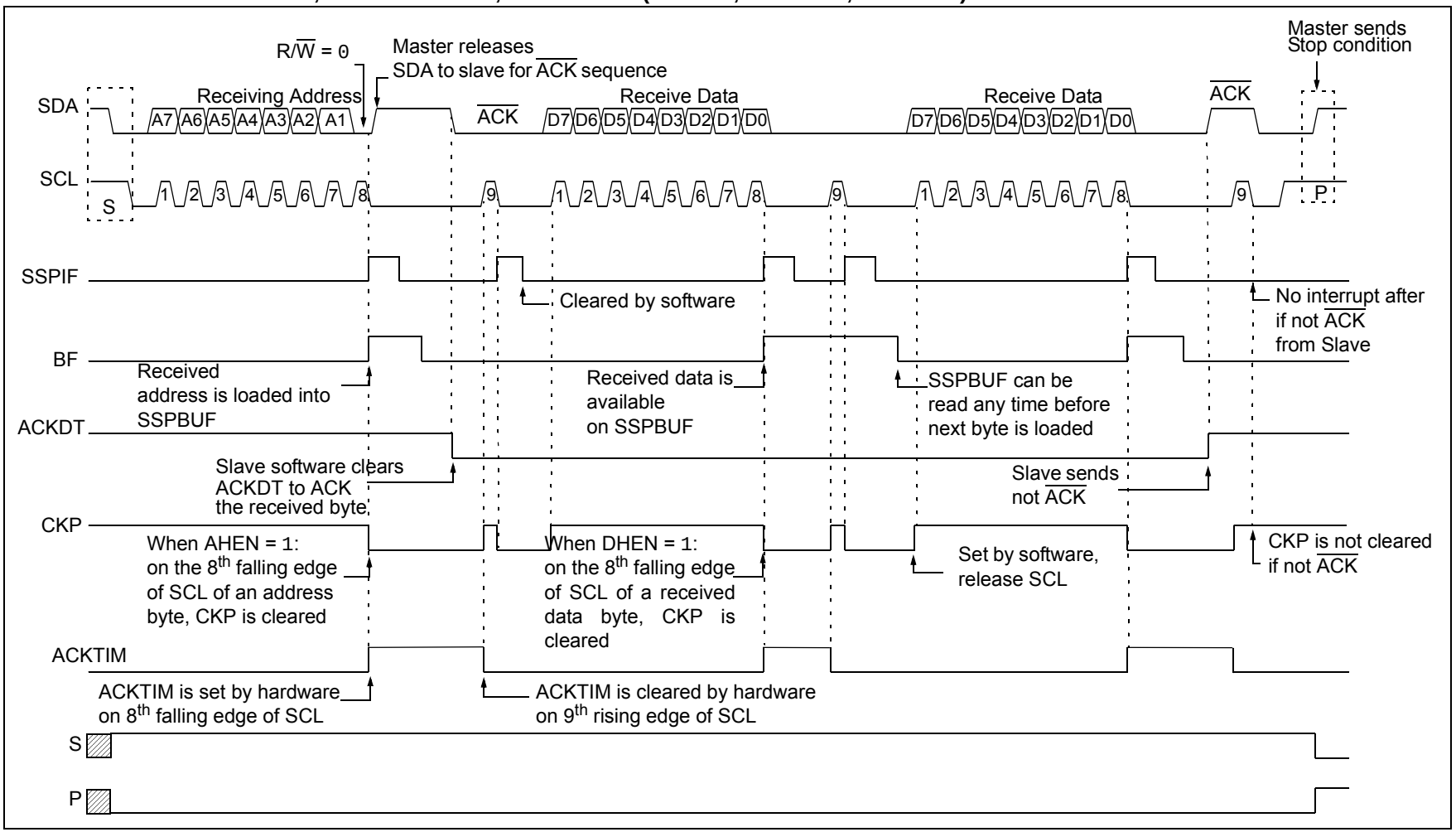


FIGURE 27-9: I²C SLAVE, 7-BIT ADDRESS, RECEPTION (SEN = 1, AHEN = 1, DHEN = 1)



27.4.4 SLAVE TRANSMISSION

When the R/$\overline{W}$ bit of the incoming address byte is set and an address match occurs, the R/$\overline{W}$ bit in the SSPSTAT register is set. The received address is loaded into the SSPBUF register and an $\overline{ACK}$ pulse is sent by the slave on the 9th bit.

Following the $\overline{ACK}$, slave hardware clears the CKP bit and the SCL pin is held low. Refer to **Section 27.4.7 "Clock Stretching"** for more details. By stretching the clock, the master will be unable to assert another clock pulse until the slave is done preparing the transmit data.

The transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then the SCL pin should be released by setting the CKP bit in the SSPCON1 register. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time.

The $\overline{ACK}$ pulse from the master-receiver is latched on the rising edge of the 9th SCL input pulse. This $\overline{ACK}$ value is copied to the ACKSTAT bit in the SSPCON2 register. If ACKSTAT is set (not $\overline{ACK}$), the data transfer is complete. In this case, when the not $\overline{ACK}$ is latched by the slave, the slave goes idle and waits for another occurrence of the Start bit. If the SDA line was low ($\overline{ACK}$), the next transmit data must be loaded into the SSPBUF register. Again, the SCL pin must be released by setting bit CKP.

An MSSP interrupt is generated for each data transfer byte. The SSPIF bit must be cleared by software, and the SSPSTAT register is used to determine the status of the byte. The SSPIF bit is set on the falling edge of the 9th clock pulse.

27.4.4.1 Slave Mode Bus Collision

A slave receives a Read request and begins shifting data out on the SDA line. If a bus collision is detected and the SBCDE bit in the SSPCON3 register is set, the BCLIF bit in the PIR register is set. Once a bus collision is detected, the slave goes idle and waits to be addressed again. The user's software can use the BCLIF bit to handle a slave bus collision.

27.4.4.2 7-Bit Transmission

A master device can transmit a read request to a slave, and then it clocks data out of the slave. The list below outlines what software for a slave will need to do to accomplish a standard transmission. Figure 27-10 can be used as a reference to this list.

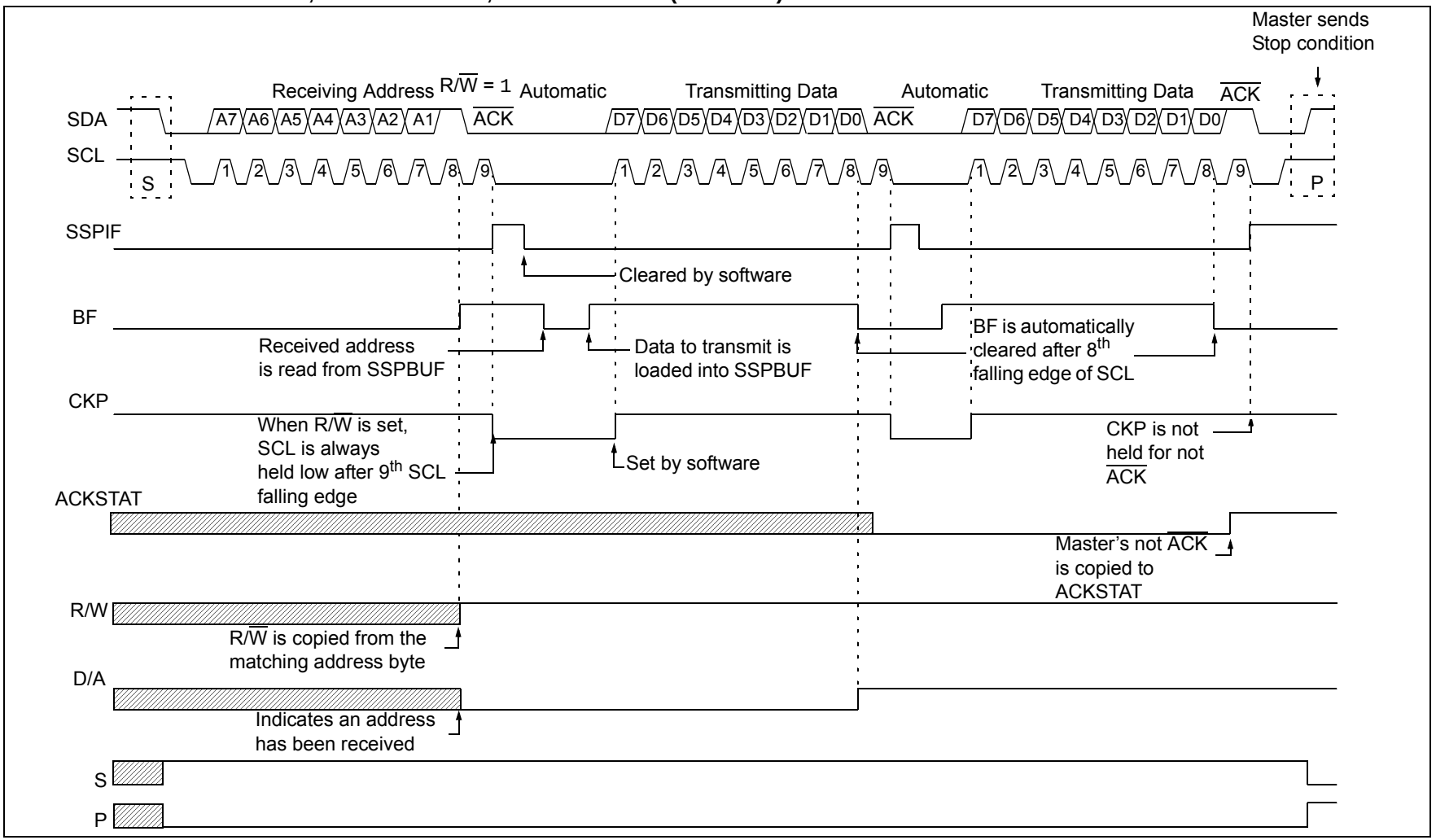
1. Master sends a Start condition on SDA and SCL.
2. S bit in the SSPSTAT register is set; SSPIF is set if interrupt on Start detect is enabled.
3. Matching address with R/$\overline{W}$ bit set is received by the Slave setting SSPIF bit.
4. Slave hardware generates an $\overline{ACK}$ and sets SSPIF.
5. SSPIF bit is cleared by user.
6. Software reads the received address from SSPBUF, clearing BF.
7. R/$\overline{W}$ is set so CKP was automatically cleared after the $\overline{ACK}$.
8. The slave software loads the transmit data into SSPBUF.
9. CKP bit is set releasing SCL, allowing the master to clock the data out of the slave.
10. SSPIF is set after the $\overline{ACK}$ response from the master is loaded into the ACKSTAT register.
11. SSPIF bit is cleared.
12. The slave software checks the ACKSTAT bit to see if the master wants to clock out more data.

Note 1: If the master $\overline{ACK}$s, the clock will be stretched.

2: ACKSTAT is the only bit updated on the rising edge of SCL (9th) rather than on the falling edge.

13. Steps 9–13 are repeated for each transmitted byte.
14. If the master sends a not $\overline{ACK}$, the clock is not held but SSPIF is still set.
15. The master sends a Restart condition or a Stop.
16. The slave is no longer addressed.

FIGURE 27-10: I²C SLAVE, 7-BIT ADDRESS, TRANSMISSION (AHEN = 0)



27.4.4.3 7-Bit Transmission with Address Hold Enabled

Setting the AHEN bit in the SSPCON3 register enables additional clock stretching and interrupt generation after the 8th falling edge of a received matching address. Once a matching address has been clocked in, CKP is cleared and the SSPIF interrupt is set.

Figure 27-11 displays a standard waveform of a 7-bit Address Slave Transmission with AHEN enabled.

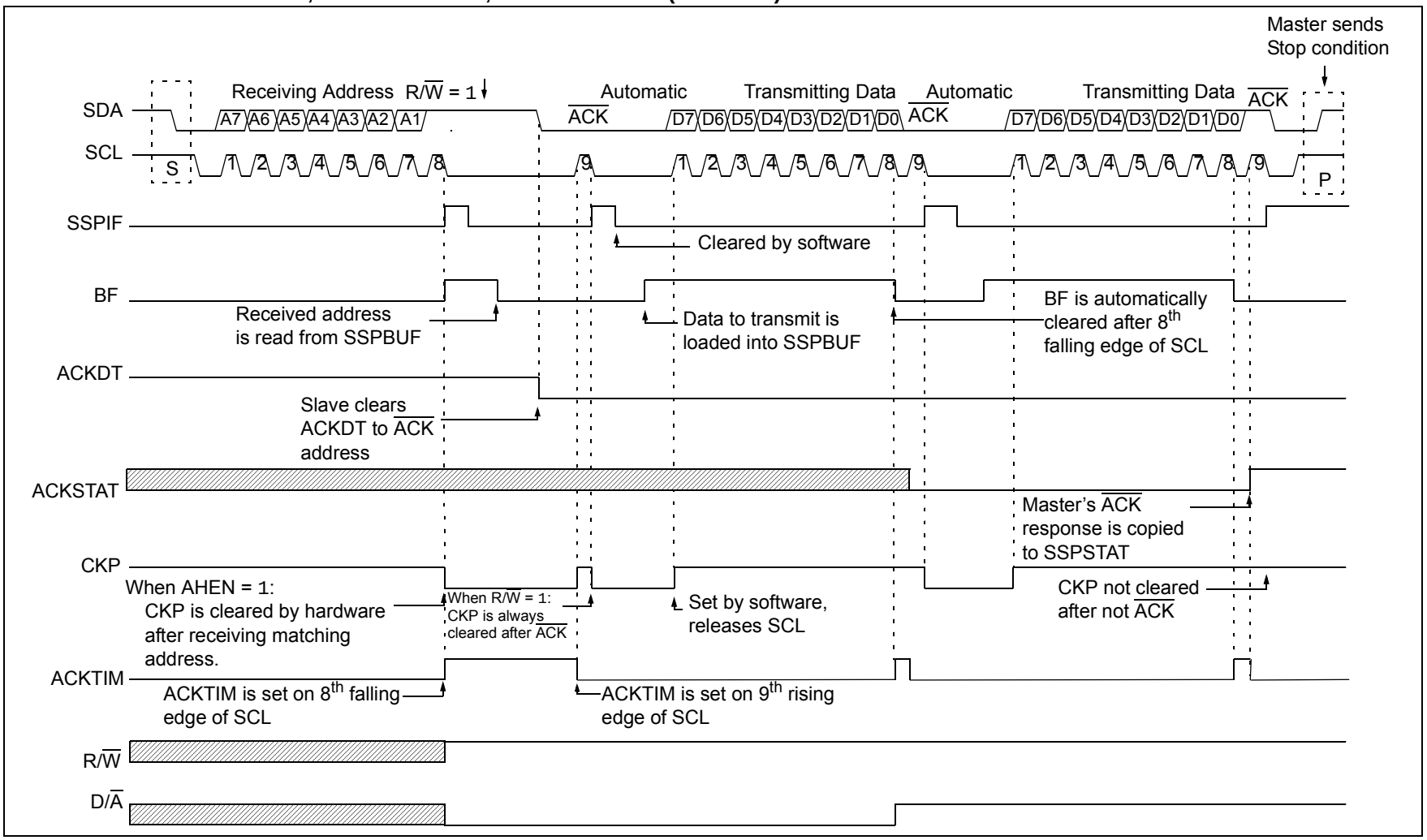
1. Bus starts idle.
2. Master sends Start condition; the S bit in the SSPSTAT register is set; SSPIF is set if interrupt on Start detect is enabled.
3. Master sends matching address with R/$\overline{W}$ bit set. After the 8th falling edge of the SCL line, the CKP bit is cleared and SSPIF interrupt is generated.
4. Slave software clears SSPIF.
5. Slave software reads ACKTIM bit in the SSPCON3 register and R/$\overline{W}$ and D/$\overline{A}$ bits in the SSPSTAT register to determine the source of the interrupt.
6. Slave reads the address value from the SSPBUF register, clearing the BF bit.
7. Slave software decides from this information if it wishes to $\overline{ACK}$ or not $\overline{ACK}$, and sets ACKDT bit in the SSPCON2 register accordingly.
8. Slave sets the CKP bit releasing SCL.
9. Master clocks in the $\overline{ACK}$ value from the slave.
10. Slave hardware automatically clears the CKP bit and sets SSPIF after the $\overline{ACK}$ if the R/$\overline{W}$ bit is set.
11. Slave software clears SSPIF.
12. Slave loads value to transmit to the master into SSPBUF setting the BF bit.

Note:	SSPBUF cannot be loaded until after the $\overline{ACK}$.

13. Slave sets CKP bit releasing the clock.
14. Master clocks out the data from the slave and sends an $\overline{ACK}$ value on the 9th SCL pulse.
15. Slave hardware copies the $\overline{ACK}$ value into the ACKSTAT bit in the SSPCON2 register.
16. Steps 10–15 are repeated for each byte transmitted to the master from the slave.
17. If the master sends a not $\overline{ACK}$, the slave releases the bus, allowing the master to send a Stop and end the communication.

Note:	Master must send a not $\overline{ACK}$ on the last byte to ensure that the slave releases the SCL line to receive a Stop.

FIGURE 27-11: I²C SLAVE, 7-BIT ADDRESS, TRANSMISSION (AHEN = 1)



27.4.5 SLAVE MODE 10-BIT ADDRESS RECEPTION

This section describes a standard sequence of events for the MSSP module configured as an I^2C Slave in 10-bit Addressing mode.

Figure 27-12 is used as a visual reference for this description.

This is a step-by-step process of what must be done by slave software to accomplish I^2C communication:

1. Bus starts idle.
2. Master sends Start condition; S bit in the SSPSTAT register is set; SSPIF is set if interrupt on Start detect is enabled.
3. Master sends matching high address with R/$\overline{W}$ bit clear; UA bit in the SSPSTAT register is set.
4. Slave sends $\overline{ACK}$ and SSPIF is set.
5. Software clears the SSPIF bit.
6. Software reads received address from SSPBUF, clearing the BF flag.
7. Slave loads low address into SSPADD, releasing SCL.
8. Master sends matching low-address byte to the Slave; UA bit is set.

Note:	Updates to the SSPADD register are not allowed until after the $\overline{ACK}$ sequence.

9. Slave sends $\overline{ACK}$ and SSPIF is set.

Note:	If the low address does not match, SSPIF and UA are still set so that the slave software can set SSPADD back to the high address. BF is not set because there is no match. CKP is unaffected.

10. Slave clears SSPIF.
11. Slave reads the received matching address from SSPBUF, clearing BF.
12. Slave loads high address into SSPADD.
13. Master clocks a data byte to the slave and clocks out the slave's $\overline{ACK}$ on the 9th SCL pulse; SSPIF is set.
14. If SEN bit in the SSPCON2 register is set, CKP is cleared by hardware and the clock is stretched.
15. Slave clears SSPIF.
16. Slave reads the received byte from SSPBUF, clearing BF.
17. If SEN is set, the slave sets CKP to release the SCL.
18. Steps 13–17 are repeated for each received byte.
19. Master sends Stop to end the transmission.

