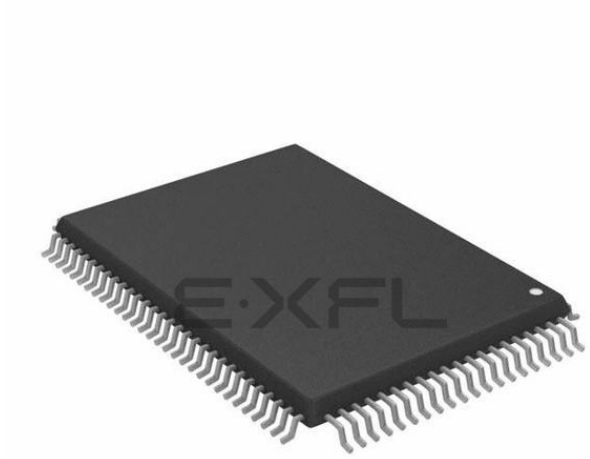

Welcome to E-XFL.COM

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	8051
Program Memory Type	ROMless
Controller Series	CY7C647xx
RAM Size	16K x 8
Interface	I²C, USB, USART
Number of I/O	40
Voltage - Supply	3.15V ~ 3.45V
Operating Temperature	0°C ~ 70°C
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	100-TQFP (14x20)
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy7c64714-100axc

Table 4-1. Special Function Registers

x	8x	9x	Ax	Bx	Cx	Dx	Ex	Fx
0	**IOA**	**IOB**	**IOC**	**IOD**	SCON1	PSW	ACC	B
1	SP	**EXIF**	**INT2CLR**	**IOE**	SBUF1			
2	DPL0	**MPAGE**	**INT4CLR**	**OEA**				
3	DPH0			**OEB**				
4	**DPL1**			**OEC**				
5	**DPH1**			**OED**				
6	**DPS**			**OEE**				
7	PCON							
8	TCON	SCON0	IE	IP	T2CON	**EICON**	**EIE**	**EIP**
9	TMOD	SBUF0						
A	TL0	**AUTOPTRH1**	**EP2468STAT**	**EP01STAT**	RCAP2L			
B	TL1	**AUTOPTRL1**	**EP24FIFOFLGS**	**GPIFTRIG**	RCAP2H			
C	TH0	**reserved**	**EP68FIFOFLGS**		TL2			
D	TH1	**AUTOPTRH2**		**GPIFSGLDATH**	TH2			
E	**CKCON**	**AUTOPTRL2**		**GPIFSGLDATLX**				
F		**reserved**	**AUTOPTRSETUP**	**GPIFSGLDATLNOX**				

4.3 I²C Bus

FX1 supports the I²C bus as a master only at 100/400 KHz. SCL and SDA pins have open-drain outputs and hysteresis inputs. These signals must be pulled up to 3.3V, even if no I²C device is connected.

4.4 Buses

All packages: 8- or 16-bit “FIFO” bidirectional data bus, multiplexed on I/O ports B and D. 128-pin package: adds 16-bit output-only 8051 address bus, 8-bit bidirectional data bus.

4.5 USB Boot Methods

During the power-up sequence, internal logic checks the I²C port for the connection of an EEPROM whose first byte is either 0xC0 or 0xC2. If found, it uses the VID/PID/DID values in the EEPROM in place of the internally stored values (0xC0), or it boot-loads the EEPROM contents into internal RAM (0xC2). If no EEPROM is detected, FX1 enumerates using internally stored descriptors. The default ID values for FX1 are VID/PID/DID (0x04B4, 0x6473, 0xAxxx where xxx=Chip revision).[2]

Table 4-2. Default ID Values for FX1

Default VID/PID/DID		
Vendor ID	0x04B4	Cypress Semiconductor
Product ID	0x6473	EZ-USB FX1
Device release	0xAnnn	Depends chip revision (nnn = chip revision where first silicon = 001)

4.6 ReNumeration™

Because the FX1’s configuration is soft, one chip can take on the identities of multiple distinct USB devices.

When first plugged into USB, the FX1 enumerates automatically and downloads firmware and USB descriptor tables over the USB cable. Next, the FX1 enumerates again, this time as a device defined by the downloaded information. This patented two-step process, called ReNumeration™, happens instantly when the device is plugged in, with no hint that the initial download step has occurred.

Two control bits in the USBCS (USB Control and Status) register control the ReNumeration process: DISCON and RENUM. To simulate a USB disconnect, the firmware sets DISCON to 1. To reconnect, the firmware clears DISCON to 0.

Before reconnecting, the firmware sets or clears the RENUM bit to indicate whether the firmware or the Default USB Device will handle device requests over endpoint zero: if RENUM = 0, the Default USB Device will handle device requests; if RENUM = 1, the firmware will.

4.7 Bus-powered Applications

The FX1 fully supports bus-powered designs by enumerating with less than 100 mA as required by the USB specification.

4.8 Interrupt System

4.8.1 INT2 Interrupt Request and Enable Registers

FX1 implements an autovector feature for INT2 and INT4. There are 27 INT2 (USB) vectors, and 14 INT4 (FIFO/GPIF) vectors. See EZ-USB Technical Reference Manual (TRM) for more details.

Note:

2. The I²C bus SCL and SDA pins must be pulled up, even if an EEPROM is not connected. Otherwise this detection method does not work properly.

4.12 Endpoint RAM

4.12.1 Size

- 3 × 64 bytes (Endpoints 0 and 1)
- 8 × 512 bytes (Endpoints 2, 4, 6, 8)

4.12.2 Organization

- EP0—Bidirectional endpoint zero, 64-byte buffer
- EP1IN, EP1OUT—64-byte buffers, bulk or interrupt
- EP2,4,6,8—Eight 512-byte buffers, bulk, interrupt, or isochronous, of which only the transfer size is available. EP4 and EP8 can be double buffered, while EP2 and 6 can be either double, triple, or quad buffered. Regardless of the physical size of the buffer, each endpoint buffer accommodates only one full-speed packet. For bulk endpoints the maximum number of bytes it can accommodate is 64, even though the physical buffer size is 512 or 1024. For an ISOCHRONOUS endpoint the maximum number of bytes it can accommodate is 1023. For endpoint configuration options, see *Figure 4-5*.

4.12.3 Setup Data Buffer

A separate 8-byte buffer at 0xE6B8-0xE6BF holds the Setup data from a CONTROL transfer.

4.12.4 Endpoint Configurations

Endpoints 0 and 1 are the same for every configuration. Endpoint 0 is the only CONTROL endpoint, and endpoint 1 can be either BULK or INTERRUPT. The endpoint buffers can be configured in any 1 of the 12 configurations shown in the vertical columns. In full-speed, BULK mode uses only the first 64 bytes of each buffer, even though memory exists for the allocation of the isochronous transfers in BULK mode the unused endpoint buffer space is not available for other operations. An example endpoint configuration would be:

EP2—1023 double buffered; EP6—64 quad buffered (column 8).

