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?

Details	
Product Status	Obsolete
Applications	USB Microcontroller
Core Processor	M8B
Program Memory Type	OTP (8kB)
Controller Series	CY7C635xx
RAM Size	256 x 8
Interface	PS/2, USB
Number of I/O	40
Voltage - Supply	4V ~ 5.5V
Operating Temperature	0°C ~ 70°C
Mounting Type	Surface Mount
Package / Case	48-BSSOP (0.295", 7.50mm Width)
Supplier Device Package	48-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy7c63513-pvc

TABLE OF CONTENTS

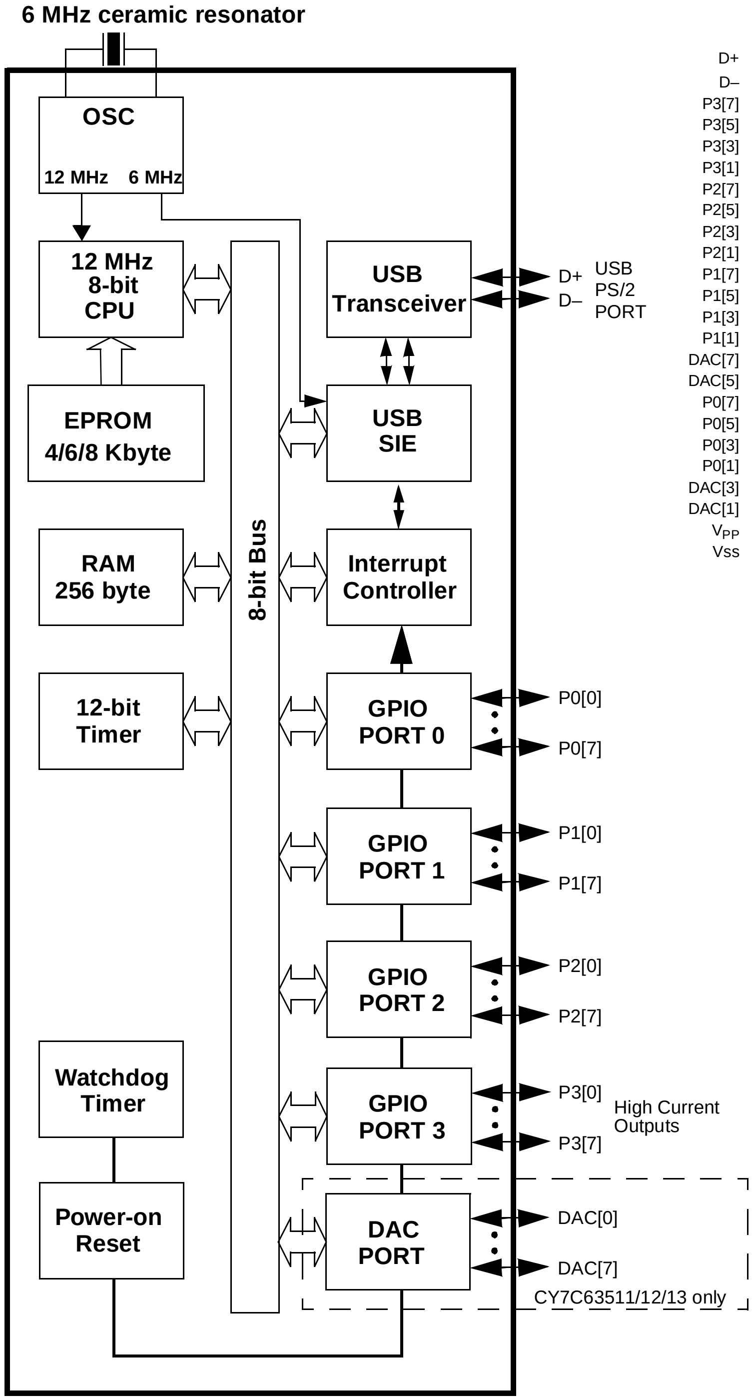
LIST OF FIGURES

LIST OF TABLES

1.0 Features

- Low-cost solution for low-speed applications with high I/O requirements such as keyboards, keyboards with integrated pointing device, gamepads, and many others.
- USB Specification Compliance
  - —Conforms to USB Specification, Version 1.0
  - —Conforms to USB HID Specification, Version 1.0
  - —Supports 1 device address and 3 data endpoints
  - —Integrated USB transceiver
- 8-bit RISC microcontroller
  - —Harvard architecture
  - —6 MHz external ceramic resonator
  - —12 MHz internal CPU clock
- Internal memory
  - —256 bytes of RAM
  - —4 Kbytes of EPROM (CY7C63411, CY7C63511)
  - —6 Kbytes of EPROM (CY7C63412, CY7C63512)
  - —8 Kbytes of EPROM (CY7C63413, CY7C63513)
- Interface can auto-configure to operate as PS2 or USB
- I/O port
  - —24 General Purpose I/O (GPIO) pins (Port 0 to 2) capable of sinking 7 mA per pin (typical)
  - —Eight GPIO pins (Port 3) capable of sinking 12 mA per pin (typical) which can drive LEDs
  - —Higher current drive is available by connecting multiple GPIO pins together to drive an common output
  - —Each GPIO port can be configured as inputs with internal pull-ups or open drain outputs or traditional CMOS outputs
  - —The CY7C63511/12/1 has an additional eight I/O pins on a DAC port which has programmable current sink outputs
  - —Maskable interrupts on all I/O pins
- 12-bit free-running timer with one microsecond clock ticks
- Watchdog timer (WDT)
- Internal power-on reset (POR)
- Improved output drivers to reduce EMI
- Operating voltage from 4.0V to 5.5VDC
- Operating temperature from 0 to 70 degrees Celsius
- CY7C63411/12/13 available in 40-pin PDIP, 48-pin SSOP for production
- CY7C63411/12/13 available in 40-pin Windowed CerDIP, 48-pin Windowed SideBraze for program development
- CY7C63511/12/13 available in 48-pin SSOP packages for production
- CY7C63511/12/13 available in 48-pin Windowed SideBraze for program development
- Industry standard programmer support

Logic Block Diagram



Pin Configurations

CY7C63511/12/13 48-pin SSOP 48-pin SideBraze

D+	1	48	V_{CC}
D–	2	47	Vss
P3[7]	3	46	P3[6]
P3[5]	4	45	P3[4]
P3[3]	5	44	P3[2]
P3[1]	6	43	P3[0]
P2[7]	7	42	P2[6]
P2[5]	8	41	P2[4]
P2[3]	9	40	P2[2]
P2[1]	10	39	P2[0]
P1[7]	11	38	P1[6]
P1[5]	12	37	P1[4]
P1[3]	13	36	P1[2]
P1[1]	14	35	P1[0]
DAC[7]	15	34	DAC[6]
DAC[5]	16	33	DAC[4]
P0[7]	17	32	P0[6]
P0[5]	18	31	P0[4]
P0[3]	19	30	P0[2]
P0[1]	20	29	P0[0]
DAC[3]	21	28	DAC[2]
DAC[1]	22	27	DAC[0]
V_{PP}	23	26	$XTAL_{OUT}$
Vss	24	25	$XTAL_{IN}$