27.4.6 10-BIT ADDRESSING WITH ADDRESS OR DATA HOLD

Reception using 10-bit addressing with AHEN or DHEN set is the same as with 7-bit modes. The only difference is the need to update the SSPADD register using the UA bit. All functionality, specifically when the CKP bit is cleared and the SCL line is held low, is the same. Figure 27-13 can be used as a reference of a slave in 10-bit addressing with AHEN set.

Figure 27-14 shows a standard waveform for a slave transmitter in 10-bit Addressing mode.

FIGURE 27-12: I²C SLAVE, 10-BIT ADDRESS, RECEPTION (SEN = 1, AHEN = 0, DHEN = 0)

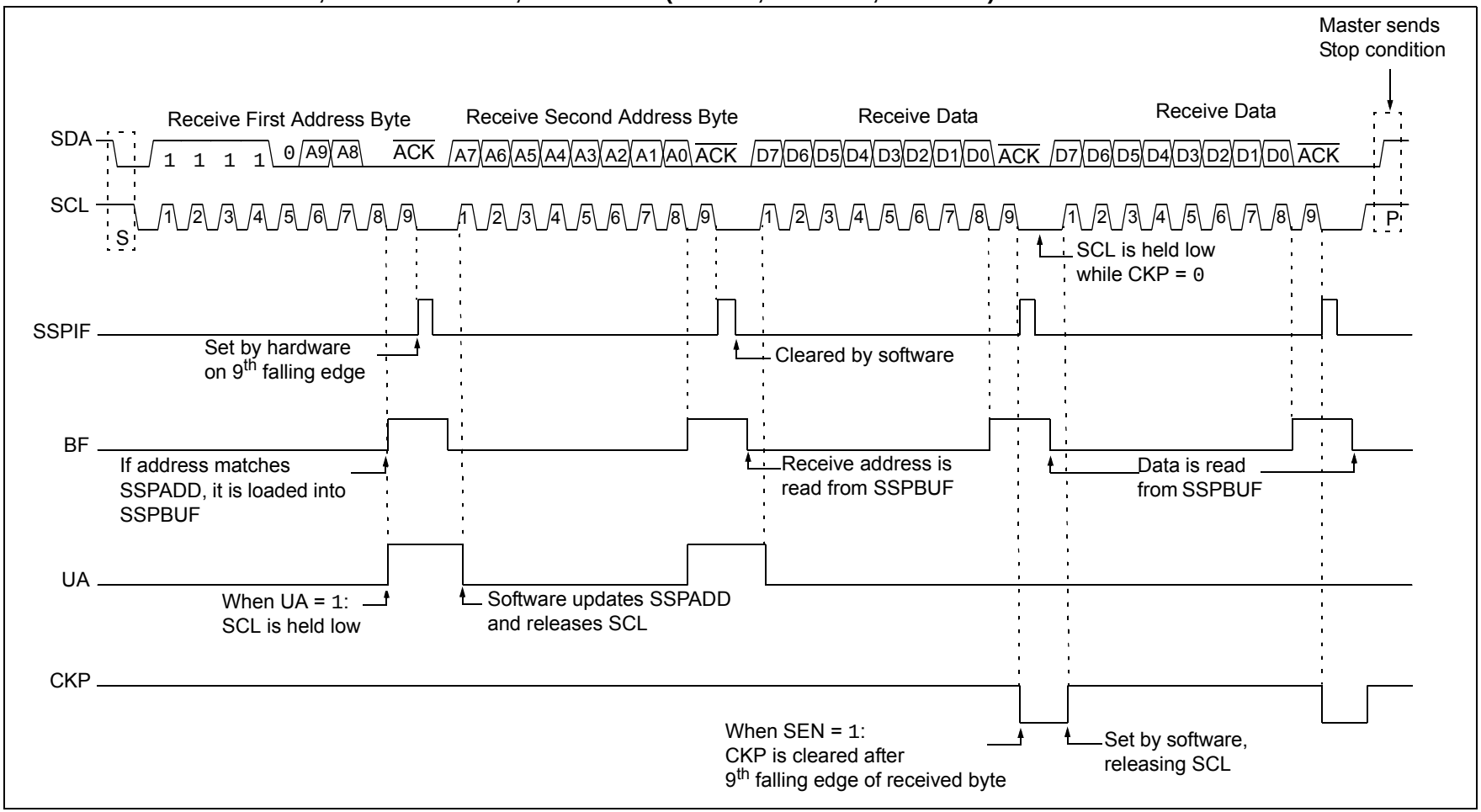


FIGURE 27-13: I²C SLAVE, 10-BIT ADDRESS, RECEPTION (SEN = 0, AHEN = 1, DHEN = 0)

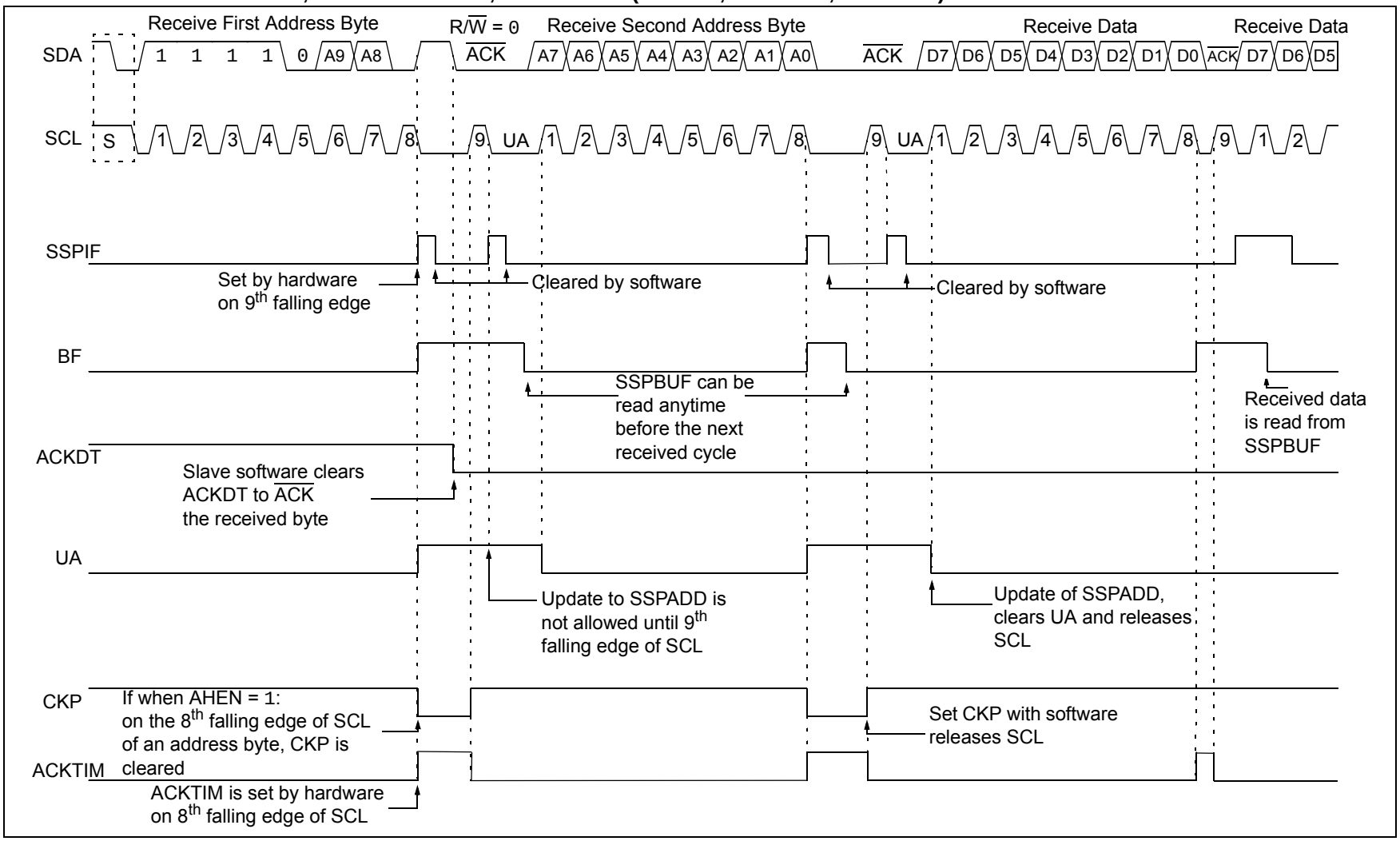
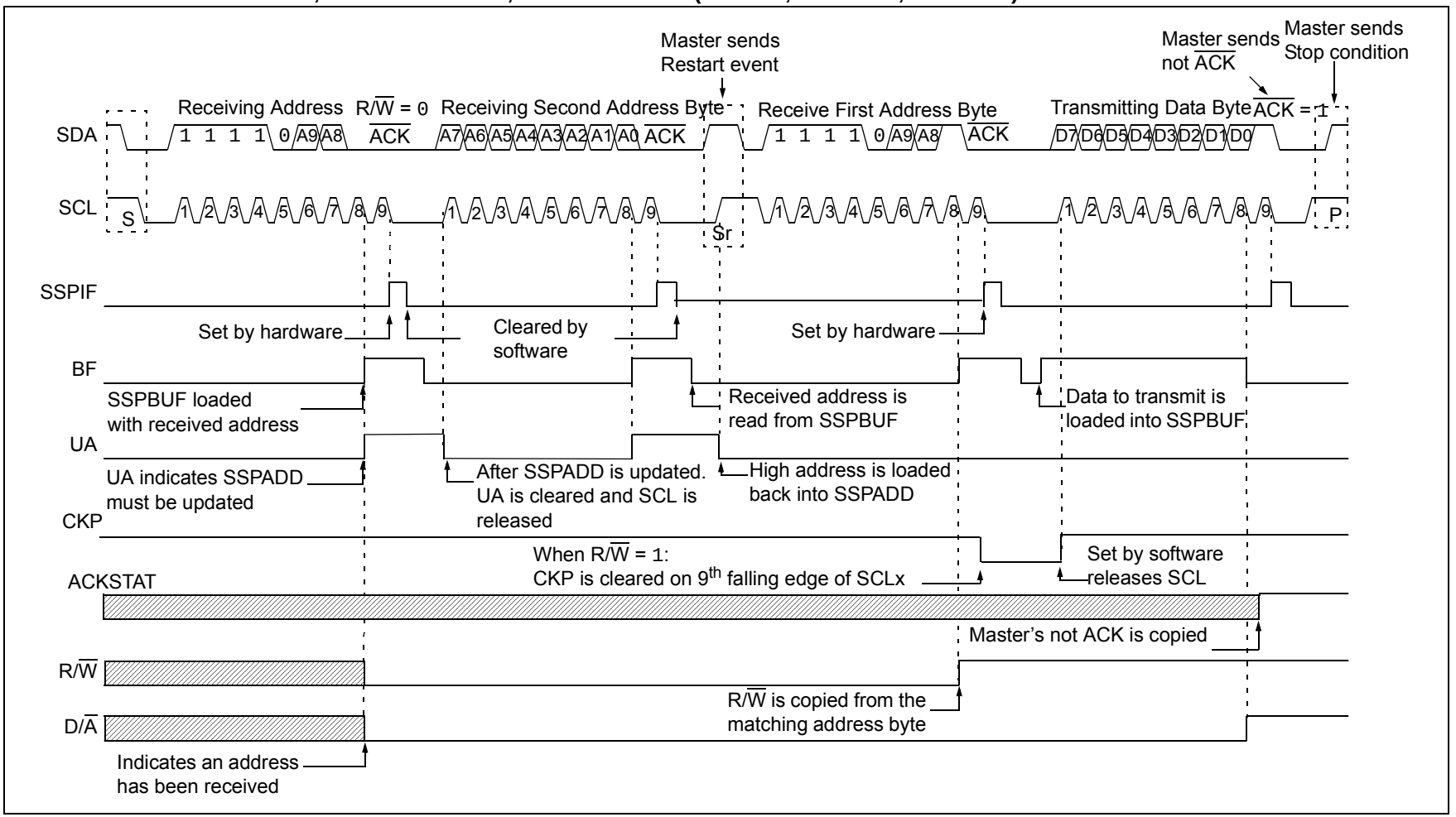


FIGURE 27-14: I²C SLAVE, 10-BIT ADDRESS, TRANSMISSION (SEN = 0, AHEN = 0, DHEN = 0)



27.4.7 CLOCK STRETCHING

Clock stretching occurs when a device on the bus holds the SCL line low, effectively pausing communication. The slave may stretch the clock to allow more time to handle data or prepare a response for the master device. A master device is not concerned with stretching, as it is stretching anytime it is active on the bus and not transferring data. Any stretching done by a slave is invisible to the master software and handled by the hardware that generates SCL.

The CKP bit in the SSPCON1 register is used to control stretching in software. Any time the CKP bit is cleared, the module will wait for the SCL line to go low and then hold it. Setting CKP will release SCL and allow more communication.

27.4.7.1 Normal Clock Stretching

Following an $\overline{ACK}$, if the R/$\overline{W}$ bit in the SSPSTAT register is set, causing a read request, the slave hardware will clear CKP. This allows the slave time to update SSPBUF with data to transfer to the master. If the SEN bit in the SSPCON2 register is set, the slave hardware will always stretch the clock after the $\overline{ACK}$ sequence. Once the slave is ready, CKP is set by software and communication resumes.

Note 1:	The BF bit has no effect on whether the clock will be stretched or not. This is different than previous versions of the module that would not stretch the clock or clear CKP if SSPBUF was read before the 9th falling edge of SCL.
2:	Previous versions of the module did not stretch the clock for a transmission if SSPBUF was loaded before the 9th falling edge of SCL. It is now always cleared for read requests.

27.4.7.2 10-Bit Addressing Mode

In 10-bit Addressing mode, when the UA bit is set, the clock is always stretched. This is the only time the SCL is stretched without CKP being cleared. SCL is released immediately after a write to SSPADD.

Note:	Previous versions of the module did not stretch the clock if the second address byte did not match.

27.4.7.3 Byte NACKing

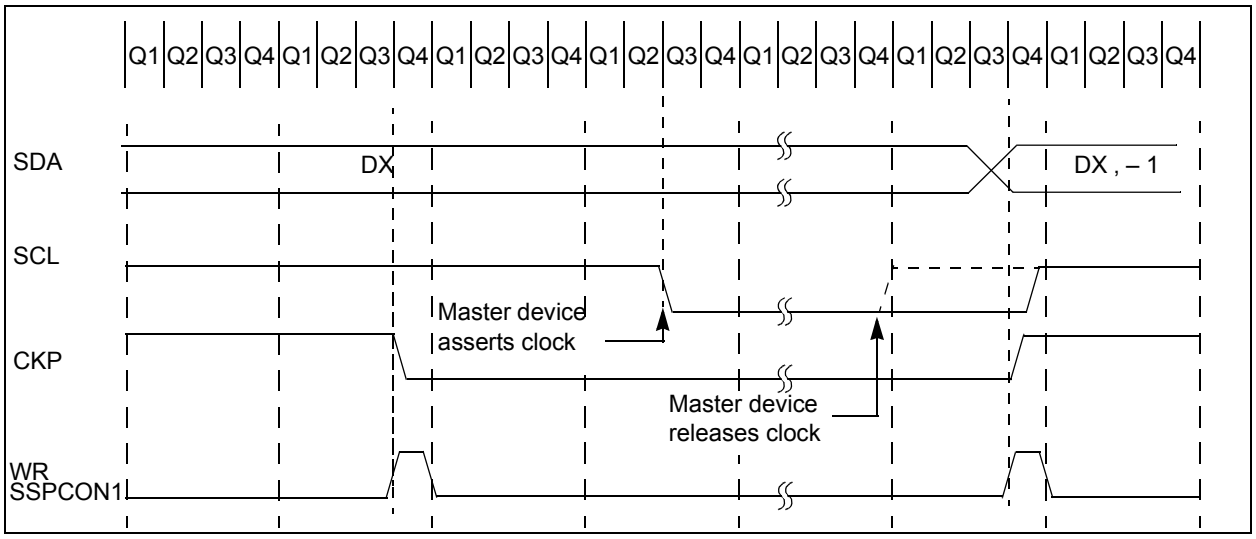
When AHEN bit in the SSPCON3 register is set, CKP is cleared by hardware after the 8th falling edge of SCL for a received matching address byte. When DHEN bit in the SSPCON3 register is set, CKP is cleared after the 8th falling edge of SCL for received data.

Stretching after the 8th falling edge of SCL allows the slave to look at the received address or data and decide if it wants to ACK the received data.

27.4.8 CLOCK SYNCHRONIZATION AND THE CKP BIT

Any time the CKP bit is cleared, the module will wait for the SCL line to go low and then hold it. However, clearing the CKP bit will not assert the SCL output low until the SCL output is already sampled low. Therefore, the CKP bit will not assert the SCL line until an external I^2C master device has already asserted the SCL line. The SCL output will remain low until the CKP bit is set and all other devices on the I^2C bus have released SCL. This ensures that a write to the CKP bit will not violate the minimum high time requirement for SCL (refer to Figure 27-16).

FIGURE 27-15: CLOCK SYNCHRONIZATION TIMING



27.4.9 GENERAL CALL ADDRESS SUPPORT

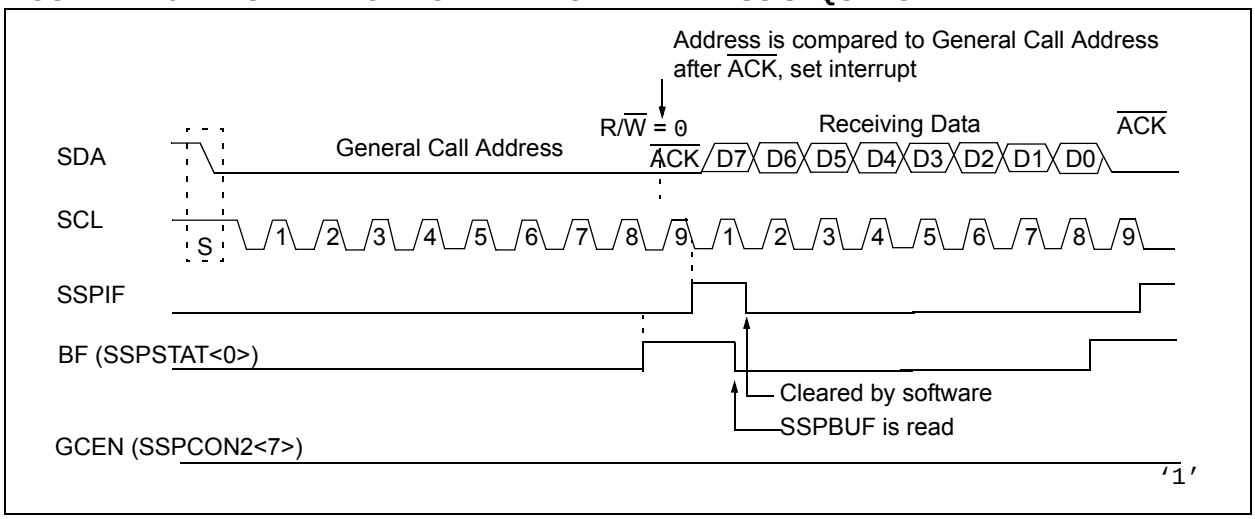
The addressing procedure for the I^2C bus is such that the first byte after the Start condition usually determines which device will be the slave addressed by the master device. The exception is the general call address, which can address all devices. When this address is used, all devices should, in theory, respond with an acknowledge.

