Section 7. Resets

HIGHLIGHTS

This section of the manual contains the following topics:

7.1 INTRODUCTION

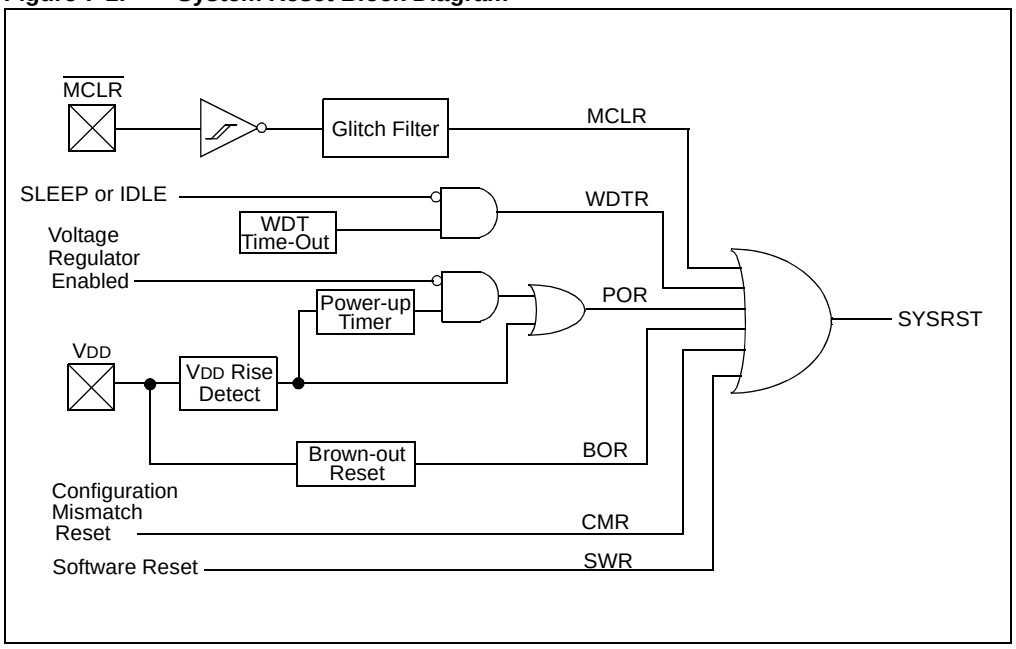
The Resets module combines all Reset sources and controls the system Reset signal SYSRST. The following is a list of device Reset sources:

- POR: Power-on Reset
- $\overline{\text{MCLR}}$: Pin Reset
- SWR: Software Reset
- WDTR: Watchdog Timer Reset
- BOR: Brown-out Reset
- CMR: Configuration Mismatch Reset

A simplified block diagram of the Reset module is shown in Figure 7-1. Any active source of Reset will make the system Reset signal active. Many registers associated with the CPU and peripherals are forced to a known “Reset state”. Most registers are unaffected by a Reset; their status is unknown on POR and unchanged by all other Resets.

Note:	Refer to the specific peripheral or the CPU section of this manual for register Reset states.

Figure 7-1: System Reset Block Diagram



7.2 CONTROL REGISTERS

All types of device Resets will set corresponding Status bits in the RCON register to indicate the type of Reset (see Register 7-1). A Power-on Reset will clear all bits, except for the BOR and POR bits (RCON<1:0>), which are set. The user may set or clear any of the bits at any time during code execution. The RCON bits serve only as Status bits. Setting a particular Reset Status bit in software will not cause a system Reset to occur.

The RCON register also has other bits associated with the Watchdog Timer and device power-saving states. For more information on the function of these bits, refer to **Section 7.4.3 "Using the RCON Status Bits"**.

The RSWRST control register has only one bit, SWRST. This bit is used to force a software Reset condition.

The Resets module consists of the following Special Function Registers (SFRs):

- RCON: Control register for Resets
  RCONCLR, RCONSET, RCONINV: Atomic Bit Manipulation Write-only Registers for RCON
- RSWRST: Data Register for Resets
  RSWRSTCLR, RSWRSTSET, RSWRSTINV: Atomic Bit Manipulation Write-only Registers for RSWRST

The following table summarizes all Resets-related registers. Corresponding registers appear after the summary, followed by a detailed description of each register.

