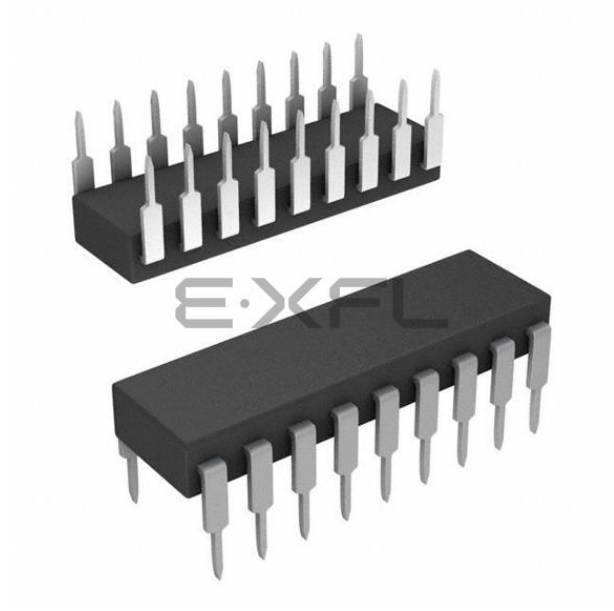


Welcome to E-XFL.COM

Embedded - Microcontrollers - Application Specific: Tailored Solutions for Precision and Performance

Embedded - Microcontrollers - Application Specific represents a category of microcontrollers designed with unique features and capabilities tailored to specific application needs. Unlike general-purpose microcontrollers, application-specific microcontrollers are optimized for particular tasks, offering enhanced performance, efficiency, and functionality to meet the demands of specialized applications.

What Are Embedded - Microcontrollers - Application Specific?

Application-specific microcontrollers are engineered to

Details	
Product Status	Obsolete
Applications	USB Microcontroller
Core Processor	M8B
Program Memory Type	OTP (3kB)
Controller Series	CY7C632xx
RAM Size	96 x 8
Interface	USB
Number of I/O	10
Voltage - Supply	3.5V ~ 5.5V
Operating Temperature	0°C ~ 70°C
Mounting Type	Through Hole
Package / Case	18-DIP (0.300", 7.62mm)
Supplier Device Package	18-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy7c63231a-pxc

TABLE OF CONTENTS

2.0 Functional Overview

2.1 *enCoRe* USB - The New USB Standard

Cypress has reinvented its leadership position in the low-speed USB market with a new family of innovative microcontrollers. Introducing...*enCoRe*™ USB—"enhanced Component Reduction." Cypress has leveraged its design expertise in USB solutions to create a new family of low-speed USB microcontrollers that enables peripheral developers to design new products with a minimum number of components. At the heart of the Cypress *enCoRe* USB technology is the breakthrough design of a crystal-less oscillator. By integrating the oscillator into the chip, an external crystal or resonator is no longer needed. We have also integrated other external components commonly found in low-speed USB applications such as pull-up resistors, wake-up circuitry, and a 3.3V regulator. All of this adds up to a lower system cost.

The family is comprised of 8-bit RISC One Time Programmable (OTP) microcontrollers. The instruction set has been optimized specifically for USB and PS/2 operations, although the microcontrollers can be used for a variety of other embedded applications.

The features up to 10 general-purpose I/O (GPIO) pins to support USB, PS/2 and other applications. The I/O pins are grouped into two ports (Port 0 to 1) where each pin can be individually configured as inputs with internal pull-ups, open drain outputs, or traditional CMOS outputs with programmable drive strength of up to 50 mA output drive. Additionally, each I/O pin can be used to generate a GPIO interrupt to the microcontroller.

The microcontrollers feature an internal oscillator. With the presence of USB traffic, the internal oscillator can be set to precisely tune to USB timing requirements (6 MHz ±1.5%). This clock generator has been optimized to reduce clock-related noise emissions (EMI), and provides the 6-MHz and 12-MHz clocks that remain internal to the microcontroller. When using the internal oscillator, XTALIN and XTALOUT can be configured as additional input pins that can be read on port 2. Optionally, an external 6-MHz ceramic resonator can be used to provide a higher precision reference if needed.

The is offered with 3 Kbytes of EPROM to minimize cost, and has 96 bytes of data RAM for stack space, user variables, and USB endpoint FIFOs.

The family includes low-voltage reset logic, a watchdog timer, a vectored interrupt controller, and a 12-bit free-running timer. The low-voltage reset (LVR) logic detects when power is applied to the device, resets the logic to a known state, and begins executing instructions at EPROM address 0x0000. LVR will also reset the part when V_{CC} drops below the operating voltage range. The watchdog timer can be used to ensure the firmware never gets stalled for more than approximately 8 ms.

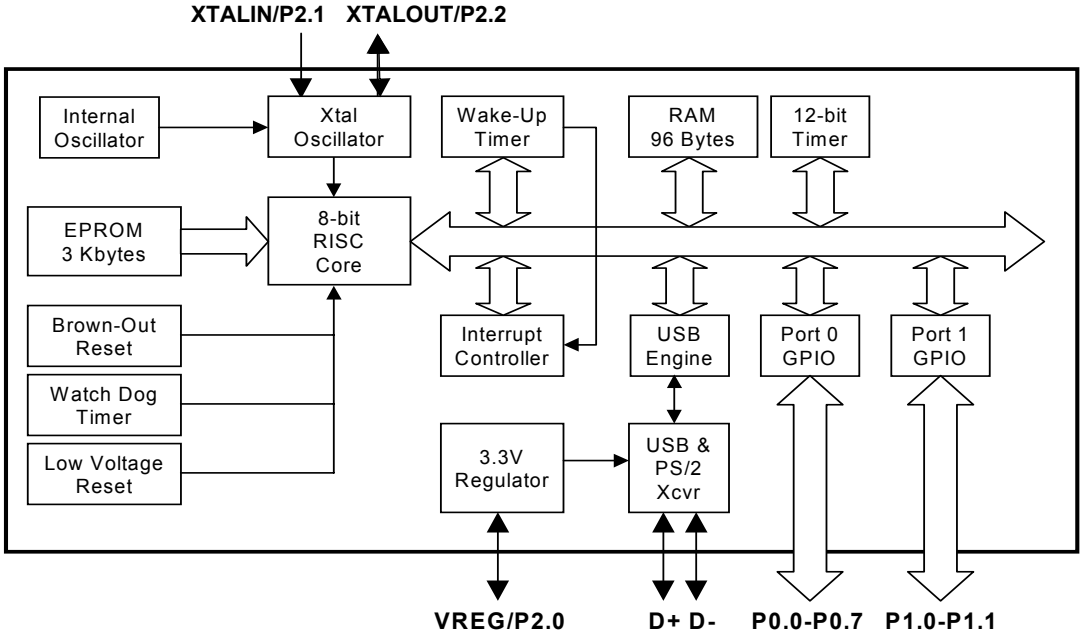
The microcontroller supports 7 maskable interrupts in the vectored interrupt controller. Interrupt sources include the USB Bus-Reset, the 128-µs and 1.024-ms outputs from the free-running timer, two USB endpoints, an internal wake-up timer and the GPIO port. The timers bits cause periodic interrupts when enabled. The USB endpoints interrupt after USB transactions complete on the bus. The GPIO port has a level of masking to select which GPIO inputs can cause a GPIO interrupt. For additional flexibility, the input transition polarity that causes an interrupt is programmable for each GPIO pin. The interrupt polarity can be either rising or falling edge.

The free-running 12-bit timer clocked at 1 MHz provides two interrupt sources as noted above (128 µs and 1.024 ms). The timer can be used to measure the duration of an event under firmware control by reading the timer at the start and end of an event, and subtracting the two values.

The CY7C63221/31A includes an integrated USB serial interface engine (SIE). The hardware supports one USB device address with two endpoints. The SIE allows the USB host to communicate with the function integrated into the microcontroller. A 3.3V regulated output pin provides a pull-up source for the external USB resistor on the D– pin. When using an external voltage regulator VREG can be configured as an input pin that can be read on port 2 (P2.0).