The general call address is a reserved address in the I^2C protocol, defined as address 0x00. When the GCEN bit in the SSPCON2 register is set, the slave module will automatically $\overline{ACK}$ the reception of this address regardless of the value stored in SSPADD. After the slave clocks in an address of all zeros with the R/$\overline{W}$ bit clear, an interrupt is generated and slave software can read SSPBUF and respond. Figure 27-17 shows a general call reception sequence.

In 10-bit Address mode, the UA bit will not be set on the reception of the general call address. The slave will prepare to receive the second byte as data, just as it would in 7-bit mode.

If the AHEN bit in the SSPCON3 register is set, just as with any other address reception, the slave hardware will stretch the clock after the 8th falling edge of SCL. The slave must then set its ACKDT value and release the clock with communication progressing as it would normally.

FIGURE 27-16: SLAVE MODE GENERAL CALL ADDRESS SEQUENCE



27.4.10 SSPMSK1 REGISTER

An SSP Mask (SSPMSK1) register is available in I^2C Slave mode as a mask for the value held in the SSPSR register during an address comparison operation. A zero ('0') bit in the SSPMSK1 register has the effect of making the corresponding bit of the received address a "don't care".

This register is reset to all '1's upon any Reset condition and, therefore, has no effect on standard SSP operation until written with a mask value.

The SSPMSK1 register is active during:

- 7-bit Address mode: address compare of A<7:1>.
- 10-bit Address mode: address compare of A<7:0> only. The SSP mask has no effect during the reception of the first (high) byte of the address.

27.5 I²C MASTER MODE

Master mode is enabled by setting and clearing the appropriate SSPM bits in the SSPCON1 register and by setting the SSPEN bit. In Master mode, the SDA and SCK pins must be configured as inputs. The MSSP peripheral hardware will override the output driver TRIS controls when necessary, to drive the pins low.

The Master mode of operation is supported by interrupt generation on the detection of the Start and Stop conditions. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSP module is disabled. Control of the I²C bus may be taken when the P bit is set or the bus is idle.

In Firmware-Controlled Master mode, user code conducts all I²C bus operations based on Start and Stop bit condition detection. Start and Stop condition detection is the only active circuitry in this mode. All other communication is done by the user's software directly manipulating the SDA and SCL lines.

The following events will cause the SSP Interrupt Flag bit (SSPIF) to be set (SSP interrupt, if enabled):

- Start condition detected
- Stop condition detected
- Data transfer byte transmitted/received
- Acknowledge transmitted/received
- Repeated Start generated

Note 1:	The MSSP module, when configured in I²C Master mode, does not allow queuing of events. For instance, the user is not allowed to initiate a Start condition and immediately write the SSPBUF register to initiate transmission before the Start condition is complete. In this case, the SSPBUF will not be written to and the WCOL bit will be set, indicating that a write to the SSPBUF did not occur.
2:	When in Master mode, Start/Stop detection is masked and an interrupt is generated when the SEN/PEN bit is cleared and the generation is complete.

27.5.1 I²C MASTER MODE OPERATION

The master device generates all of the serial clock pulses and the Start and Stop conditions. A transfer ends with a Stop condition or with a Repeated Start condition. Since the Repeated Start condition is also the beginning of the next serial transfer, the I²C bus will not be released.

In Master Transmit mode, serial data is output through SDA while SCL outputs the serial clock. The first byte transmitted contains the slave address of the receiving device (7 bits) and the Read/Write (R/$\overline{W}$) bit. In this case, the R/$\overline{W}$ bit will be logic '0'. Serial data is transmitted 8 bits at a time. After each byte is transmitted, an Acknowledge bit is received. Start and Stop conditions are output to indicate the beginning and the end of a serial transfer.

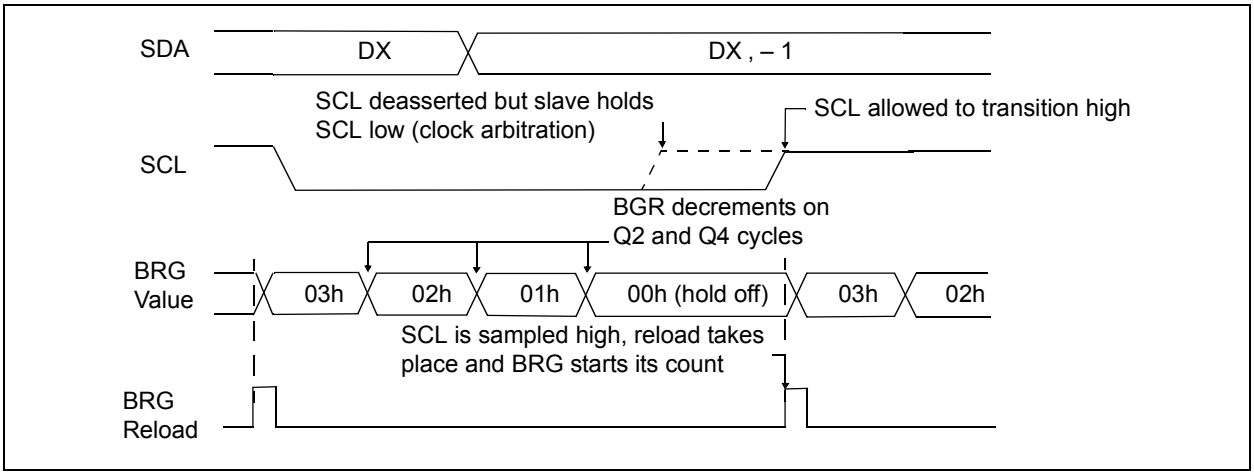
In Master Receive mode, the first byte transmitted contains the slave address of the transmitting device (7 bits) and the R/$\overline{W}$ bit. In this case, the R/$\overline{W}$ bit will be logic '1'. Thus, the first byte transmitted is a 7-bit slave address followed by a '1' to indicate the receive bit. Serial data is received via SDA, while SCL outputs the serial clock. Serial data is received 8 bits at a time. After each byte is received, an Acknowledge bit is transmitted. Start and Stop conditions indicate the beginning and end of transmission.

A Baud Rate Generator is used to set the clock frequency output on SCL. Refer to **Section 27.6 "Baud Rate Generator"** for more details.

27.5.2 CLOCK ARBITRATION

Clock arbitration occurs when the master, during any receive, transmit or Repeated Start/Stop condition, releases the SCL pin (SCL allowed to float high). When the SCL pin is allowed to float high, the Baud Rate Generator (BRG) is suspended from counting until the SCL pin is actually sampled high. When the SCL pin is sampled high, the Baud Rate Generator is reloaded with the contents of SSPADD<7:0> and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count in the event that the clock is held low by an external device (Figure 27-17).

FIGURE 27-17: BAUD RATE GENERATOR TIMING WITH CLOCK ARBITRATION



27.5.3 WCOL STATUS FLAG

If the user writes the SSPBUF when a Start, Restart, Stop, Receive or Transmit sequence is in progress, the WCOL is set and the contents of the buffer are unchanged (the write does not occur). Any time the WCOL bit is set, it indicates that an action on SSPBUF was attempted while the module was not idle.

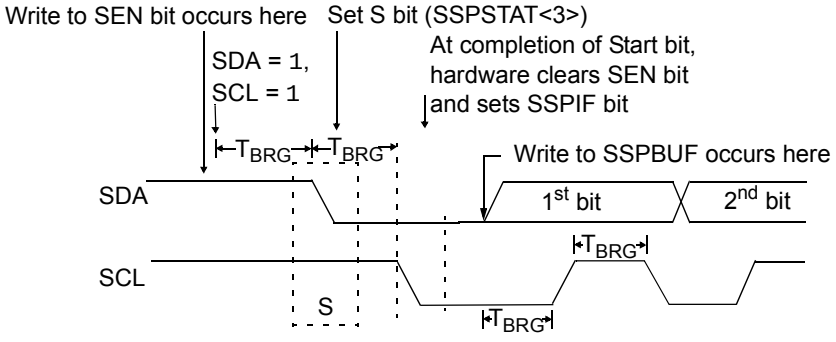
Note:	Because queuing of events is not allowed, writing to the lower 5 bits in the SSPCON2 register is disabled until the Start condition is complete.

27.5.4 I^2C MASTER MODE START CONDITION TIMING

To initiate a Start condition, the user sets the Start Enable bit, SEN, in the SSPCON2 register. If the SDA and SCL pins are sampled high, the Baud Rate Generator is reloaded with the contents of SSPADD<7:0> and starts its count. If SCL and SDA are both sampled high when the Baud Rate Generator times out (T_{BRG}), the SDA pin is driven low. The action of the SDA being driven low while SCL is high is the Start condition and causes the S bit in the SSPSTAT1 register to be set. Following this, the Baud Rate Generator is reloaded with the contents of SSPADD<7:0> and resumes its count. When the Baud Rate Generator times out (T_{BRG}), the SEN bit in the SSPCON2 register will be automatically cleared by hardware; the Baud Rate Generator is suspended, leaving the SDA line held low and the Start condition is complete.

Note 1:	If, at the beginning of the Start condition, the SDA and SCL pins are already sampled low, or if, during the Start condition, the SCL line is sampled low before the SDA line is driven low, a bus collision occurs, the Bus Collision Interrupt Flag, BCLIF, is set, the Start condition is aborted and the I^2C module is reset into its idle state.
2:	The Philips I^2C Specification states that a bus collision cannot occur on a Start.

FIGURE 27-18: FIRST START BIT TIMING



27.5.5 I²C MASTER MODE REPEATED START CONDITION TIMING

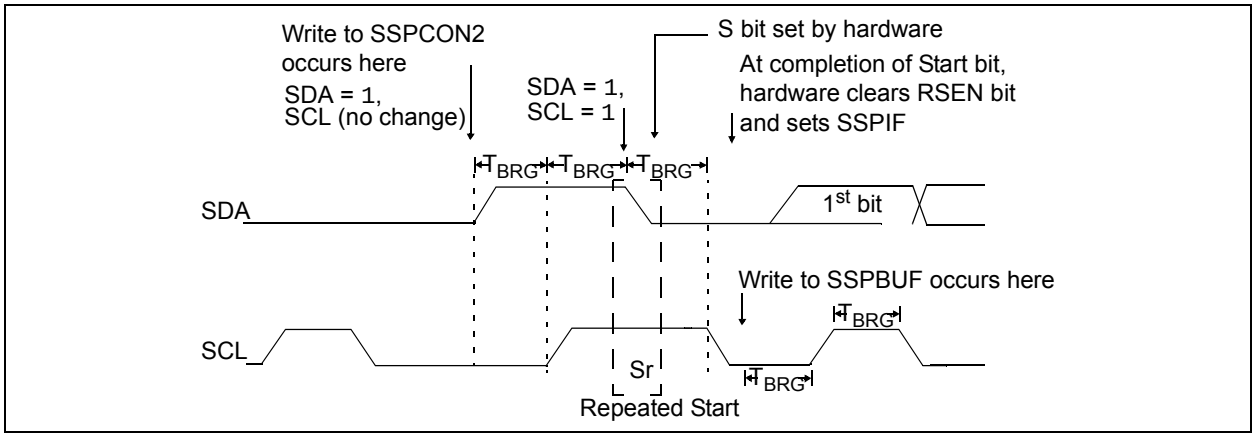
A Repeated Start condition occurs when the RSEN bit in the SSPCON2 register is programmed high and the Master state machine is no longer active. When the RSEN bit is set, the SCL pin is asserted low. When the SCL pin is sampled low, the Baud Rate Generator is loaded and begins counting. The SDA pin is released (brought high) for one Baud Rate Generator count (T_{BRG}). When the Baud Rate Generator times out, if SDA is sampled high, the SCL pin will be deasserted (brought high). When SCL is sampled high, the Baud Rate Generator is reloaded and begins counting. SDA and SCL must be sampled high for one T_{BRG}. This action is then followed by assertion of the SDA pin (SDA = 0) for one T_{BRG} while SCL is high. SCL is asserted low. Following this, the RSEN bit in the SSPCON2 register will be automatically cleared and the Baud Rate Generator will not be reloaded, leaving the SDA pin held low. As soon as a Start condition is detected on the SDA and SCL pins, the S bit in the SSPSTAT register will be set. The SSPIF bit will not be set until the Baud Rate Generator has timed out.

Note 1: If RSEN is programmed while any other event is in progress, it will not take effect.

2: A bus collision during the Repeated Start condition occurs if:

- SDA is sampled low when SCL goes from low to high.
- SCL goes low before SDA is asserted low. This may indicate that another master is attempting to transmit a data '1'.

FIGURE 27-19: REPEAT START CONDITION WAVEFORM



27.5.6 I²C MASTER MODE TRANSMISSION

Transmission of a data byte, a 7-bit address or the other half of a 10-bit address is accomplished by simply writing a value to the SSPBUF register. This action will set the Buffer Full (BF) flag bit and allow the Baud Rate Generator to begin counting and start the next transmission. Each bit of address/data will be shifted out onto the SDA pin after the falling edge of SCL is asserted. SCL is held low for one Baud Rate Generator rollover count (T_{BRG}). Data should be valid before SCL is released high. When the SCL pin is released high, it is held that way for T_{BRG}. The data on the SDA pin must remain stable for that duration and some hold time after the next falling edge of SCL. After the 8th bit is shifted out (the falling edge of the 8th clock), the BF flag is cleared and the master releases the SDA. This allows the slave device being addressed to respond with an $\overline{ACK}$ bit during the 9th bit time if an address match occurred or if data was received properly. The status of $\overline{ACK}$ is written into the ACKSTAT bit on the rising edge of the 9th clock. If the master receives an Acknowledge, the Acknowledge Status bit (ACKSTAT) is cleared. If not, the bit is set. After the 9th clock, the SSPIF bit is set and the master clock (Baud Rate Generator) is suspended until the next data byte is loaded into the SSPBUF, leaving SCL low and SDA unchanged (Figure 27-20).

After the write to the SSPBUF, each bit of the address will be shifted out on the falling edge of SCL until all seven address bits and the R/$\overline{W}$ bit are completed. On the falling edge of the 8th clock, the master will release the SDA pin, allowing the slave to respond with an Acknowledge. On the falling edge of the 9th clock, the master will sample the SDA pin to see if the address was recognized by a slave. The status of the $\overline{ACK}$ bit is loaded into the ACKSTAT status bit in the SSPCON2 register. Following the falling edge of the 9th clock transmission of the address, the SSPIF is set, the BF flag is cleared and the Baud Rate Generator is turned off until another write to the SSPBUF takes place, holding SCL low and allowing SDA to float.

27.5.6.1 BF Status Flag

In Transmit mode, the BF bit in the SSPSTAT register is set when the CPU writes to SSPBUF and is cleared when all 8 bits are shifted out.

27.5.6.2 WCOL Status Flag

If the user writes the SSPBUF when a transmit is already in progress (i.e., SSPSR is still shifting out a data byte), the WCOL is set and the contents of the buffer are unchanged (the write does not occur).

WCOL must be cleared by software before the next transmission.

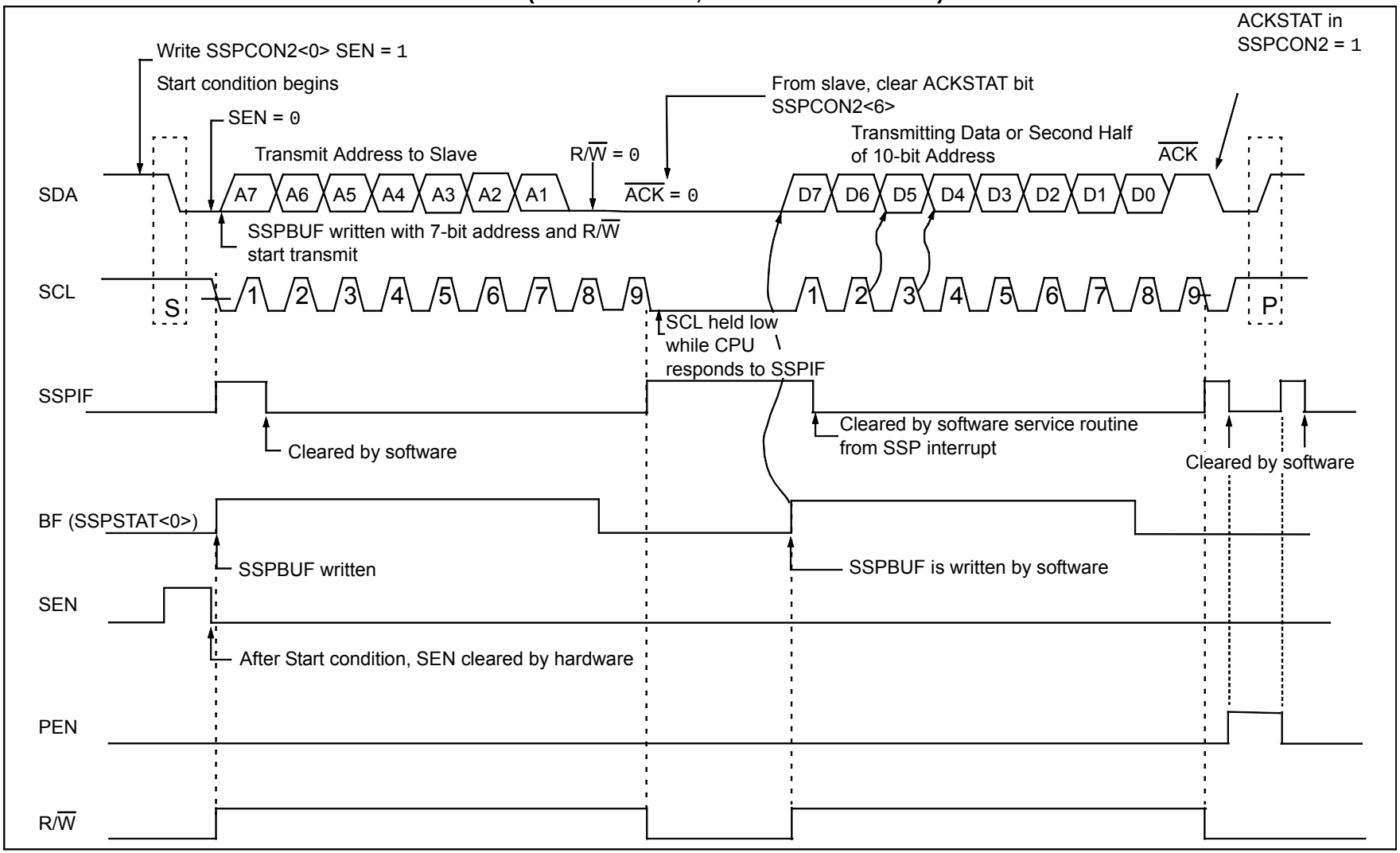
27.5.6.3 ACKSTAT Status Flag

In Transmit mode, the ACKSTAT bit in the SSPCON2 register is cleared when the slave has sent an Acknowledge ($\overline{ACK}$ = 0) and is set when the slave does not Acknowledge ($\overline{ACK}$ = 1). A slave sends an Acknowledge when it has recognized its address (including a general call) or when the slave has properly received its data.