CY7C63411/12/13 48-pin SSOP 48-pin SideBraze

D+	1	48	V_{CC}
D–	2	47	Vss
P3[7]	3	46	P3[6]
P3[5]	4	45	P3[4]
P3[3]	5	44	P3[2]
P3[1]	6	43	P3[0]
P2[7]	7	42	P2[6]
P2[5]	8	41	P2[4]
P2[3]	9	40	P2[2]
P2[1]	10	39	P2[0]
P1[7]	11	38	P1[6]
P1[5]	12	37	P1[4]
P1[3]	13	36	P1[2]
P1[1]	14	35	P1[0]
NC	15	34	NC
NC	16	33	NC
P0[7]	17	32	P0[6]
P0[5]	18	31	P0[4]
P0[3]	19	30	P0[2]
P0[1]	20	29	P0[0]
NC	21	28	NC
NC	22	27	NC
V_{PP}	23	26	$XTAL_{OUT}$
Vss	24	25	$XTAL_{IN}$

TOP VIEW

CY7C63411/12/13 40-pin PDIP 40-pin CerDIP

D+	1	40	V_{CC}
D–	2	39	V_{SS}
P3[7]	3	38	P3[6]
P3[5]	4	37	P3[4]
P3[3]	5	36	P3[2]
P3[1]	6	35	P3[0]
P2[7]	7	34	P2[6]
P2[5]	8	33	P2[4]
P2[3]	9	32	P2[2]
P2[1]	10	31	P2[0]
P1[7]	11	30	P1[6]
P1[5]	12	29	P1[4]
P1[3]	13	28	P1[2]
P1[1]	14	27	P1[0]
P0[7]	15	26	P0[6]
P0[5]	16	25	P0[4]
P0[3]	17	24	P0[2]
P0[1]	18	23	P0[0]
V_{PP}	19	22	$XTAL_{OUT}$
Vss	20	21	$XTAL_{IN}$

TOP VIEW

3.0 Pin Assignments

Name	I/O	CY7C63411/12/13		CY7C63511/12/13	Description
		40-Pin	48-Pin	48-Pin	
D+, D–	I/O	1,2	1,2	1,2	USB differential data; PS/2 clock and data signals
P0[7:0]	I/O	15,26,16,25, 17,24,18,23	17,32,18,31, 19,30,20,29	17,32,18,31, 19,30,20,29	GPIO port 0 capable of sinking 7 mA (typical)
P1[7:0]	I/O	11,30,12,29, 13,28,14,27	11,38,12,37, 13,36,14,35	11,38,12,37, 13,36,14,35	GPIO Port 1 capable of sinking 7 mA (typical)
P2[7:0]	I/O	7,34,8,33, 9,32,10,31	7,42,8,41, 9,40,10,39	7,42,8,41, 9,40,10,39	GPIO Port 2 capable of sinking 7 mA (typical)
P3[7:0]	I/O	3,38,4,37, 5,36,6,35	3,46,4,45, 5,44,6,43	3,46,4,45, 5,44,6,43	GPIO Port 3 capable of sinking 12 mA (typical)
DAC[7:0]	I/O	n/a	n/a	15,34,16,33, 21,28,22,27	DAC I/O Port with programmable current sink outputs. DAC[1:0] offer a programmable range of 3.2 to 16 mA typical. DAC[7:2] have a programmable sink current range of 0.2 to 1.0 mA typical.
$XTAL_{IN}$	IN	21	25	25	6 MHz ceramic resonator or external clock input
$XTAL_{OUT}$	OUT	22	26	26	6 MHz ceramic resonator
V_{PP}		19	23	23	Programming voltage supply, ground for normal operation
V_{CC}		40	48	48	Voltage supply
Vss		20,39	24,47	24,47	Ground

4.0 Programming Model

4.1 14-bit Program Counter (PC)

The 14-bit program counter (PC) allows access for up to 8 kilobytes of EPROM using the CY7C634/5xx architecture. The program counter is cleared during reset, such that the first instruction executed after a reset is at address 0x0000h. This is typically a jump instruction to a reset handler that initializes the application.

The lower 8 bits of the program counter are incremented as instructions are loaded and executed. The upper 6 bits of the program counter are incremented by executing an XPAGE instruction. As a result, the last instruction executed within a 256 byte “page” of sequential code should be an XPAGE instruction. The assembler directive “XPAGEON” will cause the assembler to insert XPAGE instructions automatically. As instructions can be either one or two bytes long, the assembler may occasionally need to insert a NOP followed by an XPAGE for correct execution.

The program counter of the next instruction to be executed, carry flag, and zero flag are saved as two bytes on the program stack during an interrupt acknowledge or a CALL instruction. The program counter, carry flag, and zero flag are restored from the program stack only during a RETI instruction.

Please note the program counter cannot be accessed directly by the firmware. The program stack can be examined by reading SRAM from location 0x00 and up.

4.2 8-bit Accumulator (A)

The accumulator is the general purpose, do everything register in the architecture where results are usually calculated.

4.3 8-bit Index Register (X)

The index register “X” is available to the firmware as an auxiliary accumulator. The X register also allows the processor to perform indexed operations by loading an index value into X.

4.4 8-bit Program Stack Pointer (PSP)

During a reset, the program stack pointer (PSP) is set to zero. This means the program “stack” starts at RAM address 0x00 and “grows” upward from there. Note the program stack pointer is directly addressable under firmware control, using the MOV PSP,A instruction. The PSP supports interrupt service under hardware control and CALL, RET, and RETI instructions under firmware control.

