CY7C63000
CY7C63001
CY7C63100
CY7C63101
CY7C63200
CY7C63201
Universal Serial Bus Microcontroller

October 1996 - Revised June 26, 1997

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

1.0 Features

- **Low-cost solution for low-speed USB peripherals such as mouse, joystick, and gamepad**
- **USB Specification Compliance**
  - **Conforms to USB 1.5 Mbps Specification, Version 1.0**
  - **Conforms to USB HID Specification, Version 1.0**
  - **Supports 1 device address and 2 endpoints**
- **8-bit RISC microcontroller**
  - **Harvard architecture**
  - **6 MHz external ceramic resonator or clock crystal**
  - **12 MHz internal operation**
  - **USB optimized instruction set**
- **Internal memory**
  - **128 bytes of RAM**
  - **2K bytes of EPROM (CY7C63000, CY7C63100, CY7C63200)**
  - **4K bytes of EPROM (CY7C63001, CY7C63101, CY7C63201)**
- **I/O ports**
  - **Integrated USB transceivers**
  - **Up to 16 Schmitt trigger I/O pins with internal pull-up**
  - **Up to 8 I/O pins with LED drive capability**
  - **Special purpose I/O mode supports optimization of photo transistor and LED in mouse application**
  - **Maskable Interrupts on all I/O pins**
- **8-bit free-running timer**
- **Watchdog timer (WDT)**
- **Internal power-on reset (POR)**
- **Improved output drivers to reduce EMI**
- **Operating voltage from 4.0V to 5.25VDC**
- **Operating temperature from 0 to 70 degree Celsius**
- **Available in space saving and low cost 18-pin PDIP, 20-pin PDIP, 20-pin SOIC, and 24-pin SOIC packages**
- **Windowed packages also available to support program development: 18, 20, and 24-pin Windowed CerDIP**
- **Industry standard programmer support**

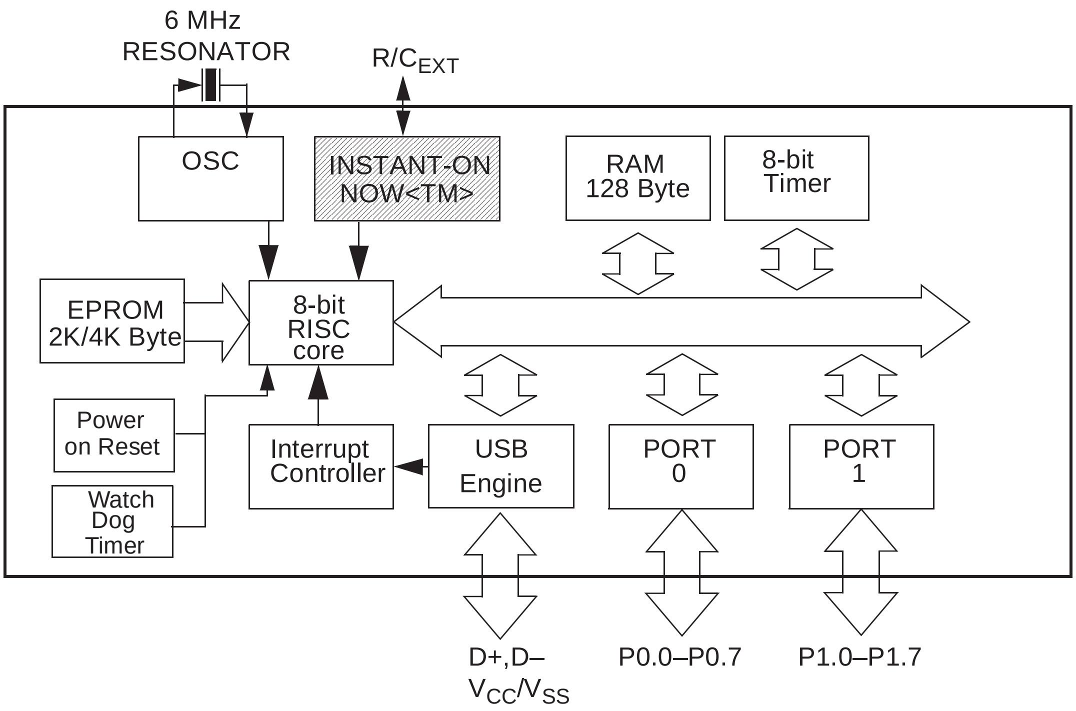
2.0 Functional Overview

The CY7C63xxx is a family of 8-bit RISC One Time Programmable (OTP) microcontrollers with a built-in 1.5-Mbps USB serial interface engine. The microcontroller features 35 instructions which are optimized for USB applications. There is 128 bytes of onboard RAM available incorporated into each microcontroller. The Cypress USB Controller accepts a 6 MHz ceramic resonator or a 6 MHz crystal as its clock source. This clock is doubled within the chip to provide a 12 MHz clock for the microprocessor.

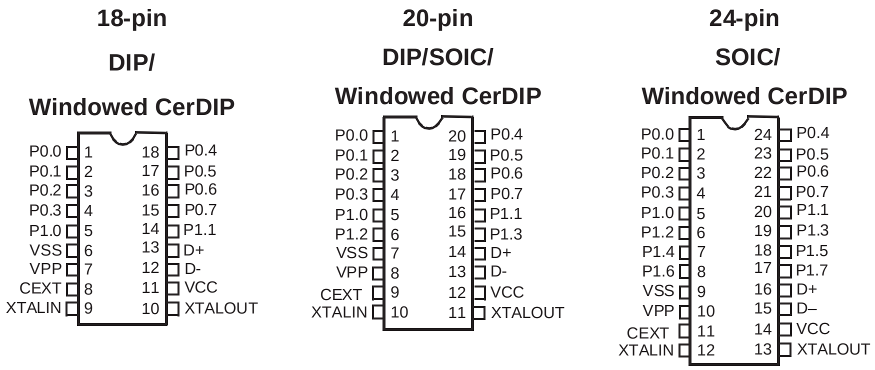
The microcontroller features two ports of up to sixteen general purpose I/Os (GPIOs). Each GPIO pin can be used to generate an interrupt to the microcontroller. Additionally, all pins in Port 1 are equipped with programmable drivers strong enough to drive LEDs. The GPIO ports feature low EMI emissions as a result of controlled rise and fall times, and unique output driver circuits in the microcontroller. The Cypress microcontrolles have a range of GPIOs to fit various applications; the CY7C630xx has twelve GPIO, the CY7C631xx has sixteen GPIO, and the CY7C632xx has ten GPIO. Notice that each part has eight ‘low-current’ ports (Port 0)with the remaining ports (Port 1) being ‘high-current’

The twelve GPIO CY7C6300x is available in is a 20-pin PDIP (-PC), 20-pin SOIC (-SC), and a 20-pin Windowed CerDIP. The sixteen GPIO CY7C6310x is available in a 24-pin SOIC (-SC) and a 24-pin Windowed CerDIP (-SC). The ten GPIO CY7C6320x is available in an 18-pin PDIP (-PC) and an 18-pin Windowed CerDIP (-WC).

Logic Block Diagram



Pin Configurations (Top View)


18-pin
DIP/
Windowed CerDIP

20-pin
DIP/SOIC/
Windowed CerDIP

24-pin
SOIC/
Windowed CerDIP

6311–1

3.0 Pin Definitions

Name	I/O	18-Pin	20-Pin	24-pin	Description
P0.0	I/O	1	1	1	Port 0 bit 0
P0.1	I/O	2	2	2	Port 0 bit 1
P0.2	I/O	3	3	3	Port 0 bit 2
P0.3	I/O	4	4	4	Port 0 bit 3
P0.4	I/O	18	20	24	Port 0 bit 4
P0.5	I/O	17	19	23	Port 0 bit 5
P0.6	I/O	16	18	22	Port 0 bit 6
P0.7	I/O	15	17	21	Port 0 bit 7
P1.0	I/O	5	5	5	Port 1 bit 0
P1.1	I/O	14	16	20	Port 1 bit 1
P1.2	I/O	–	6	6	Port 1 bit 2
P1.3	I/O	–	15	19	Port 1 bit 3
P1.4	I/O	–	–	7	Port 1 bit 4
P1.5	I/O	–	–	18	Port 1 bit 5
P1.6	I/O	–	–	8	Port 1 bit 6
P1.7	I/O	–	–	17	Port 1 bit 7
XTALIN	I	9	10	12	Crystal / Ceramic resonator in or external clock input
XTALOUT	O	10	11	13	Crystal / Ceramic resonator out
CEXT	I/O	8	9	11	Connects to external R/C timing circuit for optional suspend wakeup
D+	I/O	13	14	16	USB data+
D–	I/O	12	13	15	USB data–
VPP	–	7	8	10	Programming voltage supply, tie to ground during normal operation
V_{CC}	–	11	12	14	Voltage supply
V_{SS}	–	6	7	9	Ground

4.0 Pin Description

Name	Description
V_{DD}	1 pin. Connects to the USB power source or to a nominal 5V power supply. Actual V_{CC} range can vary between 4.0V and 5.25V
V_{SS}	1 pin. Connects to ground
V_{PP}	1 pin. Used in programming the on-chip EPROM. This pin should be tied to ground during normal operations.
XTALIN	1 pin. Input from an external ceramic resonator, crystal, or clock
XTALOUT	1 pin. Return path for the ceramic resonator or crystal
P0.0–P0.7, P1.0–P1.7	16 pins. P0.0–P0.7 are the 8 I/O lines in Port 0. P1.0–P1.7 are the 8 I/O lines in Port 1. Please note that P1.0–P1.1 are supported in the CY7C6320x and P1.0–P1.3 are supported in the CY7C6300x. All I/O pins are pulled up internally by 16KΩ resistors. However, the sink current of each pin can be programmed to one of sixteen levels. Besides functioning as general purpose I/O lines, each pin can be programmed as an interrupt input. The interrupt is edge-triggered, with programmable polarity.
D+, D–	2 pins. Open-drain I/O with 2 pins. Bidirectional USB data lines. An external 7.5 KΩ resistor must be connected between the D– pin and V_{CC} to select low-speed USB operation.
CEXT	1 pin. Open-drain output with Schmitt trigger input. The input is connected to a level-sensitive (HIGH) interrupt. CEXT may be connected to an external RC to generate a wake-up from Suspend mode. See Section 5.6.