4.12.5 Default Alternate Settings

Table 4-6. Default Alternate Settings[4, 5]

Alternate Setting	0	1	2	3
ep0	64	64	64	64
ep1out	0	64 bulk	64 int	64 int
ep1in	0	64 bulk	64 int	64 int
ep2	0	64 bulk out (2×)	64 int out (2×)	64 iso out (2×)
ep4	0	64 bulk out (2×)	64 bulk out (2×)	64 bulk out (2×)
ep6	0	64 bulk in (2×)	64 int in (2×)	64 iso in (2×)
ep8	0	64 bulk in (2×)	64 bulk in (2×)	64 bulk in (2×)

4.13 External FIFO Interface

4.13.1 Architecture

The FX1 slave FIFO architecture has eight 512-byte blocks in the endpoint RAM that directly serve as FIFO memories, and are controlled by FIFO control signals (such as IFCLK, SLCS#, SLRD, SLWR, SLOE, PKTEND, and flags). The usable size of these buffers depend on the USB transfer mode as described in Section 4.12.2.

In operation, some of the eight RAM blocks fill or empty from the SIE, while the others are connected to the I/O transfer logic. The transfer logic takes two forms, the GPIF for internally generated control signals, or the slave FIFO interface for externally controlled transfers.

4.13.2 Master/Slave Control Signals

The FX1 endpoint FIFOS are implemented as eight physically distinct 256x16 RAM blocks. The 8051/SIE can switch any of the RAM blocks between two domains, the USB (SIE) domain and the 8051-I/O Unit domain. This switching is done virtually

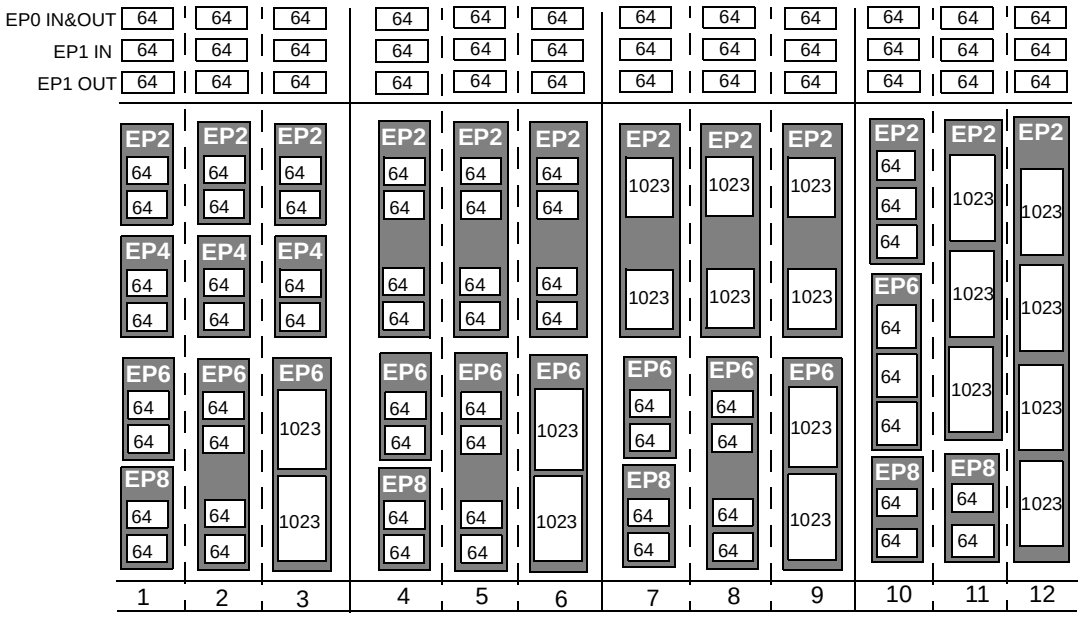

Figure 4-5. Endpoint Configuration

Notes:
4. "0" means "not implemented."
5. "2×" means "double buffered."

instantaneously, giving essentially zero transfer time between “USB FIFOS” and “Slave FIFOS.” Since they are physically the same memory, no bytes are actually transferred between buffers.

At any given time, some RAM blocks are filling/emptying with USB data under SIE control, while other RAM blocks are available to the 8051 and/or the I/O control unit. The RAM blocks operate as single-port in the USB domain, and dual-port in the 8051-I/O domain. The blocks can be configured as single, double, triple, or quad buffered as previously shown.

The I/O control unit implements either an internal-master (M for master) or external-master (S for Slave) interface.

In Master (M) mode, the GPIF internally controls FIFOADR[1..0] to select a FIFO. The RDY pins (two in the 56-pin package, six in the 100-pin and 128-pin packages) can be used as flag inputs from an external FIFO or other logic if desired. The GPIF can be run from either an internally derived clock or externally supplied clock (IFCLK), at a rate that transfers data up to 96 Megabytes/s (48-MHz IFCLK with 16-bit interface).

In Slave (S) mode, the FX1 accepts either an internally derived clock or externally supplied clock (IFCLK, max. frequency 48 MHz) and SLCS#, SLRD, SLWR, SLOE, PKTEND signals from external logic. When using an external IFCLK, the external clock must be present before switching to the external clock with the IFCLKSRC bit. Each endpoint can individually be selected for byte or word operation by an internal configuration bit, and a Slave FIFO Output Enable signal SLOE enables data of the selected width. External logic must insure that the output enable signal is inactive when writing data to a slave FIFO. The slave interface can also operate asynchronously, where the SLRD and SLWR signals act directly as strobes, rather than a clock qualifier as in synchronous mode. The signals SLRD, SLWR, SLOE and PKTEND are gated by the signal SLCS#.

4.13.3 GPIF and FIFO Clock Rates

An 8051 register bit selects one of two frequencies for the internally supplied interface clock: 30 MHz and 48 MHz. Alternatively, an externally supplied clock of 5 MHz–48 MHz feeding the IFCLK pin can be used as the interface clock. IFCLK can be configured to function as an output clock when the GPIF and FIFOs are internally clocked. An output enable bit in the IFCONFIG register turns this clock output off, if desired. Another bit within the IFCONFIG register will invert the IFCLK signal whether internally or externally sourced.

4.14 GPIF

The GPIF is a flexible 8- or 16-bit parallel interface driven by a user-programmable finite state machine. It allows the CY7C64713/4 to perform local bus mastering, and can implement a wide variety of protocols such as ATA interface, printer parallel port, and Utopia.

The GPIF has six programmable control outputs (CTL), nine address outputs (GPIFADRx), and six general-purpose ready inputs (RDY). The data bus width can be 8 or 16 bits. Each GPIF vector defines the state of the control outputs, and determines what state a ready input (or multiple inputs) must be before proceeding. The GPIF vector can be programmed to advance a FIFO to the next data value, advance an address, etc. A sequence of the GPIF vectors make up a single waveform that will be executed to perform the desired data move between the FX1 and the external device.