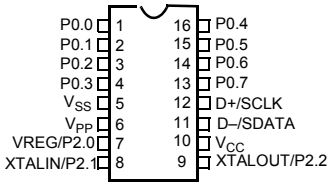
The USB D+ and D– USB pins can alternately be used as PS/2 SCLK and SDATA signals, so that products can be designed to respond to either USB or PS/2 modes of operation. PS/2 operation is supported with internal pull-up resistors on SCLK and SDATA, the ability to disable the regulator output pin, and an interrupt to signal the start of PS/2 activity. No external components are necessary for dual USB and PS/2 systems, and no GPIO pins need to be dedicated to switching between modes. Slow edge rates operate in both modes to reduce EMI.

3.0 Logic Block Diagram



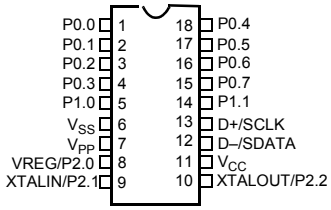
4.0 Pin Configurations

CY7C63221A
16-pin PDIP

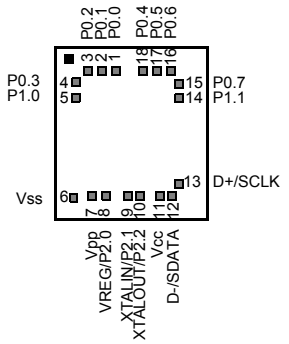


(Top View)

CY7C63231A
18-pin SOIC/PDIP



CY7C63221A-XC/XWC
DIE



5.0 Pin Assignments

Name	I/O	16-Pin	CY7C63231A/ CY7C63221A-XC 18-Pin/Pad	Description
D–/SDATA, D+/SCLK	I/O	11 12	12 13	USB differential data lines (D– and D+), or PS/2 clock and data signals (SDATA and SCLK)
P0[7:0]	I/O	1, 2, 3, 4, 13, 14, 15, 16	1, 2, 3, 4, 15, 16, 17, 18	GPIO Port 0 capable of sinking up to 50 mA/pin, or sinking controlled low or high programmable current. Can also source 2 mA current, provide a resistive pull-up, or serve as a high-impedance input.
P1[1:0]	I/O	NA	5,14	IO Port 1 capable of sinking up to 50 mA/pin, or sinking controlled low or high programmable current. Can also source 2 mA current, provide a resistive pull-up, or serve as a high-impedance input.

7.0 Instruction Set Summary

Refer to the *CYASM Assembler User's Guide* for detailed information on these instructions. Note that conditional jump instructions (i.e. JC, JNC, JZ, JNZ) take 5 cycles if jump is taken, 4 cycles if no jump.

MNEMONIC	Operand	Opcode	Cycles		MNEMONIC	Operand	Opcode	Cycles
HALT		00	7		NOP		20	4
ADD A,expr	data	01	4		INC A	acc	21	4
ADD A,[expr]	direct	02	6		INC X	x	22	4
ADD A,[X+expr]	index	03	7		INC [expr]	direct	23	7
ADC A,expr	data	04	4		INC [X+expr]	index	24	8
ADC A,[expr]	direct	05	6		DEC A	acc	25	4
ADC A,[X+expr]	index	06	7		DEC X	x	26	4
SUB A,expr	data	07	4		DEC [expr]	direct	27	7
SUB A,[expr]	direct	08	6		DEC [X+expr]	index	28	8
SUB A,[X+expr]	index	09	7		IORD expr	address	29	5
SBB A,expr	data	0A	4		IOWR expr	address	2A	5
SBB A,[expr]	direct	0B	6		POP A		2B	4
SBB A,[X+expr]	index	0C	7		POP X		2C	4
OR A,expr	data	0D	4		PUSH A		2D	5
OR A,[expr]	direct	0E	6		PUSH X		2E	5
OR A,[X+expr]	index	0F	7		SWAP A,X		2F	5
AND A,expr	data	10	4		SWAP A,DSP		30	5
AND A,[expr]	direct	11	6		MOV [expr],A	direct	31	5
AND A,[X+expr]	index	12	7		MOV [X+expr],A	index	32	6
XOR A,expr	data	13	4		OR [expr],A	direct	33	7
XOR A,[expr]	direct	14	6		OR [X+expr],A	index	34	8
XOR A,[X+expr]	index	15	7		AND [expr],A	direct	35	7
CMP A,expr	data	16	5		AND [X+expr],A	index	36	8
CMP A,[expr]	direct	17	7		XOR [expr],A	direct	37	7
CMP A,[X+expr]	index	18	8		XOR [X+expr],A	index	38	8
MOV A,expr	data	19	4		IOWX [X+expr]	index	39	6
MOV A,[expr]	direct	1A	5		CPL		3A	4
MOV A,[X+expr]	index	1B	6		ASL		3B	4
MOV X,expr	data	1C	4		ASR		3C	4
MOV X,[expr]	direct	1D	5		RLC		3D	4
reserved		1E			RRC		3E	4
XPAGE		1F	4		RET		3F	8
MOV A,X		40	4		DI		70	4
MOV X,A		41	4		EI		72	4
MOV PSP,A		60	4		RETI		73	8
CALL	addr	50 - 5F	10					
JMP	addr	80-8F	5		JC	addr	C0-CF	5 (or 4)
CALL	addr	90-9F	10		JNC	addr	D0-DF	5 (or 4)
JZ	addr	A0-AF	5 (or 4)		JACC	addr	E0-EF	7
JNZ	addr	B0-BF	5 (or 4)		INDEX	addr	F0-FF	14