27.5.6.4 Typical Transmit Sequence:

1. The user generates a Start condition by setting the SEN bit in the SSPCON2 register.
2. SSPIF is set by hardware on completion of the Start.
3. SSPIF is cleared by software.
4. The MSSP module will wait the required start time before any other operation takes place.
5. The user loads the SSPBUF with the slave address to transmit.
6. Address is shifted out the SDA pin until all 8 bits are transmitted. Transmission begins as soon as SSPBUF is written to.
7. The MSSP module shifts in the $\overline{ACK}$ bit from the slave device and writes its value into the ACKSTAT bit in the SSPCON2 register.
8. The MSSP module generates an interrupt at the end of the 9th clock cycle by setting the SSPIF bit.
9. The user loads the SSPBUF with 8 bits of data.
10. Data is shifted out the SDA pin until all 8 bits are transmitted.
11. The MSSP module shifts in the $\overline{ACK}$ bit from the slave device and writes its value into the ACKSTAT bit in the SSPCON2 register.
12. Steps 8-11 are repeated for all transmitted data bytes.
13. The user generates a Stop or Restart condition by setting the PEN or RSEN bits in the SSPCON2 register. Interrupt is generated once the Stop/Restart condition is complete.

FIGURE 27-20: I²C MASTER MODE WAVEFORM (TRANSMISSION, 7 OR 10-BIT ADDRESS)



27.5.7 I²C MASTER MODE RECEPTION

Master mode reception is enabled by programming the Receive Enable (RCEN) bit in the SSPCON2 register.

Note:	The MSSP module must be in an idle state before the RCEN bit is set or the RCEN bit will be disregarded.

The Baud Rate Generator begins counting and, upon each rollover, the state of the SCL pin changes (high-to-low/low-to-high) and data is shifted into the SSPSR. After the falling edge of the 8th clock, the receive enable flag is automatically cleared, the contents of the SSPSR are loaded into the SSPBUF, the BF flag bit is set, the SSPIF flag bit is set and the Baud Rate Generator is suspended from counting, holding SCL low. The MSSP is now in idle state awaiting the next command. When the buffer is read by the CPU, the BF flag bit is automatically cleared. The user can then send an Acknowledge bit at the end of reception by setting the Acknowledge Sequence Enable (ACKEN) bit in the SSPCON2 register.

27.5.7.1 BF Status Flag

In receive operation, the BF bit is set when an address or data byte is loaded into SSPBUF from SSPSR. It is cleared when the SSPBUF register is read.

27.5.7.2 SSPOV Status Flag

In receive operation, the SSPOV bit is set when 8 bits are received into the SSPSR and the BF flag bit is already set from a previous reception.

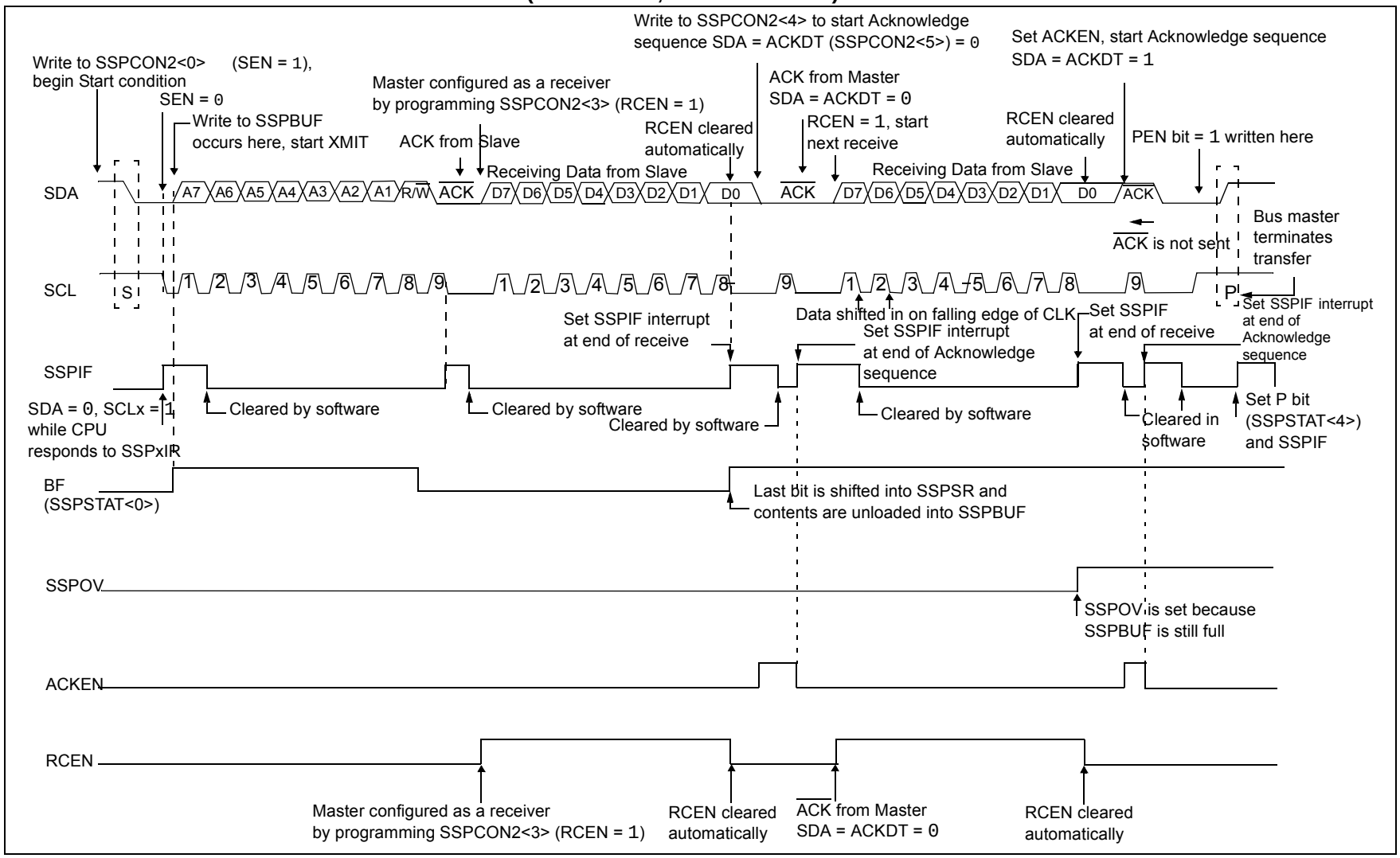
27.5.7.3 WCOL Status Flag

If the user writes the SSPBUF when a receive is already in progress (i.e., SSPSR is still shifting in a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

27.5.7.4 Typical Receive Sequence:

1. The user generates a Start condition by setting the SEN bit in the SSPCON2 register.
2. SSPIF is set by hardware on completion of the Start.
3. SSPIF is cleared by software.
4. User writes SSPBUF with the slave address to transmit and the R/$\overline{W}$ bit set.
5. Address is shifted out the SDA pin until all 8 bits are transmitted. Transmission begins as soon as SSPBUF is written to.
6. The MSSP module shifts in the $\overline{ACK}$ bit from the slave device and writes its value into the ACKSTAT bit in the SSPCON2 register.
7. The MSSP module generates an interrupt at the end of the 9th clock cycle by setting the SSPIF bit.
8. User sets the RCEN bit in the SSPCON2 register and the Master clocks in a byte from the slave.
9. After the 8th falling edge of SCL, SSPIF and BF are set.
10. Master clears SSPIF and reads the received byte from SSPUF, then clears BF.
11. Master sets $\overline{ACK}$ value sent to slave in ACKDT bit in the SSPCON2 register and initiates the $\overline{ACK}$ by setting the ACKEN bit.
12. Masters $\overline{ACK}$ is clocked out to the Slave and SSPIF is set.
13. The user clears SSPIF.
14. Steps 8-13 are repeated for each received byte from the slave.
15. Master sends a not $\overline{ACK}$ or Stop to end communication.

FIGURE 27-21: I^2C MASTER MODE WAVEFORM (RECEPTION, 7-BIT ADDRESS)



27.5.8 ACKNOWLEDGE SEQUENCE TIMING

An Acknowledge sequence is enabled by setting the Acknowledge Sequence Enable (ACKEN) bit in the SSPCON2 register. When this bit is set, the SCL pin is pulled low and the contents of the Acknowledge data bit are presented on the SDA pin. If the user wishes to generate an Acknowledge, then the ACKDT bit should be cleared. If not, the user should set the ACKDT bit before starting an Acknowledge sequence. The Baud Rate Generator then counts for one rollover period (T_{BRG}) and the SCL pin is deasserted (pulled high). When the SCL pin is sampled high (clock arbitration), the Baud Rate Generator counts for T_{BRG}. The SCL pin is then pulled low. Following this, the ACKEN bit is automatically cleared, the Baud Rate Generator is turned off and the MSSP module then goes into Idle mode (Figure 27-22).

27.5.8.1 WCOL Status Flag

If the user writes the SSPBUF when an Acknowledge sequence is in progress, WCOL is set and the contents of the buffer are unchanged (the write does not occur).

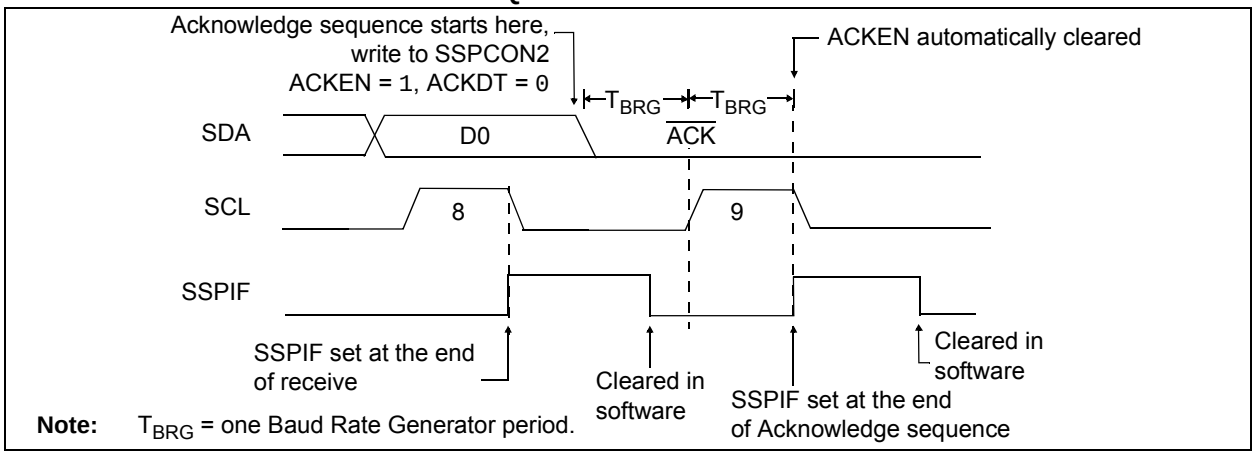
27.5.9 STOP CONDITION TIMING

A Stop bit is asserted on the SDA pin at the end of a receive/transmit by setting the Stop Sequence Enable bit, PEN, in the SSPCON2 register. At the end of a receive/transmit, the SCL line is held low after the falling edge of the 9th clock. When the PEN bit is set, the master will assert the SDA line low. When the SDA line is sampled low, the Baud Rate Generator is reloaded and counts down to '0'. When the Baud Rate Generator times out, the SCL pin will be brought high and then, one T_{BRG} (Baud Rate Generator rollover count) later, the SDA pin will be deasserted. When the SDA pin is sampled high while SCL is high, the P bit in the SSPSTAT register, is set. A T_{BRG} later, the PEN bit is cleared and the SSPIF bit is set (Figure 27-23).

27.5.9.1 WCOL Status Flag

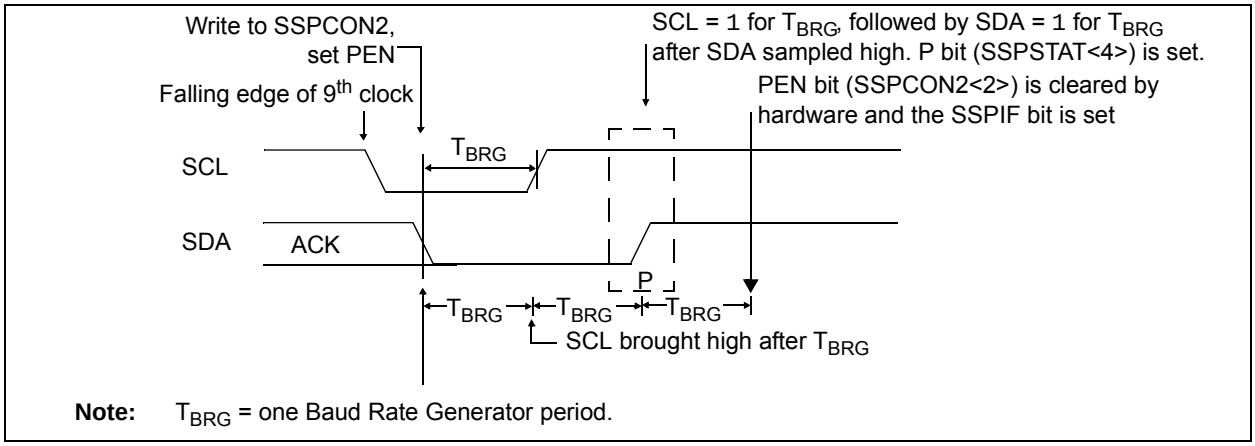
If the user writes the SSPBUF when a Stop sequence is in progress, the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

FIGURE 27-22: ACKNOWLEDGE SEQUENCE WAVEFORM



Note: T_{BRG} = one Baud Rate Generator period.

FIGURE 27-23: STOP CONDITION RECEIVE OR TRANSMIT MODE



Note: T_{BRG} = one Baud Rate Generator period.

27.5.10 SLEEP OPERATION

While in Sleep mode, the I^2C slave module can receive addresses or data and, when an address match or complete byte transfer occurs, wake the processor from Sleep (if the MSSP interrupt is enabled).

27.5.11 EFFECTS OF A RESET

A Reset disables the MSSP module and terminates the current transfer.

27.5.12 MULTI-MASTER MODE

In Multi-Master mode, the interrupt generation on the detection of the Start and Stop conditions allows the determination of when the bus is free. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSPx module is disabled. Control of the I^2C bus may be taken when the P bit in the SSPSTAT register is set or the bus is idle, with both the S and P bits clear. When the bus is busy, enabling the SSP interrupt will generate the interrupt when the Stop condition occurs.

In multi-master operation, the SDA line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed by hardware with the result placed in the BCLIF bit.

The states where arbitration can be lost are:

- Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

27.5.13 MULTI-MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA by letting SDA float high, and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin is '0', a bus collision has taken place. The master will set the Bus Collision Interrupt Flag (BCLIF) and reset the I^2C port to its Idle state (Figure 27-24).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are deasserted and the SSPBUF can be written to. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

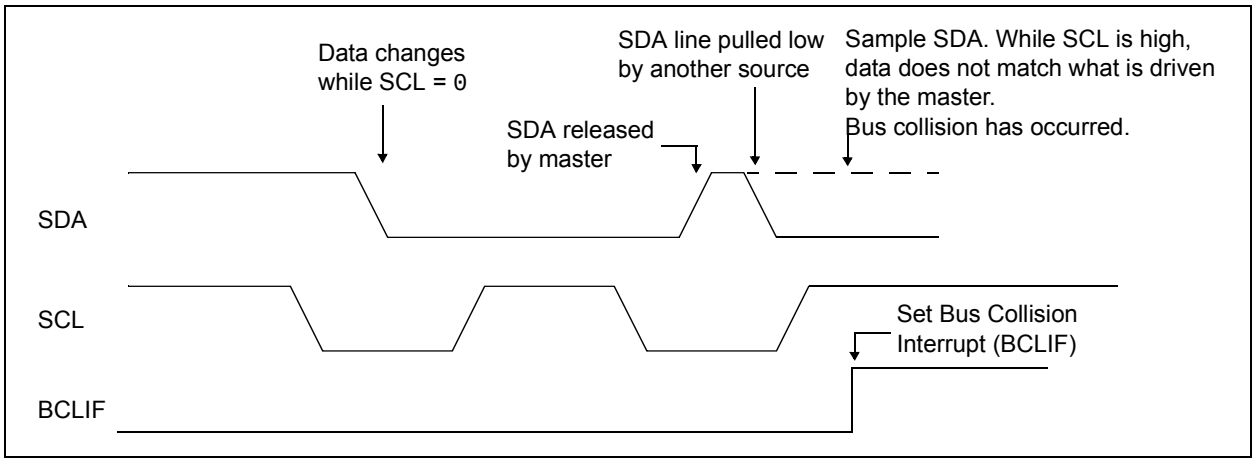
If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the SSPCON2 register are cleared. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

The master will continue to monitor the SDA and SCL pins. If a Stop condition occurs, the SSPIF bit will be set.

A write to the SSPBUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when the bus collision occurred.

In Multi-Master mode, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I^2C bus can be taken when the P bit is set in the SSPSTAT register, or the bus is idle and the S and P bits are cleared.

FIGURE 27-24: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



27.5.13.1 Bus Collision During a Start Condition

During a Start condition, a bus collision occurs if:

a) SDA or SCL are sampled low at the beginning of the Start condition (Figure 27-25)

b) SCL is sampled low before SDA is asserted low (Figure 27-26)

During a Start condition, both the SDA and the SCL pins are monitored.