Table 7-1: Reset SFR Summary

Name		Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
RCON	31:24	—	—	—	—	—	—	—	—
	23:16	—	—	—	—	—	—	—	—
	15:8	—	—	—	—	—	—	CMR	VREGS
	7:0	EXTR	SWR	—	WDTO	SLEEP	IDLE	BOR	POR
RCONCLR	31:0	Write clears selected bits in RCON, read yields undefined value							
RCONSET	31:0	Write sets selected bits in RCON, read yields undefined value							
RCONINV	31:0	Write inverts selected bits in RCON, read yields undefined value							
RSWRST	31:24	—	—	—	—	—	—	—	—
	23:16	—	—	—	—	—	—	—	—
	15:8	—	—	—	—	—	—	—	—
	7:0	—	—	—	—	—	—	—	SWRST
RSWRSTCLR	31:0	Write clears selected bits in RSWRST, read yields undefined value							
RSWRSTSET	31:0	Write sets selected bits in RSWRST, read yields undefined value							
RSWRSTINV	31:0	Write inverts selected bits in RSWRST, read yields undefined value							

Register 7-1: RCON: Reset Control Register

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 31							bit 24

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 23							bit 16

r-x	r-x	r-x	r-x	r-x	r-0	R/W-0	R/W-0
—	—	—	—	—	—	CMR	VREGS
bit 15							bit 8

R/W-0	R/W-0	r-x	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1
EXTR	SWR	—	WDTO	SLEEP	IDLE	BOR	POR
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	P = Programmable bit	r = Reserved bit
U = Unimplemented bit	-n = Bit Value at POR: (‘0’, ‘1’, x = Unknown)		

bit 31-11 **Reserved:** Write ‘0’; ignore read

bit 10 **Reserved:** Write ‘0’; ignore read

bit 9 **CMR:** Configuration Mismatch Reset Flag bit
1 = Configuration mismatch Reset has occurred
0 = Configuration mismatch Reset has not occurred
Note: This bit is set in hardware, it can only be cleared (= 0) in software.

bit 8 **VREGS:** Voltage Regulator Standby Enable bit
1 = Regulator is enabled and is on during SLEEP mode
0 = Regulator is disabled and is off during SLEEP mode

bit 7 **EXTR:** External Reset ($\overline{\text{MCLR}}$) Pin Flag bit
1 = Master Clear (pin) Reset has occurred
0 = Master Clear (pin) Reset has not occurred
Note: This bit is set in hardware, it can only be cleared (= 0) in software.

bit 6 **SWR:** Software Reset Flag bit
1 = Software Reset was executed
0 = Software Reset as not executed
Note: This bit is set in hardware, it can only be cleared (= 0) in software.

bit 5 **Reserved:** Write ‘0’; ignore read

bit 4 **WDTO:** Watchdog Timer Time-out Flag bit
1 = WDT Time-out has occurred
0 = WDT Time-out has not occurred
Note: This bit is set in hardware, it can only be cleared (= 0) in software.

bit 3 **SLEEP:** Wake From SLEEP Flag bit
1 = Device was in SLEEP mode
0 = Device was not in SLEEP mode
Note: This bit is set in hardware, it can only be cleared (= 0) in software.

bit 2 **IDLE:** Wake From IDLE Flag bit
1 = Device was in IDLE mode
0 = Device was not in IDLE mode
Note: This bit is set in hardware, it can only be cleared (= 0) in software.

Register 7-1: RCON: Reset Control Register

bit 1 **BOR:** Brown-out Reset Flag bit
User software must clear this bit to view next detection.
1 = Brown-out Reset has occurred
0 = Brown-out Reset has not occurred
Note: This bit is set in hardware, it can only be cleared (= 0) in software.

bit 0 **POR:** Power-on Reset Flag bit
User software must clear this bit to view next detection.
1 = Power-on Reset has occurred
0 = Power-on Reset has not occurred
Note: This bit is set in hardware, it can only be cleared (= 0) in software.

Register 7-2: RCONCLR: RCON Clear Register

Write clears selected bits in RCON, read yields undefined value	
bit 31	bit 0

bit 31-0 **Clears selected bits in RCON**

A write of '1' in one or more bit positions clears the corresponding bit(s) in RCON register and does not affect unimplemented or read-only bits. A write of '0' will not affect the register.

Example: `RCONCLR = 0x00008001` will clear bits 15 and 5 in RCON register.

Register 7-3: RCONSET: RCON Set Register

Write sets selected bits in RCON, read yields undefined value	
bit 31	bit 0

bit 31-0 **Sets selected bits in RCON**

A write of '1' in one or more bit positions sets the corresponding bit(s) in RCON register and does not affect unimplemented or read-only bits. A write of '0' will not affect the register.