5.0 Functional Description

The Cypress CY7C63000/1, CY7C63100/1, and CY7C63200/1 USB microcontrollers are optimized for human-interface computer peripherals such as a mouse, joystick, and gamepad. Cypress USB microcontrollers conform to the low-speed (1.5 Mbps) requirements of the USB Specification version 1.0. Each micorcontroller is a self-contained unit with a USB interface engine, USB transceivers, an 8-bit RISC microcontroller, a clock oscillator, timers, and program memories. It supports one USB device address and two end points.

The 6 MHz clock generated by the on-chip oscillator is stepped up to 12 MHz to drive the microcontroller. A RISC architecture with 35 instructions is chosen to provide the best balance between performance and product cost.

5.1 Memory Organization

The memory in the USB Controller is organized into user program memory in EPROM space and data memory in SRAM space.

5.1.1 Program Memory Organization

The 14-bit Program Counter (PC) is capable of addressing 16K bytes of program space. However, the program space of the CY7C63000, CY7C63100 and CY7C63200 is 2K bytes. For applications requiring more program space, the CY7C63001, CY7C63101 and CY7C63201 each offer 4K bytes of EPROM. The program memory space is divided into two functional groups: Interrupt Vectors and program code.

The interrupt vectors occupy the first 16 bytes of the program space. Each vector is 2 bytes long. After a reset, the Program Counter points to location zero of the program space. *Figure 5-1* shows the organization of the Program memory Space.

5.1.2 Security Fuse Bit

The Cypress USB microcontroller includes a security fuse bit. When the security fuse is programmed, the EPROM program memory outputs 0xFF to the EPROM programmer, thus protecting the user’s code.

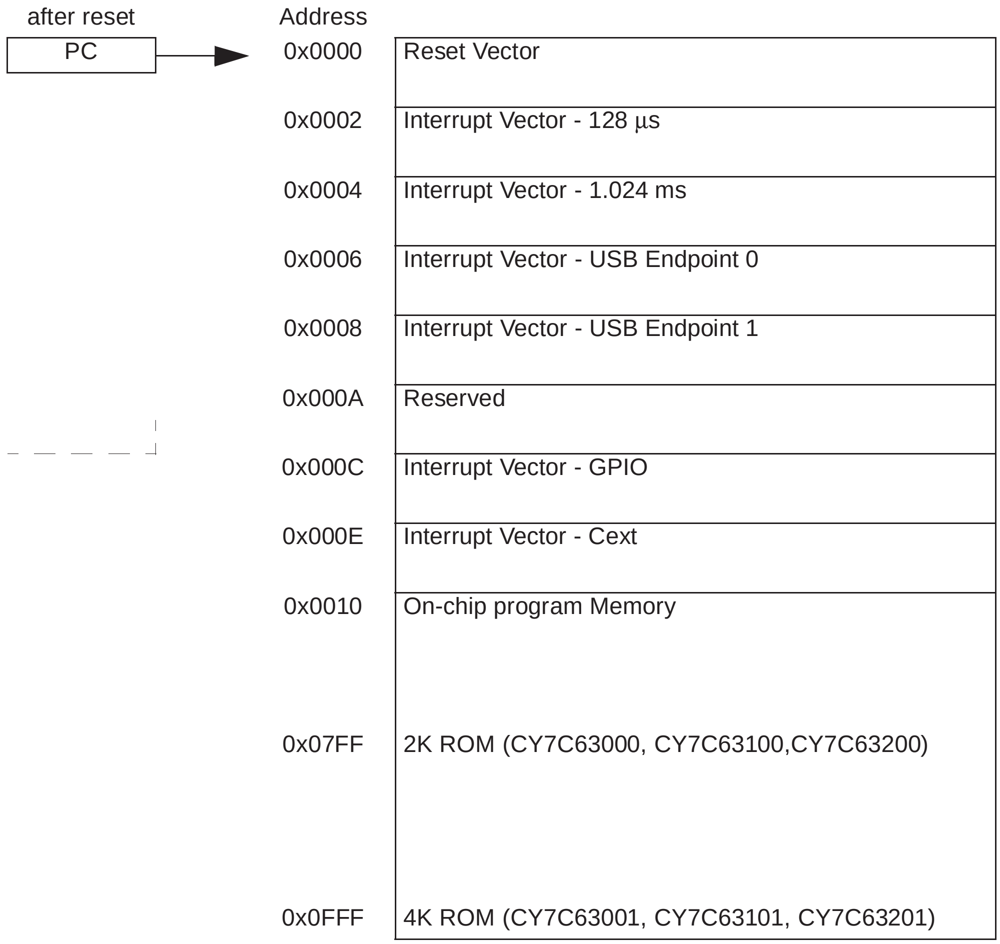


Figure 5-1. Program Memory Space

5.1.3 Data Memory Organization

The USB Controller includes 128 bytes of data RAM. The upper 16 bytes of the data memory are used as USB FIFOs for End Point 0 and End Point 1. Each end point is associated with an 8-byte FIFO.

The USB controller includes two pointers into data RAM, the Program Stack Pointer (PSP) and the Data Stack Pointer (DSP). The value of PSP after reset is 0x00. The PSP is incremented by 2 whenever a CALL instruction is executed and it is decremented by 2 whenever a RET instruction is used.

The DSP is pre-decremented by 1 whenever a PUSH instruction is executed and it is incremented by 1 after a POP instruction is used. The default value of the DSP after reset is 0x00, which would cause the first PUSH to write into USB FIFO space for End Point 1. Therefore, the DSP should be mapped to a location such as 0x70 before initiating any data stack operations. Refer to the Reset section for more information about DSP re-mapping after reset. *Figure 5-2* illustrates the Data Memory Space.

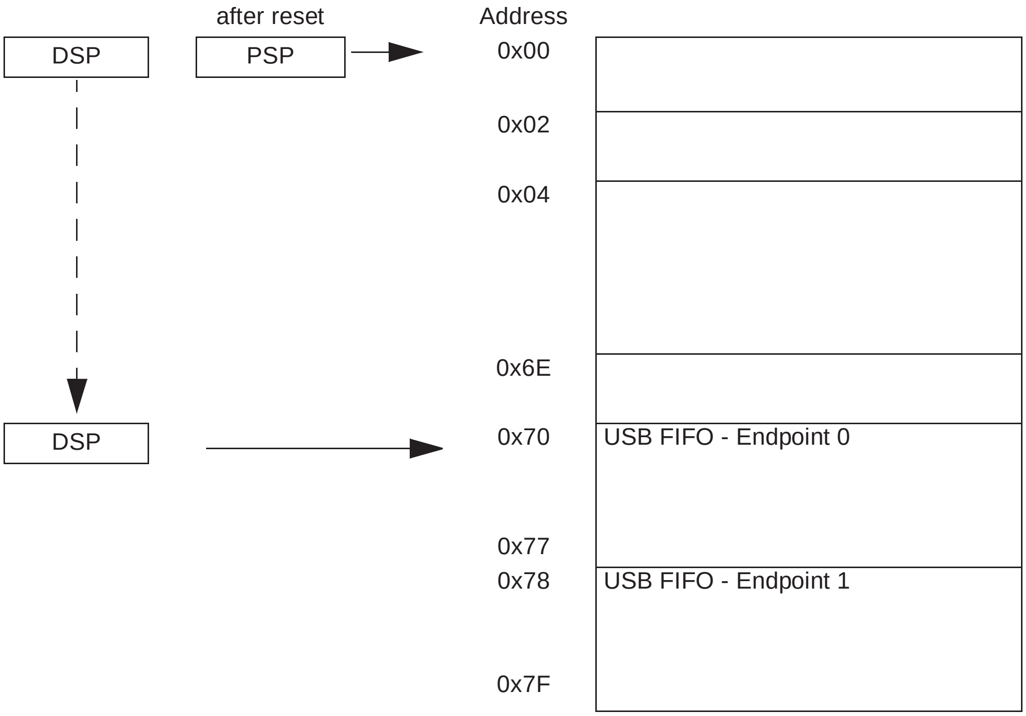


Figure 5-2. Data Memory Space

5.2 I/O Register Summary

I/O registers are accessed via the I/O Read (IORD) and I/O Write (IOWR, IOWX) instructions.