4.14.1 Six Control OUT Signals

The 100- and 128-pin packages bring out all six Control Output pins (CTL0-CTL5). The 8051 programs the GPIF unit to define the CTL waveforms. The 56-pin package brings out three of these signals, CTL0–CTL2. CTLx waveform edges can be programmed to make transitions as fast as once per clock (20.8 ns using a 48-MHz clock).

4.14.2 Six Ready IN Signals

The 100- and 128-pin packages bring out all six Ready inputs (RDY0–RDY5). The 8051 programs the GPIF unit to test the RDY pins for GPIF branching. The 56-pin package brings out two of these signals, RDY0–1.

4.14.3 Nine GPIF Address OUT Signals

Nine GPIF address lines are available in the 100- and 128-pin packages, GPIFADR[8..0]. The GPIF address lines allow indexing through up to a 512-byte block of RAM. If more address lines are needed, I/O port pins can be used.

4.14.4 Long Transfer Mode

In master mode, the 8051 appropriately sets GPIF transaction count registers (GPIFTCB3, GPIFTCB2, GPIFTCB1, or GPIFTCB0) for unattended transfers of up to 2^{32} transactions. The GPIF automatically throttles data flow to prevent under or overflow until the full number of requested transactions complete. The GPIF decrements the value in these registers to represent the current status of the transaction.

4.15 ECC Generation

The EZ-USB FX1 can calculate ECCs (Error-Correcting Codes) on data that passes across its GPIF or Slave FIFO interfaces. There are two ECC configurations: Two ECCs, each calculated over 256 bytes (SmartMedia™ Standard); and one ECC calculated over 512 bytes.

The ECC can correct any one-bit error or detect any two-bit error.

Note: To use the ECC logic, the GPIF or Slave FIFO interface must be configured for byte-wide operation.

4.15.1 ECC Implementation

The two ECC configurations are selected by the ECCM bit:

4.15.1.1 ECCM = 0

Two 3-byte ECCs, each calculated over a 256-byte block of data. This configuration conforms to the SmartMedia Standard.

Write any value to ECCRESET, then pass data across the GPIF or Slave FIFO interface. The ECC for the first 256 bytes of data will be calculated and stored in ECC1. The ECC for the next 256 bytes will be stored in ECC2. After the second ECC is calculated, the values in the ECCx registers will not change until ECCRESET is written again, even if more data is subsequently passed across the interface.

4.15.1.2 ECCM = 1

One 3-byte ECC calculated over a 512-byte block of data.

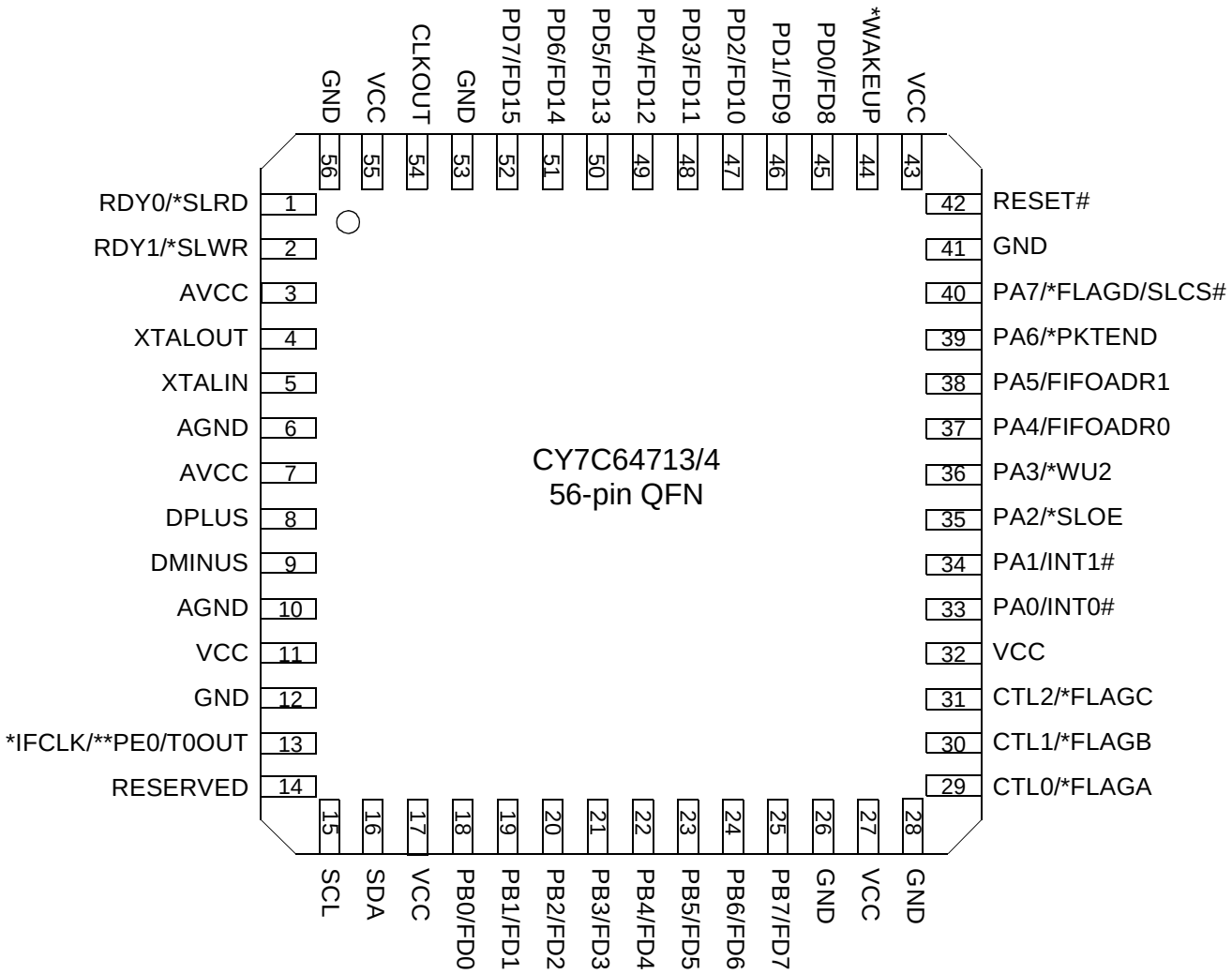

Figure 5-4. CY7C64713/4 56-pin QFN Pin Assignment

* denotes programmable polarity

Table 5-1. FX1 Pin Definitions (continued)[8]