If the SDA pin is already low or the SCL pin is already low, all of the following occur:

- the Start condition is aborted
- the BCLIF flag is set and
- the MSSP module is reset to its Idle state (Figure 27-25)

The Start condition begins with the SDA and SCL pins deasserted. When the SDA pin is sampled high, the Baud Rate Generator is loaded and counts down. If the SCL pin is sampled low while SDA is high, a bus collision occurs because it is assumed that another master is attempting to drive a data '1' during the Start condition.

If the SDA pin is sampled low during this count, the BRG is reset and the SDA line is asserted early (Figure 27-27). If, however, a '1' is sampled on the SDA pin, the SDA pin is asserted low at the end of the BRG count. The Baud Rate Generator is then reloaded and counts down to zero; if the SCL pin is sampled as '0' during this time, a bus collision does not occur. At the end of the BRG count, the SCL pin is asserted low.

Note:	The reason why bus collision is not a factor during a Start condition is that no two bus masters can assert a Start condition at the exact same time. Therefore, one master will always assert SDA before the other. This condition does not cause a bus collision because the two masters must be allowed to arbitrate the first address following the Start condition. If the address is the same, arbitration must be allowed to continue into the data portion, Repeated Start or Stop conditions.

FIGURE 27-25: BUS COLLISION DURING A START CONDITION (SDA ONLY)

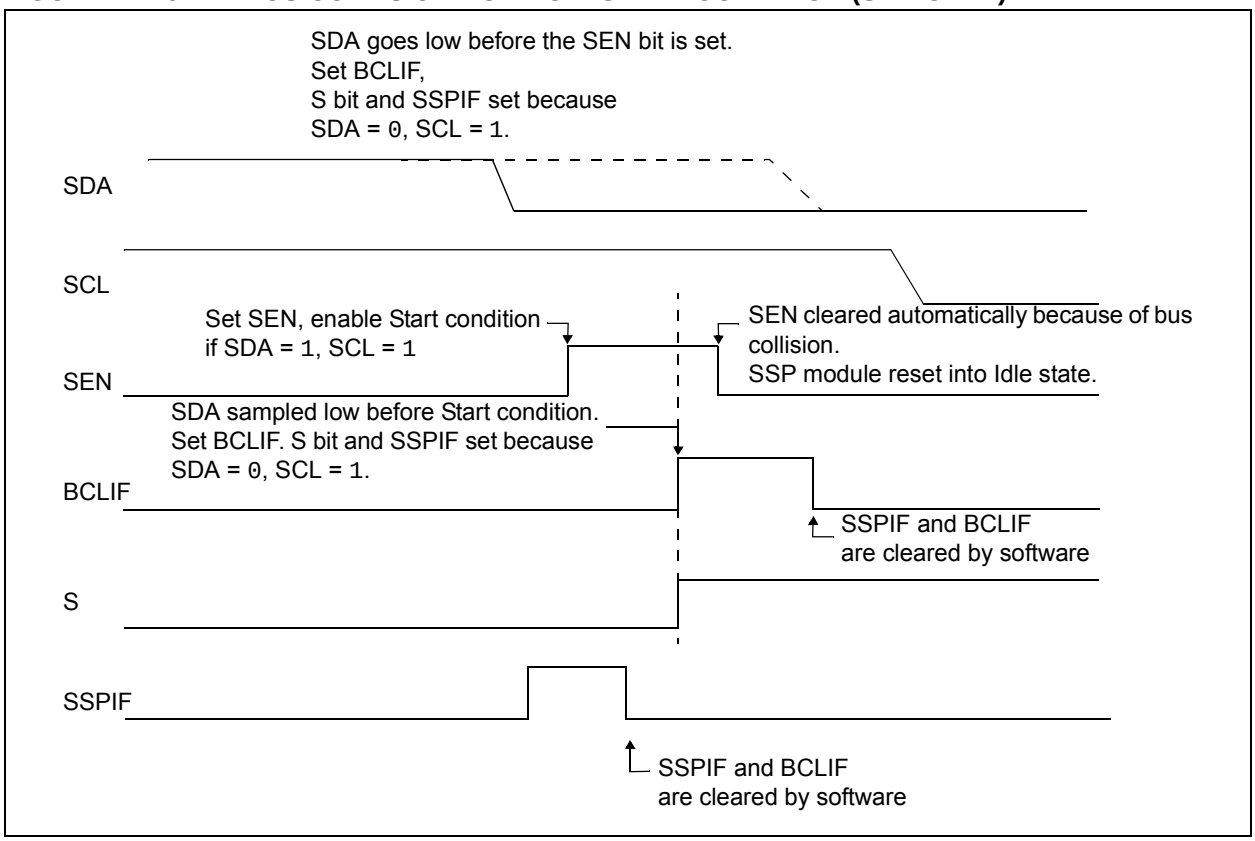


FIGURE 27-26: BUS COLLISION DURING A START CONDITION (SCL = 0)

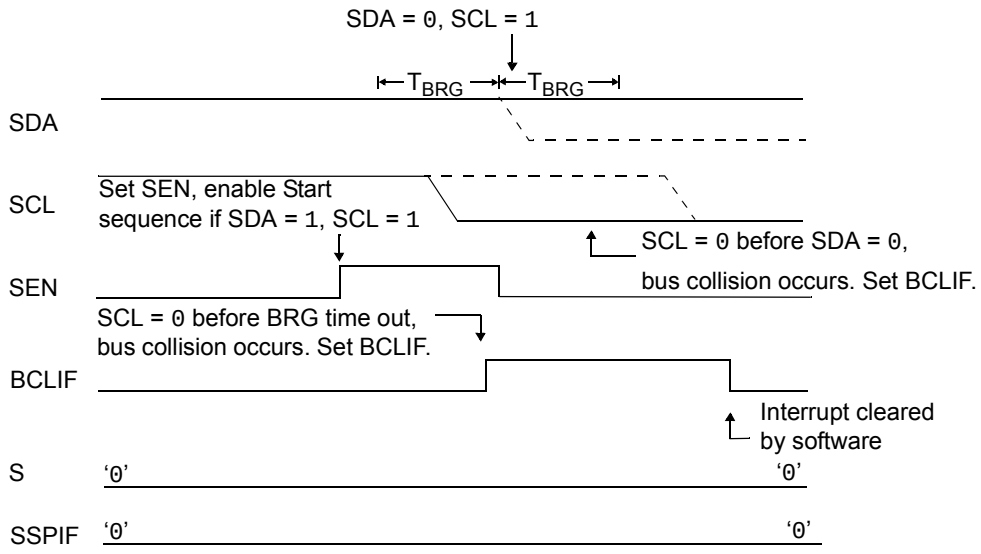
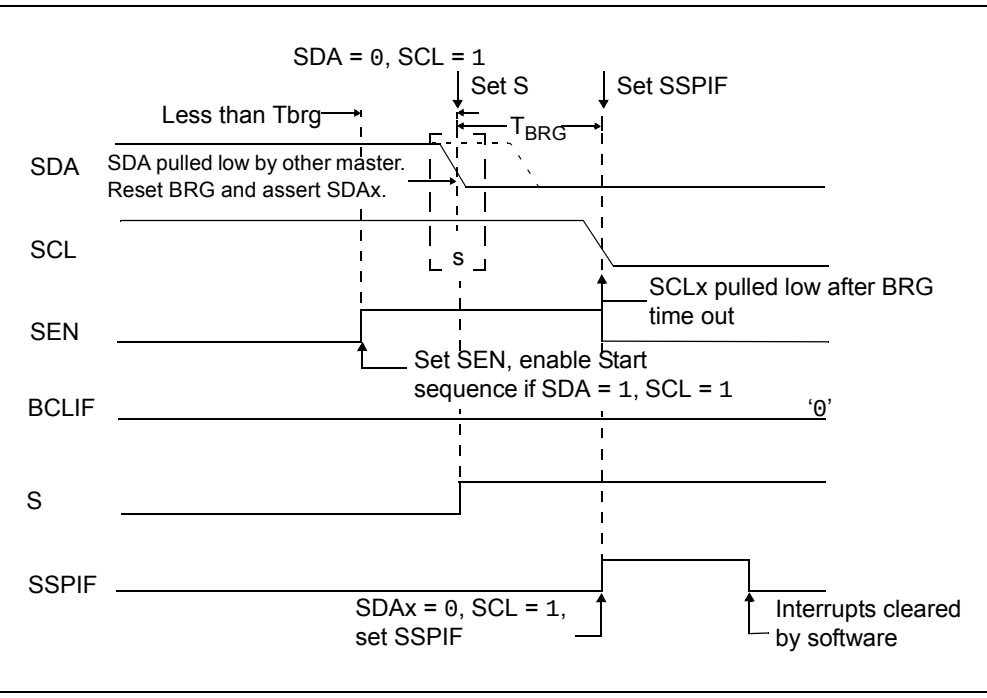


FIGURE 27-27: BRG RESET DUE TO SDA ARBITRATION DURING START CONDITION



27.5.13.2 Bus Collision During a Repeated Start Condition

During a Repeated Start condition, a bus collision occurs if:

a) A low level is sampled on SDA when SCL goes from low level to high level

b) SCL goes low before SDA is asserted low, indicating that another master is attempting to transmit a data '1'

When the user releases SDA and the pin is allowed to float high, the BRG is loaded with SSPADD and counts down to zero. The SCL pin is then deasserted and, when sampled high, the SDA pin is sampled.

If SDA is low, a bus collision has occurred (i.e., another master is attempting to transmit a data '0'; see Figure 27-28). If SDA is sampled high, the BRG is reloaded and begins counting. If SDA goes from high to low before the BRG times out, no bus collision occurs because no two masters can assert SDA at exactly the same time.

If SCL goes from high to low before the BRG times out and SDA has not already been asserted, a bus collision occurs. In this case, another master is attempting to transmit a data '1' during the Repeated Start condition (refer to Figure 27-29).

If, at the end of the BRG time out, both SCL and SDA are still high, the SDA pin is driven low and the BRG is reloaded and begins counting. At the end of the count, regardless of the status of the SCL pin, the SCL pin is driven low and the Repeated Start condition is complete.

FIGURE 27-28: BUS COLLISION DURING A REPEATED START CONDITION (CASE 1)

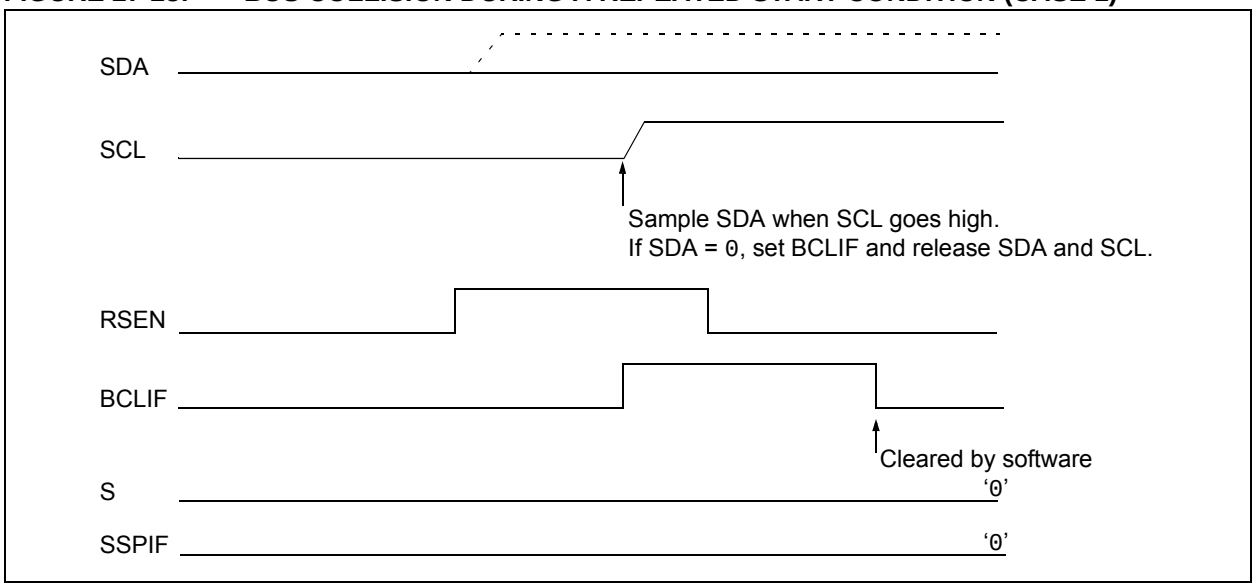
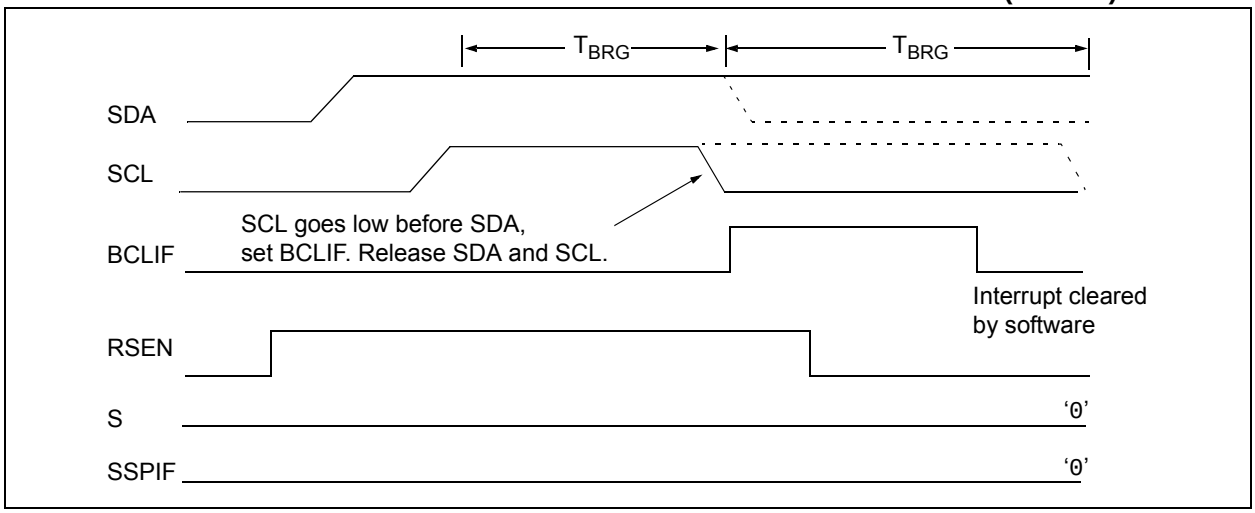


FIGURE 27-29: BUS COLLISION DURING A REPEATED START CONDITION (CASE 2)



27.5.13.3 Bus Collision During a Stop Condition

Bus collision occurs during a Stop condition if:

a) After the SDA pin has been deasserted and allowed to float high, SDA is sampled low after the BRG has timed out.

b) After the SCL pin is deasserted, SCL is sampled low before SDA goes high.

The Stop condition begins with SDA asserted low. When SDA is sampled low, the SCL pin is allowed to float. When the pin is sampled high (clock arbitration), the Baud Rate Generator is loaded with SSPADD and counts down to 0. After the BRG times out, SDA is sampled. If SDA is sampled low, a bus collision has occurred. This is due to another master attempting to drive a data '0' (Figure 27-30). If the SCL pin is sampled low before SDA is allowed to float high, a bus collision occurs. This is another case of another master attempting to drive a data '0' (Figure 27-31).

FIGURE 27-30: BUS COLLISION DURING A STOP CONDITION (CASE 1)

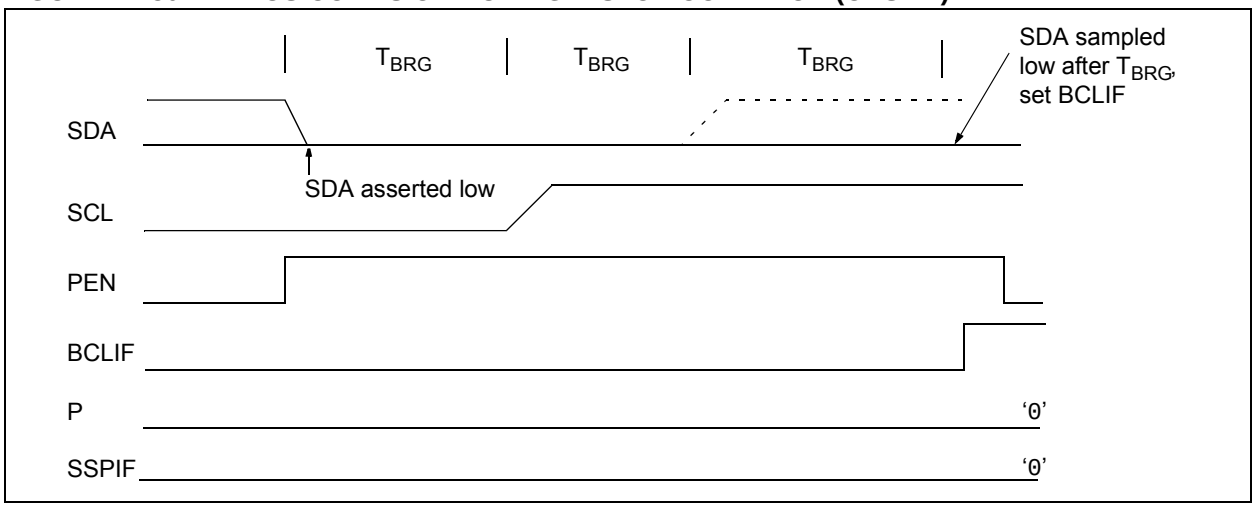


FIGURE 27-31: BUS COLLISION DURING A STOP CONDITION (CASE 2)

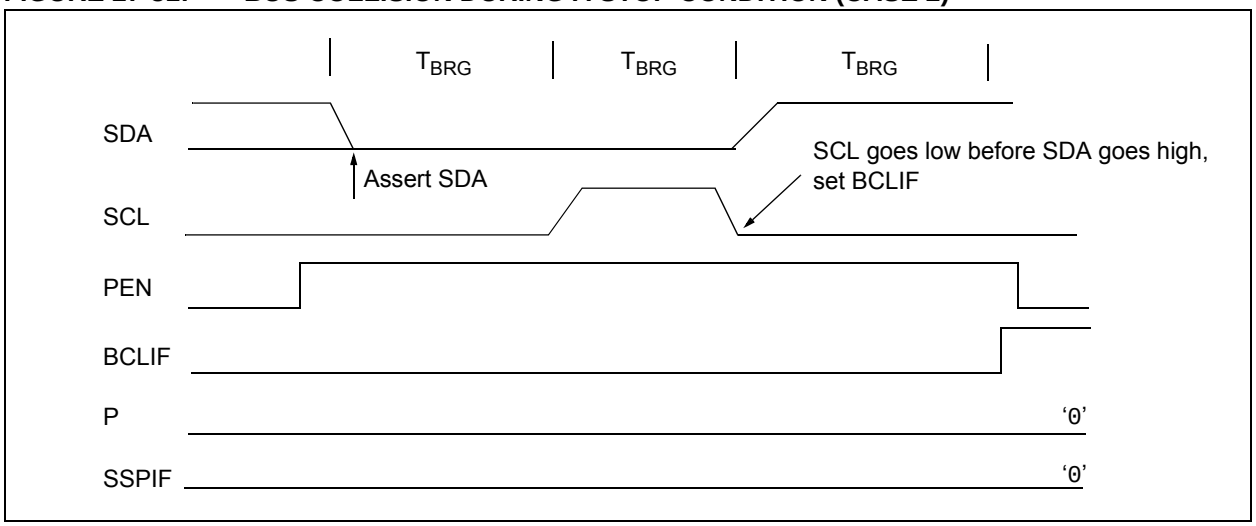


TABLE 27-2: SUMMARY OF REGISTERS ASSOCIATED WITH I²C OPERATION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on Page:
INTCON	GIE	PEIE	T0IE	INTE	IOCE	T0IF	INTF	IOCF	102
PIE1	TXIE	RCIE	BCLIE	SSPIE	CC2IE	CC1IE	TMR2IE	TMR1IE	103
PIR1	TXIF	RCIF	BCLIF	SSPIF	CC2IF	CC1IF	TMR2IF	TMR1IF	105
TRISGPA	TRISA7	TRISA6	TRISA5	—	TRISA3	TRISA2	TRISA1	TRISA0	119
TRISGPB	TRISB7	TRISB6	TRISB5	TRISB4	—	—	TRISB1	TRISB0	125
SSPADD	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0	215
SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								175*
SSPCON1	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	212
SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	213
SSPCON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	214
SSPMSK1	MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0	215
SSPSTAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	211
SSPMSK2	MSK27	MSK26	MSK25	MSK24	MSK23	MSK22	MSK21	MSK20	216
SSPADD2	ADD27	ADD26	ADD25	ADD24	ADD23	ADD22	ADD21	ADD20	216

Legend: – = unimplemented, read as '0'. Shaded cells are not used by the MSSP module in I²C mode.