6.0 Memory Organization

6.1 Program Memory Organization

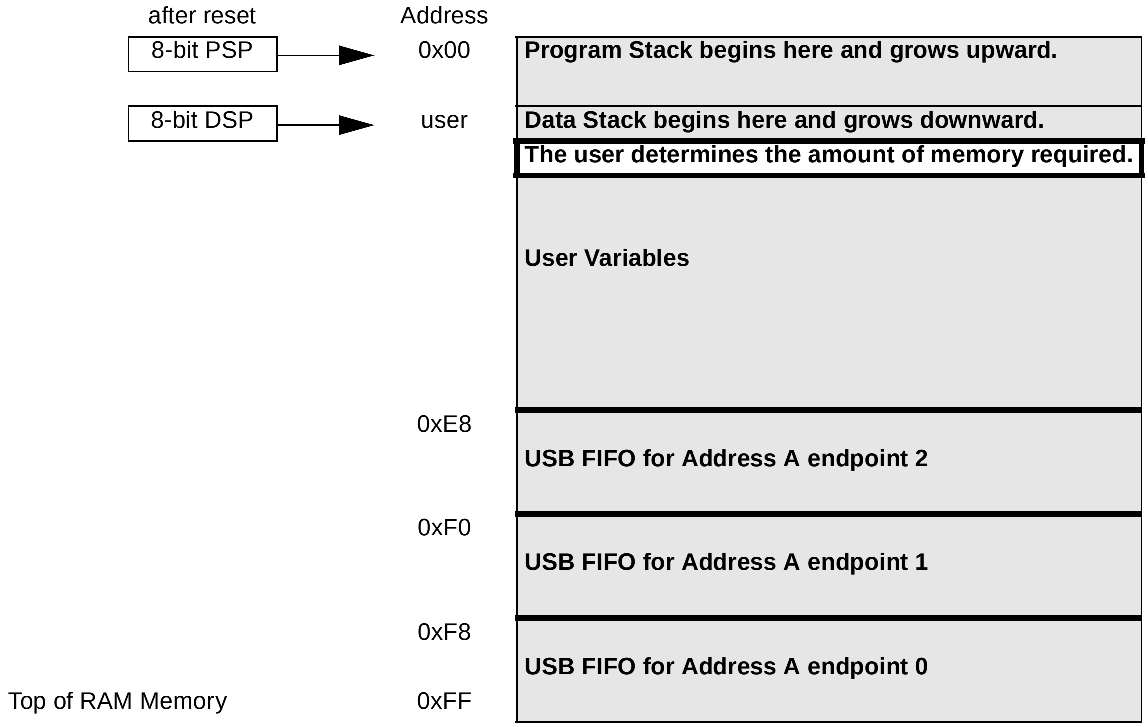
after reset: 14-bit PC → 0x0000

Address	
0x0000	Program execution begins here after a reset.
0x0002	USB Bus Reset interrupt vector
0x0004	128 µs timer interrupt vector
0x0006	1.024 ms timer interrupt vector
0x0008	USB address A endpoint 0 interrupt vector
0x000A	USB address A endpoint 1 interrupt vector
0x000C	USB address A endpoint 2 interrupt vector
0x000E	Reserved
0x0010	Reserved
0x0012	Reserved
0x0014	DAC interrupt vector
0x0016	GPIO interrupt vector
0x0018	Reserved
0x001A	**Program Memory begins here**
0x0FFF	**4 KB PROM ends here (CY7C63411,CY7C63511)**
0x17FF	**6 KB PROM ends here (CY7C63412, CY7C63512)**
0x1FDF	(8K - 32 bytes) **8 KB PROM ends here (CY7C63413, CY7C63513)**

Figure 6-1. Program Memory Space with Interrupt Vector Table

6.2 Data Memory Organization

The CY7C63411/12/13 and CY7C63511/12/13 microcontrollers provide 256 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:



6.3 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 (e.g., IOWX 0h) means the I/O port is selected solely by the contents of X.

Table 6-1. I/O Register Summary

Register Name	I/O Address	Read/Write	Function
Port 0 Data	0x00	R/W	GPIO Port 0
Port 1 Data	0x01	R/W	GPIO Port 1
Port 2 Data	0x02	R/W	GPIO Port 2
Port 3 Data	0x03	R/W	GPIO Port 3
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 2 Interrupt Enable	0x06	W	Interrupt enable for pins in Port 2
Port 3 Interrupt Enable	0x07	W	Interrupt enable for pins in Port 3
GPIO Configuration	0x08	R/W	GPIO Ports Configurations
USB Device Address A	0x10	R/W	USB Device Address A
EP A0 Counter Register	0x11	R/W	USB Address A, Endpoint 0 counter register
EP A0 Mode Register	0x12	R/W	USB Address A, Endpoint 0 configuration register
EP A1 Counter Register	0x13	R/W	USB Address A, Endpoint 1 counter register
EP A1 Mode Register	0x14	R/C	USB Address A, Endpoint 1 configuration register
EP A2 Counter Register	0x15	R/W	USB address A, Endpoint 2 counter register
EP A2 Mode Register	0x16	R/C	USB address A, Endpoint 2 configuration register
USB Status & Control	0x1F	R/W	USB up-stream port traffic status and control register
Global Interrupt Enable	0x20	R/W	Global interrupt enable register
Endpoint Interrupt Enable	0x21	R/W	USB endpoint interrupt enables
Timer (LSB)	0x24	R	Lower 8 bits of free-running timer (1 MHz)
Timer (MSB)	0x25	R	Upper 4 bits of free-running timer that are latched when the lower 8 bits are read.
WDR Clear	0x26	W	Watch Dog Reset clear
DAC Data	0x30	R/W	DAC I/O
DAC Interrupt Enable	0x31	W	Interrupt enable for each DAC pin.
DAC Interrupt Polarity	0x32	W	Interrupt polarity for each DAC pin
DAC Isink	0x38-0x3F	W	One four bit sink current register for each DAC pin.
Processor Status & Control	0xFF	R/W	Microprocessor status and control

7.0 Clocking

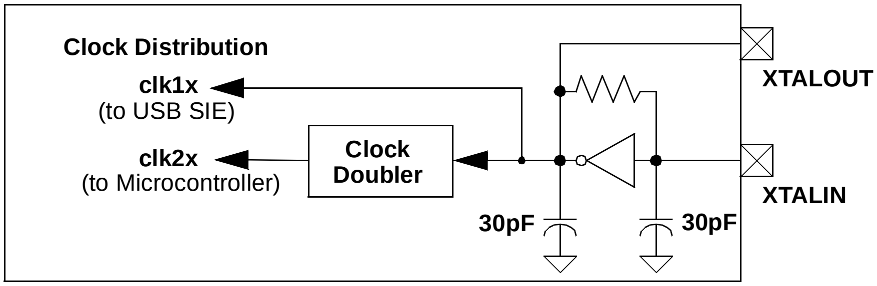


Figure 7-1. Clock Oscillator On-chip Circuit

The $XTAL_{IN}$ and $XTAL_{OUT}$ are the clock pins to the microcontroller. The user can connect a low-cost ceramic resonator or an external oscillator can be connected to these pins to provide a reference frequency for the internal clock distribution and clock doubler.

An external 6 MHz clock can be applied to the $XTAL_{IN}$ pin if the $XTAL_{OUT}$ pin is left open. Please note that grounding the $XTAL_{OUT}$ pin is not permissible as the internal clock is effectively shorted to ground.

8.0 Reset

The USB Controller supports three types of resets. All registers are restored to their default states during a reset. The USB Device Addresses are set to 0 and all interrupts are disabled. In addition, the Program Stack Pointer (PSP) and Data Stack Pointer (DSP) are set to 0x00. For USB applications, the firmware should set the DSP below 0xE8h to avoid a memory conflict with RAM dedicated to USB FIFOs. The assembly instructions to do this are shown below:

```
Mov A, E8h      ; Move 0xE8 hex into Accumulator
Swap A,dsp      ; swap accumulator value into dsp register
```

The three reset types are:

1. Power-On Reset (POR)
2. Watch Dog Reset (WDR)
3. USB Bus Reset (non hardware reset)

The occurrence of a reset is recorded in the Processor Status and Control Register located at I/O address 0xFF. Bits 4, 5, and 6 are used to record the occurrence of POR, USB Reset, and WDR respectively. The firmware can interrogate these bits to determine the cause of a reset.

The microcontroller begins execution from ROM address 0x0000h after a POR 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. That means the reset handler in firmware should initialize the hardware and begin executing the “main” loop of code. Attempting to execute either a RET or RETI in the reset handler will cause unpredictable execution results.

8.1 Power-On Reset (POR)

Power-On Reset (POR) occurs every time the V_{CC} voltage to the device ramps from 0V to an internally defined trip voltage (Vrst), of approximately 1/2 full supply voltage. In addition to the normal reset initialization noted under “Reset,” bit 4 (PORS) of the Processor Status and Control Register is set to “1” to indicate to the firmware that a power on reset occurred. The POR event forces the GPIO ports into input mode (high impedance), and the state of Port 3 bit 7 is used to control how the part will respond after the POR releases.

If Port 3 bit 7 is high (pulled to V_{CC}) and the USB IO are at the idle state (DM high and DP low) the part will go into a semi-permanent power down/suspend mode, waiting for the USB IO to go to one of Bus Reset, K (resume) or SE0. If Port 3 bit 7 is still high when the part comes out of suspend, then a 128 us timer starts, delaying CPU operation until the ceramic resonator has stabilized.