Table 5-1. I/O Register Summary

Register Name	I/O Address	Read/Write	Function
Port 0 Data	0x00	R/W	General purpose I/O Port (low current)
Port 1 Data	0x01	R/W	General purpose I/O Port (high current)
Port 0 Interrupt Enable	0x04	W	Interrupt enable for pins in Port 0
Port 1 Interrupt Enable	0x05	W	Interrupt enable for pins in Port 1
Port 0 Pull-up	0x08	W	Pull-up resistor control for Port 0 pins
Port 1 Pull-up	0x09	W	Pull-up resistor control for Port 1 pins
USB EP 0 TX Config.	0x10	R/W	USB End Point 0 transmit configuration
USB EP 1 TX Config.	0x11	R/W	USB End Point 1 transmit configuration
USB Device Address	0x12	R/W	USB device address
USB Status & Control	0x13	R/W	USB status and control
USB EP 0 RX Status	0x14	R/W	USB End Point 0 receive status

Table 5-1. I/O Register Summary (continued)

Register Name	I/O Address	Read/Write	Function
Global Interrupt Enable	0x20	R/W	Global Interrupt Enable
Watch Dog Timer	0x21	W	Watch Dog Timer clear
Cext Clear	0x22	R/W	External R-C Timing circuit control
Timer	0x23	R	Free-running timer
Port 0 Isink	0x30-0x37	W	Input sink current control for Port 0 pins. There is one Isink register for each pin. Address of the Isink register for pin 0 is located at 0x30 and the register address for pin 7 is located at 0x37
Port 1 Isink	0x38-0x3B	W	Input sink current control for Port 1 pins. There is one Isink register for each pin. Address of the Isink register for pin 0 is located at 0x38 and the register address for pin 3 is located at 0x3B
Status & Control	0xFF	R/W	Processor status and control

5.3 Reset

The USB Controller supports three types of resets. All registers are restored to their default states during a reset. The USB Device Address is set to 0 and all interrupts are disabled. In addition, the Program Stack Pointer (PSP) is set to 0x00 and the Data Stack Pointer (DSP) is set to 0x00. The user should set the DSP to location 0x70 to reserve 16 bytes of FIFO space. The assembly instructions to do so are:

```
Mov A, 70h     ; Move 70 hex into Accumulator, use 70 instead of 6F because the dsp is
               ; always decremented by 1 before data transfer in the PUSH instruction
Swap A, dsp    ; Move Accumulator value into dsp
```

The three reset types are:

1. Power On Reset (POR)
2. Watch Dog Reset (WDR)
3. USB Reset

The occurrence of a reset is recorded in the Status and Control Register located at I/O address 0xFF (*Figure 5-3*). Reading and writing this register are supported by the IORD and IOWR instructions. Bits 1, 2, and 7 are reserved and must be written as zeros during a write. During a read, reserved bit positions should be ignored. Bits 4, 5, and 6 are used to record the occurrence of POR, USB and WDR Reset respectively. The firmware can interrogate these bits to determine the cause of a reset. Bit 0 is the “Run” control, clearing this bit will stop the microcontroller. Once this bit is set to low, only a reset can set this bit HIGH.

The microcontroller resumes execution from ROM address 0X00 after a reset unless the Suspend bit (bit 3) of the Status and Control Register is set. Setting the Suspend bit stops the clock oscillator and the interrupt timers as well as powering-down the microcontroller. The detection of any USB activity will terminate the suspend condition.

7	6	5	4	3	2	1	0
W	R/W	R/W	R/W	R/W	W	W	R/W
Reserved	Watch Dog Reset	USB Reset	Power-on Reset	Suspend	Reserved	Reserved	Run

Figure 5-3. Status and Control Register (Address 0xFF)

5.3.1 Power-On Reset (POR)

Power On Reset (POR) occurs every time the power to the device is switched on. Bit 4 of the Status and Control Register is set to record this event (the register contents are set to 00011001 by the POR). The USB Controller is placed in suspended mode at the end of POR to conserve power (most device functions such as the clock oscillator, the timers, and the interrupt logic are turned off in the suspend mode). Only a non-idle USB Bus state will terminate the suspend mode and begin normal operations.

5.3.2 Watch Dog Reset (WDR)

The Watch Dog Timer Reset (WDR) occurs when the Most Significant Bit of the 4-bit Watch Dog Timer Register transitions from LOW to HIGH. Writing any value to the write-only Watch Dog Restart Register at 0x21 will clear the timer. The Watch Dog timer is clocked by a 1.024 ms clock from the free running timer. If 8 clocks occur between writes to the timer, a WDR occurs. Bit 6 of the Status and Control Register will be set to record the event. A Watch Dog Timer Reset lasts for 8.192 ms after which the microcontroller begins execution at ROM address 0x00. The USB transmitter is disabled by a Watch Dog Reset because the USB Device Address Register is cleared. Otherwise, the USB Controller would respond to all address 0 transactions. The transmitter remains disabled until the WDR bit in the Status and Control Register is reset to 0 by firmware.

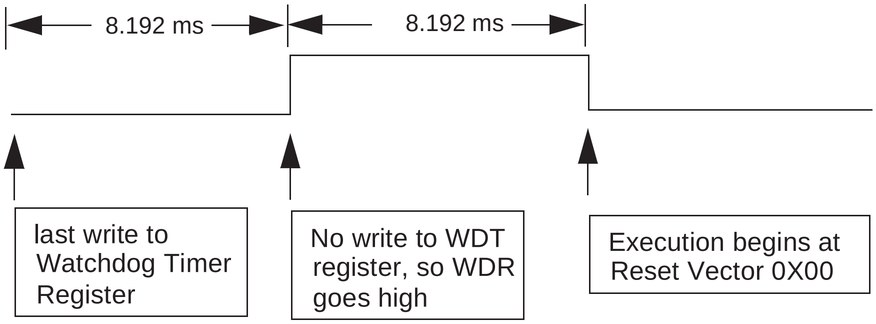


Figure 5-4. Watch Dog Reset (WDR)

5.3.3 USB Bus Reset

The USB Controller recognizes a USB Reset when a Single Ended Zero (SE0) condition persists for longer than 8 micro-seconds. SE0 is defined as the condition in which both the D+ line and the D– line are LOW. Bit 5 of the Status and Control Register will be set to record this event. If the USB reset happens while the device is suspended (such as after a POR), the suspend condition will be cleared and the clock oscillator will be restarted. However, the microcontroller is not released until the USB reset is removed.

5.4 On-chip Timer

The USB Controller is equipped with an 8-bit free-running timer driven by a clock one-sixth the crystal frequency. Bits 0 through 7 of the counter are readable from the read-only Timer Register located at I/O address 0x23. The Timer Register is cleared during a Power-On Reset. *Figure 5-5* illustrates the format of this register and *Figure 5-6* is its block diagram.

With a 6 MHz crystal, the timer resolution is 1 µs.

The timer generates two interrupts: the 128 µs interrupt and the 1.024 ms interrupt.

7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R
Count 7	Count 6	Count 5	Count 4	Count 3	Count 2	Count 1	Count 0

Figure 5-5. Timer Register (Address 0x23)

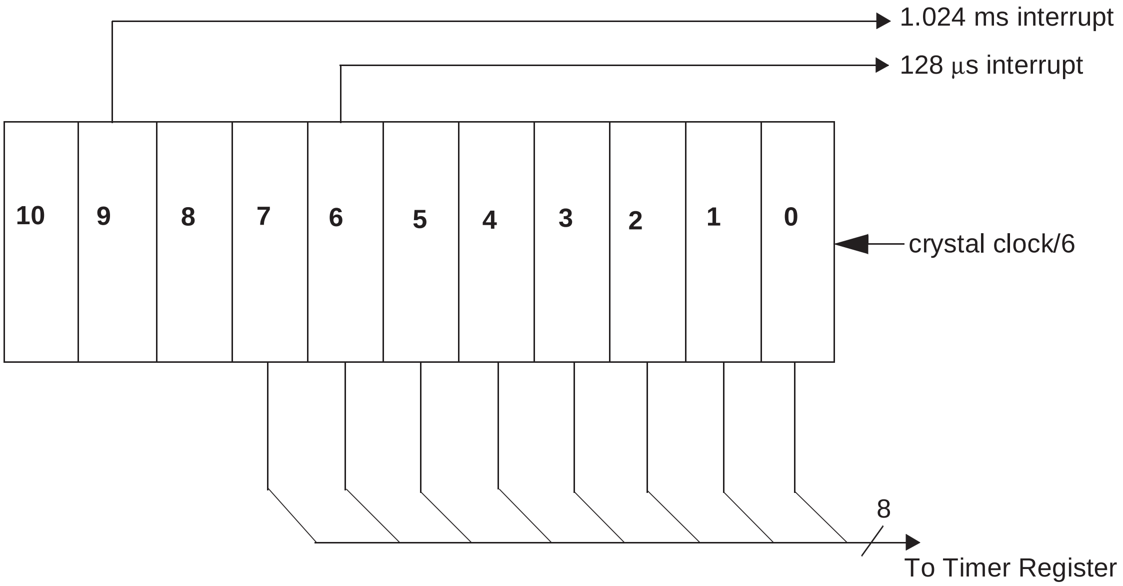


Figure 5-6. Timer Block Diagram

5.5 General Purpose I/O Ports

Interface with peripherals is conducted via 12 GPIO signals. These 12 signals are divided into two ports: Port 0 and Port 1. Port 0 contains eight lines (P0.0–P0.7) and Port 1 contains up to eigth lines (P1.0–P1.7), depending on the package. Both ports can be accessed by the IORD, IOWR and IOWX instructions. The Port 0 data register is located at I/O address 0x00 while the Port 1 data register is located at I/O address 0x01. The contents of both registers are set HIGH during a reset. Refer to *Figures 5-7* and *5-8* for the formats of the data registers. In addition to supporting general input/output functions, each I/O line can trigger an interrupt to the microcontroller. Please refer to the interrupt section for more details.