8.0 Memory Organization

8.1 Program Memory Organization

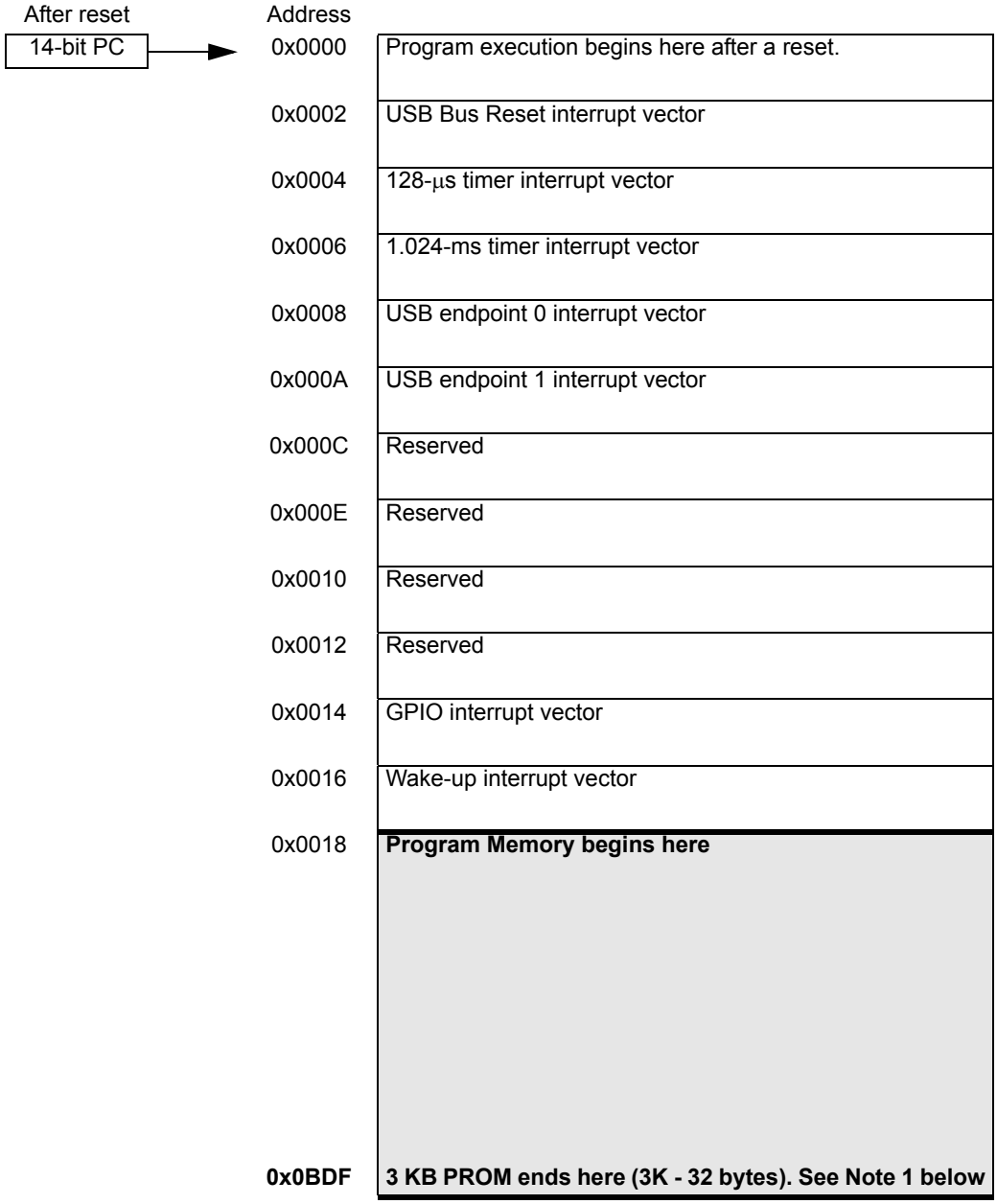
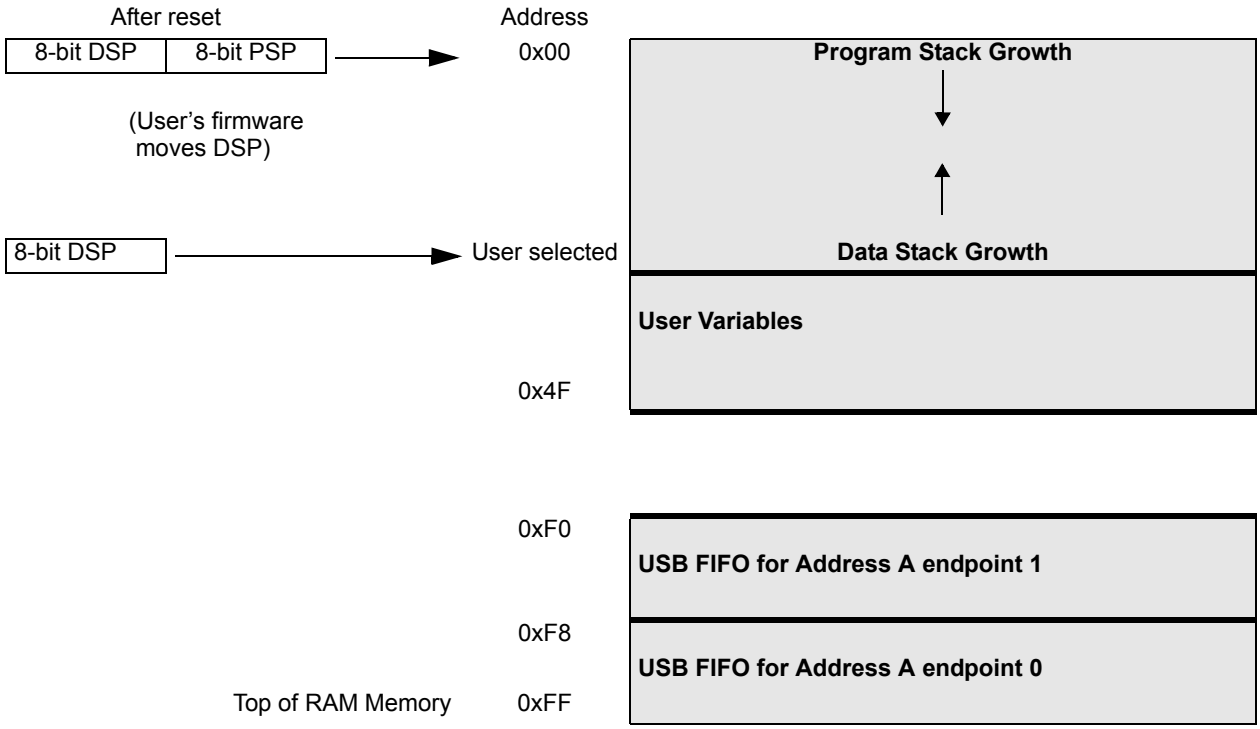


Figure 8-1. Program Memory Space with Interrupt Vector Table

Note:
1. The upper 32 bytes of the 3K PROM are reserved. Therefore, user's program must not over-write this space.

8.2 Data Memory Organization

The microcontroller provides 96 bytes of data RAM. In normal usage, the SRAM is partitioned into four areas: program stack, data stack, user variables and USB endpoint FIFOs as shown below:



8.3 I/O Register Summary

I/O registers are accessed via the I/O Read (IORD) and I/O Write (IOWR, IOWX) instructions. IORD reads the selected port into the accumulator. IOWR writes data from the accumulator to the selected port. Indexed I/O Write (IOWX) adds the contents of X to the address in the instruction to form the port address and writes data from the accumulator to the specified port. Note that specifying address 0 with IOWX (e.g., IOWX 0h) means the I/O port is selected solely by the contents of X.

Note: All bits of all registers are cleared to all zeros on reset, except the Processor Status and Control Register (*Figure 18-1*). *All registers not listed are reserved, and should never be written by firmware. All bits marked as reserved should always be written as 0 and be treated as undefined by reads.*

Table 8-1. I/O Register Summary

Register Name	I/O Address	Read/Write	Function	Fig.
Port 0 Data	0x00	R/W	GPIO Port 0	12-2
Port 1 Data	0x01	R/W	GPIO Port 1	12-3
Port 2 Data	0x02	R	Auxiliary input register for D+, D–, VREG, XTALIN, XTALOUT	12-8
Port 0 Interrupt Enable	0x04	W	Interrupt enable for pins in Port 0	19-4
Port 1 Interrupt Enable	0x05	W	Interrupt enable for pins in Port 1	19-5
Port 0 Interrupt Polarity	0x06	W	Interrupt polarity for pins in Port 0	19-6
Port 1 Interrupt Polarity	0x07	W	Interrupt polarity for pins in Port 1	19-7
Port 0 Mode0	0x0A	W	Controls output configuration for Port 0	12-4
Port 0 Mode1	0x0B	W		12-5
Port 1 Mode0	0x0C	W	Controls output configuration for Port 1	12-6
Port 1 Mode1	0x0D	W		12-7
USB Device Address	0x10	R/W	USB Device Address register	14-1
EP0 Counter Register	0x11	R/W	USB Endpoint 0 counter register	14-4
EP0 Mode Register	0x12	R/W	USB Endpoint 0 configuration register	14-2
EP1 Counter Register	0x13	R/W	USB Endpoint 1 counter register	14-4
EP1 Mode Register	0x14	R/W	USB Endpoint 1 configuration register	14-3
USB Status & Control	0x1F	R/W	USB status and control register	13-1
Global Interrupt Enable	0x20	R/W	Global interrupt enable register	19-1
Endpoint Interrupt Enable	0x21	R/W	USB endpoint interrupt enables	19-2
Timer (LSB)	0x24	R	Lower 8 bits of free-running timer (1 MHz)	17-1
Timer (MSB)	0x25	R	Upper 4 bits of free-running timer	17-2
WDR Clear	0x26	W	Watch Dog Reset clear	-
Clock Configuration	0xF8	R/W	Internal / External Clock configuration register	9-2
Processor Status & Control	0xFF	R/W	Processor status and control	18-1

9.0 Clocking

The chip can be clocked from either the internal on-chip clock, or from an oscillator based on an external resonator/crystal, as shown in *Figure 9-1*. No additional capacitance is included on chip at the XTALIN/OUT pins. Operation is controlled by the Clock Configuration Register, *Figure 9-2*.