If Port 3 bit 7 was low (pulled to V_{SS}) the part will start a 128 ms timer, delaying CPU operation until V_{CC} has stabilized, then continuing to run as reset.

Firmware should clear the POR Status (PORS) bit in register FFh before going into suspend as this status bit selects the 128 μs or 128 ms start-up timer value as follows: IF Port 3 bit 7 is high then 128 μs is always used; ELSE if PORS is high then 128 ms is used; ELSE 128 μs is used.

- Fill and empty the FIFOs
- Suspend/Resume coordination
- Verify and select Data toggle values

11.1 USB Enumeration

The enumeration sequence is shown below:

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.

11.2 PS/2 Operation

PS/2 operation is possible with the CY7C634/5xx series through the use of firmware and several operating modes. The first enabling feature:

1. USB Bus reset on D+ and D– is an interrupt that can be disabled;
2. USB traffic can be disabled via bit7 of the USB register;
3. D+ and D– can be monitored and driven via firmware as independent port bits.

Bits 5 and 4 of the Upstream Status and Control register are directly connected to the D+ and D– USB pins of the CY7C634/5xx. These pins constantly monitor the levels of these signals with CMOS input thresholds. Firmware can poll and decode these signals as PS/2 clock and data.

Bits [2:0] defaults to ‘000’ at reset which allows the USB SIE to control output on D+ and D–. Firmware can override the SIE and directly control the state of these pins via these 3 control bits. Since PS/2 is an open drain signalling protocol, these modes allow all 4 PS/2 states to be generated on the D+ and D– pins

11.3 USB Port Status and Control

USB status and control is regulated by the USB Status and Control Register located at I/O address 0x1Fh as shown in *Figure 11-1*. 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	R	R/W	R/W	R/W	R/W
Reserved	Reserved	D+	D–	Bus Activity	Control Bit 2	Control Bit 1	Control Bit 0

Figure 11-1. USB Status and Control Register 0x1Fh

The Bus Activity bit is a “sticky” bit that indicates if any non-idle USB event has occurred on the USB bus. The user firmware should check and clear this bit periodically to detect any loss of bus activity. Writing a “0” to the Bus Activity bit clears it while writing a “1” preserves the current value. In other words, the 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. The following table shows how the control bits are encoded for this register.

Control Bits	Control action
000	Not forcing (SIE controls driver)
001	Force K (D+ high, D– low)
010	Force J (D+ low, D– high)
011	Force SE0 (D+ low, D– low)
100	Force SE0 (D– low, D+ low)
101	Force D– low, D+ HiZ
110	Force D– HiZ, D+ low
111	Force D– HiZ, D+ HiZ

12.0 USB Device

USB Device Address A includes three endpoints: EPA0, EPA1, and EPA2. End Point 0 (EPA0) allows the USB host to recognize, set-up, and control the device. In particular, EPA0 is used to receive and transmit control (including set-up) packets.

12.1 USB Ports

The USB Controller provides one USB device address with three endpoints. The USB Device Address Register contents are cleared during a reset, setting the USB device address to zero and marking this address as disabled. *Figure 12-1* shows the format of the USB Address Register.

Device Address Enable	Device Address Bit 6	Device Address Bit 5	Device Address Bit 4	Device Address Bit 3	Device Address Bit 2	Device Address Bit 1	Device Address Bit 0

Figure 12-1. USB Device Address Register 0x10h (read/write)

Bit 7 (Device Address Enable) in the USB Device Address Register must be set by firmware before the serial interface engine (SIE) will respond to USB traffic to this address. The Device Address in bits [6:0] must be set by firmware during the USB enumeration process to an address assigned by the USB host that does not equal zero. This register is cleared by a hardware reset or the USB bus reset.

12.2 Device Endpoints (3)

The USB controller communicates with the host using dedicated FIFOs, one per endpoint. Each endpoint FIFO is implemented as 8 bytes of dedicated SRAM. There are three endpoints defined for Device “A” that are labeled “EPA0,” “EPA1,” and EPA2.”

All USB devices are required to have an endpoint number 0 (EPA0) that is used to initialize and control the USB device. End Point 0 provides access to the device configuration information and allows generic USB status and control accesses. End Point 0 is bidirectional as the USB controller can both receive and transmit data.

The endpoint mode registers are cleared during reset. The EPA0 endpoint mode register uses the format shown below:

Endpoint 0 set-up Received	Endpoint 0 In Received	Endpoint 0 Out Received	Acknowledge	Mode Bit 3	Mode Bit 2	Mode Bit 1	Mode Bit 0

Figure 12-2. USB Device EPA0 Mode Register 0x12h (read/write)

Bits[7:5] in the endpoint 0 mode registers (EPA0) are “sticky” status bits that are set by the SIE to report the type of token that was most recently received. The sticky bits must be cleared by firmware as part of the USB processing.

The endpoint mode registers for EPA1 and EPA2 do not use bits [7:5] as shown below:

Reserved	Reserved	Reserved	Acknowledge	Mode Bit 3	Mode Bit 2	Mode Bit 1	Mode Bit 0

Figure 12-3. USB Device Endpoint Mode Registers 0x14h, 0x16h (read/write)

The ‘Acknowledge’ bit is set whenever the SIE engages in a transaction that completes with an ‘ACK’ packet.

The ‘set-up’ PID status (bit[7]) is forced high from the start of the data packet phase of the set-up transaction, until the start of the ACK packet returned by the SIE. The CPU is prevented from clearing this bit during this interval, and subsequently until the CPU first does a IORD to this endpoint 0 mode register.

Bits[6:0] of the endpoint 0 mode register are locked from CPU IOWR operations only if the SIE has updated one of these bits, which the SIE does only at the end of a packet transaction (set-up ... Data ... ACK, or Out ... Data ... ACK, or In ... Data ... ACK). The CPU can unlock these bits by doing a subsequent I/O read of this register.

Firmware must do an IORD after an IOWR to an endpoint 0 register to verify that the contents have changed and that the SIE has not updated these values.

While the ‘set-up’ bit is set, the CPU cannot write to the DMA buffers at memory locations 0xE0 through 0xE7 and 0xF8 through 0xFF. This prevents an incoming set-up transaction from conflicting with a previous In data buffer filling operation by firmware.

The mode bits (bits [3:0]) in an Endpoint Mode Register control how the endpoint responds to USB bus traffic. The mode bit encoding is shown in Section 16.

The format of the endpoint Device counter registers is shown below:

Data 0/1 Toggle	Data Valid	Reserved	Reserved	Byte count Bit 3	Byte count Bit 2	Byte count Bit 1	Byte count Bit 0

Figure 12-4. USB Device Counter Registers 0x11h, 0x13h, 0x15h (read/write)

Bits 0 to 3 indicate the number of data bytes to be transmitted during an IN packet, valid values are 0 to 8 inclusive. Data Valid bit 6 is used for OUT and set-up tokens only. Data 0/1 Toggle bit 7 selects the DATA packet’s toggle state: 0 for DATA0, 1 for DATA1.