7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0

Figure 5-7. Port 0 Data Register (Address 0x00)

7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0

Figure 5-8. Port 1 Data Register (Address 0x01)

Each GPIO line includes an internal 16 KΩ resistor. This resistor provides both the pull-up function and slew control. Two factors govern the enabling and disabling of each resistor: the state of its associated Port Pull-up register bit and the state of the Data Register bit. The control bits in the Port Pull-up register are active LOW.

The output is HIGH when a “1” is written to the Data Register and the Port Pull-up register is “0”. Writing a “0” to the Data Register will disable the Pull-up resistor and output a LOW regardless of the setting in the Port Pull-up Register. The output will go to a high-Z state if the Data Register bit and the Port Pull-up Register bit are both “1”. *Figure 5-9* illustrates the block diagram of one I/O line. The Port Isink Register is used to control the output current level and it is described later in this section. *Table 5-2* is the Output Control truth table.

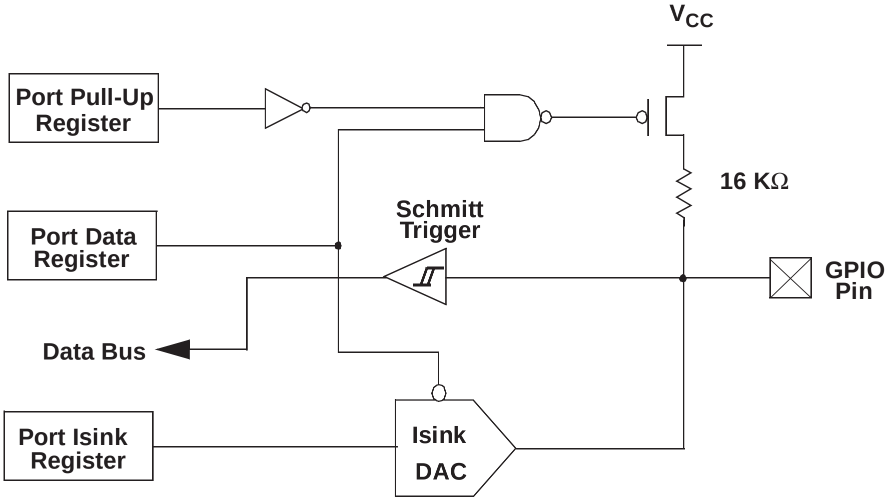


Figure 5-9. Block Diagram of an I/O Line

Table 5-2. Output Control Truth Table

Data Register	Port Pull-up Register	Output at I/O Pin
0	0	Sink Current (‘0’)
0	1	Sink Current (‘0’)
1	0	Pull-up Resistor (‘1’)
1	1	Hi-Z

To configure a GPIO pin as an input, a “1” should be written to the Port Data Register bit associated with that pin to disable the pull down function of the Isink DAC (see *Figure 5-9*).When the Port Data Register is read, the bit value will be a “1” if the voltage on the pin is greater than the Schmitt trigger threshold and “0” if below the threshold. In applications where an internal pull-up is required, the 16K-Ω pull-up resistor can be engaged by writing a “0” to the appropriate bit in the Port Pull-Up Register.

Both Port 0 and Port 1 Pull-up registers are write only (see *Figures 5-10* and *5-11*). Port 0 Pull-up is located at I/O address 0x08 and Port 1 Pull-up is mapped to address 0x09. The contents of the Port Pull-up registers are cleared during reset, allowing the outputs to be controlled by the state of the Data Registers. The Port pull-up registers also selects the polarity of transition that generates a GPIO interrupt. A “0” selects a HIGH to LOW transition while a “1” selects a LOW to HIGH transition.

7	6	5	4	3	2	1	0
W	W	W	W	W	W	W	W
Pull P0.7	Pull P0.6	Pull P0.5	Pull P0.4	Pull P0.3	Pull P0.2	Pull P0.1	Pull P0.0

Figure 5-10. Port 0 Pull-Up Register (Address 0x08)

7	6	5	4	3	2	1	0
W	W	W	W	W	W	W	W
Pull P1.7	Pull P1.6	Pull P1.5	Pull P1.4	Pull P1.3	Pull P1.2	Pull P1.1	Pull P1.0

Figure 5-11. Port 1 Pull-Up Register (Address 0x09)

Writing a “0” to the Data Register will drive the output LOW. Instead of providing a fixed output drive, the USB Controller allows the user to select an output sink current level for each I/O pin. The sink current of each output is controlled by a dedicated Port Isink Register. The lower 4 bits of this register contain a code selecting one of sixteen sink current levels. The upper 4 bits of the register are ignored. The format of the Port Isink Register is shown in *Figure 5-12*.

7	6	5	4	3	2	1	0
W	W	W	W	W	W	W	W
Isink7	Isink6	Isink5	Isink4	Isink3	Isink2	Isink1	Isink0

Figure 5-12. Port Isink Register for One GPIO Line

Port 0 is a low current port suitable for connecting photo transistors. Port 1 is a high current port capable of LED drive. See section 7.0 for current ranges. 0000 is the lowest drive strength. 1111 is the highest.

The write-only sink current control registers for Port 0 outputs are assigned from I/O address 0x30 to 0x37 with the control bits for P00 starting at 0x30. Port 1 sink current control registers continue from I/O address 0x38 to 0x3B. All sink current control registers are cleared during a reset, resulting in the minimum drive setting.

5.6 Instant-on Feature (Suspend Mode)

The USB Controller can be placed in a low-power state by setting the Suspend bit (bit 3) of the Status and Control register. Almost all logic blocks in the device are turned off except the USB receiver, the GPIO interrupt logic, and the Cext interrupt logic. The clock oscillator as well as the free-running and watch dog timers are shut down.

The suspend mode will be terminated when one of the three following conditions occur:

1. USB activity
2. A GPIO interrupt
3. Cext interrupt

The clock oscillator, GPIO and timers restart immediately on exiting suspend mode. The USB engine and microcontroller return to a fully functional state at most 256 us later. The microcontroller will execute the instruction following the I/O write that placed the device into suspend mode before servicing any interrupt requests.

Both the GPIO interrupt and the Cext interrupt allow the USB Controller to wake-up periodically and poll potentiometers, optics, and other system components while maintaining a very low average power consumption.

To use Cext to generate an “Instant-on” interrupt, the pin is connected to ground with an external capacitor and connected to VCC with an external resistor. A “0” is written to the Cext register located at I/O address 0x22 to discharge the capacitor. A “1” is then written to disable the open-drain output driver. A Schmitt trigger input circuit monitors the input and generates a wake-up interrupt when the input voltage rises above the input threshold. By changing the values of the external resistor and capacitor, the user can fine tune the charge rate of the R-C timing circuit. The format of the Cext register is shown in *Figure 5-13*. Reading the register returns the value of the Cext pin. During a reset, the Cext is HIGH.

7	6	5	4	3	2	1	0
							R/W
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Cext

Figure 5-13. The Cext Register (Address 0x22)

5.7 XTALIN/XTALOUT

XTALIN and XTALOUT are the crystal oscillator pins. A 6 MHz crystal or ceramic resonator should be connected to these pins.The feedback capacitors and bias resistor are internal to the IC.

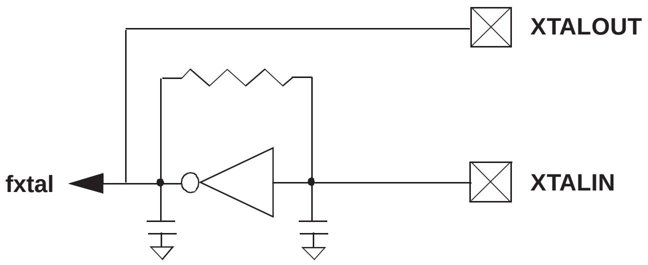


Figure 5-14. Clock Oscillator On-chip Circuit

5.8 Interrupts

Interrupts are generated by the General Purpose I/O lines, the Cext pin, the internal timer, and the USB engine. All interrupts except Reset are maskable by the Global Interrupt Enable Register. Access to this register is accomplished via IORD, IOWR and IOWX instructions to address 0x20. Writing a “1” to a bit position enables the interrupt associated with that position. During a reset, the contents the Interrupt Enable Register are cleared, disabling all interrupts. *Figure 5-15* illustrates the format of the Global Interrupt Enable Register.

7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Wake-up Interrupt Enable	GPIO Interrupt Enable	Reserved	USB EP1 Interrupt Enable	USB EP0 Interrupt Enable	1.024 ms Interrupt Enable	128 us Interrupt Enable	Reserved

Figure 5-15. Global Interrupt Enable Register (Address 0x20)

The interrupt controller contains a separate latch for each interrupt except the Wake-up interrupt. When an interrupt is generated it is latched as a pending interrupt. It will stay as a pending interrupt until it is serviced or a reset occurs. The Wake-up interrupt is not latched, and is pending whenever the Cext pin is high. A pending interrupt will only generate an interrupt request if it is enabled in the Global Interrupt Enable Register. The highest priority interrupt request will be serviced following the execution of the current instruction.

When servicing an interrupt, the hardware will first disable all interrupts by clearing the Global Interrupt Enable Register. Next, the interrupt latch of the current interrupt is cleared. This is followed by a CALL instruction to the ROM address associated with the interrupt being serviced (i.e., the Interrupt Vector). The instruction in the interrupt table is typically a JMP instruction to the address of the Interrupt Service Routine (ISR). The user can re-enable interrupts in the interrupt service routine by writing to the appropriate bits in the Global Interrupt Enable Register. Interrupts can be nested to a level limited only by the available stack space.