* Page provides register information.

27.6 Baud Rate Generator

The MSSP module has a Baud Rate Generator available for clock generation in the I^2C Master mode. The Baud Rate Generator (BRG) reload value is placed in the SSPADD register. When a write occurs to SSPBUF, the Baud Rate Generator will automatically begin counting down.

Once the given operation is complete, the internal clock will automatically stop counting and the clock pin will remain in its last state.

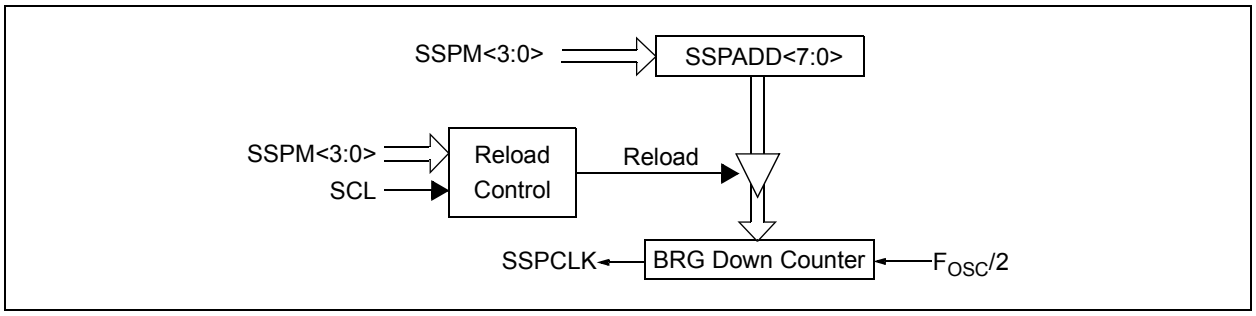
An internal signal “Reload” in Figure 27-32 triggers the value from SSPADD to be loaded into the BRG counter. This occurs twice for each oscillation of the module clock line. The logic dictating when the reload signal is asserted depends on the mode the MSSP is being operated in.

Table 27-3 demonstrates clock rates based on instruction cycles and the BRG value loaded into SSPADD.

EQUATION 27-1:

$$F_{CLOCK} = \frac{F_{OSC}}{(SSPADD + 1)(4)}$$

FIGURE 27-32: BAUD RATE GENERATOR BLOCK DIAGRAM



Note:	Values of 0x00, 0x01 and 0x02 are not valid for SSPADD when used as a Baud Rate Generator for I^2C. This is an implementation limitation.

TABLE 27-3: MSSP CLOCK RATE W/BRG

F_{OSC}	F_{CY}	BRG Value	F_{CLOCK} (2 Rollovers of BRG)
8 MHz	2 MHz	04h	400 kHz(1)
8 MHz	2 MHz	0Bh	166 kHz
8 MHz	2 MHz	13h	100 kHz

Note 1: The I^2C interface does not conform to the 400 kHz I^2C specification (which applies to rates greater than 100 kHz) in all details, but may be used with care where higher rates are required by the application.

REGISTER 27-1: SSPSTAT: SSP STATUS REGISTER

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
bit 7							bit 0

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

bit 7 **SMP:** Data Input Sample bit
1 = Slew rate control disabled for standard-speed mode (100 kHz and 1 MHz)
0 = Slew rate control enabled for high-speed mode (400 kHz)

bit 6 **CKE:** Clock Edge Select bit
1 = Enable input logic so that thresholds are compliant with SM bus specification
0 = Disable SM bus specific inputs

bit 5 **D/$\overline{A}$:** Data/$\overline{\text{Address}}$ bit
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4 **P:** Stop bit
(This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a Stop bit has been detected last (this bit is '0' on Reset)
0 = Stop bit was not detected last

bit 3 **S:** Start bit
(This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a Start bit has been detected last (this bit is '0' on Reset)
0 = Start bit was not detected last

bit 2 **R/$\overline{W}$:** Read/$\overline{\text{Write}}$ bit information
This bit holds the R/$\overline{W}$ bit information following the last address match. This bit is only valid from the address match to the next Start bit, Stop bit, or not $\overline{\text{ACK}}$ bit.
In I^2C Slave mode:
1 = Read
0 = Write
In I^2C Master mode:
1 = Transmit is in progress
0 = Transmit is not in progress
OR-ing this bit with SEN, RSEN, PEN, RCEN or ACKEN will indicate if the MSSP is in Idle mode.

bit 1 **UA:** Update Address bit (10-bit I^2C mode only)
1 = Indicates that the user needs to update the address in the SSPADD register
0 = Address does not need to be updated

bit 0 **BF:** Buffer Full status bit
Receive:
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty
Transmit:
1 = Data transmit in progress (does not include the ACK and Stop bits), SSPBUF is full
0 = Data transmit complete (does not include the ACK and Stop bits), SSPBUF is empty

REGISTER 27-2: SSPCON1: SSP CONTROL REGISTER 1

R/C/HS-0	R/C/HS-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>			
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n = Value at POR
'1' = Bit is set	'0' = Bit is cleared	HS = Bit is set by hardware C = User cleared

bit 7 **WCOL:** Write Collision Detect bit

Master mode:

1 = A write to the SSPBUF register was attempted while the I^2C conditions were not valid for a transmission to be started

0 = No collision

Slave mode:

1 = The SSPBUF register is written while it is still transmitting the previous word (must be cleared in software)

0 = No collision

bit 6 **SSPOV:** Receive Overflow Indicator bit(1)

1 = A byte is received while the SSPBUF register is still holding the previous byte. SSPOV is a "don't care" in Transmit mode (must be cleared in software).

0 = No overflow

bit 5 **SSPEN:** Synchronous Serial Port Enable bit

In both modes, when enabled, these pins must be properly configured as input or output

1 = Enables the serial port and configures the SDA and SCL pins as the source of the serial port pins(2)

0 = Disables serial port and configures these pins as I/O port pins

bit 4 **CKP:** Clock Polarity Select bit

In I^2C Slave mode: SCL release control

1 = Enable clock

0 = Holds clock low (clock stretch). (Used to ensure data setup time.)

In I^2C Master mode: Unused in this mode

bit 3-0 **SSPM<3:0>:** Synchronous Serial Port Mode Select bits

0000 = Reserved
0001 = Reserved
0010 = Reserved
0011 = Reserved
0100 = Reserved
0101 = Reserved
0110 = I^2C Slave mode, 7-bit address
0111 = I^2C Slave mode, 10-bit address
1000 = I^2C Master mode, clock = $F_{OSC}/(4 * (SSPADD+1))$(3)
1001 = Reserved
1010 = Reserved
1011 = I^2C Firmware-Controlled Master mode (Slave idle)
1100 = Reserved
1101 = Reserved
1110 = I^2C Slave mode, 7-bit address with Start and Stop bit interrupts enabled
1111 = I^2C Slave mode, 10-bit address with Start and Stop bit interrupts enabled

Note 1: In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.

2: When enabled, the SDA and SCL pins must be configured as inputs.

3: SSPADD values of 0, 1 or 2 are not supported for I^2C mode.

REGISTER 27-3: SSPCON2: SSP CONTROL REGISTER 2

R/W-0/0	R-0/0	R/W-0/0	R/S/HS-0/0	R/S/HS-0/0	R/S/HS-0/0	R/S/HS-0/0	R/W/HS-0/0
GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	H = Bit is set by hardware S = User set

bit 7 **GCEN:** General Call Enable bit (in I^2C Slave mode only)
1 = Enable interrupt when a general call address (0x00 or 00h) is received in the SSPSR register
0 = General call address disabled

bit 6 **ACKSTAT:** Acknowledge Status bit (in I^2C mode only)
1 = Acknowledge was not received
0 = Acknowledge was received

bit 5 **ACKDT:** Acknowledge Data bit (in I^2C mode only)
In Receive mode:
Value transmitted when the user initiates an Acknowledge sequence at the end of a receive
1 = Not Acknowledge
0 = Acknowledge

bit 4 **ACKEN:** Acknowledge Sequence Enable bit (in I^2C Master mode only)
In Master Receive mode:
1 = Initiate Acknowledge sequence on SDA and SCL pins and transmit ACKDT data bit. Automatically cleared by hardware.
0 = Acknowledge sequence idle

bit 3 **RCEN:** Receive Enable bit (in I^2C Master mode only)
1 = Enables Receive mode for I^2C
0 = Receive idle

bit 2 **PEN:** Stop Condition Enable bit (in I^2C Master mode only)
SCK Release Control:
1 = Initiate Stop condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Stop condition idle

bit 1 **RSEN:** Repeated Start Condition Enabled bit (in I^2C Master mode only)
1 = Initiate Repeated Start condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Repeated Start condition idle

bit 0 **SEN:** Start Condition Enabled bit (in I^2C Master mode only)
In Master mode:
1 = Initiate Start condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Start condition idle
In Slave mode:
1 = Clock stretching is enabled for both Slave Transmit and Slave Receive (stretch enabled)
0 = Clock stretching is disabled

Note 1: For bits ACKEN, RCEN, PEN, RSEN, SEN: If the I^2C module is not in the Idle mode, this bit may not be set (no spooling) and the SSPBUF may not be written (or writes to the SSPBUF are disabled).

REGISTER 27-4: SSPCON3: SSP CONTROL REGISTER 3

R-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **ACKTIM:** Acknowledge Time status bit (I^2C mode only)(2)
- 1 = Indicates the I^2C bus is in an Acknowledge sequence, set on 8th falling edge of SCL clock
- 0 = Not an Acknowledge sequence, cleared on 9th rising edge of SCL clock

bit 6 **PCIE**: Stop Condition Interrupt Enable bit (I^2C mode only)
- 1 = Enable interrupt on detection of Stop condition
- 0 = Stop detection interrupts are disabled(1)

bit 5 **SCIE**: Start Condition Interrupt Enable bit (I^2C mode only)
- 1 = Enable interrupt on detection of Start or Restart conditions
- 0 = Start detection interrupts are disabled(1)

bit 4 **BOEN:** Buffer Overwrite Enable bit

In I^2C Master mode:
This bit is ignored.

In I^2C Slave mode:
- 1 = SSPBUF is updated and $\overline{ACK}$ is generated for a received address/data byte, ignoring the state of the SSPOV bit only if the BF bit = 0.
- 0 = SSPBUF is only updated when SSPOV is clear.

bit 3 **SDAHT:** SDA Hold Time Selection bit
- 1 = Minimum of 300 ns hold time on SDA after the falling edge of SCL
- 0 = Minimum of 100 ns hold time on SDA after the falling edge of SCL

bit 2 **SBCDE:** Slave Mode Bus Collision Detect Enable bit (I^2C Slave mode only)

If, on the rising edge of SCL, SDA is sampled low when the module outputs a high state, the BCLIF bit in the PIR1 register is set and bus goes idle.
- 1 = Enable slave bus collision interrupts
- 0 = Slave bus collision interrupts are disabled

bit 1 **AHEN:** Address Hold Enable bit (I^2C Slave mode only)
- 1 = Following the 8th falling edge of SCL for a matching received address byte; CKP bit in the SSPCON1 register will be cleared and the SCL will be held low.
- 0 = Address holding is disabled

bit 0 **DHEN:** Data Hold Enable bit (I^2C Slave mode only)
- 1 = Following the 8th falling edge of SCL for a received data byte; slave hardware clears the CKP bit in the SSPCON1 register and SCL is held low.
- 0 = Data holding is disabled

Note 1: This bit has no effect in Slave modes where Start and Stop condition detection is explicitly listed as enabled.

2: The ACKTIM status bit is only active when the AHEN bit or DHEN bit is set.

REGISTER 27-5: SSPMSK1: SSP MASK REGISTER 1

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
MSK<7:0>							
bit 7							bit 0

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

bit 7-1 **MSK<7:1>:** Mask bits
1 = The received address bit n is compared to SSPADD<n> to detect I^2C address match
0 = The received address bit n is not used to detect I^2C address match

bit 0 **MSK<0>:** Mask bit for I^2C Slave mode, 10-bit Address
I^2C Slave mode, 10-bit address (SSPM<3:0> = 0111 or 1111):
1 = The received address bit 0 is compared to SSPADD<0> to detect I^2C address match
0 = The received address bit 0 is not used to detect I^2C address match I^2C Slave mode, 7-bit address, the bit is ignored

REGISTER 27-6: SSPADD: MSSP ADDRESS AND BAUD RATE REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADD<7:0>							
bit 7							bit 0

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

Master mode:

bit 7-0 **ADD<7:0>:** Baud Rate Clock Divider bits
SCL pin clock period = ((ADD<7:0> + 1) * 4)/F_{OSC}

10-Bit Slave mode — Most Significant Address byte:

bit 7-3 **Not used:** Unused for Most Significant Address byte. Bit state of this register is a "don't care". Bit pattern sent by master is fixed by I^2C specification and must be equal to '11110'. However, those bits are compared by hardware and are not affected by the value in this register.

bit 2-1 **ADD<2:1>:** Two Most Significant bits of 10-bit address.

bit 0 **Not used:** Unused in this mode. Bit state is a "don't care".

10-Bit Slave mode — Least Significant Address byte:

bit 7-0 **ADD<7:0>:** Eight Least Significant bits of 10-bit address

7-Bit Slave mode:

bit 7-1 **ADD<7:1>:** 7-bit address

bit 0 **Not used:** Unused in this mode. Bit state is a "don't care".

REGISTER 27-7: SSPMSK2: SSP MASK REGISTER 2

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
MSK2<7:0>							
bit 7							bit 0

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

bit 7-1 **MSK2<7:1>:** Mask bits
1 = The received address bit n is compared to SSPADD2<n> to detect I^2C address match
0 = The received address bit n is not used to detect I^2C address match

bit 0 **MSK2<0>:** Mask bit for I^2C Slave mode, 10-bit Address
I^2C Slave mode, 10-bit address (SSPM<3:0> = 0111 or 1111):
1 = The received address bit 0 is compared to SSPADD2<0> to detect I^2C address match
0 = The received address bit 0 is not used to detect I^2C address match I^2C Slave mode, 7-bit address, the bit is ignored

REGISTER 27-8: SSPADD2: MSSP ADDRESS 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADD2<7:0>							
bit 7							bit 0

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

Master mode:

bit 7-0 **ADD2<7:0>:** Baud Rate Clock Divider bits
SCL pin clock period = ((ADD<7:0> + 1) * 4)/F_{OSC}

10-Bit Slave mode — Most Significant Address byte:

bit 7-3 **Not used:** Unused for Most Significant Address byte. Bit state of this register is a "don't care". Bit pattern sent by master is fixed by I^2C specification and must be equal to '11110'. However, those bits are compared by hardware and are not affected by the value in this register.

bit 2-1 **ADD2<2:1>:** Two Most Significant bits of 10-bit address

bit 0 **Not used:** Unused in this mode. Bit state is a "don't care".

10-Bit Slave mode — Least Significant Address byte:

bit 7-0 **ADD2<7:0>:** Eight Least Significant bits of 10-bit address

7-Bit Slave mode:

bit 7-1 **ADD2<7:1>:** 7-bit address

bit 0 **Not used:** Unused in this mode. Bit state is a "don't care".

28.0 INSTRUCTION SET SUMMARY

The MCP19122/3 instruction set is highly orthogonal and comprises three basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

Each instruction is a 14-bit word divided into an **opcode,** which specifies the instruction type, and one or more **operands,** which further specify the operation of the instruction. The formats for each of the categories is presented in Figure 28-1, while the various opcode fields are summarized in Table 28-1.

Table 28-2 lists the instructions recognized by the MPASM™ assembler.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, the result is placed in the file register specified in the instruction.

For **bit-oriented** instructions, 'b' represents a bit field designator, which selects the bit affected by the operation, while 'f' represents the address of the file in which the bit is located.

For **literal and control** operations, 'k' represents an 8-bit or 11-bit constant, or literal value.

One instruction cycle consists of four oscillator periods; for an oscillator frequency of 4 MHz, this gives a normal instruction execution time of 1 µs. All instructions are executed within a single instruction cycle, unless a conditional test is true, or the program counter is changed as a result of an instruction. When this occurs, the execution takes two instruction cycles, with the second cycle executed as a NOP.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

28.1 Read-Modify-Write Operations

Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (RMW) operation. The register is read, the data is modified, and the result is stored according to either the instruction or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

For example, a CLRF PORTA instruction will read PORTGPA, clear all the data bits, then write the result back to PORTGPA. This example would have the unintended consequence of clearing the condition that set the IOCF flag.