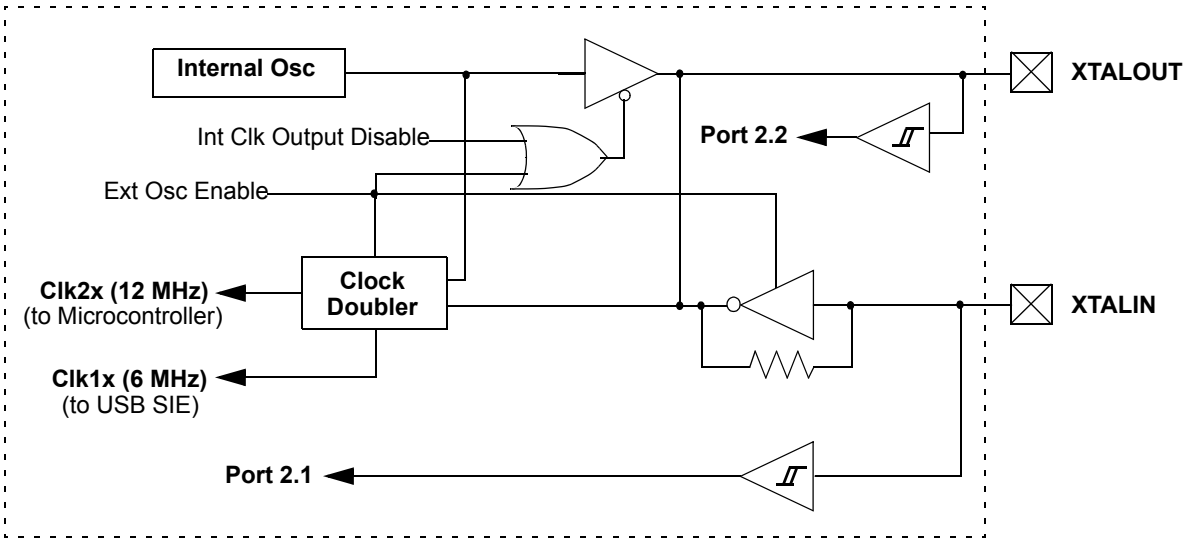


Figure 9-1. Clock Oscillator On-chip Circuit

Bit #	7	6	5	4	3	2	1	0
Bit Name	Ext. Clock Resume Delay	Wake-up Timer Adjust Bit [2:0]			Low-voltage Reset Disable	Precision USB Clocking Enable	Internal Clock Output Disable	External Oscillator Enable
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Figure 9-2. Clock Configuration Register (Address 0xF8)

Bit 7: Ext. Clock Resume Delay

External Clock Resume Delay bit selects the delay time when switching to the external oscillator from the internal oscillator mode, or when waking from suspend mode with the external oscillator enabled.

1 = 4 ms delay.

0 = 128 μs delay.

The delay gives the oscillator time to start up. The shorter time is adequate for operation with ceramic resonators, while the longer time is preferred for start-up with a crystal. (These times **do not include** an initial oscillator start-up time which depends on the resonating element. This time is typically 50–100 μs for ceramic resonators and 1–10 ms for crystals). Note that this bit only selects the delay time for the external clock mode. When waking from suspend mode with the internal oscillator (Bit 0 is LOW), the delay time is only 8 μs in addition to a delay of approximately 1 μs for the oscillator to start.

Bit [6:4]: Wake-up Timer Adjust Bit [2:0]

The Wake-up Timer Adjust Bits are used to adjust the Wake-up timer period.

If the Wake-up interrupt is enabled in the Global Interrupt Enable Register, the microcontroller will generate wake-up interrupts periodically. The frequency of these periodical wake-up interrupts is adjusted by setting the Wake-up Timer Adjust Bit [2:0], as described in Section 11.2. One common use of the wake-up interrupts is to generate periodical wake-up events during suspend mode to check for changes, such as looking for movement in a mouse, while maintaining a low average power.

Bit 3: Low-voltage Reset Disable

When V_{CC} drops below V_{LVR} (see Section 23.0 for the value of V_{LVR}) and the Low-voltage Reset circuit is enabled, the microcontroller enters a partial suspend state for a period of t_{START} (see Section 24.0 for the value of t_{START}). Program

Start-up times for the external oscillator depend on the resonating device. Ceramic-resonator-based oscillators typically start in less than 100 μs, while crystal-based oscillators take longer, typically 1 to 10 ms. Board capacitance should be minimized on the XTALIN and XTALOUT pins by keeping the traces as short as possible.

An external 6-MHz clock can be applied to the XTALIN pin if the XTALOUT pin is left open.

10.0 Reset

The USB Controller supports three types of resets. The effects of the reset are listed below. The reset types are:

1. Low-voltage Reset (LVR)
2. Brown-out Reset (BOR)
3. Watchdog Reset (WDR)

The occurrence of a reset is recorded in the Processor Status and Control Register (*Figure 18-1*). Bits 4 (Low-voltage or Brown-out Reset bit) and 6 (Watchdog Reset bit) are used to record the occurrence of LVR/BOR and WDR respectively. The firmware can interrogate these bits to determine the cause of a reset.

The microcontroller begins execution from ROM address 0x0000 after a LVR, BOR, or WDR reset. Although this looks like interrupt vector 0, there is an important difference. Reset processing does NOT push the program counter, carry flag, and zero flag onto program stack. Attempting to execute either a RET or RETI in the reset handler will cause unpredictable execution results.

The following events take place on reset. More details on the various resets are given in the following sections.

1. All registers are reset to their default states (all bits cleared, except in Processor Status and Control Register).
2. GPIO and USB pins are set to high-impedance state.
3. The VREG pin is set to high-impedance state.
4. Interrupts are disabled.
5. USB operation is disabled and must be enabled by firmware if desired, as explained in Section 14.1.
6. For a BOR or LVR, the external oscillator is disabled and Internal Clock mode is activated, followed by a time-out period t_{START} for V_{CC} to stabilize. A WDR does not change the clock mode, and there is no delay for V_{CC} stabilization on a WDR. Note that the External Oscillator Enable (Bit 0, *Figure 9-2*) will be cleared by a WDR, but it does not take effect until suspend mode is entered.
7. The Program Stack Pointer (PSP) and Data Stack Pointer (DSP) reset to address 0x00. Firmware should move the DSP for USB applications, as explained in Section 6.5.
8. Program execution begins at address 0x0000 after the appropriate time-out period.

10.1 Low-voltage Reset (LVR)

When V_{CC} is first applied to the chip, the internal oscillator is started and the Low-voltage Reset is initially enabled by default. At the point where V_{CC} has risen above V_{LVR} (see Section 23.0 for the value of V_{LVR}), an internal counter starts counting for a period of t_{START} (see Section 24.0 for the value of t_{START}). During this t_{START} time, the microcontroller enters a partial suspend state to wait for V_{CC} to stabilize before it begins executing code from address 0x0000.

As long as the LVR circuit is enabled, this reset sequence repeats whenever the V_{CC} pin voltage drops below V_{LVR}. The LVR can be disabled by firmware by setting the Low-voltage Reset Disable bit in the Clock Configuration Register (*Figure 9-2*). In addition, the LVR is automatically disabled in suspend mode to save power. If the LVR was enabled before entering suspend mode, it becomes active again once the suspend mode ends.

When LVR is disabled during normal operation (e.g., by writing ‘0’ to the Low-voltage Reset Disable bit in the Clock Configuration Register), the chip may enter an unknown state if V_{CC} drops below V_{LVR}. Therefore, LVR should be enabled at all times during normal operation. If LVR is disabled (e.g., by firmware or during suspend mode), a secondary low-voltage monitor, BOR, becomes active, as described in the next section. The LVR/BOR Reset bit of the Processor Status and Control Register (*Figure 18-1*), is set to ‘1’ if either a LVR or BOR has occurred.