Example: `RCONSET = 0x00008001` will set bits 15 and 5 in RCON register.

Register 7-4: RCONINV: RCON Invert Register

Write inverts selected bits in RCON, read yields undefined value	
bit 31	bit 0

bit 31-0 **Inverts selected bits in RCON**

A write of '1' in one or more bit positions inverts the corresponding bit(s) in RCON register and does not affect unimplemented or read-only bits. A write of '0' will not affect the register.

Example: `RCONINV = 0x00008001` will invert bits 15 and 5 in RCON register.

Register 7-5: RSWRST: Software Reset Register

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 31							bit 24

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 23							bit 16

r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 15							bit 8

r-x	r-x	r-x	r-x	r-x	r-x	r-x	W-0
—	—	—	—	—	—	—	SWRST
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	P = Programmable bit	r = Reserved bit
U = Unimplemented bit	-n = Bit Value at POR: (‘0’, ‘1’, x = Unknown)		

bit 31-1 **Reserved:** Write ‘0’; ignore read

bit 0 **SWRST:** Software Reset Trigger bit
1 = Enable software Reset event
0 = No effect

Note: The system unlock sequence must be performed before the SWRST bit can be written. See **7.3.4 “Software Reset (SWR)”**.

Register 7-6: RSWRSTCLR: RSWRST Clear Register

Write clears selected bits in RSWRST, read yields undefined value	
bit 31	bit 0

bit 31-0 **Clears selected bits in RSWRST**

A write of '1' in one or more bit positions clears the corresponding bit(s) in RSWRST register and does not affect unimplemented or read-only bits. A write of '0' will not affect the register.

Example: `RSWRSTCLR = 0x00008001` will clear bits 15 and 5 in RSWRST register.

Register 7-7: RSWRSTSET: RSWRST Set Register

Write sets selected bits in RSWRST, read yields undefined value	
bit 31	bit 0

bit 31-0 **Sets selected bits in RSWRST**

A write of '1' in one or more bit positions sets the corresponding bit(s) in RSWRST register and does not affect unimplemented or read-only bits. A write of '0' will not affect the register.

Example: `RSWRSTSET = 0x00008001` will set bits 15 and 5 in RSWRST register.

Register 7-8: RSWRSTINV: RSWRST Invert Register

Write inverts selected bits in RSWRST, read yields undefined value	
bit 31	bit 0

bit 31-0 **Inverts selected bits in RSWRST**

A write of '1' in one or more bit positions inverts the corresponding bit(s) in RSWRST register and does not affect unimplemented or read-only bits. A write of '0' will not affect the register.

Example: `RSWRSTINV = 0x00008001` will invert bits 15 and 5 in RSWRST register.

7.3 MODES OF OPERATION

7.3.1 System Reset

The PIC32MX Internal System Reset (SYSRST) can be generated from multiple Reset sources, such as POR (Power-on Reset), BOR (Brown-out Reset), MCLR (Master Clear Reset), WDTO (Watchdog Time-out Reset), SWR (Software Reset) and CMR (Configuration Mis-match Reset). A system Reset is active at first POR and asserted until device configuration settings are loaded and the clock oscillator sources become stable. The system Reset is then de-asserted allowing the CPU to start fetching code after 8 system clock cycles (SYSCLK).

BOR, MCLR and WDTO Resets are asynchronous events and to avoid SFR (Special Function Register) and RAM corruptions, the system Reset is synchronized with the system clock. All other Reset events are synchronous.

7.3.2 Power-on Reset (POR)

A power-on event generates an internal Power-on Reset pulse when a VDD rise is detected above VPOR. The device supply-voltage-characteristics must meet the specified starting-voltage and rise-rate requirements to generate the POR pulse. In particular, VDD must fall below VPOR before a new POR is initiated. For more information on the VPOR and VDD rise-rate specifications, refer to the Electrical Characteristics section of the specific device data sheet for details.

For those PIC32MX variants that have the on-chip voltage regulator enabled, the Power-up Timer (PWRT) is automatically disabled. For those PIC32MX variants that have the on-chip voltage regulator disabled, the core is supplied from an external power supply and the Power-up Timer is automatically enabled and is used to extend the duration of a power-up sequence. The PWRT adds a fixed 64 ms nominal delay at device start-up. Hence, the Power-on delay can either be the on-chip voltage regulator output delay, designated as TPU, or the power-up timer delay, designated as TPWRT.