128 TQFP	100 TQFP	56 QFN	Name	Type	Default	Description
99	77	42	RESET#	Input	N/A	**Active LOW Reset**. Resets the entire chip. See section 4.9 ”Reset and Wakeup” on page 5 for more details.
35			EA	Input	N/A	**External Access**. This pin determines where the 8051 fetches code between addresses 0x0000 and 0x3FFF. If EA = 0 the 8051 fetches this code from its internal RAM. IF EA = 1 the 8051 fetches this code from external memory.
12	11	5	XTALIN	Input	N/A	**Crystal Input**. Connect this signal to a 24-MHz parallel-resonant, fundamental mode crystal and load capacitor to GND. It is also correct to drive XTALIN with an external 24 MHz square wave derived from another clock source. When driving from an external source, the driving signal should be a 3.3V square wave.
11	10	4	XTALOUT	Output	N/A	**Crystal Output**. Connect this signal to a 24-MHz parallel-resonant, fundamental mode crystal and load capacitor to GND. If an external clock is used to drive XTALIN, leave this pin open.
1	100	54	CLKOUT	O/Z	12 MHz	**CLKOUT:** 12-, 24- or 48-MHz clock, phase locked to the 24-MHz input clock. The 8051 defaults to 12-MHz operation. The 8051 may three-state this output by setting CPUCS.1 = 1.
Port A						
82	67	33	PA0 or INT0#	I/O/Z	I (PA0)	Multiplexed pin whose function is selected by PORTACFG.0 **PA0** is a bidirectional IO port pin. **INT0#** is the active-LOW 8051 INT0 interrupt input signal, which is either edge triggered (IT0 = 1) or level triggered (IT0 = 0).
83	68	34	PA1 or INT1#	I/O/Z	I (PA1)	Multiplexed pin whose function is selected by: PORTACFG.1 **PA1** is a bidirectional IO port pin. **INT1#** is the active-LOW 8051 INT1 interrupt input signal, which is either edge triggered (IT1 = 1) or level triggered (IT1 = 0).
84	69	35	PA2 or SLOE	I/O/Z	I (PA2)	Multiplexed pin whose function is selected by two bits: IFCONFIG[1:0]. **PA2** is a bidirectional IO port pin. **SLOE** is an input-only output enable with programmable polarity (FIFOPINPOLAR.4) for the slave FIFOs connected to FD[7..0] or FD[15..0].
85	70	36	PA3 or WU2	I/O/Z	I (PA3)	Multiplexed pin whose function is selected by: WAKEUP.7 and OEA.3 **PA3** is a bidirectional I/O port pin. **WU2** is an alternate source for **USB Wakeup,** enabled by WU2EN bit (WAKEUP.1) and polarity set by WU2POL (WAKEUP.4). If the 8051 is in suspend and WU2EN = 1, a transition on this pin starts up the oscillator and interrupts the 8051 to allow it to exit the suspend mode. Asserting this pin inhibits the chip from suspending, if WU2EN = 1.
89	71	37	PA4 or FIFOADR0	I/O/Z	I (PA4)	Multiplexed pin whose function is selected by: IFCONFIG[1..0]. **PA4** is a bidirectional I/O port pin. **FIFOADR0** is an input-only address select for the slave FIFOs connected to FD[7..0] or FD[15..0].
90	72	38	PA5 or FIFOADR1	I/O/Z	I (PA5)	Multiplexed pin whose function is selected by: IFCONFIG[1..0]. **PA5** is a bidirectional I/O port pin. **FIFOADR1** is an input-only address select for the slave FIFOs connected to FD[7..0] or FD[15..0].
91	73	39	PA6 or PKTEND	I/O/Z	I (PA6)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] bits. **PA6** is a bidirectional I/O port pin. **PKTEND** is an input used to commit the FIFO packet data to the endpoint and whose polarity is programmable via FIFOPINPOLAR.5.

Table 5-1. FX1 Pin Definitions (continued)[8]

128 TQFP	100 TQFP	56 QFN	Name	Type	Default	Description
77	62		PC5 or GPIFADR5	I/O/Z	I (PC5)	Multiplexed pin whose function is selected by PORTCCFG.5 **PC5** is a bidirectional I/O port pin. **GPIFADR5** is a GPIF address output pin.
78	63		PC6 or GPIFADR6	I/O/Z	I (PC6)	Multiplexed pin whose function is selected by PORTCCFG.6 **PC6** is a bidirectional I/O port pin. **GPIFADR6** is a GPIF address output pin.
79	64		PC7 or GPIFADR7	I/O/Z	I (PC7)	Multiplexed pin whose function is selected by PORTCCFG.7 **PC7** is a bidirectional I/O port pin. **GPIFADR7** is a GPIF address output pin.
PORT D						
102	80	45	PD0 or FD[8]	I/O/Z	I (PD0)	Multiplexed pin whose function is selected by the IFCONFIG[1..0] and EPxFIFOCFG.0 (wordwide) bits. **FD[8]** is the bidirectional FIFO/GPIF data bus.
103	81	46	PD1 or FD[9]	I/O/Z	I (PD1)	Multiplexed pin whose function is selected by the IFCONFIG[1..0] and EPxFIFOCFG.0 (wordwide) bits. **FD[9]** is the bidirectional FIFO/GPIF data bus.
104	82	47	PD2 or FD[10]	I/O/Z	I (PD2)	Multiplexed pin whose function is selected by the IFCONFIG[1..0] and EPxFIFOCFG.0 (wordwide) bits. **FD[10]** is the bidirectional FIFO/GPIF data bus.
105	83	48	PD3 or FD[11]	I/O/Z	I (PD3)	Multiplexed pin whose function is selected by the IFCONFIG[1..0] and EPxFIFOCFG.0 (wordwide) bits. **FD[11]** is the bidirectional FIFO/GPIF data bus.
121	95	49	PD4 or FD[12]	I/O/Z	I (PD4)	Multiplexed pin whose function is selected by the IFCONFIG[1..0] and EPxFIFOCFG.0 (wordwide) bits. **FD[12]** is the bidirectional FIFO/GPIF data bus.
122	96	50	PD5 or FD[13]	I/O/Z	I (PD5)	Multiplexed pin whose function is selected by the IFCONFIG[1..0] and EPxFIFOCFG.0 (wordwide) bits. **FD[13]** is the bidirectional FIFO/GPIF data bus.
123	97	51	PD6 or FD[14]	I/O/Z	I (PD6)	Multiplexed pin whose function is selected by the IFCONFIG[1..0] and EPxFIFOCFG.0 (wordwide) bits. **FD[14]** is the bidirectional FIFO/GPIF data bus.
124	98	52	PD7 or FD[15]	I/O/Z	I (PD7)	Multiplexed pin whose function is selected by the IFCONFIG[1..0] and EPxFIFOCFG.0 (wordwide) bits. **FD[15]** is the bidirectional FIFO/GPIF data bus.
Port E						
108	86		PE0 or T0OUT	I/O/Z	I (PE0)	Multiplexed pin whose function is selected by the PORTECFG.0 bit. **PE0** is a bidirectional I/O port pin. **T0OUT** is an active-HIGH signal from 8051 Timer-counter0. T0OUT outputs a high level for one CLKOUT clock cycle when Timer0 overflows. If Timer0 is operated in Mode 3 (two separate timer/counters), T0OUT is active when the low byte timer/counter overflows.
109	87		PE1 or T1OUT	I/O/Z	I (PE1)	Multiplexed pin whose function is selected by the PORTECFG.1 bit. **PE1** is a bidirectional I/O port pin. **T1OUT** is an active-HIGH signal from 8051 Timer-counter1. T1OUT outputs a high level for one CLKOUT clock cycle when Timer1 overflows. If Timer1 is operated in Mode 3 (two separate timer/counters), T1OUT is active when the low byte timer/counter overflows.
110	88		PE2 or T2OUT	I/O/Z	I (PE2)	Multiplexed pin whose function is selected by the PORTECFG.2 bit. **PE2** is a bidirectional I/O port pin. **T2OUT** is the active-HIGH output signal from 8051 Timer2. T2OUT is active (HIGH) for one clock cycle when Timer/Counter 2 overflows.