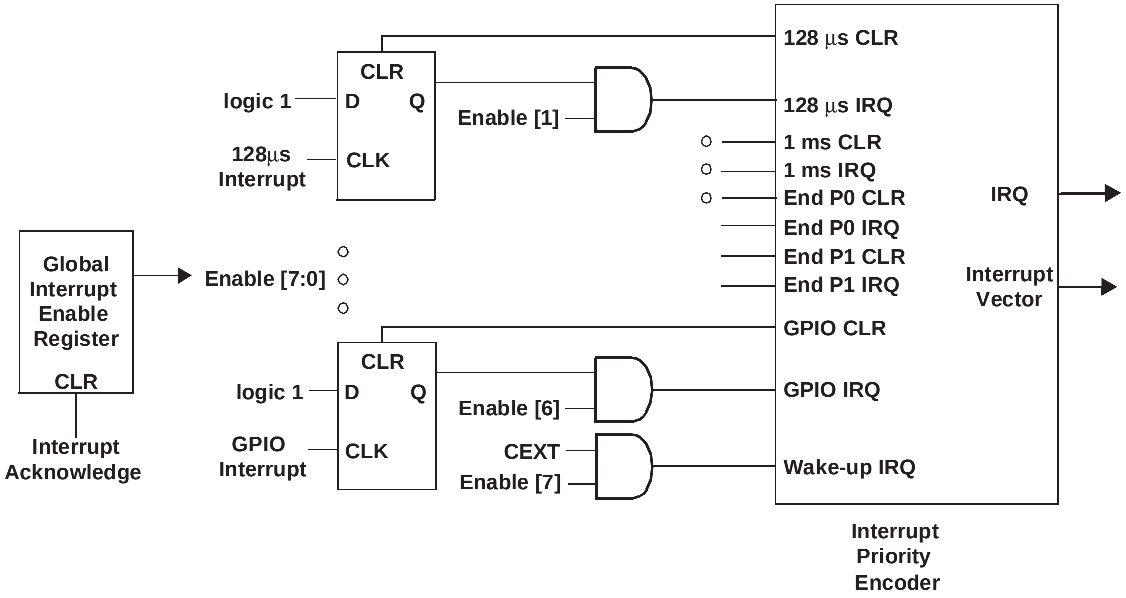


Figure 5-16. Interrupt Controller Logic Block Diagram

The Program Counter value as well as the Carry and Zero flags (CF, ZF) are automatically stored onto the Program Stack by the CALL instruction as part of the interrupt acknowledge process. The user firmware is responsible for insuring that the processor state is preserved and restored during an interrupt. For example the PUSH A instruction should be used as the first command in the ISR to save the accumulator value and the POP A instruction should be used just before the RET instruction to restore the accumulator value. The program counter CF and ZF are restored when the RET instruction is executed.

The Interrupt Vectors supported by the USB Controller are listed in *Table 5-3*. Interrupt Vector 0 (Reset) has the highest priority, Interrupt Vector 7 has the lowest priority. Because the JMP instruction is 2 bytes long, the interrupt vectors occupy 2 bytes.

Table 5-3. Interrupt Vector Assignments

Interrupt Vector Number	ROM Address	Function
0	0x00	Reset
1	0x02	128 µs timer interrupt
2	0x04	1.024 ms timer interrupt
3	0x06	USB end point 0 interrupt
4	0x08	USB end point 1 interrupt
5	0x0A	Reserved
6	0x0C	GPIO interrupt
7	0x0E	Wake-up interrupt

5.8.1 Interrupt Latency

Interrupt latency can be calculated from the following equation:

Interrupt Latency = (Number of clock cycles remaining in the current instruction) + (10 clock cycles for the CALL instruction) + (5 clock cycles for the JMP instruction)

For example, if a 5 clock cycle instruction such as JC is being executed when an interrupt occurs, the first instruction of the Interrupt Service Routine will execute a min. of 16 clocks (1+10+5) or a max. of 20 clocks (5+10+5) after the interrupt is issued. The interrupt latches are sampled at the rising edge of the last clock cycle in the current instruction.

5.8.2 GPIO Interrupt

The General Purpose I/O interrupts are generated by signal transitions at the Port 0 and Port 1 I/O pins. GPIO interrupts are edge sensitive with programmable interrupt polarities. Setting a bit HIGH in the Port Pull-up Register (see *Figure 5-10* and *5-11*) selects a LOW to HIGH interrupt trigger for the corresponding port pin. Setting a bit LOW activates a HIGH to LOW interrupt trigger. Each GPIO interrupt is maskable on a per-pin basis by a dedicated bit in the Port Interrupt Enable Register. Writing a “1” enables the interrupt. *Figure 5-17* and *Figure 5-18* illustrate the format of the Port Interrupt Enable Registers for Port 0 and Port 1 located at I/O address 0x04 and 0x05 respectively. These write only registers are cleared during reset, thus disabling all GPIO interrupts.

7	6	5	4	3	2	1	0
W	W	W	W	W	W	W	W
P0.7 Int En	P0.6 Int En	P0.5 Int En	P0.4 Int En	P0.3 Int En	P0.2 Int En	P0.1 Int En	P0.0 Int En

Figure 5-17. Port 0 Interrupt Enable Register (Address 0x04)

7	6	5	4	3	2	1	0
W	W	W	W	W	W	W	W
P1.7 Int En	P1.6 Int En	P1.5 Int En	P1.4 Int En	P1.3 Int En	P1.2 Int En	P1.1 Int En	P1.0 Int En

Figure 5-18. Port 1 Interrupt Enable Register (Address 0x05)

A block diagram of the GPIO interrupt logic is shown in *Figure 5-19*. The bit setting in the Port Pull-up Register selects the interrupt polarity. If the selected signal polarity is detected on the I/O pin a HIGH signal is generated. If the Port Interrupt Enable bit for this pin is HIGH and no other port pins are requesting interrupts, then the 12-input OR gate will issue a LOW to HIGH signal to clock the GPIO interrupt flip flop. The output of the flip flop is further qualified by the Global GPIO Interrupt Enable bit before it is processed by the Interrupt Priority Encoder. Both the GPIO interrupt flip flop and the Global GPIO Enable bit are cleared during GPIO interrupt acknowledge by on-chip hardware.

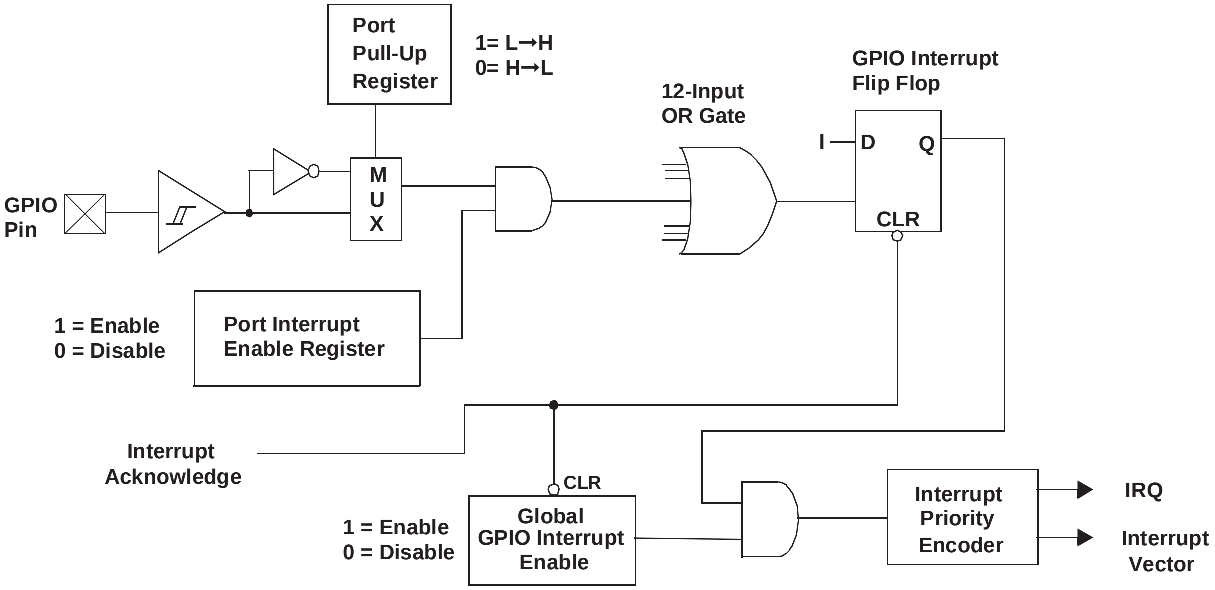


Figure 5-19. GPIO Interrupt Logic Block Diagram

Please note that if one port pin triggered an interrupt, no other port pins can cause a GPIO interrupt until that port pin has returned to its inactive (non-trigger) state or its corresponding port interrupt enable bit is cleared. The USB Controller does not assign interrupt priority to different port pins and the Port Interrupt Enable Registers are not cleared during the interrupt acknowledge process. When a GPIO interrupt is serviced, the ISR must poll the ports to determine which pin caused the interrupt.

5.8.3 USB Interrupt

A USB End Point 0 interrupt is generated after the host has written data to End Point 0 or after the USB Controller has transmitted a packet from End Point 0 and receives an ACK from the host. An OUT packet from the host which is NAKd by the USB Controller will not generate an interrupt. This interrupt is masked by the USB EP0 Interrupt Enable bit (bit 3) of the Global Interrupt Enable Register.