At this point the POR event has expired, but the device Reset is still asserted while device configuration settings are loaded and the clock oscillator sources are configured and the clock monitoring circuitry waits for the oscillator source to become stable. The clock source used by the PIC32MX device when exiting from Reset is always selected from the FNOSC<2:0> bits in the DEVCFG1 Configuration Word. This additional delay depends on the clock and can include delays for TOSC, TLOCK and TFSCM. For details on the oscillator, PLL and Fail-Safe clock monitoring, refer to **Section 6.3.5 “Fail-Safe Clock Monitor Operation”**.

After these delays expire, the system Reset SYSRST is de-asserted. Before allowing the CPU to start code execution, 8 system clock cycles are required before the synchronized Reset to the CPU core is de-asserted.

The power-on event sets the BOR and POR Status bits (RCON<1:0>).

Refer to the Electrical Characteristics section of the specific device data sheet for more information on the values of the delay parameters.

Note:	When the device exits the Reset condition (begins normal operation), the device operating parameters (voltage, frequency, temperature, etc.) must be within their operating ranges; otherwise, the device will not function correctly. The user must ensure that the delay between the time power is first applied and the time system Reset is released is long enough to get all operating parameters within specification.

7.3.3 $\overline{\text{MCLR}}$ Reset

Whenever the $\overline{\text{MCLR}}$ pin is driven low, the Reset event is synchronized with the system clock SYSCLK before asserting the system Reset SYSRST, provided the input pulse on $\overline{\text{MCLR}}$ is longer than a certain minimum width, as specified in the Electrical Characteristics section of the specific device data sheet for details.

$\overline{\text{MCLR}}$ provides a filter to minimize the effects of noise and to avoid unwanted Reset events. The EXTR Status bit (RCON<7>) is set to indicate the $\overline{\text{MCLR}}$ Reset.

7.3.4 Software Reset (SWR)

The PIC32MX CPU core doesn't provide a specific RESET "instruction"; however, a hardware Reset can be performed in software (Software Reset) by executing a software Reset-command sequence. The software Reset command acts like a $\overline{\text{MCLR}}$ Reset. The software Reset sequence requires the system unlock sequence to be executed before the SWRST bit can be written. Refer to **Section 6.3.6 "Clock Switching Operation"** regarding the system unlock details. A software Reset is performed as follows:

- Write the system unlock sequence
- Set bit SWRST (RSWRST<0>) = 1
- Read the RSWRST register
- Follow with "while(1);" or 4 "NOP" instructions

Writing a '1' to RSWRST register sets bit SWRST, arming the software Reset. The subsequent read of the RSWRST register triggers the software Reset, which should occur on the next clock cycle following the read operation. To ensure no other user code is executed before the Reset event occurs, it is recommended that 4 'NOP' instructions or a "while(1);" statement be placed after the READ instruction.

The SWR Status bit (RCON<6>) is set to indicate the Software Reset.

Example 7-1: Software Reset Command Sequence

```
/*  The following code illustrates a software Reset */

    /* perform a system unlock sequence */
    SYSTEMUnlock();

    /* set SWRST bit to arm reset */
    RSWRSTSET = 1;

    /* read RSWRST register to trigger reset */
    volatile int* p = &RSWRST;
    *p;

    /* prevent any unwanted code execution until reset occurs*/
    while(1);
```


7.3.5 Watchdog Timer Reset

Note:	In this manual, a distinction is made between a power mode as it is used in a specific module, and a power mode as it is used by the device, e.g., Sleep mode of the Comparator and SLEEP mode of the CPU. To indicate which type of power mode is intended, uppercase and lowercase letters (Sleep, Idle, Debug) signify a module power mode, and all uppercase letters (SLEEP, IDLE, DEBUG) signify a device power mode.

A Watchdog Timer (WDT) Reset event is synchronized with the system clock SYSCLK before asserting the system Reset. Note that a WDT Time-out during SLEEP or IDLE mode will wake-up the processor and branch to the PIC32MX Reset vector, but not reset the processor. The only bits affected are WDTO and SLEEP or IDLE in the RCON register. Refer to **Section 9. "Watchdog Timer and Power-up Timer"** in this manual.