TABLE 28-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1). The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1.
PC	Program Counter
$\overline{TO}$	Time-Out bit
C	Carry bit
DC	Digit carry bit
Z	Zero bit
$\overline{PD}$	Power-Down bit

FIGURE 28-1: GENERAL FORMAT FOR INSTRUCTIONS

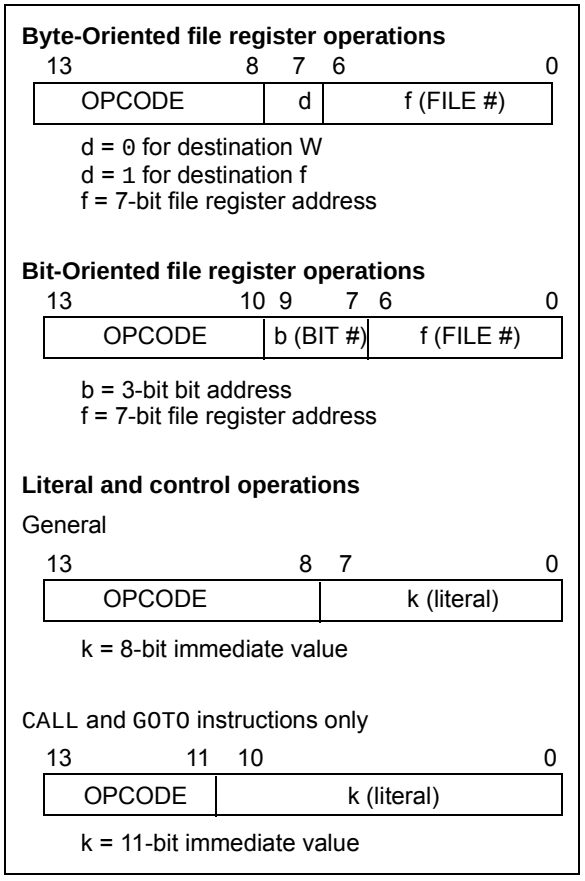


TABLE 28-2: MCP19122/3 INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS						
ADDWF	f, d	Add W and f	1	00 0111 dfff ffff	C, DC, Z	1, 2
ANDWF	f, d	AND W with f	1	00 0101 dfff ffff	Z	1, 2
CLRF	f	Clear f	1	00 0001 lfff ffff	Z	2
CLRW	–	Clear W	1	00 0001 0xxx xxxx	Z	
COMF	f, d	Complement f	1	00 1001 dfff ffff	Z	1, 2
DECF	f, d	Decrement f	1	00 0011 dfff ffff	Z	1, 2
DECFSZ	f, d	Decrement f, Skip if 0	1**(2)**	00 1011 dfff ffff		1, 2, 3
INCF	f, d	Increment f	1	00 1010 dfff ffff	Z	1, 2
INCFSZ	f, d	Increment f, Skip if 0	1**(2)**	00 1111 dfff ffff		1, 2, 3
IORWF	f, d	Inclusive OR W with f	1	00 0100 dfff ffff	Z	1, 2
MOVF	f, d	Move f	1	00 1000 dfff ffff	Z	1, 2
MOVWF	f	Move W to f	1	00 0000 lfff ffff		
NOP	–	No Operation	1	00 0000 0xx0 0000		
RLF	f, d	Rotate Left f through Carry	1	00 1101 dfff ffff	C	1, 2
RRF	f, d	Rotate Right f through Carry	1	00 1100 dfff ffff	C	1, 2
SUBWF	f, d	Subtract W from f	1	00 0010 dfff ffff	C, DC, Z	1, 2
SWAPF	f, d	Swap nibbles in f	1	00 1110 dfff ffff		1, 2
XORWF	f, d	Exclusive OR W with f	1	00 0110 dfff ffff	Z	1, 2
BIT-ORIENTED FILE REGISTER OPERATIONS						
BCF	f, b	Bit Clear f	1	01 00bb bfff ffff		1, 2
BSF	f, b	Bit Set f	1	01 01bb bfff ffff		1, 2
BTFSC	f, b	Bit Test f, Skip if Clear	1 **(2)**	01 10bb bfff ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 **(2)**	01 11bb bfff ffff		3
LITERAL AND CONTROL OPERATIONS						
ADDLW	k	Add literal and W	1	11 111x kkkk kkkk	C, DC, Z	
ANDLW	k	AND literal with W	1	11 1001 kkkk kkkk	Z	
CALL	k	Call Subroutine	2	10 0kkk kkkk kkkk		
CLRWDT	–	Clear Watchdog Timer	1	00 0000 0110 0100	$\overline{TO}$, $\overline{PD}$	
GOTO	k	Go to address	2	10 1kkk kkkk kkkk		
IORLW	k	Inclusive OR literal with W	1	11 1000 kkkk kkkk	Z	
MOVLW	k	Move literal to W	1	11 00xx kkkk kkkk		
RETFIE	–	Return from interrupt	2	00 0000 0000 1001		
RETLW	k	Return with literal in W	2	11 01xx kkkk kkkk		
RETURN	–	Return from Subroutine	2	00 0000 0000 1000		
SLEEP	–	Go into Standby mode	1	00 0000 0110 0011	$\overline{TO}$, $\overline{PD}$	
SUBLW	k	Subtract W from literal	1	11 110x kkkk kkkk	C, DC, Z	
XORLW	k	Exclusive OR literal with W	1	11 1010 kkkk kkkk	Z	

Note 1: When an I/O register is modified as a function of itself (e.g., `MOVF PORTA, 1`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 module.

3: If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a `NOP`.

28.2 Instruction Descriptions

ADDLW	Add literal and W
Syntax:	[*label*] ADDLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) + k → (W)
Status Affected:	C, DC, Z
Description:	The contents of the W register are added to the eight-bit literal 'k' and the result is placed in the W register.

ADDWF	Add W and f
Syntax:	[*label*] ADDWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) + (f) → (destination)
Status Affected:	C, DC, Z
Description:	Add the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

ANDLW	AND literal with W
Syntax:	[*label*] ANDLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) .AND. (k) → (W)
Status Affected:	Z
Description:	The contents of W register are AND'ed with the eight-bit literal 'k'. The result is placed in the W register.

ANDWF	AND W with f
Syntax:	[*label*] ANDWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) .AND. (f) → (destination)
Status Affected:	Z
Description:	AND the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

BCF	Bit Clear f
Syntax:	[*label*] BCF f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	0 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is cleared.

BSF	Bit Set f
Syntax:	[*label*] BSF f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	1 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is set.

BTFSC	Bit Test f, Skip if Clear
Syntax:	[*label*] BTFSC f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	skip if (f<b>) = 0
Status Affected:	None
Description:	If bit 'b' in register 'f' is '1', the next instruction is executed. If bit 'b' in register 'f' is '0', the next instruction is discarded, and a NOP is executed instead, making this a two-cycle instruction.

BTFSS	Bit Test f, Skip if Set
Syntax:	[*label*] BTFSS f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b < 7
Operation:	skip if (f<b>) = 1
Status Affected:	None
Description:	If bit 'b' in register 'f' is '0', the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a two-cycle instruction.

CALL	Call Subroutine
Syntax:	[*label*] CALL k
Operands:	0 ≤ k ≤ 2047
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<4:3>) → PC<12:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The 11-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two-cycle instruction.

CLRF	Clear f
Syntax:	[*label*] CLRF f
Operands:	0 ≤ f ≤ 127
Operation:	00h → (f) 1 → Z
Status Affected:	Z
Description:	The contents of register 'f' are cleared and the Z bit is set.

CLRW	Clear W
Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h → (W) 1 → Z
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

CLRWDT	Clear Watchdog Timer
Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h → WDT 0 → WDT prescaler, 1 → $\overline{TO}$ 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

COMF	Complement f
Syntax:	[*label*] COMF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	($\bar{f}$) → (destination)
Status Affected:	Z
Description:	The contents of register 'f' are complemented. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

DECF	Decrement f
Syntax:	[*label*] DECF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f) - 1 → (destination)
Status Affected:	Z
Description:	Decrement register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

DECFSZ	Decrement f, Skip if 0
Syntax:	[*label*] DECFSZ f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - 1 → (destination); skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are decremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'. If the result is '1', the next instruction is executed. If the result is '0', then a NOP is executed instead, making it a two-cycle instruction.

GOTO	Unconditional Branch
Syntax:	[*label*] GOTO k
Operands:	$0 \le k \le 2047$
Operation:	k → PC<10:0> PCLATH<4:3> → PC<12:11>
Status Affected:	None
Description:	GOTO is an unconditional branch. The eleven-bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two-cycle instruction.

INCF	Increment f
Syntax:	[*label*] INCF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) + 1 → (destination)
Status Affected:	Z
Description:	The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

INCFSZ	Increment f, Skip if 0
Syntax:	[*label*] INCFSZ f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) + 1 → (destination), skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'. If the result is '1', the next instruction is executed. If the result is '0', a NOP is executed instead, making it a two-cycle instruction.

IORLW	Inclusive OR literal with W
Syntax:	[*label*] IORLW k
Operands:	$0 \le k \le 255$
Operation:	(W) .OR. k → (W)
Status Affected:	Z
Description:	The contents of the W register are OR'ed with the 8-bit literal 'k'. The result is placed in the W register.

IORWF	Inclusive OR W with f
Syntax:	[*label*] IORWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) .OR. (f) → (destination)
Status Affected:	Z
Description:	Inclusive OR the W register with register 'f'. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

MOVF	Move f
Syntax:	[*label*] MOVF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) → (dest)
Status Affected:	Z
Description:	The contents of register 'f' is moved to a destination dependent upon the status of 'd'. If d = 0, destination is W register. If d = 1, the destination is file register 'f' itself. d = 1 is useful to test a file register since Status flag Z is affected.
Words:	1
Cycles:	1
Example:	MOVF FSR, 0 After Instruction W = value in FSR register Z = 1

MOVLW	Move literal to W
Syntax:	[*label*] MOVLW k
Operands:	$0 \le k \le 255$
Operation:	k → (W)
Status Affected:	None
Description:	The eight-bit literal 'k' is loaded into W register. The "don't cares" will assemble as '0's.
Words:	1
Cycles:	1
Example:	MOVLW 0x5A After Instruction W = 0x5A

MOVWF	Move W to f
Syntax:	[*label*] MOVWF f
Operands:	$0 \le f \le 127$
Operation:	(W) → (f)
Status Affected:	None
Description:	Move data from W register to register 'f'.
Words:	1
Cycles:	1
Example:	MOVW OPTION F Before Instruction OPTION = 0xFF W = 0x4F After Instruction OPTION = 0x4F W = 0x4F

NOP	No Operation
Syntax:	[*label*] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.
Words:	1
Cycles:	1
Example:	NOP

RETFIE	Return from Interrupt
Syntax:	[*label*] RETFIE
Operands:	None
Operation:	TOS → PC, 1 → GIE
Status Affected:	None
Description:	Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two-cycle instruction.
Words:	1
Cycles:	2
Example:	RETFIE After Interrupt PC = TOS GIE = 1

RETLW	Return with literal in W
Syntax:	[*label*] RETLW k
Operands:	0 ≤ k ≤ 255
Operation:	k → (W); TOS → PC
Status Affected:	None
Description:	The W register is loaded with the 8-bit literal ‘k’. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.
Words:	1
Cycles:	2
Example:	CALL TABLE;W contains ;table offset ;value GOTO DONE TABLE • • ADDWF PC ;W = offset RETLW k1 ;Begin table RETLW k2 ; • • • RETLW kn ;End of table DONE Before Instruction W = 0x07 After Instruction W = value of k8

RETURN	Return from Subroutine
Syntax:	[*label*] RETURN
Operands:	None
Operation:	TOS → PC
Status Affected:	None
Description:	Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.

RLF **Rotate Left f through Carry**

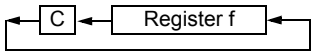
Syntax: [*label*] RLF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the left through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is stored back in register 'f'.



Words: 1

Cycles: 1

Example:

```
RLF     REG1,0
```

Before Instruction
```
REG1   =   1110
0110
C      =   0
```
After Instruction
```
REG1   =   1110
0110
W      =   1100
1100
C      =   1
```

RRF **Rotate Right f through Carry**

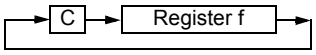
Syntax: [*label*] RRF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the right through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.



SLEEP **Enter Sleep mode**

Syntax: [*label*] SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT prescaler,
1 → $\overline{TO}$,
0 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Description: The power-down Status bit, $\overline{PD}$ is cleared. Time-out Status bit, $\overline{TO}$ is set. Watchdog Timer and its prescaler are cleared.
The processor is put into Sleep mode with the oscillator stopped.

SUBLW **Subtract W from literal**

Syntax: [*label*] SUBLW k

Operands: $0 \le k \le 255$

Operation: k - (W) → (W)

Status Affected: C, DC, Z

Description: The W register is subtracted (two's complement method) from the 8-bit literal 'k'. The result is placed in the W register.

Result	Condition
C = 0	W > k
C = 1	W ≤ k
DC = 0	W<3:0> > k<3:0>
DC = 1	W<3:0> ≤ k<3:0>

SUBWF	Subtract W from f
Syntax:	[*label*] SUBWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) - (W) → (destination)
Status Affected:	C, DC, Z
Description:	Subtract (two's complement method) W register from register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

C = 0	W > f
C = 1	W ≤ f
DC = 0	W<3:0> > f<3:0>
DC = 1	W<3:0> ≤ f<3:0>

SWAPF	Swap Nibbles in f
Syntax:	[*label*] SWAPF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f<3:0>) → (destination<7:4>), (f<7:4>) → (destination<3:0>)
Status Affected:	None
Description:	The upper and lower nibbles of register 'f' are exchanged. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed in register 'f'.

XORLW	Exclusive OR literal with W
Syntax:	[*label*] XORLW k
Operands:	$0 \le k \le 255$
Operation:	(W) .XOR. k → (W)
Status Affected:	Z
Description:	The contents of the W register are XOR'ed with the 8-bit literal 'k'. The result is placed in the W register.

XORWF	Exclusive OR W with f
Syntax:	[*label*] XORWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) .XOR. (f) → (destination)
Status Affected:	Z
Description:	Exclusive OR the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

NOTES:

29.0 IN-CIRCUIT SERIAL PROGRAMMING™ (ICSP™)

ICSP programming allows customers to manufacture circuit boards with unprogrammed devices. Programming can be done after the assembly process, allowing the device to be programmed with the most recent firmware or a custom firmware. Five pins are needed for ICSP programming:

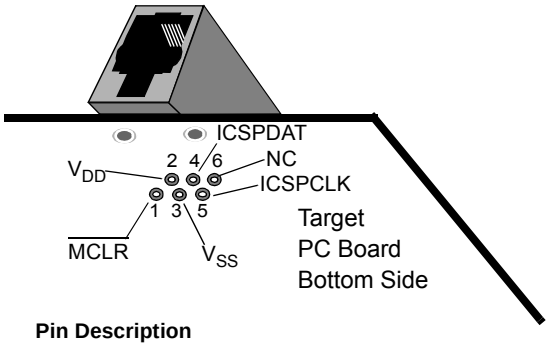
- ICSPCLK
- ICSPDAT
- $\overline{MCLR}$
- V_{DD}
- V_{SS}

In Program/Verify mode the Program Memory, User IDs and the Configuration Words are programmed through serial communications. The ICSPDAT pin is a bidirectional I/O used for transferring the serial data and the ICSPCLK pin is the clock input. The device is placed into a Program/Verify mode by holding the ICSPDAT and ICSPCLK pins low, while raising the $\overline{MCLR}$ pin from V_{IL} to V_{IHH}.

29.1 Common Programming Interfaces

Connection to a target device is typically done through an ICSP header. A commonly found connector on development tools is the RJ-11 in the 6P6C (6-pin, 6-connector) configuration. See Figure 29-1.

FIGURE 29-1: ICD RJ-11 STYLE CONNECTOR INTERFACE

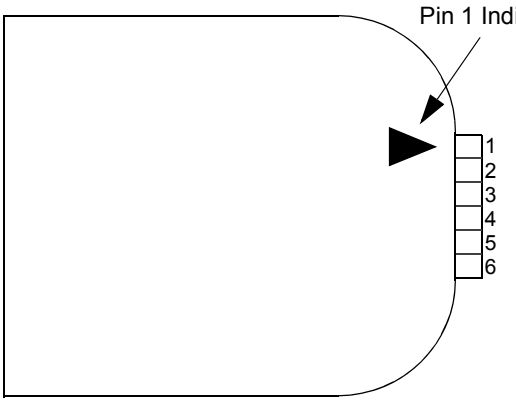


Pin Description

1	=	1 = $\overline{MCLR}$
2	=	2 = V_{DD}Target
3	=	3 = V_{SS} (ground)
4	=	4 = ICSPDAT
5	=	5 = ICSPCLK
6	=	6 = No Connect

Another connector often found in use with the PICkit™ programmers is a standard 6-pin header with 0.1 inch spacing. Refer to Figure 29-2.

FIGURE 29-2: PICkit-STYLE CONNECTOR INTERFACE



Pin Description*

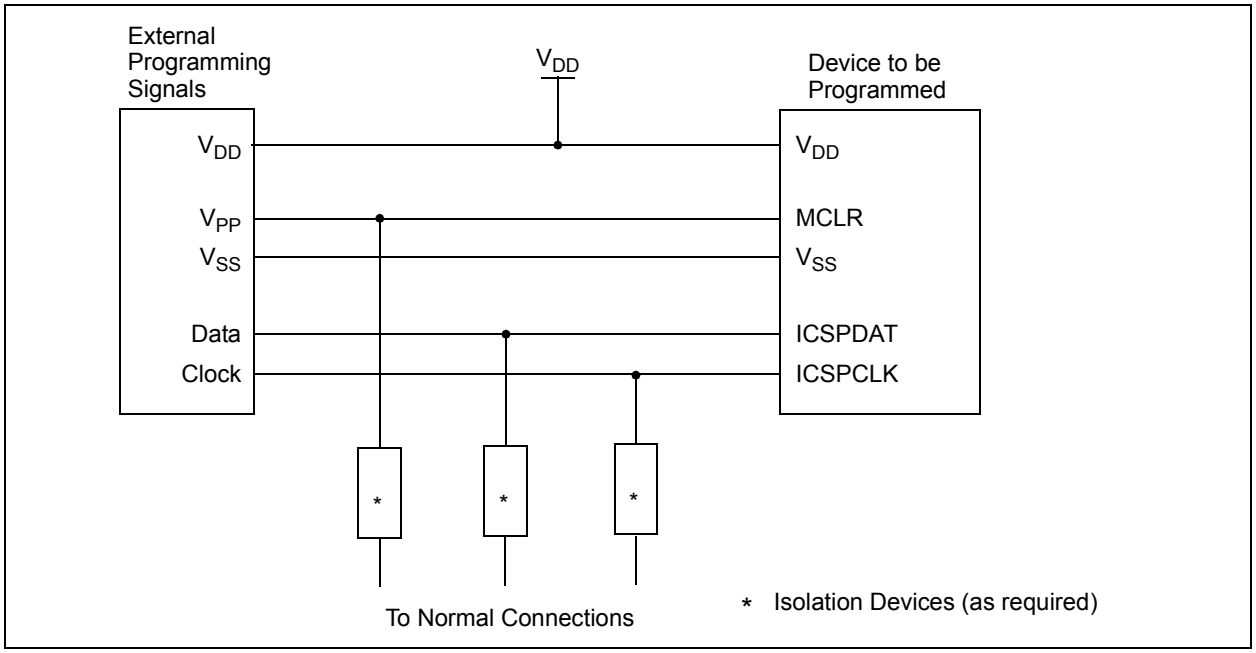
1	=	1 = $\overline{MCLR}$
2	=	2 = V_{DD}Target
3	=	3 = V_{SS} (ground)
4	=	4 = ICSPDAT
5	=	5 = ICSPCLK
6	=	6 = No Connect

* The 6-pin header (0.100" spacing) accepts 0.025" square pins.