A USB End Point 1 interrupt is generated after the USB Controller has transmitted a packet from End Point 1 and has received an ACK from the host. This interrupt is masked by the USB EP1 Interrupt Enable bit (bit 4) of the Global Interrupt Enable Register.

5.8.4 Timer Interrupt

There are two timer interrupts: the 128 µs interrupt and the 1.024 ms interrupt. They are masked by bits 1 and 2 of the Global Interrupt Enable Register respectively. The user should disable both timer interrupts before going into the suspend mode to avoid possible conflicts between servicing the interrupts first or the suspend request first.

5.8.5 Wake-up Interrupt

A wake-up interrupt is generated when the Cext pin is HIGH. It is level sensitive and is not latched to the interrupt controller. It can be masked by the Wake-up Interrupt Enable bit (bit 7) of the Global Interrupt Enable Register. This interrupt can be used to perform periodic checks on attached peripherals when the USB Controller is placed in the low-power suspend mode. See the Instant-On Feature section for more details.

5.9 USB Engine

The USB engine includes the Serial Interface Engine (SIE) and the low-speed USB I/O transceivers. The SIE block performs most of the USB interface functions with only minimal support from the microcontroller core. Two end points are supported. End Point 0 is used to receive and transmit control (including setup) packets while End Point 1 is only used to transmit data packets.

The USB SIE processes USB bus activity at the transaction level independently. It does all the NRZI encoding/decoding and bit stuffing/unstuffing. It also determines token type, checks address and endpoint values, generates and checks CRC values and controls the flow of data bytes between the bus and the End Point FIFOs.

The firmware handles higher level and function specific tasks. During control transfers the firmware must interpret device requests and respond correctly. It also must coordinate Suspend/Resume, verify and select DATA toggle values, and perform function specific tasks.

The USB engine and the firmware communicate though the End Point FIFOs, USB End Point interrupts, and the USB registers described in the sections below.

5.9.1 USB Enumeration Process

The USB Controller provides a USB Device Address Register at I/O location 0x12. Reading and writing this register is achieved via the IORD and IOWR instructions. The register contents are cleared during a reset, setting the USB address of the USB Controller to 0. *Figure 5-20* shows the format of the USB Address Register.

7	6	5	4	3	2	1	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reserved	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0

Figure 5-20. USB Device Address Register (Address 0x12)

Typical enumeration steps:

1. The host computer sends a SETUP packet followed by a DATA packet to USB address 0 requesting the Device descriptor.
2. The USB Controller decodes the request and retrieves its Device descriptor from the program memory space.
3. The host computer performs a control read sequence and the USB Controller responds by sending the Device descriptor over the USB bus.
4. After receiving the descriptor, the host computer sends a SETUP packet followed by a DATA packet to address 0 assigning a new USB address to the device.
5. The USB Controller stores the new address in its USB Device Address Register after the no-data control sequence completes.
6. The host sends a request for the Device descriptor using the new USB address.
7. The USB Controller decodes the request and retrieves the Device descriptor from the program memory.
8. The host performs a control read sequence and the USB Controller responds by sending its Device descriptor over the USB bus.
9. The host generates control reads to the USB Controller to request the Configuration and Report descriptors.
10. The USB Controller retrieves the descriptors from its program space and returns the data to the host over the USB.
11. Enumeration is complete after the host has received all the descriptors.

5.9.2 End Point 0

All USB devices are required to have an end point number 0 that is used to initialize and manipulate the device. End Point 0 provides access to the device’s configuration information and allows generic USB status and control accesses.

End Point 0 can receive and transmit data. Both receive and transmit data share the same 8-byte End Point 0 FIFO located at data memory space 0x70 to 0x77. Received data may overwrite the data previously in the FIFO.

5.9.2.1 End Point 0 Receive

After receiving a packet and placing the data into the End Point 0 FIFO, the USB Controller updates the USB End Point 0 RX register to record the receive status and then generates an USB End Point 0 interrupt. The format of the End Point 0 RX Register is shown in *Figure 5-21*.

7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
Count 3	Count 2	Count 1	Count 0	Data Toggle	IN	OUT	SETUP

Figure 5-21. USB End Point 0 RX Register (Address 0x14)

This is a read/write register located at I/O address 0x14. Any write to this register will clear all bits except bit 3 which remains unchanged. All bits are cleared during reset.

Bit 0 is set to 1 when a SETUP token for End Point 0 is received. Once set to a 1 this bit remains high until it is cleared by an I/O write or a reset. While the data following a SETUP is being received by the USB engine, this bit will not be cleared by an I/O write. User firmware writes to the USB FIFOs are disabled when bit 0 is set. This prevents SETUP data from being overwritten.

Bits 1 and 2 are updated whenever a valid token is received on End Point 0. Bit 1 is set to 1 if an OUT token is received and cleared to 0 if any other token is received. Bit 2 is set to 1 if an IN token is received and cleared to 0 if any other token is received.

Bit 3 shows the Data Toggle status of DATA packets received on End Point 0. This bit is updated for DATA following SETUP tokens and for DATA following OUT tokens if Stall (bit 5 of 0x10) is not set and either EnableOuts or StatusOuts (bits 3 and 4 of 0x13) are set.

Bits 4 to 7 are the count of the number of bytes received in a DATA packet. The two CRC bytes are included in the count, so the count value is two greater than the number of data bytes received. The count is always updated and the data is always stored in the FIFO for DATA packets following a SETUP token. The count for DATA following an OUT token is updated if Stall (bit 5 of 0x10) is 0 and either EnableOuts or StatusOuts (bits 3 and 4 of 0x13) are 1. The DATA following an OUT will be written into the FIFO if EnableOuts is set to 1 and Stall and StatusOuts are 0.

A maximum of 8 bytes are written into the End Point 0 FIFO. If there are less the 8 bytes of data the CRC is written into the FIFO.

Due to register space limitations, the Receive Data Invalid bit is located in the USB End Point 0 TX Configuration Register. Refer to the End Point 0 Transmit section for details. This bit is set by the SIE if an error is detected in a received DATA packet.

The table below summarizes the USB Engine response to SETUP and OUT transactions on End Point 0. In the Data Packet column ‘Error’ represents a packet with a CRC, PID or bit stuffing error, or a packet with more than 8 bytes of data. ‘Valid’ is a packet without an Error. ‘Status’ is a packet that is a valid control read Status stage, while ‘N/Status’ is not a correct Status stage (see section 5.9.4). The ‘Stall’ bit is described is section 5.9.2.2. The ‘StatusOuts’ and ‘EnableOuts’ bits are described in section 5.9.4.

Control Bit Settings			Received Packets		USB Engine Response				
Stall	Status Out	Enable Out	Token Type	Data Packet	FIFO Write	Toggle Update	Count Update	Interrupt	Reply
-	-	-	SETUP	Valid	Yes	Yes	Yes	Yes	ACK
-	-	-	SETUP	Error	Yes	Yes	Yes	Yes	None
0	0	1	OUT	Valid	Yes	Yes	Yes	Yes	ACK
0	0	1	OUT	Error	Yes	Yes	Yes	Yes	None
0	0	0	OUT	Valid	No	No	No	No	NAK
0	0	0	OUT	Error	No	No	No	No	None
1	0	0	OUT	Valid	No	No	No	No	STALL
1	0	0	OUT	Error	No	No	No	No	None
0	1	0	OUT	Status	No	Yes	Yes	Yes	ACK
0	1	0	OUT	N/Status	No	Yes	Yes	Yes	STALL
0	1	0	OUT	Error	No	Yes	No	No	None

Figure 5-22. USB Engine Response to SETUP and OUT transactions on End Point 0

5.9.2.2 End Point 0 Transmit

The USB End Point 0 TX Register located at I/O address 0x10 controls data transmission from End Point 0 (see *Figure 5-23*). This is a read/write register. All bits are cleared during reset.

7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Enable Respond to IN packets	Data 1/0	Stall	Data Invalid	Count 3	Count 2	Count 1	Count 0

Figure 5-23. USB End Point 0 TX Configuration Register (Address 0x10)

Bits 0 to 3 indicate the numbers of data bytes to be transmitted during an IN packet, valid values are 0 to 8 inclusive.

Bit 4 indicates that a received DATA packet error (CRC, PID, or bitstuffing error) occurred during a SETUP or OUT data phase.

Setting the Stall bit (bit 5) will stall IN and OUT packets. This bit is cleared whenever a SETUP packet is received by End Point 0.

Bit 6 (Data 1/0) must be set to either 0 or 1 to select the DATA packet’s toggle state, 0 for DATA0, 1 for DATA1.

After the transmit data has been loaded into the FIFO, bit 6 should be set according to the data toggle state and bit 7 set to "1". This enables the USB Controller to respond to an IN packet. Bit 7 is cleared and an End Point 0 interrupt is generated by the SIE once the host acknowledges the data transmission. Bit 7 is also cleared when a SETUP token is received. The Interrupt Service Routine can check bit 7 to confirm that the data transfer was successful.

5.9.3 End Point 1

End Point 1 is capable of transmit only. The data to be transmitted is stored in the 8-byte End Point 1 FIFO located at data memory space 0x78 to 0x7F.

5.9.3.1 End Point 1 Transmit

Transmission is controlled by the USB End Point 1 TX Register located at I/O address 0x11 (see *Figure 5-24*). This is a read/write register. All bits are cleared during reset.

7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Enable Respond to IN packets	Data 1/0	Stall	End Point 1 Enable	Count 3	Count 2	Count 1	Count 0

Figure 5-24. USB End Point 1 TX Configuration Register (Address 0x11)

Bits 0 to 3 indicate the numbers of data bytes to be transmitted during an IN packet, valid values are 0 to 8 inclusive.