7.3.6 Brown-out Reset

PIC32MX family devices have a simple brown-out capability. If the voltage supplied to the regulator is inadequate to maintain a regulated level, the regulator Reset circuitry will generate a BOR event which is synchronized with the system clock SYSCLK before asserting the system Reset. This event is captured by the BOR flag bit (RCON<1>). Refer to the Electrical Characteristics section of the specific device data sheet for further details.

7.3.7 Configuration Mismatch Reset

To maintain the integrity of the stored configuration values, all device Configuration bits are loaded and implemented as a complementary set of bits. As the Configuration Words are being loaded, for each bit loaded as '1', a complementary value of '0', is stored into its corresponding background word location and vice versa. The bit pairs are compared every time the Configuration Words are loaded, including SLEEP mode. During this comparison, if the Configuration bit values are not found opposite to each other, a configuration mismatch event is generated which causes a device Reset.

If a device Reset occurs as a result of a configuration mismatch, the CMR Status bit (RCON<9>) is set.

7.4 EFFECTS OF VARIOUS RESETS

The Reset value for the Reset Control register, RCON, will depend on the type of device Reset, as indicated in Table 7-2.

Table 7-2: Status Bits, Their Significance and the Initialization Condition for RCON Register

Condition	Program Counter	EXTR	SWR	WDTO	SLEEP	IDLE	CMR	BOR	POR
Power-on Reset	BFC0_0000h	0	0	0	0	0	0	1	1
Brown-out Reset	BFC0_0000h	0	0	0	0	0	0	1	u
$\overline{\text{MCLR}}$ Reset during Run Mode	BFC0_0000h	1	u	u	u	u	u	u	u
$\overline{\text{MCLR}}$ Reset during IDLE Mode	BFC0_0000h	1	u	u	u	1(1)	u	u	u
$\overline{\text{MCLR}}$ Reset during SLEEP Mode	BFC0_0000h	1	u	u	1(1)	u	u	u	u
Software Reset Command	BFC0_0000h	u	1	u	u	u	u	u	u
Configuration Word Mismatch Reset	BFC0_0000h	u	u	u	u	u	1	u	u
WDT Time-out Reset during Run Mode	BFC0_0000h	u	u	1	u	u	u	u	u
WDT Time-out Reset during IDLE Mode	BFC0_0000h	u	u	1	u	1(1)	u	u	u
WDT Time-out Reset during SLEEP Mode	BFC0_0000h	u	u	1	1(1)	u	u	u	u
Interrupt Exit from IDLE Mode	Vector	u	u	u	u	1(1))	u	u	u
Interrupt Exit from SLEEP Mode	Vector	u	u	u	1(1)	u	u	u	u

Legend: u = unchanged

Note 1: SLEEP and IDLE bits states defined by previously executed WAIT instruction.

7.4.1 Special Function Register Reset States

Most of the Special Function Registers (SFRs) associated with the PIC32MX CPU and peripherals are reset to a particular value at a device Reset. Reset values are specified in the corresponding section of this manual.

The Reset value for the Reset Control register, RCON, will depend on the type of device Reset.

7.4.2 Configuration Word Register Reset States

All Reset conditions force the configuration settings to be re-loaded. The POR Reset sets all the Configuration Word register locations = 1 before loading the configuration settings. For all other Reset conditions, the Configuration Word register locations are not reset prior to being re-loaded. This difference in behavior accommodates $\overline{\text{MCLR}}$ assertions during DEBUG mode without affecting the state of the DEBUG operations.

Independent of the source of a Reset, the system clock is always re-loaded and is specified by the FNOSC<2:0> value in the DEVCFG1 Configuration Word. When the device is executing code, the user may change the primary system clock source by using the OSCCON register. Refer to **Section 6. "Oscillators"** in this manual for further details.

7.4.3 Using the RCON Status Bits

The user can read the RCON register after any system Reset to determine the cause of the Reset. Table 7-3 provides a summary of the Reset flag bit operation.

Note:	The Status bits in the RCON register should be cleared after they are read so that the next RCON register value after a device Reset will be meaningful.

Table 7-3: Reset Flag Bit Operation

Flag Bit	Set by:	Cleared by:
POR (RCON<0>)	POR	user software
BOR (RCON<1>)	POR, BOR	user software
EXTR (RCON<7>)	$\overline{\text{MCLR}}$ Reset	user software, POR, BOR
SWR (RCON<6>)	Software Reset command	user software, POR, BOR
CMR (RCON<9>)	Configuration mis-match	user software, POR, BOR
WDTO (RCON<4>)	WDT time-out	user software, POR, BOR
SLEEP (RCON<3>)	`WAIT` instruction	user software, POR, BOR
IDLE (RCON<2>)	`WAIT` instruction	user software, POR, BOR

Note: All Reset flag bits may be set or cleared by the user software.