10.2 Brown-out Reset (BOR)

The Brown-out Reset (BOR) circuit is always active and behaves like the POR. BOR is asserted whenever the V_{CC} voltage to the device is below an internally defined trip voltage of approximately 2.5V. The BOR re-enables LVR. That is, once V_{CC} drops and trips BOR, the part remains in reset until V_{CC} rises above V_{LVR}. At that point, the t_{START} delay occurs before normal operation resumes, and the microcontroller starts executing code from address 0x00 after the t_{START} delay.

In suspend mode, only the BOR detection is active, giving a reset if V_{CC} drops below approximately 2.5V. Since the device is suspended and code is not executing, this lower reset voltage is safe for retaining the state of all registers and memory. Note that in suspend mode, LVR is disabled as discussed in Section 10.1.

10.3 Watchdog Reset (WDR)

The Watchdog Timer Reset (WDR) occurs when the internal Watchdog timer rolls over. Writing any value to the write-only Watchdog Reset Register at address 0x26 will clear the timer. The timer will roll over and WDR will occur if it is not cleared within t_{WATCH} (see *Figure 10-1*) of the last clear. Bit 6 (Watchdog Reset bit) of the Processor Status and Control Register is set to record this event (see Section 18.0 for more details). A Watchdog Timer Reset lasts for typically 2–4 ms after which the microcontroller begins execution at ROM address 0x0000.

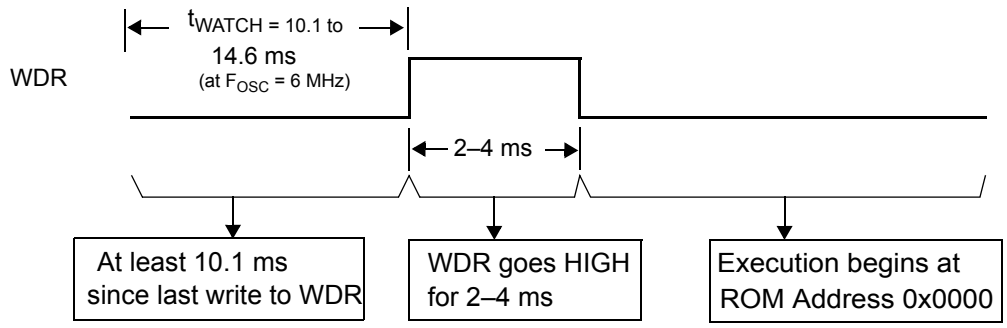


Figure 10-1. Watchdog Reset (WDR, Address 0x26)

11.0 Suspend Mode

The parts support a versatile low-power suspend mode. In suspend mode, only an enabled interrupt or a LOW state on the D–/SDATA pin will wake the part. Two options are available. For lowest power, all internal circuits can be disabled, so only an external event will resume operation. Alternatively, a low-power internal wake-up timer can be used to trigger the wake-up interrupt. This timer is described in Section 11.2, and can be used to periodically poll the system to check for changes, such as looking for movement in a mouse, while maintaining a low average power.

The is placed into a low-power state by setting the Suspend bit of the Processor Status and Control Register (*Figure 18-1*). All logic blocks in the device are turned off except the GPIO interrupt logic, the D–/SDATA pin input receiver, and (optionally) the wake-up timer. The clock oscillators, as well as the free-running and watchdog timers are shut down. Only the occurrence of an enabled GPIO interrupt, wake-up interrupt, SPI slave interrupt, or a LOW state on the D–/SDATA pin will wake the part from suspend (D– LOW indicates non-idle USB activity). Once one of these resuming conditions occurs, clocks will be restarted and the device returns to full operation after the oscillator is stable and the selected delay period expires. This delay period is determined by selection of internal vs. external clock, and by the state of the Ext. Clock Resume Delay as explained in Section 9.0.

In suspend mode, any enabled and pending interrupt will wake the part up. The state of the Interrupt Enable Sense bit (Bit 2, *Figure 18-1*) does not have any effect. As a result, any interrupts not intended for waking from suspend should be disabled through the Global Interrupt Enable Register and the USB End Point Interrupt Enable Register (Section 19.0).

If a resuming condition exists when the suspend bit is set, the part will still go into suspend and then awake after the appropriate delay time. The Run bit in the Processor Status and Control Register must be set for the part to resume out of suspend.

Once the clock is stable and the delay time has expired, the microcontroller will execute the instruction following the I/O write that placed the device into suspend mode before servicing any interrupt requests.

To achieve the lowest possible current during suspend mode, all I/O should be held at either V_{CC} or ground. In addition, the GPIO *bit* interrupts (*Figure 19-4* and *Figure 19-5*) should be disabled for any pins that are not being used for a wake-up interrupt. This should be done even if the main GPIO Interrupt Enable (*Figure 19-1*) is off.

Typical code for entering suspend is shown below:

```
...         ; All GPIO set to low-power state (no floating pins, and bit interrupts disabled unless using for wake-up)
...         ; Enable GPIO and/or wake-up timer interrupts if desired for wake-up
...         ; Select clock mode for wake-up (see Section 11.1)
mov a, 09h  ; Set suspend and run bits
iowr FFh    ; Write to Status and Control Register - Enter suspend, wait for GPIO/wake-up interrupt or USB activity
nop         ; This executes before any ISR
...         ; Remaining code for exiting suspend routine
```

11.1 Clocking Mode on Wake-up from Suspend

When exiting suspend on a wake-up event, the device can be configured to run in either Internal or External Clock mode. The mode is selected by the state of the External Oscillator Enable bit in the Clock Configuration Register (*Figure 9-2*). Using the

Bit 3: USB Bus Activity

The Bus Activity bit is a “sticky” bit that detects any non-idle USB event has occurred on the USB bus. Once set to HIGH by the SIE to indicate the bus activity, this bit retains its logical HIGH value until firmware clears it. Writing a ‘0’ to this bit clears it; writing a ‘1’ preserves its value. The user firmware should check and clear this bit periodically to detect any loss of bus activity. Firmware can clear the Bus Activity bit, but only the SIE can set it. The 1.024-ms timer interrupt service routine is normally used to check and clear the Bus Activity bit.

1 = There has been bus activity since the last time this bit was cleared. This bit is set by the SIE.

0 = No bus activity since last time this bit was cleared (by firmware).

Bit [2:0]: D+/D– Forcing Bit [2:0]

Forcing bits allow firmware to directly drive the D+ and D– pins, as shown in *Table 13-1*. Outputs are driven with controlled edge rates in these modes for low EMI. For forcing the D+ and D– pins in USB mode, D+/D– Forcing Bit 2 should be 0. Setting D+/D– Forcing Bit 2 to ‘1’ puts both pins in an open-drain mode, preferred for applications such as PS/2 or LED driving.

Table 13-1. Control Modes to Force D+/D– Outputs

D+/D– Forcing Bit [2:0]	Control Action	Application
000	Not forcing (SIE controls driver)	Any Mode
001	Force K (D+ HIGH, D– LOW)	USB Mode
010	Force J (D+ LOW, D– HIGH)	
011	Force SE0 (D– LOW, D+ LOW)	
100	Force D– LOW, D+ LOW	PS/2 Mode[2]
101	Force D– LOW, D+ HiZ	
110	Force D– HiZ, D+ LOW	
111	Force D– HiZ, D+ HiZ	

Note:
2. For PS/2 operation, the D+/D- Forcing Bit [2:0] = 111b mode must be set initially (one time only) before using the other PS/2 force modes.

14.0 USB Device

The supports one USB Device Address with two endpoints: EP0 and EP1.

14.1 USB Address Register

The USB Device Address Register contains a 7-bit USB address and one bit to enable USB communication. This register is cleared during a reset, setting the USB device address to zero and marking this address as disabled. *Figure 14-1* shows the format of the USB Address Register.

Bit #	7	6	5	4	3	2	1	0
Bit Name	Device Address Enable	Device Address Bit						
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Figure 14-1. USB Device Address Register (Address 0x10)

In either USB or PS/2 mode, this register is cleared by both hardware resets and the USB bus reset. See Section 19.3 for more information on the USB Bus Reset - PS/2 interrupt.

Bit 7: Device Address Enable

This bit must be enabled by firmware before the serial interface engine (SIE) will respond to USB traffic at the address specified in Bit [6:0].