Bit 4 must be set before End Point 1 can be used. If this bit is cleared, the USB Controller will ignore all traffic to End Point 1.

Setting the Stall bit (bit 5) will stall IN and OUT packets until this bit is cleared.

Bit 6 (Data 1/0) must be set to either 0 or 1 depending on the data packet's toggle state, 0 for DATA0, 1 for DATA1.

After the transmit data has been loaded into the FIFO, bit 6 should be set according to the data toggle state and bit 7 set to "1". This enables the USB Controller to respond to an IN packet. Bit 7 is cleared and an End Point 1 interrupt is generated by the SIE once the host acknowledges the data transmission.

5.9.4 USB Status and Control

USB status and control is regulated by USB Status and Control Register located at I/O address 0x13 as shown in *Figure 5-25*. This is a read/write register. All reserved bits must be written to zero. All bits in the register are cleared during reset.

7	6	5	4	3	2	1	0
			R/W	R/W		R/W	R/W
Reserved	Reserved	Reserved	Enable Outs	StatusOuts	Reserved	Force Resume	Bus Activity

Figure 5-25. USB Status and Control Register (Address 0x13)

Bit 0 will be set by the SIE if any USB activity except idle (D+ LOW, D– HIGH) is detected. The user program should check and clear this bit periodically to detect any loss of bus activity. Writing a 0 to this bit clears it. Writing a 1 does not change its value.

Bit 1 is used to force the on-chip USB transmitter to the K state which will send a Resume signal to the host.

Bit 2 is a reserved bit that must be set to 0.

Bit 3 is used to automatically respond to the Status stage OUT of a control read transfer on End Point 0. A valid Status stage OUT contains a DATA1 packet with 0 bytes of data. If the StatusOuts bit is set, the USB engine will respond to a valid Status stage OUT with an ACK, and any other OUT with a STALL. The data is not written into the FIFO when this bit is set. This bit is cleared when a SETUP token is received by End Point 0.

Bit 4 is used to enable the receiving of End Point 0 OUT packets. When this bit is set to 1, the data from an OUT transaction to be written into the End Point 0 FIFO and the USB engine responds with an ACK. If this bit is 0, data will not be written to the FIFO and the response is a NAK. This bit is cleared following a SETUP or OUT transaction.

5.10 Instruction Set Summary

Table 5-4. Instruction Set Map

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	OE	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
IPRET	addr	1E	13		RRC		3E	4
XPAGE		1F	4		RET		3F	8
JMP	addr	8x	5		JC	addr	Cx	5
CALL	addr	9x	10		JNC	addr	Dx	5
JZ	addr	Ax	5		JACC	addr	Ex	7
JNZ	addr	Bx	5		INDEX	addr	Fx	14

6.0 Absolute Maximum Ratings

Storage Temperature	–65°C to +150°C
Ambient Temperature with Power Applied	–0°C to +70°C
Supply voltage on V_{CC} relative to V_{SS}	–0.5V to +7.0V
DC input voltage	–0.5V to $+V_{CC}$+0.5V
DC voltage applied to outputs in High Z state	–0.5V to $+V_{CC}$+0.5V
Max. output current into Port 1 pins	60 mA
Max. output current into non-Port 1 pins	10 mA
Power dissipation	300 mW
Static discharge voltage	>2000V
Latch-up current	>200 mA

7.0 DC Characteristics Fosc = 6 MHz; Operating Temperature = 0 to 70°C

	Parameter	Min	Max	Units	Conditions
	General				
V_{cc}	Operating Voltage	4.0	5.25	V	
V_{max}	Maximum applied voltage	–0.5	6.5	V	
I_{CC}	V_{cc} Operating Supply Current		50	mA	
I_{SB1}	Supply Current - Suspend Mode		100	µA	Oscillator off, D– > Voh min
I_{SB2}	Supply Current - Start-up Mode		4	mA	V_{cc} = 5.0V
V_{pp}	Programming Voltage (disabled)	–0.4	0.4	V	
t_{start}	Resonator Start-up Interval		256	µs	V_{cc} = 5.0V, ceramic resonator
t_{int1}	Internal timer #1 interrupt period	128	128	µs	
t_{int2}	Internal timer #2 interrupt period	1.024	1.024	ms	
t_{watch}	WatchDog timer period	7.168	8.192	ms	
	Power On Reset				
V_{rst}	POR Voltage	2.0	3.4	V	NOTE [2, 6]
t_{vccs}	VCC reset slew	0.5	100	ms	linear ramp V_{CC}: 0 to Vrst
	USB Interface				
V_{oh}	Static Output High	2.8	3.6	V	15k ± 5% Ω to Gnd [3, 4]
V_{ol}	Static Output Low		0.3	V	NOTE 4
	General Purpose I/O				
R_{up}	Pull-up resistance	8K	24K	Ωs	
$I_{sink0(0)}$	Port 0 sink current (0), lowest current	0.1	0.3	mA	Vout = 2.0 V DC, Port 0 only [4]
$I_{sink0(F)}$	Port 0 sink current (F), highest current	0.5	1.5	mA	Vout = 2.0 V DC, Port 0 only [4]
$I_{sink1(0)}$	Port 1 sink current (0), lowest current	1.6	4.8	mA	Vout = 2.0 V DC, Port 1 only [4]
$I_{sink1(F)}$	Port 1 sink current (F), highest current	8	24	mA	Vout = 2.0 V DC, Port 1 only [4]
I_{range}	Sink current max/min	4.5	5.5		Vout = 2.0 V DC, Port 0 or 1 [1, 10]
I_{lin}	Differential nonlinearity		0.5	lsb	Port 0 or Port 1 [5]
I_{il}	Input leakage current		50	nA	CEXT only
I_{ol}	Sink current	6	18	mA	CEXT only

Notes:

1. Per Table 7-6 of revision 1.0 of USB specification, for Cload of 100–350pF.
2. Power on Reset will occur until the voltage on V_{CC} increases above Vrst.
3. Rx: external idle resistor, 7.5 KΩ, 2%, to V_{CC}.

7.0 DC Characteristics (continued) Fosc = 6 MHz; Operating Temperature = 0 to 70°C

	Parameter	Min	Max	Units	Conditions
t_{sink}	Current sink response time		0.8	µs	Full scale transition
	General Purpose I/O (continued)				
T_{ratio}	Tracking Ratio Port1 to Port0	13.6	18.4		Vout = 2.0V[9]
I_{max}	Port 1 max sink current		60	mA	Summed over all Port 1 bits
P_{max}	Port 1 & CEXT sink mode dissipation		25	mW	Per pin
V_{ith}	Input Threshold Voltage	45%	65%	Vcc	All ports and Cext [7]
V_H	Input Hysteresis Voltage	6%	12%	Vcc	All ports and Cext [8]
V_{OL1}	Output LOW Voltage, Cext pin		0.4	V	V_{CC} = Min., I_{OL} = 2mA
V_{OL2}	Output LOW Voltage, Cext pin		2.0	V	V_{CC} = Min., I_{OL} = 5mA

8.0 Switching Characteristics

Parameter	Description	Min.	Max.	Unit
t_{CYC}	Input clock cycle time	166.67	166.67	ns
t_{CH}	Clock HIGH time	0.45 t_{CYC}		ns
t_{CL}	Clock LOW time	0.45 t_{CYC}		ns
t_r	Transition Rise Time [1, 4, 8]	75	300	ns
t_f	Transition Fall Time [1, 4, 8]	75	300	ns

Notes:

4. 4.35 V to 5.25 V V_{CC}.
5. Measured as largest step size vs nominal according to measured full scale and zero programmed values.
6. POR can occur only once per applied V_{CC}, if V_{CC} drops below Vrst, POR will **not** re-occur. V_{CC} must return to 0.0V before POR will be re-applied on a subsequent V_{CC} ramp.
7. Low to High transition
8. This parameter is guaranteed, but not tested.
9. $T_{ratio} = I_{sink1(n)}/I_{sink0(n)}$ for the same n
10. $I_{range} = I_{sink(F)}/I_{sink(O)}$ for port 0 or 1 output