Table 5-1. FX1 Pin Definitions (continued)[8]

128 TQFP	100 TQFP	56 QFN	Name	Type	Default	Description
32	26	13	IFCLK	I/O/Z	Z	Interface Clock, used for synchronously clocking data into or out of the slave FIFOs. IFCLK also serves as a timing reference for all slave FIFO control signals and GPIF. When internal clocking is used (IFCONFIG.7 = 1) the IFCLK pin can be configured to output 30/48 MHz by bits IFCONFIG.5 and IFCONFIG.6. IFCLK may be inverted, whether internally or externally sourced, by setting the bit IFCONFIG.4 = 1.
28	22		INT4	Input	N/A	**INT4** is the 8051 INT4 interrupt request input signal. The INT4 pin is edge-sensitive, active HIGH.
106	84		INT5#	Input	N/A	**INT5#** is the 8051 INT5 interrupt request input signal. The INT5 pin is edge-sensitive, active LOW.
31	25		T2	Input	N/A	**T2** is the active-HIGH T2 input signal to 8051 Timer2, which provides the input to Timer2 when C/T2 = 1. When C/T2 = 0, Timer2 does not use this pin.
30	24		T1	Input	N/A	**T1** is the active-HIGH T1 signal for 8051 Timer1, which provides the input to Timer1 when C/T1 is 1. When C/T1 is 0, Timer1 does not use this bit.
29	23		T0	Input	N/A	**T0** is the active-HIGH T0 signal for 8051 Timer0, which provides the input to Timer0 when C/T0 is 1. When C/T0 is 0, Timer0 does not use this bit.
53	43		RXD1	Input	N/A	**RXD1**is an active-HIGH input signal for 8051 UART1, which provides data to the UART in all modes.
52	42		TXD1	Output	H	**TXD1**is an active-HIGH output pin from 8051 UART1, which provides the output clock in sync mode, and the output data in async mode.
51	41		RXD0	Input	N/A	**RXD0** is the active-HIGH RXD0 input to 8051 UART0, which provides data to the UART in all modes.
50	40		TXD0	Output	H	**TXD0** is the active-HIGH TXD0 output from 8051 UART0, which provides the output clock in sync mode, and the output data in async mode.
42			CS#	Output	H	**CS#** is the active-LOW chip select for external memory.
41	32		WR#	Output	H	**WR#** is the active-LOW write strobe output for external memory.
40	31		RD#	Output	H	**RD#** is the active-LOW read strobe output for external memory.
38			OE#	Output	H	**OE#** is the active-LOW output enable for external memory.
33	27	14	Reserved	Input	N/A	**Reserved**. Connect to ground.
101	79	44	WAKEUP	Input	N/A	**USB Wakeup**. If the 8051 is in suspend, asserting this pin starts up the oscillator and interrupts the 8051 to allow it to exit the suspend mode. Holding WAKEUP asserted inhibits the EZ-USB FX1 chip from suspending. This pin has programmable polarity (WAKEUP.4).
36	29	15	SCL	OD	Z	**Clock** for the I^2C interface. Connect to VCC with a 2.2K resistor, even if no I^2C peripheral is attached.
37	30	16	SDA	OD	Z	**Data** for **I^2C interface**. Connect to VCC with a 2.2K resistor, **even if no I^2C peripheral is attached**.
2	1	55	VCC	Power	N/A	**VCC**. Connect to 3.3V power source.
26	20	11	VCC	Power	N/A	**VCC**. Connect to 3.3V power source.
43	33	17	VCC	Power	N/A	**VCC**. Connect to 3.3V power source.
48	38		VCC	Power	N/A	**VCC**. Connect to 3.3V power source.
64	49	27	VCC	Power	N/A	**VCC**. Connect to 3.3V power source.
68	53		VCC	Power	N/A	**VCC**. Connect to 3.3V power source.
81	66	32	VCC	Power	N/A	**VCC**. Connect to 3.3V power source.
100	78	43	VCC	Power	N/A	**VCC**. Connect to 3.3V power source.
107	85		VCC	Power	N/A	**VCC**. Connect to 3.3V power source.

6.0 Register Summary

FX1 register bit definitions are described in the EZ-USB TRM in greater detail.