13.0 12-bit Free-running Timer

The 12-bit timer provides two interrupts (128 μs and 1.024 ms) and allows the firmware to directly time events that are up to 4 ms in duration. The lower 8 bits of the timer can be read directly by the firmware. Reading the lower 8 bits latches the upper 4 bits into a temporary register. When the firmware reads the upper 4 bits of the timer, it is actually reading the count stored in the temporary register. The effect of this logic is to ensure a stable 12-bit timer value can be read, even when the two reads are separated in time.

13.1 Timer (LSB)

Timer Bit 7	Timer Bit 6	Timer Bit 5	Timer Bit 4	Timer Bit 3	Timer Bit 2	Timer Bit 1	Timer Bit 0

Figure 13-1. Timer Register 0x24h (read only)

13.2 Timer (MSB)

Reserved	Reserved	Reserved	Reserved	Timer Bit 11	Timer Bit 10	Timer Bit 9	Timer Bit 8

Figure 13-2. Timer Register 0x25h (read only)

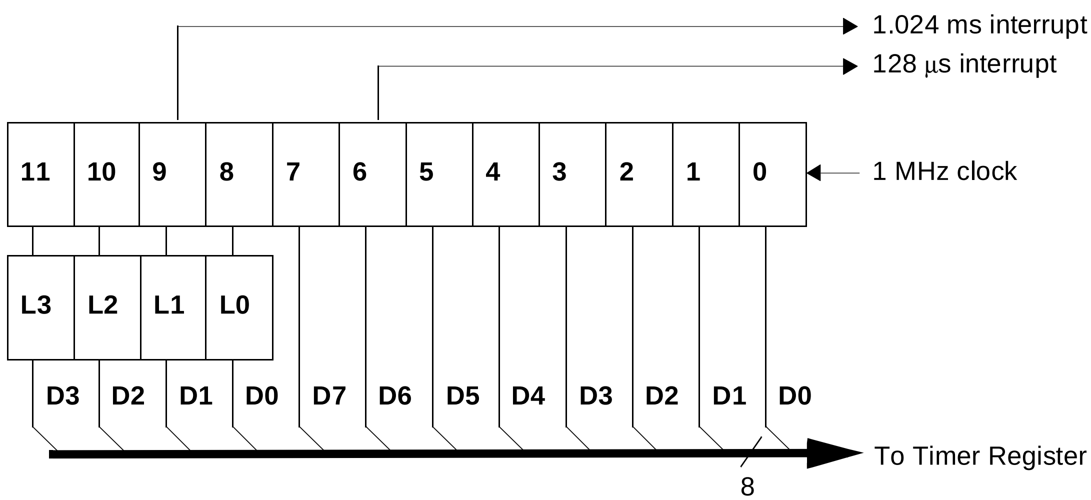


Figure 13-3. Timer Block Diagram

15.2.2 Timer Interrupt

There are two timer interrupts: the 128 µs interrupt and the 1.024 ms interrupt. 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.

15.2.3 USB Endpoint Interrupts

There are three USB endpoint interrupts, one per endpoint. The USB endpoints interrupt after the either the USB host or the USB controller sends a packet to the USB.

15.2.4 DAC Interrupt

Each DAC I/O pin can generate an interrupt, if enabled.The interrupt polarity for each DAC I/O pin is programmable. A positive polarity is a rising edge input while a negative polarity is a falling edge input. All of the DAC pins share a single interrupt vector, which means the firmware will need to read the DAC port to determine which pin or pins caused an interrupt.

Please note that if one DAC pin triggered an interrupt, no other DAC pins can cause a DAC interrupt until that pin has returned to its inactive (non-trigger) state or the corresponding interrupt enable bit is cleared. The USB Controller does not assign interrupt priority to different DAC pins and the DAC Interrupt Enable Register is not cleared during the interrupt acknowledge process.

15.2.5 GPIO Interrupt

Each of the 32 GPIO pins can generate an interrupt, if enabled. The interrupt polarity can be programmed for each GPIO port as part of the GPIO configuration. All of the GPIO pins share a single interrupt vector, which means the firmware will need to read the GPIO ports with enabled interrupts to determine which pin or pins caused an interrupt.

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.

16.0 Truth Tables

Table 16-1. USB Register Mode Encoding

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	“Forced from Setup on Control endpoint, from modes other than 0000”
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
Isochronous Out	**0101**	ignore	ignore	always	“(available to low speed devices, future USB spec enhancements)” In Only
Status In Only	**0110**	accept	TX 0	stall	For Control Endpoints
Isochronous In	**0111**	ignore	TX cnt	ignore	“(available to low speed devices, future USB spec enhancements)”
Nak Out	**1000**	ignore	ignore	NAK	An ACK from mode 1001 --> 1000
Ack Out	**1001**	ignore	ignore	ACK	This mode is changed by SIE on issuance of ACK --> 1000
Nak Out - Status In	**1010**	accept	TX 0	NAK	An ACK from mode 1011 --> 1010
Ack Out - Status In	**1011**	accept	TX 0	ACK	This mode is changed by SIE on issuance of ACK --> 1010
Nak In	**1100**	ignore	NAK	ignore	An ACK from mode 1101 --> 1100
Ack In	**1101**	ignore	TX cnt	ignore	This mode is changed by SIE on issuance of ACK --> 1100
Nak In - Status Out	**1110**	accept	NAK	check	An ACK from mode 1111 --> 111 Ack In - Status Out
Ack In - Status Out	**1111**	accept	TX cnt	Check	This mode is changed by SIE on issuance of ACK -->1110

The ‘In’ column represents the SIE’s response to the token type.

A disabled endpoint will remain such until firmware changes it, and all endpoints reset to disabled.

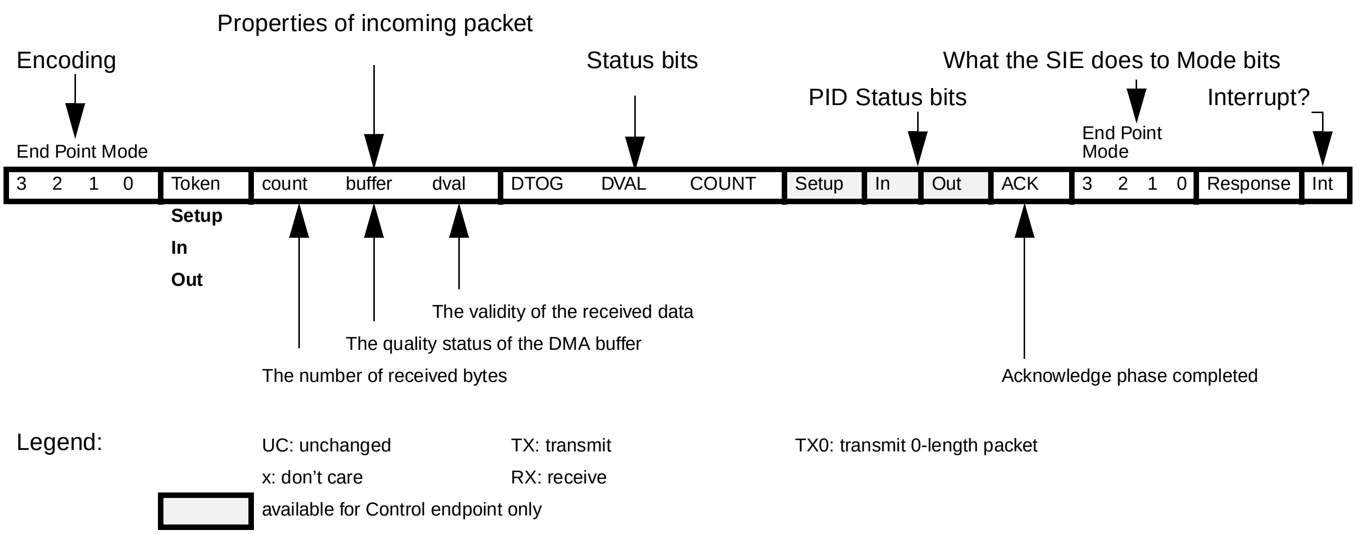
Any Setup packet to an enabled and accepting endpoint will be changed by the SIE to 0001 (NAKing).Any mode which indicates the acceptance of a Setup will acknowledge it.