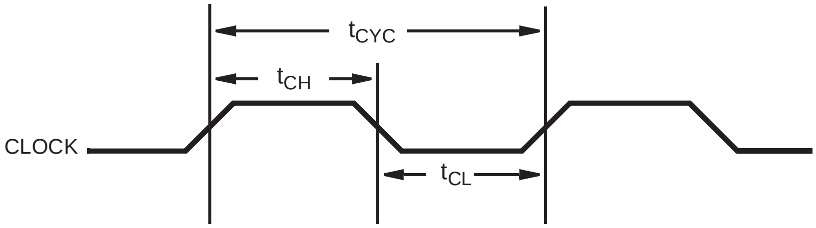


Figure 8-1. Clock Timing

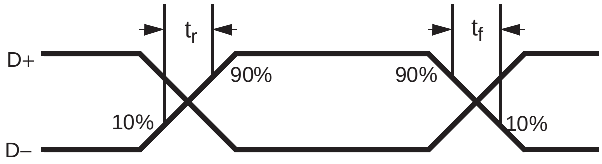


Figure 8-2. USB Data Signal Timing

9.0 Ordering Information

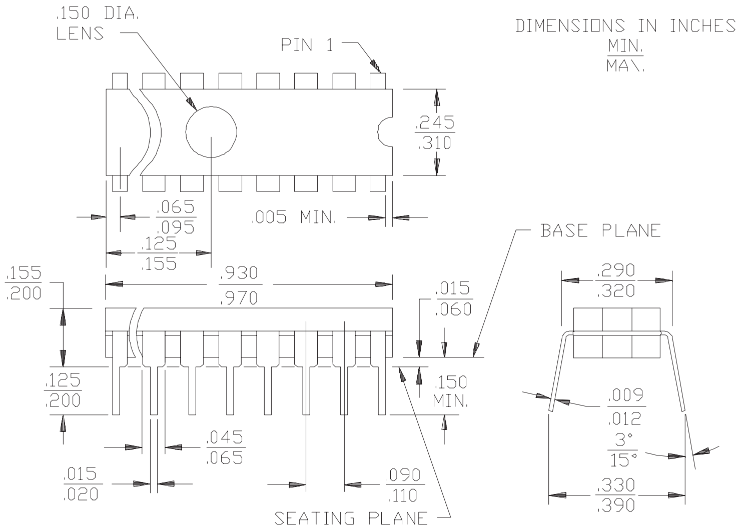
Ordering Code	EPROM Size	Number of GPIO	Package Type	Operating Range
CY7C63000-PC	2KB	12	20-Pin (300-Mil) PDIP	Commercial
CY7C63000-SC	2KB	12	20-Pin (300-Mil) SOIC	Commercial
CY7C63001-PC	4KB	12	20-Pin (300-Mil) PDIP	Commercial
CY7C63001-SC	4KB	12	20-Pin (300-Mil) SOIC	Commercial
CY7C63001-WC	4KB	12	20-Pin (300-Mil) Windowed CerDIP	Commercial
CY7C63100-SC	2KB	16	24-Pin (300-Mil) SOIC	Commercial
CY7C63101-SC	4KB	16	24-Pin (300-Mil) SOIC	Commercial
CY7C63101-WC	4KB	16	24-Pin (300-Mil) Windowed CerDIP	Commercial
CY7C63200-PC	2KB	10	18-Pin (300-Mil) PDIP	Commercial
CY7C63201-PC	4KB	10	18-Pin (300-Mil) PDIP	Commercial
CY7C63201-WC	4KB	10	18-Pin (300-Mil) Windowed CerDIP	Commercial

Document #: 38-00557-D

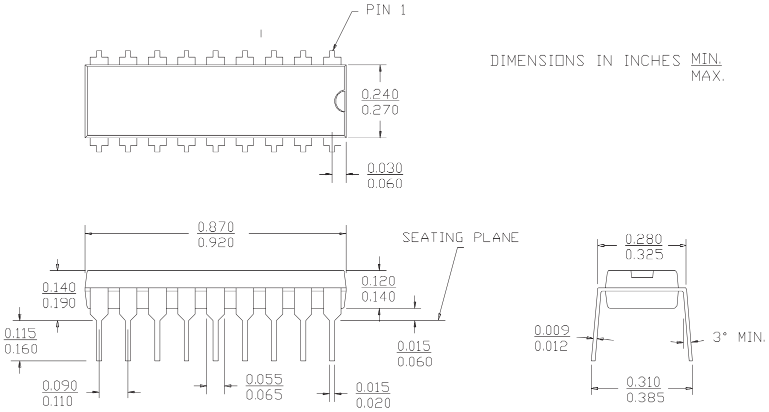
Package Diagrams

20-Lead (300-Mil) Windowed CerDIP W6

MIL-STD-1835 D-8 Config. A

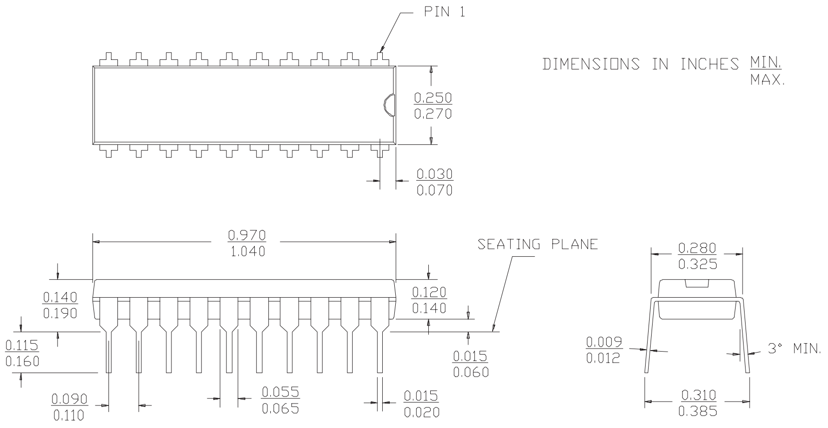


18-Lead (300-Mil) Molded DIP

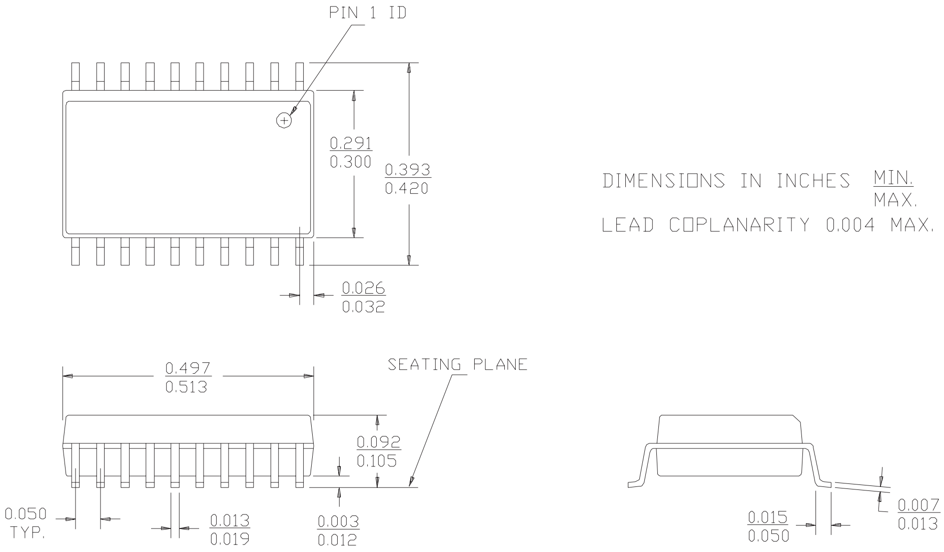


Package Diagrams (continued)

20-Lead (300-Mil) Molded DIP

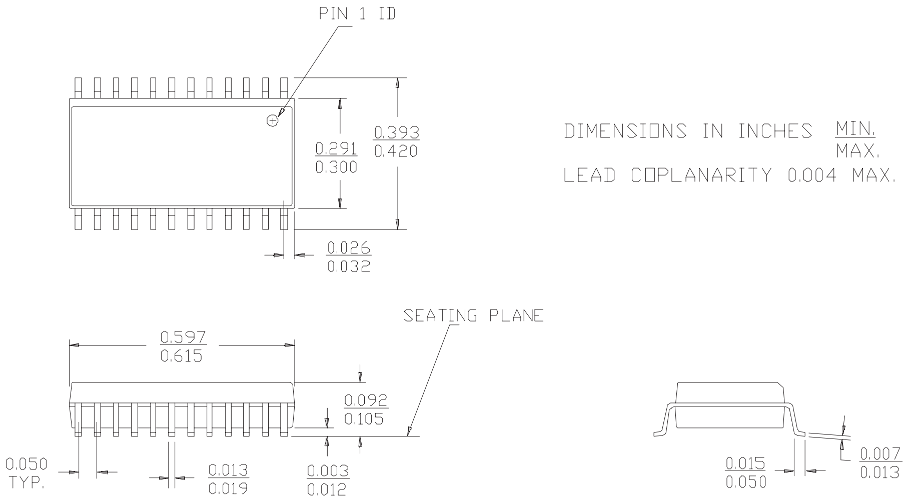


20-Lead (300-Mil) Molded SOIC



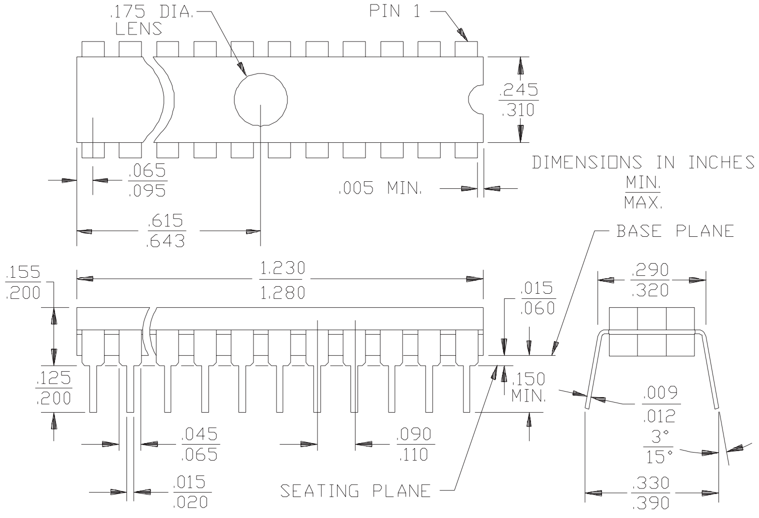
Package Diagrams (continued)

24-Lead (300-Mil) Molded SOIC



24-Lead (300-Mil) Windowed CerDIP W14

MIL-STD-1835 D-9 Config. A



© Cypress Semiconductor Corporation, 1997. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress Semiconductor product. Nor does it convey or imply any license under patent or other rights. Cypress Semiconductor does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress Semiconductor products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress Semiconductor against all charges.