Table 6-1. FX1 Register Summary

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
		GPIF Waveform Memories											
E400	128	WAVEDATA	GPIF Waveform Descriptor 0, 1, 2, 3 data	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E480	128	reserved											
		GENERAL CONFIGURATION											
E600	1	CPUCS	CPU Control & Status	0	0	PORTCSTB	CLKSPD1	CLKSPD0	CLKINV	CLKOE	8051RES	00000010	rrbbbbbr
E601	1	IFCONFIG	Interface Configuration (Ports, GPIF, slave FIFOs)	IFCLKSRC	3048MHZ	IFCLKOE	IFCLKPOL	ASYNC	GSTATE	IFCFG1	IFCFG0	10000000	RW
E602	1	PINFLAGSAB[9]	Slave FIFO FLAGA and FLAGB Pin Configuration	FLAGB3	FLAGB2	FLAGB1	FLAGB0	FLAGA3	FLAGA2	FLAGA1	FLAGA0	00000000	RW
E603	1	PINFLAGSCD[9]	Slave FIFO FLAGC and FLAGD Pin Configuration	FLAGD3	FLAGD2	FLAGD1	FLAGD0	FLAGC3	FLAGC2	FLAGC1	FLAGC0	00000000	RW
E604	1	FIFORESET[9]	Restore FIFOS to default state	NAKALL	0	0	0	EP3	EP2	EP1	EP0	xxxxxxxx	W
E605	1	BREAKPT	Breakpoint Control	0	0	0	0	BREAK	BPPULSE	BPEN	0	00000000	rrrrbbbr
E606	1	BPADDRH	Breakpoint Address H	A15	A14	A13	A12	A11	A10	A9	A8	xxxxxxxx	RW
E607	1	BPADDRL	Breakpoint Address L	A7	A6	A5	A4	A3	A2	A1	A0	xxxxxxxx	RW
E608	1	UART230	230 Kbaud internally generated ref. clock	0	0	0	0	0	0	230UART1	230UART0	00000000	rrrrrrbb
E609	1	FIFOPINPOLAR[9]	Slave FIFO Interface pins polarity	0	0	PKTEND	SLOE	SLRD	SLWR	EF	FF	00000000	rrbbbbbb
E60A	1	REVID	Chip Revision	rv7	rv6	rv5	rv4	rv3	rv2	rv1	rv0	RevA 00000001	R
E60B	1	REVCTL[9]	Chip Revision Control	0	0	0	0	0	0	dyn_out	enh_pkt	00000000	rrrrrrbb
		UDMA											
E60C	1	GPIFHOLDAMOUNT	MSTB Hold Time (for UDMA)	0	0	0	0	0	0	HOLDTIME1	HOLDTIME0	00000000	rrrrrrbb
	3	reserved											
		ENDPOINT CONFIGURATION											
E610	1	EP1OUTCFG	Endpoint 1-OUT Configuration	VALID	0	TYPE1	TYPE0	0	0	0	0	10100000	brbbrrrr
E611	1	EP1INCFG	Endpoint 1-IN Configuration	VALID	0	TYPE1	TYPE0	0	0	0	0	10100000	brbbrrrr
E612	1	EP2CFG	Endpoint 2 Configuration	VALID	DIR	TYPE1	TYPE0	SIZE	0	BUF1	BUF0	10100010	bbbbbrbb
E613	1	EP4CFG	Endpoint 4 Configuration	VALID	DIR	TYPE1	TYPE0	0	0	0	0	10100000	bbbbrrrr
E614	1	EP6CFG	Endpoint 6 Configuration	VALID	DIR	TYPE1	TYPE0	SIZE	0	BUF1	BUF0	11100010	bbbbbrbb
E615	1	EP8CFG	Endpoint 8 Configuration	VALID	DIR	TYPE1	TYPE0	0	0	0	0	11100000	bbbbrrrr
	2	reserved											
E618	1	EP2FIFOCFG[9]	Endpoint 2 / slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E619	1	EP4FIFOCFG[9]	Endpoint 4 / slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61A	1	EP6FIFOCFG[9]	Endpoint 6 / slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61B	1	EP8FIFOCFG[9]	Endpoint 8 / slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61C	4	reserved											
E620	1	EP2AUTOINLENH[9]	Endpoint 2 AUTOIN Packet Length H	0	0	0	0	0	PL10	PL9	PL8	00000010	rrrrrbbb
E621	1	EP2AUTOINLENL[9]	Endpoint 2 AUTOIN Packet Length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E622	1	EP4AUTOINLENH[9]	Endpoint 4 AUTOIN Packet Length H	0	0	0	0	0	0	PL9	PL8	00000010	rrrrrrbb
E623	1	EP4AUTOINLENL[9]	Endpoint 4 AUTOIN Packet Length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E624	1	EP6AUTOINLENH[9]	Endpoint 6 AUTOIN Packet Length H	0	0	0	0	0	PL10	PL9	PL8	00000010	rrrrrbbb
E625	1	EP6AUTOINLENL[9]	Endpoint 6 AUTOIN Packet Length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E626	1	EP8AUTOINLENH[9]	Endpoint 8 AUTOIN Packet Length H	0	0	0	0	0	0	PL9	PL8	00000010	rrrrrrbb
E627	1	EP8AUTOINLENL[9]	Endpoint 8 AUTOIN Packet Length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E628	1	ECCCFG	ECC Configuration	0	0	0	0	0	0	0	ECCM	00000000	rrrrrrrb
E629	1	ECCRESET	ECC Reset	x	x	x	x	x	x	x	x	00000000	W
E62A	1	ECC1B0	ECC1 Byte 0 Address	LINE15	LINE14	LINE13	LINE12	LINE11	LINE10	LINE9	LINE8	11111111	R
E62B	1	ECC1B1	ECC1 Byte 1 Address	LINE7	LINE6	LINE5	LINE4	LINE3	LINE2	LINE1	LINE0	11111111	R
E62C	1	ECC1B2	ECC1 Byte 2 Address	COL5	COL4	COL3	COL2	COL1	COL0	LINE17	LINE16	11111111	R
E62D	1	ECC2B0	ECC2 Byte 0 Address	LINE15	LINE14	LINE13	LINE12	LINE11	LINE10	LINE9	LINE8	11111111	R
E62E	1	ECC2B1	ECC2 Byte 1 Address	LINE7	LINE6	LINE5	LINE4	LINE3	LINE2	LINE1	LINE0	11111111	R

Note:

9. Read and writes to these register may require synchronization delay, see Technical Reference Manual for “Synchronization Delay.”

Table 6-1. FX1 Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
E6CB	1	FLOWSTB	Flowstate Strobe Configuration	SLAVE	RDYASYNC	CTLTOGL	SUSTAIN	0	MSTB2	MSTB1	MSTB0	00100000	RW
E6CC	1	FLOWSTBEDGE	Flowstate Rising/Falling Edge Configuration	0	0	0	0	0	0	FALLING	RISING	00000001	rrrrrrbb
E6CD	1	FLOWSTBPERIOD	Master-Strobe Half-Period	D7	D6	D5	D4	D3	D2	D1	D0	00000010	RW
E6CE	1	GPIFTCB3[9]	GPIF Transaction Count Byte 3	TC31	TC30	TC29	TC28	TC27	TC26	TC25	TC24	00000000	RW
E6CF	1	GPIFTCB2[9]	GPIF Transaction Count Byte 2	TC23	TC22	TC21	TC20	TC19	TC18	TC17	TC16	00000000	RW
E6D0	1	GPIFTCB1[9]	GPIF Transaction Count Byte 1	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8	00000000	RW
E6D1	1	GPIFTCB0[9]	GPIF Transaction Count Byte 0	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0	00000001	RW
	2	reserved										00000000	RW
		reserved											
		reserved											
E6D2	1	EP2GPIFFLGSEL[9]	Endpoint 2 GPIF Flag select	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6D3	1	EP2GPIFPFSTOP	Endpoint 2 GPIF stop transaction on prog. flag	0	0	0	0	0	0	0	FIFO2FLAG	00000000	RW
E6D4	1	EP2GPIFTRIG[9]	Endpoint 2 GPIF Trigger	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	reserved											
		reserved											
		reserved											
E6DA	1	EP4GPIFFLGSEL[9]	Endpoint 4 GPIF Flag select	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6DB	1	EP4GPIFPFSTOP	Endpoint 4 GPIF stop transaction on GPIF Flag	0	0	0	0	0	0	0	FIFO4FLAG	00000000	RW
E6DC	1	EP4GPIFTRIG[9]	Endpoint 4 GPIF Trigger	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	reserved											
		reserved											
		reserved											
E6E2	1	EP6GPIFFLGSEL[9]	Endpoint 6 GPIF Flag select	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6E3	1	EP6GPIFPFSTOP	Endpoint 6 GPIF stop transaction on prog. flag	0	0	0	0	0	0	0	FIFO6FLAG	00000000	RW
E6E4	1	EP6GPIFTRIG[9]	Endpoint 6 GPIF Trigger	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	reserved											
		reserved											
		reserved											
E6EA	1	EP8GPIFFLGSEL[9]	Endpoint 8 GPIF Flag select	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6EB	1	EP8GPIFPFSTOP	Endpoint 8 GPIF stop transaction on prog. flag	0	0	0	0	0	0	0	FIFO8FLAG	00000000	RW
E6EC	1	EP8GPIFTRIG[9]	Endpoint 8 GPIF Trigger	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	reserved											
E6F0	1	XGPIFSGLDATH	GPIF Data H (16-bit mode only)	D15	D14	D13	D12	D11	D10	D9	D8	xxxxxxxx	RW
E6F1	1	XGPIFSGLDATLX	Read/Write GPIF Data L & trigger transaction	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E6F2	1	XGPIFSGLDATL-NOX	Read GPIF Data L, no transaction trigger	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	R
E6F3	1	GPIFREADYCFG	Internal RDY, Sync/Async, RDY pin states	INTRDY	SAS	TCXRDY5	0	0	0	0	0	00000000	bbbrrrrr
E6F4	1	GPIFREADYSTAT	GPIF Ready Status	0	0	RDY5	RDY4	RDY3	RDY2	RDY1	RDY0	00xxxxxx	R
E6F5	1	GPIFABORT	Abort GPIF Waveforms	x	x	x	x	x	x	x	x	xxxxxxxx	W
E6F6	2	reserved											
		ENDPOINT BUFFERS											
E740	64	EP0BUF	EP0-IN/-OUT buffer	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E780	64	EP10UTBUF	EP1-OUT buffer	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E7C0	64	EP1INBUF	EP1-IN buffer	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
	2048	reserved											RW
F000	1023	EP2FIFOBUF	64/1023-byte EP 2 / slave FIFO buffer (IN or OUT)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
F400	64	EP4FIFOBUF	64 byte EP 4 / slave FIFO buffer (IN or OUT)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
F600	64	reserved											
F800	1023	EP6FIFOBUF	64/1023-byte EP 6 / slave FIFO buffer (IN or OUT)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
FC00	64	EP8FIFOBUF	64 byte EP 8 / slave FIFO buffer (IN or OUT)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
FE00	64	reserved											
xxxx		I²C Configuration Byte		0	DISCON	0	0	0	0	0	400KHZ	xxxxxxxx [11]	n/a
		Special Function Registers (SFRs)											
80	1	IOA[10]	Port A (bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW

10.4 Data Memory Write

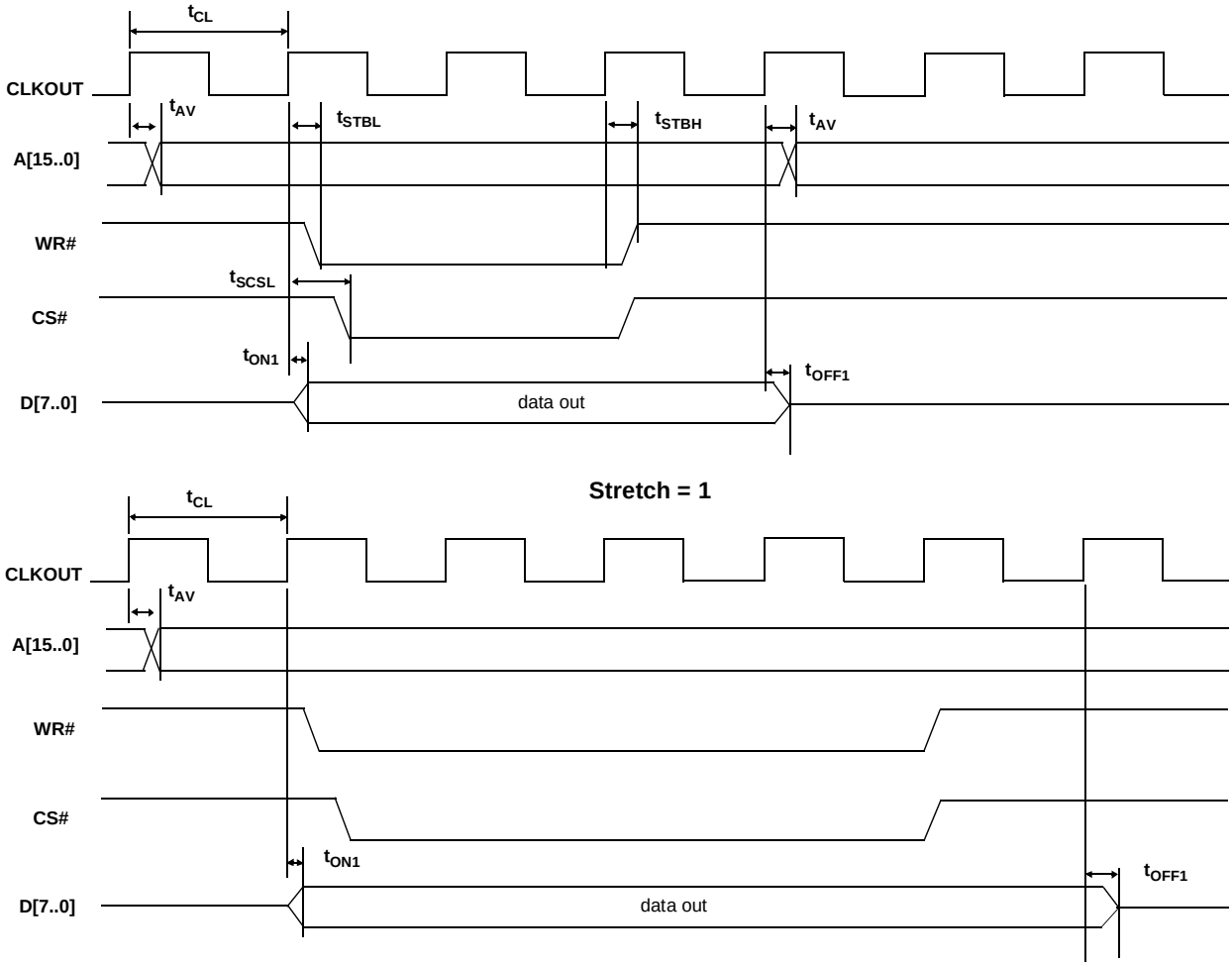

Figure 10-3. Data Memory Write Timing Diagram

Table 10-3. Data Memory Write Parameters

Parameter	Description	Min.	Max.	Unit	Notes
t_{AV}	Delay from Clock to Valid Address	0	10.7	ns	
t_{STBL}	Clock to WR Pulse LOW	0	11.2	ns	
t_{STBH}	Clock to WR Pulse HIGH	0	11.2	ns	
t_{SCSL}	Clock to CS Pulse LOW		13.0	ns	
t_{ON1}	Clock to Data Turn-on	0	13.1	ns	
t_{OFF1}	Clock to Data Hold Time	0	13.1	ns	

When using the AUTPOPTR1 or AUTOPTR2 to address external memory, the address of AUTOPTR1 will only be active while either RD# or WR# are active. The address of AUTOPTR2 will be active throughout the cycle and meet the above address valid time for which is based on the stretch value.