Most modes that control transactions involving an ending ACK will be changed by the SIE to a corresponding mode which NAKs follow on packets.

A Control endpoint has three extra status bits for PID (Setup, In and Out), but must be placed in the correct mode to function as such. Also a non-Control endpoint can be made to act as a Control endpoint if it is placed in a non appropriate mode!

A ‘check’ on an Out token during a Status transaction checks to see that the Out is of zero length and has a Data Toggle (DTOG) of 1.

Table 16-2. Decode table for*Table 16-3*: “Details of Modes for Differing Traffic Conditions”



The response of the SIE can be summarized as follows:
(1) the SIE will only respond to valid transactions, and will ignore non-valid ones;
(2) the SIE will generate IRQ when a valid transaction is completed or when the DMA buffer is corrupted
(3) an incoming Data packet is valid if the count is <= 10 (CRC inclusive) and passes all error checking;
(4) a Setup will be ignored by all non Control endpoints (in appropriate modes);
(5) an In will be ignored by an Out configured endpoint and visa versa.

The In and Out PID status is updated at the end of a transaction.
The Setup PID status is updated at the beginning of the Data packet phase.

The entire EndPoint 0 mode and the Count register are locked to CPU writes at the end of any transaction in which an ACK is transferred. These registers are only unlocked upon a CPU read of these registers, and only if that read happens after the transaction completes. This represents about a 1 µs window to which to the CPU is locked from register writes to these USB registers. Normally the firmware does a register read at the beginning of the ISR to unlock and get the mode register informati on. The interlock on the Mode and Count registers ensures that the firmware recognizes the changes that the SIE might have made during the previous transaction!

18.0 DC Characteristics

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

	Parameter	Min	Max	Units	Conditions
	General				
$V_{CC\ (1)}$	Operating Voltage	4.0	5.5	V	Non USB activity (note 1)
$V_{CC\ (2)}$	Operating Voltage	4.35	5.25	V	USB activity (note 2)
I_{CC1}	V_{cc} Operating Supply Current		40	mA	V_{cc}=5.5V
I_{CC2}	Vcc = 4.35 V		15	mA	
I_{SB1}	Supply Current - Suspend Mode		20	µA	Oscillator off, D– > Voh min
$I_{SB2\ (12)}$	Supply Current - Start-up Mode		10	mA	Vcc = 5.0V (note 11)
V_{PP}	Programming Voltage (disabled)	–0.4	0.4	V	
T_{start}	Resonator Start-up Interval		256	µs	Vcc = 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	8.192	14.33	ms	
I_{il}	Input leakage current		1	µA	any pin
I_{sm}	Max Iss IO sink current		60	mA	Cumulative across all ports (note 10)
	Power On Reset				
t_{vccs}	Vcc reset slew	0.01	200	ms	linear ramp: 0 to 4.35V (notes 4,5)
	USB Interface				
V_{oh}	Static Output High	2.8	3.6	V	15k ±5% ohms to Gnd (notes 2,6)
V_{ol}	Static Output Low		0.3	V	
V_{di}	Differential Input Sensitivity	0.2		V	|(D+)–(D–)|
V_{cm}	Differential Input Common Mode Range	0.8	2.5	V	9-1
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	µs	0 V < V_{in}<3.3 V
R_{pu}	Bus Pull-up resistance	7.35K	7.65	kΩ	7.5 kΩ ± 2%
R_{pd}	Bus Pull-down resistance	14.25	15.75	kΩ	15 kΩ ± 5%
	General Purpose I/O Interface				
R_{up}	Pull-up resistance	4.9K	9.1K	Ohms	
V_{ith}	Input threshold voltage	45%	65%	V_{CC}	All ports, low to high edge
V_H	Input hysteresis voltage	6%	12%	V_{CC}	All ports, high to low edge
I_{ol}	Sink current	7.2	16.5	mA	Port 3, Vout = 1.0V (note 1)
I_{ol}	Sink current	3.5	10.6	mA	Port 0,1,2, Vout = 2.0V (note 1)
I_{oh}	Source current	1.4	7.5	mA	Voh = 2.4V (all ports 0,1,2,3) (note 1)
	DAC Interface				
R_{up}	Pull-up resistance	8.0K	20.0K	Ohms	
$I_{sink0(0)}$	DAC[7:2] sink current (0)	0.1	0.3	mA	Vout = 2.0 V DC (note 2)
$I_{sink0(F)}$	DAC[7:2] sink current (F)	0.5	1.5	mA	Vout = 2.0 V DC (note 2)
$I_{sink1(0)}$	DAC[1:0] sink current (0)	1.6	4.8	mA	Vout = 2.0 V DC (note 2)
$I_{sink1(F)}$	DAC[1:0] sink current (F)	8	24	mA	Vout = 2.0 V DC (note 2)
I_{range}	Programmed Isink ratio: max/min	4	6		Vout = 2.0 V DC (notes 2,12)
I_{lin}	Differential nonlinearity		0.5	lsb	any pin (note 7)
t_{sink}	Current sink response time		0.8	µs	Full scale transition
T_{ratio}	Tracking ratio DAC[1:0] to DAC[7:2]	14	20		Vout = 2.0V (note 9)

19.0 Switching Characteristics

Parameter	Description	Min.	Max.	Unit	Conditions
	Clock				
t_{CYC}	Input clock cycle time	165.0	168.3	ns	
t_{CH}	Clock HIGH time	0.45 t_{CYC}		ns	
t_{CL}	Clock LOW time	0.45 t_{CYC}		ns	
	USB Driver Characteristics				
t_r	Transition Rise Time (notes 2,3,8)	75		ns	CLoad = 50 pF
t_r	Transition Rise Time (notes 2,3,8)		300	ns	CLoad = 350 pF
t_f	Transition Fall Time (notes 2,3,8)	75		ns	CLoad = 50 pF
t_f	Transition Fall Time (notes 2,3,8)		300	ns	CLoad = 350 pF
t_{rfm}	Rise/Fall Time Matching	80	120	%	t_r/t_f (note 15)
V_{crs}	Output Signal Crossover Voltage	1.3	2.0	V	
	USB Data Timing				
t_{drate}	Low Speed Data Rate	1.4775	1.5225	Mbs	Ave. Bit Rate (1.5Mb/s ± 1.5%)
t_{djr1}	Receiver Data Jitter Tolerance	–75	75	ns	To Next Transition, (note 13)
t_{djr2}	Receiver Data Jitter Tolerance	–45	45	ns	For Paired Transitions, (note 13)
t_{deop}	Differential to EOP transition Skew	–40	100	ns	(note 13)
t_{eopr1}	EOP Width at receiver	165		ns	Rejects as EOP, (notes 13,14)
t_{eopr2}	EOP Width at receiver	675		ns	Accepts as EOP, (note 13)
t_{eopt}	Source EOP Width	1.25	1.50	µs	
t_{udj1}	Differential Driver Jitter	–95	95	ns	To next transition, *Figure 19-5*
t_{udj2}	Differential Driver Jitter	–150	150	ns	To paired transition, *Figure 19-5*