1 = Enable USB device address.

0 = Disable USB device address.

Bit [6:0]: Device Address Bit[6:0]

These bits must be set by firmware during the USB enumeration process (i.e., SetAddress) to the non-zero address assigned by the USB host.

14.2 USB Control Endpoint

All USB devices are required to have an endpoint number 0 (EP0) that is used to initialize and control the USB device. EP0 provides access to the device configuration information and allows generic USB status and control accesses. EP0 is bidirectional, as the device can both receive and transmit data. EP0 uses an 8-byte FIFO at SRAM locations 0xF8-0xFF, as shown in Section 8.2.

The EP0 endpoint mode register uses the format shown in *Figure 14-2*.

Bit #	7	6	5	4	3	2	1	0
Bit Name	SETUP Received	IN Received	OUT Received	ACKed Transaction	Mode Bit			
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Figure 14-2. Endpoint 0 Mode Register (Address 0x12)

The SIE provides a locking feature to prevent firmware from overwriting bits in the USB Endpoint 0 Mode Register. Writes to the register have no effect from the point that Bit[6:0] of the register are updated (by the SIE) until the firmware reads this register. The CPU can unlock this register by reading it.

Because of these hardware-locking features, firmware should perform an read after a write to the USB Endpoint 0 Mode Register and USB Endpoint 0 Count Register (*Figure 14-4*) to verify that the contents have changed as desired, and that the SIE has not updated these values.

Bit [7:4] of this register are cleared by any non-locked write to this register, regardless of the value written.

Bit 7: SETUP Received

1 = A valid SETUP packet has been received. This bit is forced HIGH from the start of the data packet phase of the SETUP transaction until the start of the ACK packet returned by the SIE. The CPU is prevented from clearing this bit during this interval.

While this bit is set to '1', the CPU cannot write to the EP0 FIFO. This prevents firmware from overwriting an incoming SETUP transaction before firmware has a chance to read the SETUP data.

0 = No SETUP received. This bit is cleared by any non-locked writes to the register.

Bit 6: IN Received

1 = A valid IN packet has been received. This bit is updated to '1' after the last received packet in an IN transaction. This bit is cleared by any non-locked writes to the register.

0 = No IN received. This bit is cleared by any non-locked writes to the register.

Bit 5: OUT Received

1 = A valid OUT packet has been received. This bit is updated to '1' after the last received packet in an OUT transaction. This bit is cleared by any non-locked writes to the register.

0 = No OUT received. This bit is cleared by any non-locked writes to the register.

Bit 4: ACKed Transaction

The ACKed Transaction bit is set whenever the SIE engages in a transaction to the register's endpoint that completes with an ACK packet.

1 = The transaction completes with an ACK

0 = The transaction does not complete with an ACK

Bit [3:0]: Mode Bit[3:0]

The endpoint modes determine how the SIE responds to USB traffic that the host sends to the endpoint. For example, if the endpoint Mode Bits [3:0] are set to 0001 which is NAK IN/OUT mode as shown in *Table 20-1*, the SIE will send NAK handshakes in response to any IN or OUT token sent to this endpoint. In this NAK IN/OUT mode, the SIE will send an ACK handshake when the host sends a SETUP token to this endpoint. The mode encoding is shown in *Table 20-1*. Additional information on the mode bits can be found in *Table 20-2* and *Table 20-3*. These modes give the firmware total control on how to respond to different tokens sent to the endpoints from the host.

In addition, the Mode Bits are automatically changed by the SIE in response to many USB transactions. For example, if the Mode Bit [3:0] are set to 1011 which is ACK OUT-NAK IN mode as shown in *Table 20-1*, the SIE will change the endpoint Mode Bit [3:0] to NAK IN/OUT (0001) mode after issuing an ACK handshake in response to an OUT token. Firmware needs to update the mode for the SIE to respond appropriately.

14.3 USB Non-Control Endpoints

The feature one non-control endpoint, endpoint 1 (EP1). The EP1 Mode Register does not have the locking mechanism of the EP0 Mode Register. The EP1 Mode Register uses the format shown in *Figure 14-3*. EP1 uses an 8-byte FIFO at SRAM locations 0xF0–0xF7 as shown in Section 8.2.

Bit #	7	6	5	4	3	2	1	0
Bit Name	STALL	Reserved		ACKed Transaction	Mode Bit			
Read/Write	R/W	-	-	R/C	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Figure 14-3. USB Endpoint EP1 Mode Registers (Address 0x14)

Bit 7: STALL

1 = The SIE will stall an OUT packet if the Mode Bits are set to ACK-OUT, and the SIE will stall an IN packet if the mode bits are set to ACK-IN. See Section 20.0 for the available modes.

0 = This bit must be set to LOW for all other modes.

Bit [6:5]: Reserved. Must be written to zero during register writes.

Bit 4: ACKed Transaction

The ACKed transaction bit is set whenever the SIE engages in a transaction to the register's endpoint that completes with an ACK packet.

1 = The transaction completes with an ACK.

0 = The transaction does not complete with an ACK.

20.0 USB Mode Tables

The following tables give details on mode setting for the USB Serial Interface Engine (SIE) for both the control endpoint (EP0) and non-control endpoint (EP1).

Table 20-1. USB Register Mode Encoding for Control and Non-Control Endpoint

Mode	Encoding	SETUP	IN	OUT	Comments
Disable	**0000**	Ignore	Ignore	Ignore	Ignore all USB traffic to this endpoint
NAK IN/OUT	**0001**	Accept	NAK	NAK	On Control endpoint, after successfully sending an ACK handshake to a SETUP packet, the SIE forces the endpoint mode (from modes other than 0000) to 0001. The mode is also changed by the SIE to 0001 from mode 1011 on issuance of ACK handshake to an OUT.
Status OUT Only	**0010**	Accept	STALL	Check	For Control endpoints
STALL IN/OUT	**0011**	Accept	STALL	STALL	For Control endpoints
Ignore IN/OUT	**0100**	Accept	Ignore	Ignore	For Control endpoints
Reserved	**0101**	Ignore	Ignore	Always	Reserved
Status IN Only	**0110**	Accept	TX 0 Byte	STALL	For Control Endpoints
Reserved	**0111**	Ignore	TX Count	Ignore	Reserved
NAK OUT	**1000**	Ignore	Ignore	NAK	In mode 1001, after sending an ACK handshake to an OUT, the SIE changes the mode to 1000
ACK OUT($STALL^{[3]}=0$) ACK OUT($STALL^{[3]}=1$)	**1001** **1001**	Ignore Ignore	Ignore Ignore	ACK STALL	This mode is changed by the SIE to mode 1000 on issuance of ACK handshake to an OUT
NAK OUT - Status IN	**1010**	Accept	TX 0 Byte	NAK	
ACK OUT - NAK IN	**1011**	Accept	NAK	ACK	This mode is changed by the SIE to mode 0001 on issuance of ACK handshake to an OUT
NAK IN	**1100**	Ignore	NAK	Ignore	An ACK from mode 1101 changes the mode to 1100
ACK IN($STALL^{[3]}=0$) ACK IN($STALL^{[3]}=1$)	**1101** **1101**	Ignore Ignore	TX Count STALL	Ignore Ignore	This mode is changed by the SIE to mode 1100 on issuance of ACK handshake to an IN
NAK IN - Status OUT	**1110**	Accept	NAK	Check	An ACK from mode 1111 changes the mode to 1110
ACK IN - Status OUT	**1111**	Accept	TX Count	Check	This mode is changed by the SIE to mode 1110 on issuance of ACK handshake to an IN

Note:
3. STALL bit is the bit 7 of the USB Non-Control Device Endpoint Mode registers. Refer to Section 14.3 for more explanation.

Mode Column:

The 'Mode' column contains the mnemonic names given to the modes of the endpoint. The mode of the endpoint is determined by the 4 bit binaries in the 'Encoding' column as discussed below. The Status IN and Status OUT modes represent the status IN or OUT stage of the control transfer.