7.4.4 Device Reset to Code Execution Start Time

The delay between the end of a Reset event and when the device actually begins to execute code is determined by two main factors: the type of Reset, and the system clock source coming out of the Reset. The code execution start time for various types of device Resets are summarized in Table 7-4. Individual delays are characterized in the Electrical Characteristics section of the specific device data sheet for details.

Table 7-4: Code Execution Start Time for Various Device Resets

Reset Type	Clock Source	Power-Up Delay(1)(2)(3)	System Clock Delay(4)(5)	FSCM Delay(6)
POR	EC, FRC, FRCDIV, LPRC	(TPU or TPWRT) + TSYSDLY	—	—
	ECPLL, FRCPLL	(TPU or TPWRT) + TSYSDLY	TLOCK	TFSCM
	XT, HS, SOSC	(TPU or TPWRT) + TSYSDLY	TOST	TFSCM
	XTPLL, HSPLL	(TPU or TPWRT) + TSYSDLY	TOST + TLOCK	TFSCM
BOR	EC, FRC, FRCDIV, LPRC	TSYSDLY	—	—
	ECPLL, FRCPLL	TSYSDLY	TLOCK	TFSCM
	XT, HS, SOSC	TSYSDLY	TOST	TFSCM
	XTPLL	TSYSDLY	TOST + TLOCK	TFSCM
$\overline{\text{MCLR}}$, CMR, SWR, WDTO	Any Clock	TSYSDLY	—	—

Note 1: TPU = Power-up Period with on-chip regulator enabled.

2: TPWRT = Power-up Period (POWER-UP TIMER) with on-chip regulator disabled.

3: TSYSDLY = Time required to reload Device Configuration Fuses plus 8 SYSCLK cycles.

4: TOST = Oscillator Start-up Timer.

5: TLOCK = PLL lock time.

6: TFSCM = Fail-Safe Clock Monitor delay.

Note: For parameter specifictions, see Section 30.2 “AC Characteristics and Timing Parameters.”

7.5 DESIGN TIPS

Question 1: ***How can I use the RCON register to determine the source of the device reset?***

Answer: Initialization code after a Reset can examine the RCON register and confirm the source of the Reset. In certain applications, this information can be used to take appropriate action to correct the problem that caused the Reset to occur. All Reset Status bits in the RCON register should be cleared after reading them to ensure the RCON value will provide meaningful results after the next device Reset.

```
int main(void)
{
    //... perform application specific startup tasks

    // next, check the cause of the Reset
    if(RCON & 0x0003)
    {
        // execute a Power-on-Reset handler
        // ...
    }
    else if(RCON & 0x0002)
    {
        // execute a Brown-out-Reset handler
        // ...
    }
    else if(RCON & 0x0080)
    {
        // execute a Master Clear Reset handler
        // ...
    }
    else if(RCON & 0x0040)
    {
        // execute a Software Reset handler
        // ...
    }
    else if (RCON & 0x0200)
    {
        // execute a Configuration Mismatch Reset handler
        // ...
    }
    else if (RCON & 0x0010)
    {
        // execute Watchdog Timeout Reset handler
        // ...
    }

    //... perform other application specific tasks

    while(1);
}
```


7.6 RELATED APPLICATION NOTES

This section lists application notes that are related to this section of the manual. These application notes may not be written specifically for the PIC32MX device family, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to Resets are:

Title	Application Note #
No related application notes at this time.	N/A

Note: Please visit the Microchip web site (www.microchip.com) for additional application notes and code examples for the PIC32MX family of devices.

7.7 REVISION HISTORY

Revision A (September 2007)

This is the initial released version of this document.

Revision B (October 2007)

Updated document to remove Confidential status.

Revision C (April 2008)

Revised status to Preliminary; Revised U-0 to r-x.

Revision D (June 2008)

Revised Figure 7-2; Deleted Figure 7-3; Revised Sections 7.3.2, 7.3.3, 7.3.4; Revised Table 7-4; Delete Figure 7.2 and 7.3; Change Reserved bits from “Maintain as” to “Write”.

Revision E (July 2008)

Revised Section 7.3.2, 7.3.3, 7.3.6, 7.4.4.