Notes:
1. Functionality is guaranteed of this Vcc range, except USB transmitter, and DACs.
2. USB transmitter functionality is guaranteed over this Vcc range, as well as DAC outputs.
3. Per Table 7-6 of revision 1.0 of USB specification, for C_{LOAD} of 50 – 350 pF.
4. Port 3 bit 7 controls whether the parts goes into suspend after a POR event or waits 128ms to begin running.
5. POR can occur only once per applied V_{CC}, if V_{CC} drops below Vrst, POR will **not** re-occur. VCC must return to 0.0V before POR will be re-applied on a subsequent V_{CC} ramp. Vrst is nominally 1/2 Vcc but is not specified.
6. Rx: external idle resistor, 7.5 KΩ, 2%, to V_{CC}.
7. Measured as largest step size vs. nominal according to measured full scale and zero programmed values.
8. This parameter is guaranteed, but not tested.
9. T_{ratio} = Isink1[1:0](n)/Isink0[7:2](n) for the same n, programmed.
10. Total current cumulative across all Port pins flowing to Vss is limited to minimize Ground-Drop noise effects.
11. Tested under static conditions.
12. Irange: Isinkn(15)/ Isinkn(0) for the same pin.
13. Measured at cross-over point of differential data signals
14. USB Specification indicates 330 ns.
15. Tested at 200 pF.