Encoding Column:

The contents of the 'Encoding' column represent the Mode Bits [3:0] of the Endpoint Mode Registers (*Figure 14-2* and *Figure 14-3*). The endpoint modes determine how the SIE responds to different tokens that the host sends to the endpoints. For example, if the Mode Bits [3:0] of the Endpoint 0 Mode Register (*Figure 14-2*) are set to '0001', which is NAK IN/OUT mode as shown in Table 20-1 above, the SIE of the part will send an ACK handshake in response to SETUP tokens and NAK any IN or OUT tokens. For more information on the functionality of the Serial Interface Engine (SIE), see Section 13.0.

SETUP, IN, and OUT Columns:

Depending on the mode specified in the 'Encoding' column, the 'SETUP', 'IN', and 'OUT' columns contain the device SIE's responses when the endpoint receives SETUP, IN, and OUT tokens respectively.

A 'Check' in the Out column means that upon receiving an OUT token the SIE checks to see whether the OUT is of zero length and has a Data Toggle (Data1/0) of 1. If these conditions are true, the SIE responds with an ACK. If any of the above conditions is not met, the SIE will respond with either a STALL or Ignore. Table 20-3 gives detailed analysis of all possible cases.

A 'TX Count' entry in the IN column means that the SIE will transmit the number of bytes specified in the Byte Count Bit [3:0] of the Endpoint Count Register (*Figure 14-4*) in response to any IN token.

Table 20-3. Details of Modes for Differing Traffic Conditions(continued)

3	2	1	0	token	count	buffer	dval	DTOG	DVAL	COUNT	SETUP	IN	OUT	ACK	3	2	1	0	response	int
1	1	1	0	OUT	x	UC	invalid	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
1	1	1	0	IN	x	UC	x	UC	UC	UC	UC	1	UC	UC	NoChange				NAK	yes
Status OUT Only																				
0	0	1	0	OUT	2	UC	valid	1	1	updates	UC	UC	1	1	NoChange				ACK	yes
0	0	1	0	OUT	2	UC	valid	0	1	updates	UC	UC	1	UC	0	0	1	1	STALL	yes
0	0	1	0	OUT	!=2	UC	valid	updates	1	updates	UC	UC	1	UC	0	0	1	1	STALL	yes
0	0	1	0	OUT	> 10	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
0	0	1	0	OUT	x	UC	invalid	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
0	0	1	0	IN	x	UC	x	UC	UC	UC	UC	1	UC	UC	0	0	1	1	STALL	yes
OUT Endpoint																				
ACK OUT, STALL Bit = 0 (*Figure 14-3*)																				
1	0	0	1	OUT	<= 10	data	valid	updates	1	updates	UC	UC	1	1	1	0	0	0	ACK	yes
1	0	0	1	OUT	> 10	junk	x	updates	updates	updates	UC	UC	1	UC	NoChange				Ignore	yes
1	0	0	1	OUT	x	junk	invalid	updates	0	updates	UC	UC	1	UC	NoChange				Ignore	yes
1	0	0	1	IN	x	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
ACK OUT, STALL Bit = 1 (*Figure 14-3*)																				
1	0	0	1	OUT	<= 10	UC	valid	UC	UC	UC	UC	UC	1	UC	NoChange				STALL	yes
1	0	0	1	OUT	> 10	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
1	0	0	1	OUT	x	UC	invalid	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
1	0	0	1	IN	x	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
NAK OUT																				
1	0	0	0	OUT	<= 10	UC	valid	UC	UC	UC	UC	UC	1	UC	NoChange				NAK	yes
1	0	0	0	OUT	> 10	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
1	0	0	0	OUT	x	UC	invalid	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
1	0	0	0	IN	x	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
Reserved																				
0	1	0	1	OUT	x	updates	updates	updates	updates	updates	UC	UC	1	1	NoChange				RX	yes
0	1	0	1	IN	x	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
IN Endpoint																				
ACK IN, STALL Bit = 0 (*Figure 14-3*)																				
1	1	0	1	OUT	x	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
1	1	0	1	IN	x	UC	x	UC	UC	UC	UC	1	UC	1	1	1	0	0	ACK (back)	yes
ACK IN, STALL Bit = 1 (*Figure 14-3*)																				
1	1	0	1	OUT	x	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
1	1	0	1	IN	x	UC	x	UC	UC	UC	UC	1	UC	UC	NoChange				STALL	yes
NAK IN																				
1	1	0	0	OUT	x	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
1	1	0	0	IN	x	UC	x	UC	UC	UC	UC	1	UC	UC	NoChange				NAK	yes
Reserved																				
0	1	1	1	Out	x	UC	x	UC	UC	UC	UC	UC	UC	UC	NoChange				Ignore	no
0	1	1	1	IN	x	UC	x	UC	UC	UC	UC	1	UC	UC	NoChange				TX	yes

21.0 Register Summary

	Address	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Read/Write)/ Both(B)	Default/ Reset
GPIO CONFIGURATION PORTS 0, 1, AND 2	0x00	Port 0 Data	P0								BBBBBBBB	00000000
	0x01	Port 1 Data	Reserved						P1[1:0]		------BB	00000000
	0x02	Port 2 Data	Reserved		D+(SCLK) State	D- (SDATA) State	Reserved	P2.2(Int Clk Mode only)	P2.1 (Int Clk Mode only)	P2.0 Vreg Pin State	--RR-RRR	00000000
	0x0A	GPIO Port 0 Mode 0	P0[7:0] Mode0								WWWWWWWW	00000000
	0x0B	GPIO Port 0 Mode 1	P0[7:0] Mode1								WWWWWWWW	00000000
	0x0C	GPIO Port 1 Mode 0	Reserved						P1[1:0] Mode0		------WW	00000000
	0x0D	GPIO Port 1 Mode 1	Reserved						P1[1:0] Mode1		------WW	00000000
	0x04	Port 0 Interrupt Enable	P0[7:0] Interrupt Enable								WWWWWWWW	00000000
	0x05	Port 1 Interrupt Enable	Reserved						P1[1:0] Interrupt Enable		------WW	00000000
	0x06	Port 0 Interrupt Polarity	P0[7:0] Interrupt Polarity								WWWWWWWW	00000000
	0x07	Port 1 Interrupt Polarity	Reserved						P1[1:0] Interrupt Polarity		------WW	00000000
Clock Config.	0xF8	Clock Configuration	Ext. Clock Resume Delay	Wake-up Timer Adjust Bit [2:0]			Low Voltage Reset Disable	Precision USB Clocking Enable	Internal Clock Output Disable	External Oscillator Enable	BBBBBBBB	00000000
ENDPOINT 0, I AND 2 CONFIGURATION	0x10	USB Device Address	Device Address Enable	Device Address							BBBBBBBB	00000000
	0x12	EP0 Mode	SETUP Received	IN Received	OUT Received	ACKed Transaction	Mode Bit				BBBBBBBB	00000000
	0x14	EP1 Mode Register	STALL	Reserved		ACKed Transaction	Mode Bit				B--BBBBB	00000000
	0x11, 0x13	EP0 and 1Counter	Data 0/1 Toggle	Data Valid	Reserved		Byte Count				BB--BBBB	00000000
USB-SC	0x1F	USB Status and Control	PS/2 Pull-up Enable	VREG Enable	USB Reset-PS/2 Activity Interrupt Mode	Reserved	USB Bus Activity	D+/D- Forcing Bit			BBB-BBBB	00000000
INTERRUPT	0x20	Global Interrupt Enable	Wake-up Interrupt Enable	GPIO Interrupt Enable	Reserved			1.024 ms Interrupt Enable	128 µs Interrupt Enable	USB Bus Reset-PS/2 Activity Intr. Enable	BB---BBB	00000000
	0x21	Endpoint Interrupt Enable	Reserved						EP1 Interrupt Enable	EP0 Interrupt Enable	------BB	00000000
TIMER	0x24	Timer LSB	Timer Bit [7:0]								RRRRRRRR	00000000
	0x25	Timer (MSB)	Reserved				Timer Bit [11:8]				----RRRR	00000000
PROC SC.	0xFF	Process Status & Control	IRQ Pending	Watch Dog Reset	Bus Interrupt Event	LVR/BOR Reset	Suspend	Interrupt Enable Sense	Reserved	Run	RBBBBR-B	See Section 18.0