For additional interface recommendations, refer to your specific device programmer manual prior to PCB design.

It is recommended that isolation devices be used to separate the programming pins from other circuitry. The type of isolation is highly dependent on the specific application and may include devices, such as resistors, diodes, or even jumpers. See Figure 29-3 for more information.

FIGURE 29-3: TYPICAL CONNECTION FOR ICSP PROGRAMMING



29.2 In-Circuit Debugger

In-circuit debugging requires access to the ICDCLK, ICDDATA and MCLR pins. These pins are only available on the MCP19123 device.

30.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICkit™ 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits and Starter Kits
- Third-party development tools

30.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

30.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16, and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

30.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

30.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

30.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

30.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

30.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradeable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to 3 meters) interconnection cables.

30.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

30.9 PICkit 3 In-Circuit Debugger/Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

30.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at V_{DDMIN} and V_{DDMAX} for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

30.11 Demonstration/Development Boards, Evaluation Kits and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC and flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

30.12 Third-Party Development Tools

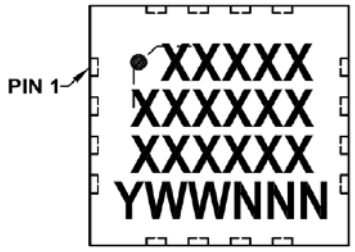
Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device programmers and gang programmers from companies such as SoftLog and CCS
- Software tools from companies such as Gimpel and Trace Systems
- Protocol analyzers from companies such as Saleae and Total Phase
- Demonstration boards from companies such as MikroElektronika, Digilent® and Olimex
- Embedded Ethernet solutions from companies such as EZ Web Lynx, WIZnet and IPLogika®

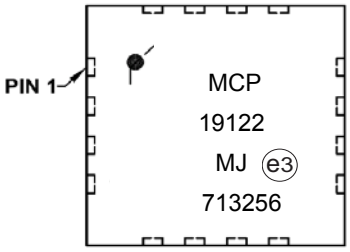
31.0 PACKAGING INFORMATION

31.1 Package Marking Information

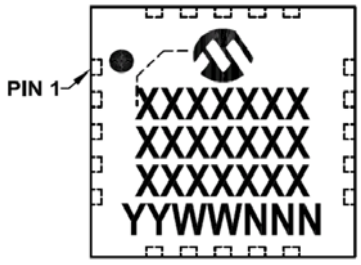
24-Lead QFN (4x4 mm) (**MCP19122** only)



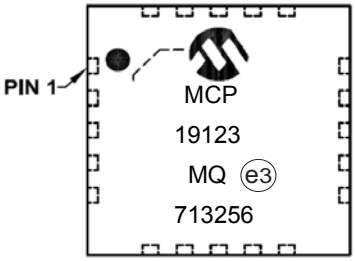
Example



28-Lead QFN (5x5 mm) (**MCP19123** only)



Example

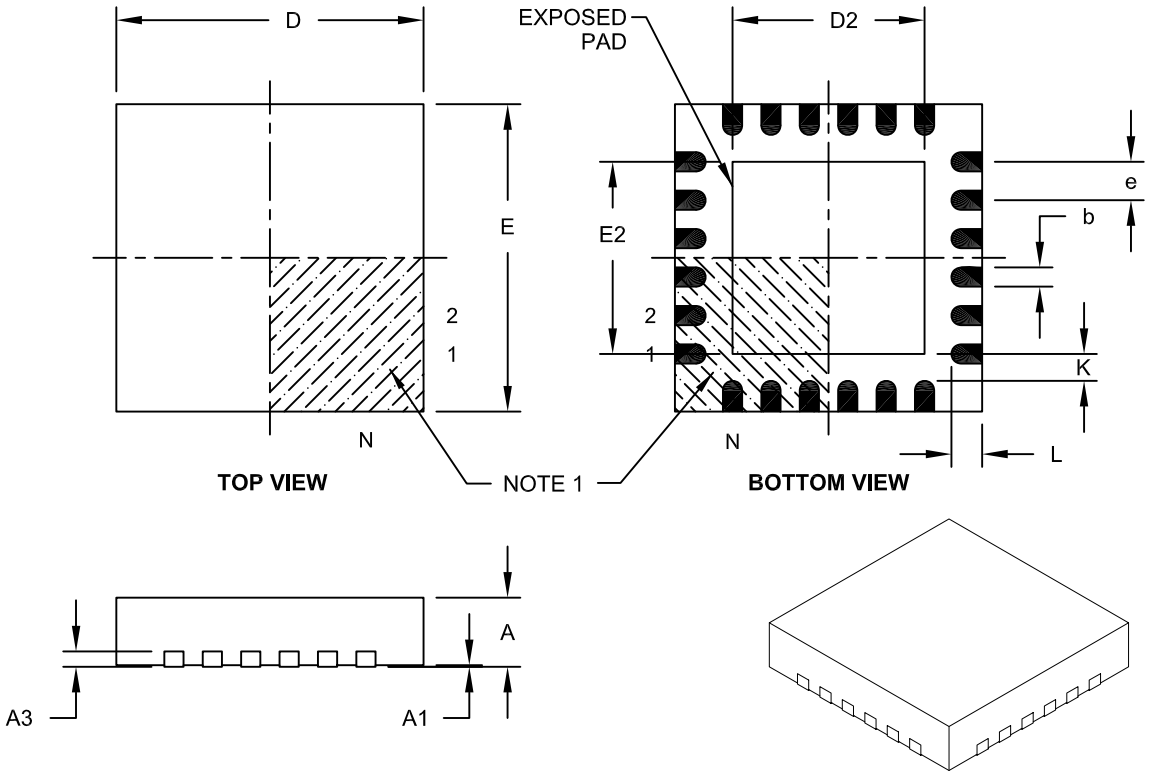


Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

24-Lead Plastic Quad Flat, No Lead Package (MJ) – 4x4x0.9 mm Body [QFN]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		24	
Pitch	e		0.50 BSC	
Overall Height	A	0.80	0.85	0.90
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		4.00 BSC	
Exposed Pad Width	E2	2.40	2.50	2.60
Overall Length	D		4.00 BSC	
Exposed Pad Length	D2	2.40	2.50	2.60
Contact Width	b	0.20	0.25	0.30
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	-	-

Notes:

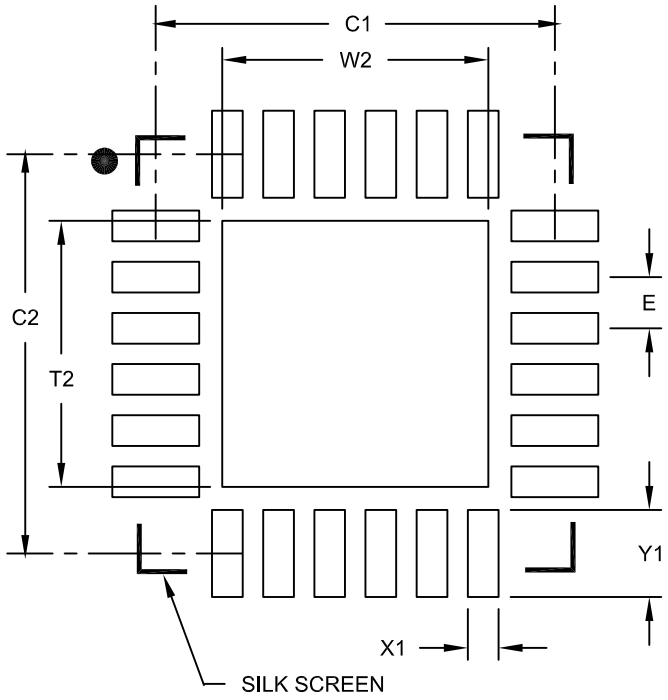
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-143A

24-Lead Plastic Quad Flat, No Lead Package (MJ) - 4x4 mm Body [QFN]

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		0.50 BSC	
Optional Center Pad Width	W2			2.60
Optional Center Pad Length	T2			2.60
Contact Pad Spacing	C1		3.90	
Contact Pad Spacing	C2		3.90	
Contact Pad Width	X1			0.30
Contact Pad Length	Y1			0.85

Notes:

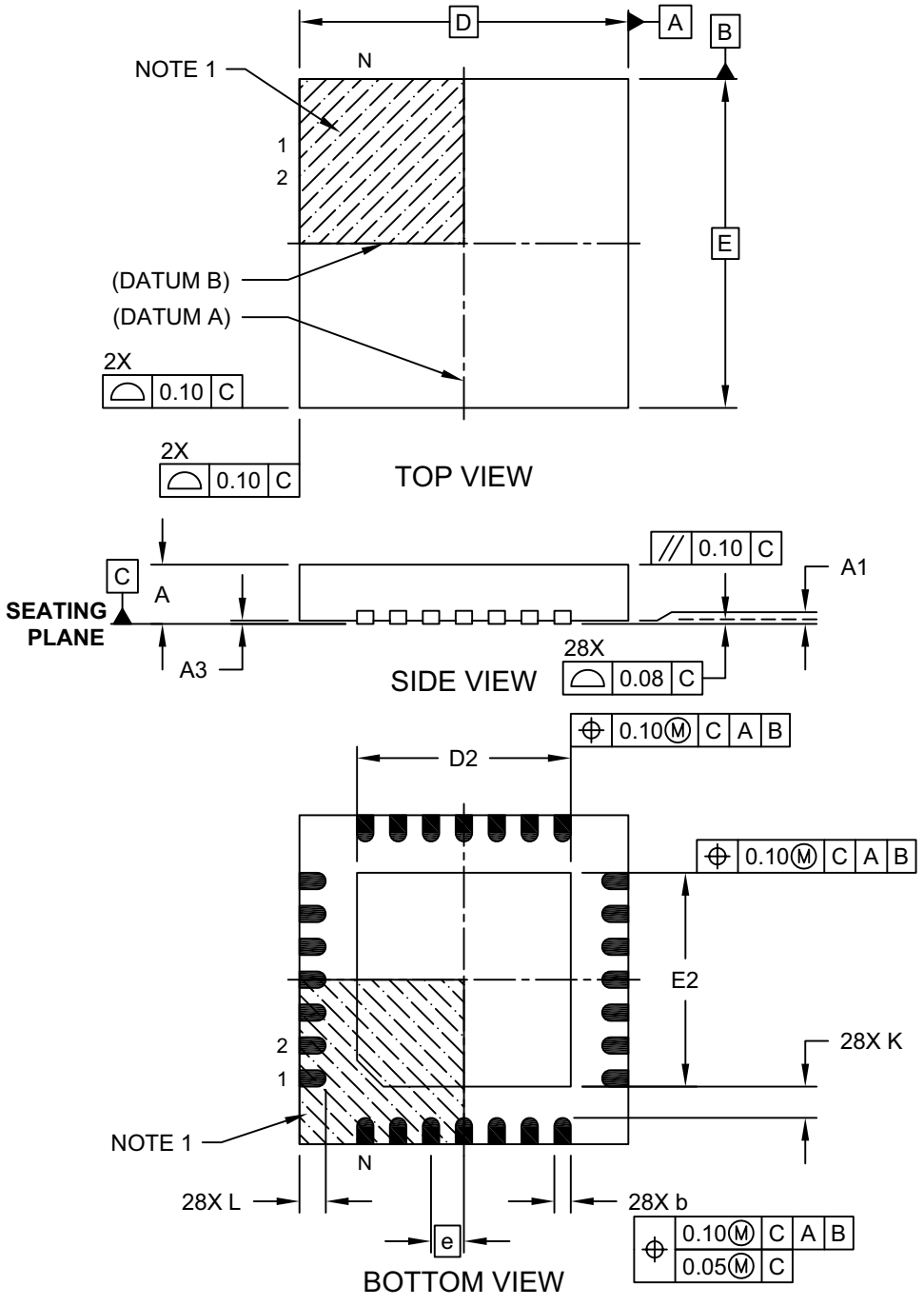
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2143B

28-Lead Plastic Quad Flat, No Lead Package (MQ) – 5x5x0.9 mm Body [QFN or VQFN]

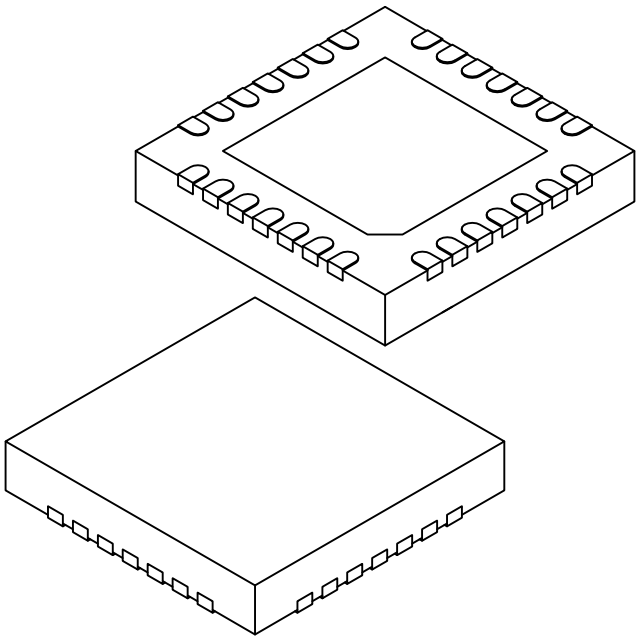
Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-140C Sheet 1 of 2

28-Lead Plastic Quad Flat, No Lead Package (MQ) – 5x5x0.9 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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		0.50 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		5.00 BSC	
Exposed Pad Width	E2	3.15	3.25	3.35
Overall Length	D		5.00 BSC	
Exposed Pad Length	D2	3.15	3.25	3.35
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.35	0.40	0.45
Contact-to-Exposed Pad	K	0.20	-	-

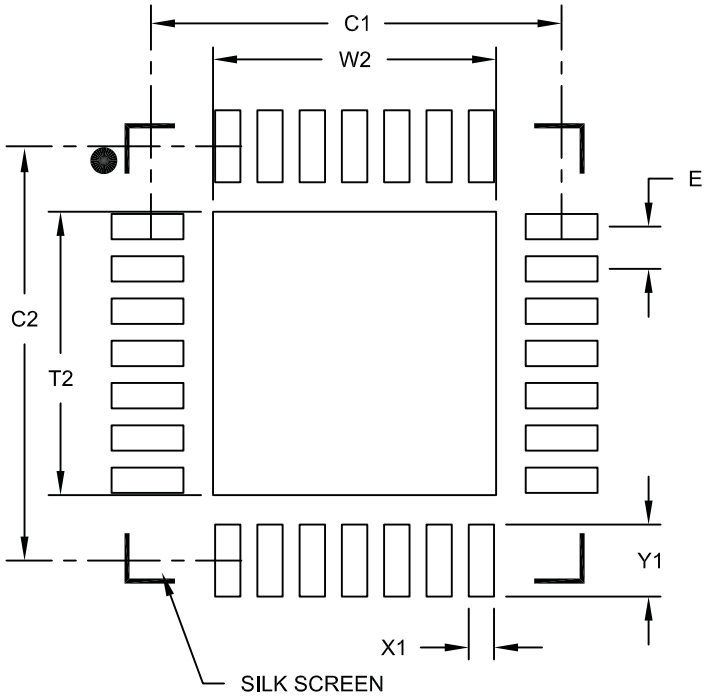
Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-140C Sheet 2 of 2

28-Lead Plastic Quad Flat, No Lead Package (MQ) – 5x5 mm Body [QFN] Land Pattern With 0.55 mm Contact Length

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		0.50 BSC	
Optional Center Pad Width	W2			3.35
Optional Center Pad Length	T2			3.35
Contact Pad Spacing	C1		4.90	
Contact Pad Spacing	C2		4.90	
Contact Pad Width (X28)	X1			0.30
Contact Pad Length (X28)	Y1			0.85

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing C04-2140A

APPENDIX A: REVISION HISTORY

Revision A (May 2017)

- Original Release of this Document.

NOTES:

INDEX

U

W

NOTES:

THE MICROCHIP WEB SITE

Microchip provides online support via our web site at www.microchip.com. This web site makes files and information easily available to customers. The web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://www.microchip.com/support

NOTES:

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	[X](1)	-	X	/XX
Device	Tape and Reel Option		Temperature Range	Package

Device:	MCP19122: Digitally Enhanced Power Analog Controller with Integrated Synchronous Driver MCP19123: Digitally Enhanced Power Analog Controller with Integrated Synchronous Driver
Tape and Reel Option:	Blank = Standard packaging (tube) T = Tape and Reel
Temperature Range:	E = -40°C to +125°C (Extended)
Package:	MJ = 24-Lead Plastic Quad Flat, No Lead Package - 4x4 mm Body (QFN) MQ = 28-Lead Plastic Quad Flat, No Lead Package - 5x5x mm Body (QFN)

Examples:

a) MCP19122-E/MJ: Extended temperature, 24 LD QFN 4x4 package

b) MCP19122T-E/MJ: Tape and Reel, Extended temperature, 24 LD QFN 4x4 package

a) MCP19123-E/MQ: Extended temperature, 28 LD QFN 5x5 package

b) MCP19123T-E/MQ: Tape and Reel, Extended temperature, 28 LD QFN 5x5 package

Note 1: Tape and Reel identifier only appears in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option.

NOTES:

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Trademarks

ISBN: 978-1-5224-1700-2

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon

Hong Kong
Tel: 852-2943-5100
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115
Fax: 86-571-8792-8116

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-3326-8000
Fax: 86-21-3326-8021

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

ASIA/PACIFIC

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-3019-1500

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

France - Saint Cloud
Tel: 33-1-30-60-70-00

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7289-7561

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820