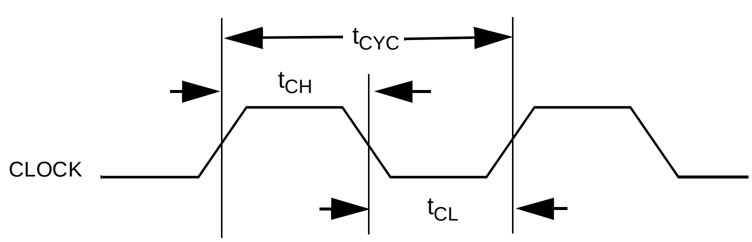


Figure 19-1. Clock Timing

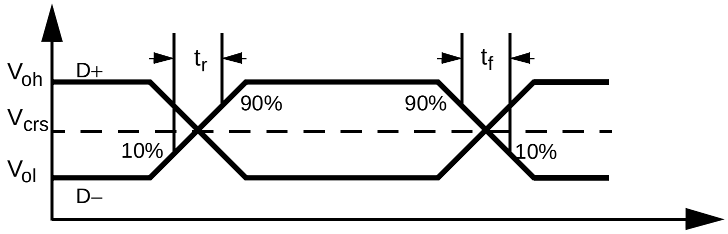


Figure 19-2. USB Data Signal Timing

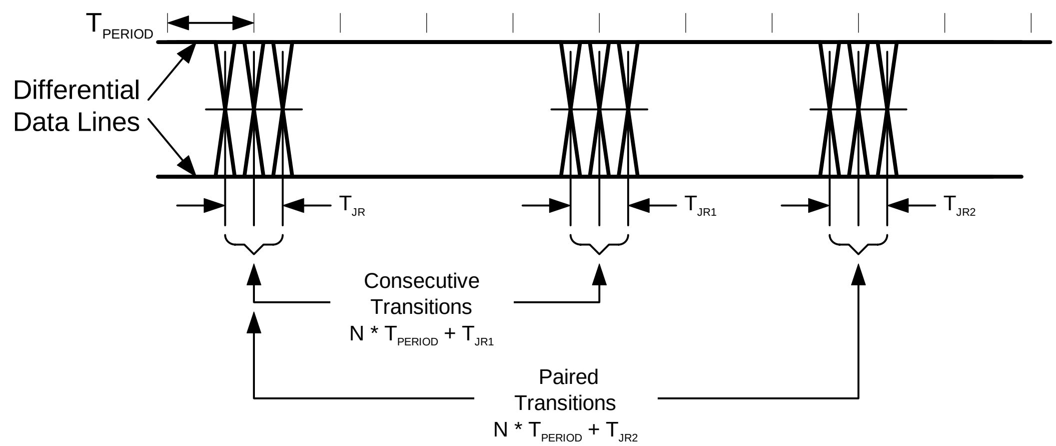


Figure 19-3. Receiver Jitter Tolerance

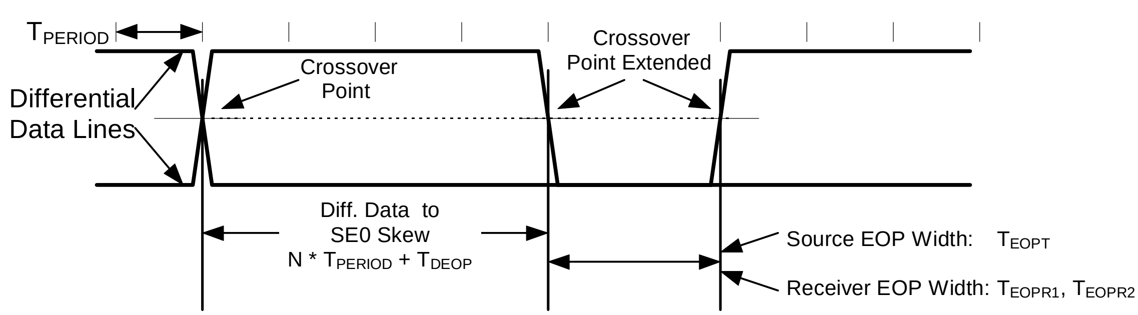


Figure 19-4. Differential to EOP Transition Skew and EOP Width

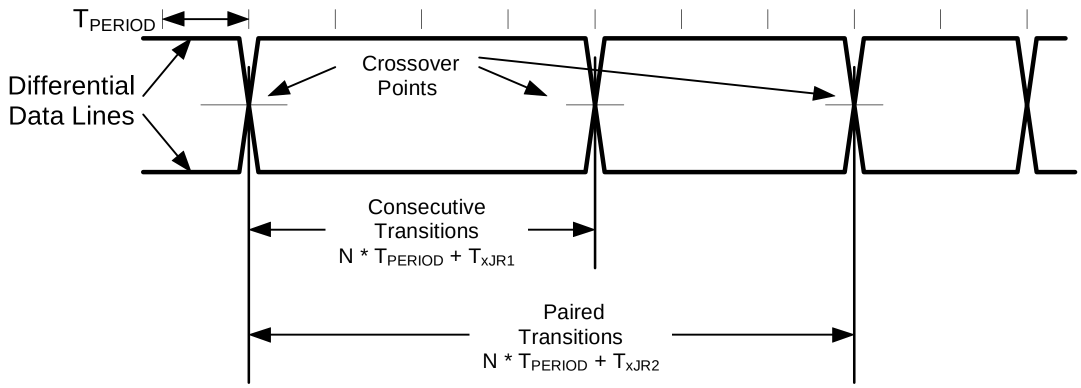


Figure 19-5. Differential Data Jitter

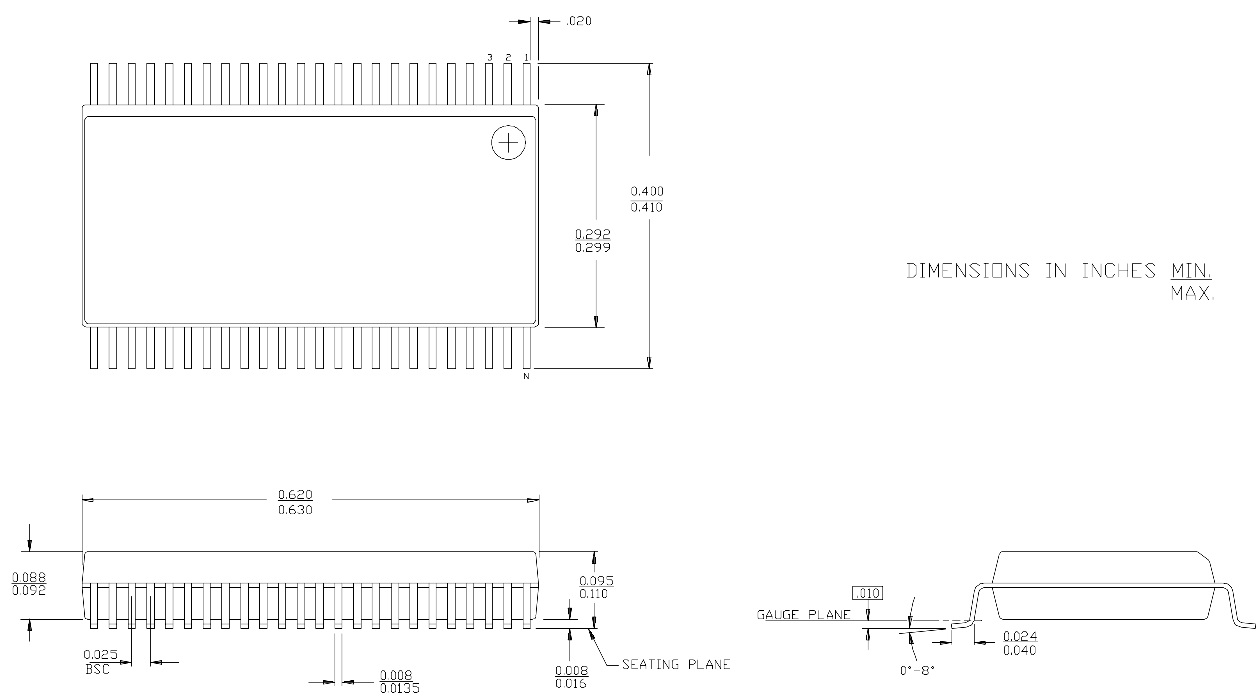
20.0 Ordering Information

Ordering Code	EPROM Size	Package Name	Package Type	Operating Range
CY7C63411-PC	4 KB	P17	40-Pin (600-Mil) PDIP	Commercial
CY7C63411-PVC	4 KB	O48	48-Lead Shrunk Small Outline Package	Commercial
CY7C63412-PC	6 KB	P17	40-Pin (600-Mil) PDIP	Commercial
CY7C63412-PVC	6 KB	O48	48-Lead Shrunk Small Outline Package	Commercial
CY7C63413-PC	8 KB	P17	40-Pin (600-Mil) PDIP	Commercial
CY7C63413-PVC	8 KB	O48	48-Lead Shrunk Small Outline Package	Commercial
CY7C63413-WC	8 KB	W18	40-Pin (600-Mil) Windowed CerDIP	Commercial
CY7C63413-WVC	8 KB	W48	48-Pin Windowed SideBraze	Commercial
CY7C63511-PVC	4 KB	O48	48-Lead Shrunk Small Outline Package	Commercial
CY7C63512-PVC	6 KB	O48	48-Lead Shrunk Small Outline Package	Commercial
CY7C63513-PVC	8 KB	O48	48-Lead Shrunk Small Outline Package	Commercial
CY7C63513-WVC	8 KB	W48	48-Pin Windowed SideBraze	Commercial

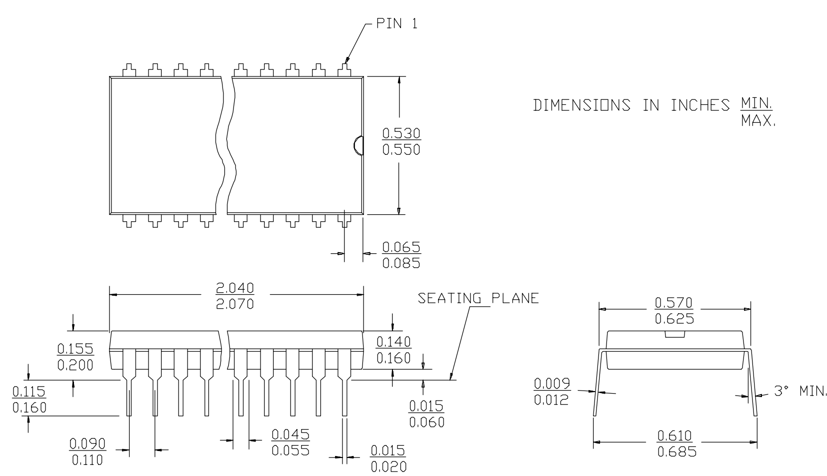
Document #: 38-00589-D

21.0 Package Diagrams

48-Lead Shrunk Small Outline Package O48

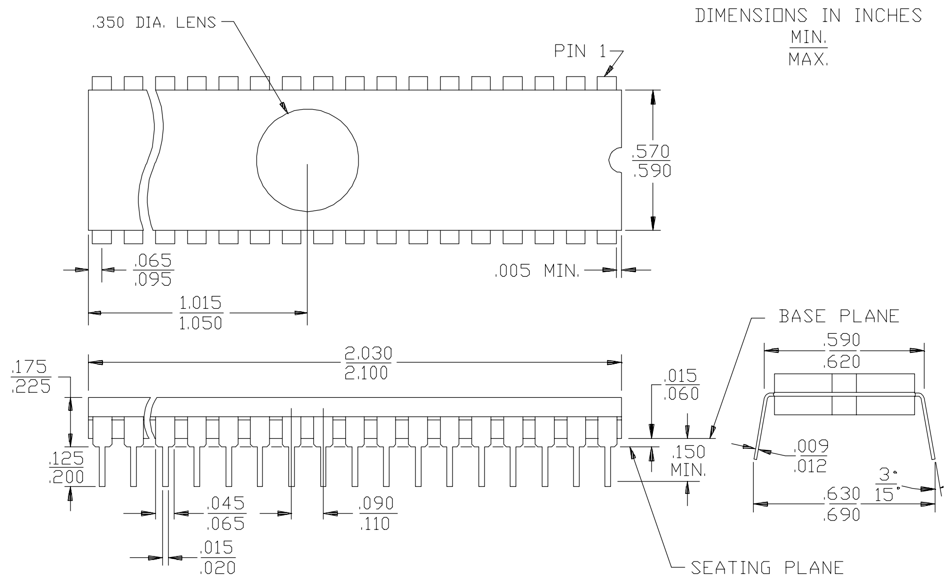


40-Lead (600-Mil) Molded DIP P17



21.0 Package Diagrams (continued)

40-Lead (600-Mil) Windowed CerDIP W18



48-Lead (600-Mil) Windowed Sidebraze W48