	Parameter	Min	Max	Units	Conditions
V_{DI}	Differential Input Sensitivity	0.2		V	|(D+)–(D–)|
V_{CM}	Differential Input Common Mode Range	0.8	2.5	V	
V_{SE}	Single Ended Receiver Threshold	0.8	2.0	V	
C_{IN}	Transceiver Capacitance		20	pF	
I_{LO}	Hi-Z State Data Line Leakage	–10	10	µA	0 V < V_{in}<3.3 V (D+ or D– pins)
R_{PU}	External Bus Pull-up resistance (D–)	1.274	1.326	kΩ	1.3 kΩ ±2% to VREG[11]
R_{PD}	External Bus Pull-down resistance	14.25	15.75	kΩ	15 kΩ ±5% to Gnd
	PS/2 Interface				
V_{OLP}	Static Output Low		0.4	V	Isink = 5 mA, SDATA or SCLK pins
R_{PS2}	Internal PS/2 Pull-up Resistance	3	7	kΩ	SDATA, SCLK pins, PS/2 Enabled
	General Purpose I/O Interface				
R_{UP}	Pull-up Resistance	8	24	kΩ	
V_{ICR}	Input Threshold Voltage, CMOS mode	40%	60%	V_{CC}	Low to high edge, Port 0 or 1
V_{ICF}	Input Threshold Voltage, CMOS mode	35%	55%	V_{CC}	High to low edge, Port 0 or 1
V_{HC}	Input Hysteresis Voltage, CMOS mode	3%	10%	V_{CC}	High to low edge, Port 0 or 1
V_{ITTL}	Input Threshold Voltage, TTL mode	0.8	2.0	V	Ports 0, 1, and 2
V_{OL1A} V_{OL1B}	Output Low Voltage, high drive mode		0.8 0.4	V V	I_{OL1} = 50 mA, Ports 0 or 1[4] I_{OL1} = 25 mA, Ports 0 or 1[4]
V_{OL2}	Output Low Voltage, medium drive mode		0.4	V	I_{OL2} = 8 mA, Ports 0 or 1[4]
V_{OL3}	Output Low Voltage, low drive mode		0.4	V	I_{OL3} = 2 mA, Ports 0 or 1[4]
V_{OH}	Output High Voltage, strong drive mode	V_{CC}–2		V	Port 0 or 1, I_{OH} = 2 mA[4]
R_{XIN}	Pull-down resistance, XTALIN pin	50		kΩ	Internal Clock Mode only

Note:
11. The 200Ω internal resistance at the VREG pin gives a standard USB pull-up using this value. Alternately, a 1.5 kΩ, 5% pull-up from D– to an external 3.3V supply can be used.

24.0 Switching Characteristics

Parameter	Description	Min.	Max.	Unit	Conditions
	Internal Clock Mode				
F_{ICLK}	Internal Clock Frequency	5.7	6.3	MHz	Internal Clock Mode enabled
F_{ICLK2}	Internal Clock Frequency, USB mode	5.91	6.09	MHz	Internal Clock Mode enabled, Bit 2 of register 0xF8h is set (Precision USB Clocking)[12]
	External Oscillator Mode				
T_{CYC}	Input Clock Cycle Time	164.2	169.2	ns	USB Operation, with External ±1.5% Ceramic Resonator or Crystal
T_{CH}	Clock HIGH Time	0.45 t_{CYC}		ns	
T_{CL}	Clock LOW Time	0.45 t_{CYC}		ns	
	Reset Timing				
t_{START}	Time-out Delay after LVR/BOR	24	60	ms	
t_{WAKE}	Internal Wake-up Period	1	5	ms	Enabled Wake-up Interrupt[13]
t_{WATCH}	WatchDog Timer Period	10.1	14.6	ms	F_{OSC} = 6 MHz
	USB Driver Characteristics				
T_R	Transition Rise Time	75		ns	CLoad = 200 pF (10% to 90%[4])
T_R	Transition Rise Time		300	ns	CLoad = 600 pF (10% to 90%[4])
T_F	Transition Fall Time	75		ns	CLoad = 200 pF (10% to 90%[4])
T_F	Transition Fall Time		300	ns	CLoad = 600 pF (10% to 90%[4])
T_{RFM}	Rise/Fall Time Matching	80	125	%	t_r/t_f[4, 14]
V_{CRS}	Output Signal Crossover Voltage[17]	1.3	2.0	V	CLoad = 200 to 600 pF[4]
	USB Data Timing				
T_{DRATE}	Low Speed Data Rate	1.4775	1.5225	Mb/s	Ave. Bit Rate (1.5 Mb/s ±1.5%)
T_{DJR1}	Receiver Data Jitter Tolerance	–75	75	ns	To Next Transition[15]
T_{DJR2}	Receiver Data Jitter Tolerance	–45	45	ns	For Paired Transitions[15]
T_{DEOP}	Differential to EOP transition Skew	–40	100	ns	Note 15
T_{EOPR2}	EOP Width at Receiver	670		ns	Accepts as EOP[15]
T_{EOPT}	Source EOP Width	1.25	1.50	µs	
T_{UDJ1}	Differential Driver Jitter	–95	95	ns	To next transition, *Figure 24-5*
T_{UDJ2}	Differential Driver Jitter	–150	150	ns	To paired transition, *Figure 24-5*
T_{LST}	Width of SE0 during Diff. Transition		210	ns	
	Non-USB Mode Driver Characteristics				Note 16
T_{FPS2}	SDATA / SCK Transition Fall Time	50	300	ns	CLoad = 150 pF to 600 pF

Notes:
12. Initially F_{ICLK2}=F_{ICLK} until a USB packet is received.
13. Wake-up time for Wake-up Adjust Bits cleared to 000b (minimum setting)
14. Tested at 200 pF.
15. Measured at cross-over point of differential data signals.
16. Non-USB Mode refers to driving the D–/SDATA and/or D+/SCLK pins with the Control Bits of the USB Status and Control Register, with Control Bit 2 HIGH.
17. Per the USB 2.0 Specification, Table 7.7, Note 10, the first transition from the Idle state is excluded.

Table 26-1 below shows the die pad coordinates for the CY7C63221A-XC. The center location of each bond pad is relative to the center of the die which has coordinate (0,0) as shown above.

Table 26-1. CY7C63221A-XC Probe Pad Coordinates in microns ((0,0) to bond pad centers)

Pad Number	Pin Name	X (microns)	Y (microns)
1	P0.0	–351.75	995.00
2	P0.1	–543.20	995.00
3	P0.2	–734.65	995.00
4	P0.3	–861.05	779.25
5	P1.0	–861.05	587.80
6	Vss	–861.05	–949.65
7	Vpp	–468.20	–968.10
8	VREG	–300.40	–968.10
9	XTALIN	63.30	–968.10
10	XTALOUT	207.50	–968.10
11	Vcc	594.60	–968.10
12	D–	771.35	–968.10
13	D+	844.05	–863.10
14	P1.1	861.05	581.95
15	P0.7	861.05	773.95
16	P0.6	720.15	995.00
17	P0.5	528.70	995.00
18	P0.4	337.25	995.00

Document History Page

Document Title: CY7C63221/31A enCoRe™ Low-speed USB Peripheral Controller Document Number: 38-08028				
REV.	**ECN NO.**	**Issue Date**	**Orig. of Change**	**Description of Change**
**	116226	06/17/02	DSG	Change from Spec number: 38-01049 to 38-08028
*A	116976	10/23/02	BON	Reformat. Add note 5 to 24.0. Add DIE sale, Section 21.0. Change *Figure 9-1*
*B	270731	See ECN	BON	Replaced the 16-Lead (300-Mil) Molded SOIC S1 graphic with the correct 18-Lead (300-Mil) Molded SOIC S1 one in the Package Diagram Section and corrected part numbers for lead-free packages.