10.7 Slave FIFO Synchronous Read

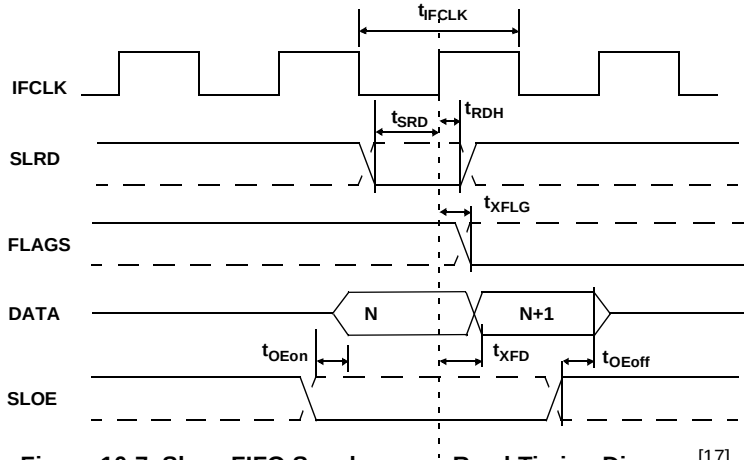

Figure 10-7. Slave FIFO Synchronous Read Timing Diagram[17]

Table 10-6. Slave FIFO Synchronous Read Parameters with Internally Sourced IFCLK[19]

Parameter	Description	Min.	Max.	Unit
t_{IFCLK}	IFCLK Period	20.83		ns
t_{SRD}	SLRD to Clock Setup Time	18.7		ns
t_{RDH}	Clock to SLRD Hold Time	0		ns
t_{OEon}	SLOE Turn-on to FIFO Data Valid		10.5	ns
t_{OEoff}	SLOE Turn-off to FIFO Data Hold		10.5	ns
t_{XFLG}	Clock to FLAGS Output Propagation Delay		9.5	ns
t_{XFD}	Clock to FIFO Data Output Propagation Delay		11	ns

Table 10-7. Slave FIFO Synchronous Read Parameters with Externally Sourced IFCLK[19]

Parameter	Description	Min.	Max.	Unit
t_{IFCLK}	IFCLK Period	20.83	200	ns
t_{SRD}	SLRD to Clock Setup Time	12.7		ns
t_{RDH}	Clock to SLRD Hold Time	3.7		ns
t_{OEon}	SLOE Turn-on to FIFO Data Valid		10.5	ns
t_{OEoff}	SLOE Turn-off to FIFO Data Hold		10.5	ns
t_{XFLG}	Clock to FLAGS Output Propagation Delay		13.5	ns
t_{XFD}	Clock to FIFO Data Output Propagation Delay		15	ns

configured to operate in auto mode and it is desired to send two packets back to back:

- A full packet (full defined as the number of bytes in the FIFO meeting the level set in AUTOINLEN register) committed automatically followed by
- A short one byte/word packet committed manually using the PKTEND pin.

In this particular scenario, the developer must make sure to assert PKTEND at least one clock cycle after the rising edge that caused the last byte/word to be clocked into the previous auto committed packet. *Figure 10-12* below shows this scenario. X is the value the AUTOINLEN register is set to when the IN endpoint is configured to be in auto mode.

Figure 10-12 shows a scenario where two packets are being committed. The first packet gets committed automatically when the number of bytes in the FIFO reaches X (value set in AUTOINLEN register) and the second one byte/word short packet being committed manually using PKTEND. Note that there is at least one IFCLK cycle timing between asserting PKTEND and clocking of the last byte of the previous packet (causing the packet to be committed automatically). Failing to adhere to this timing, will result in the FX2 failing to send the one byte/word short packet.

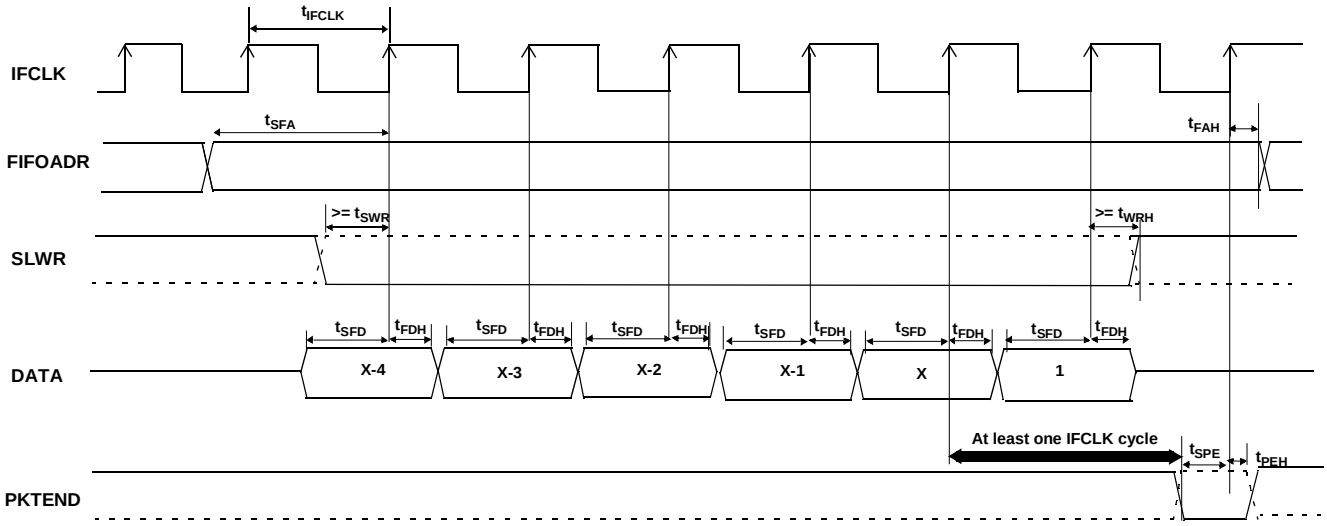

Figure 10-12. Slave FIFO Synchronous Write Sequence and Timing Diagram

10.12 Slave FIFO Asynchronous Packet End Strobe

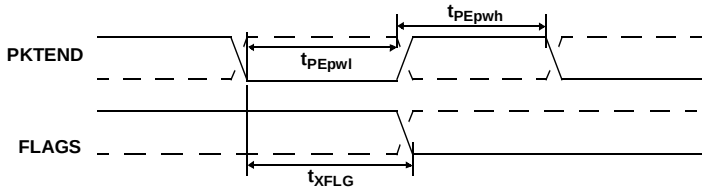

Figure 10-13. Slave FIFO Asynchronous Packet End Strobe Timing Diagram[17]

Table 10-14. Slave FIFO Asynchronous Packet End Strobe Parameters[20]

Parameter	Description	Min.	Max.	Unit
t_{PEpwl}	PKTEND Pulse Width LOW	50		ns
t_{PWpwh}	PKTEND Pulse Width HIGH	50		ns
t_{XFLG}	PKTEND to FLAGS Output Propagation Delay		115	ns

10.13 Slave FIFO Output Enable

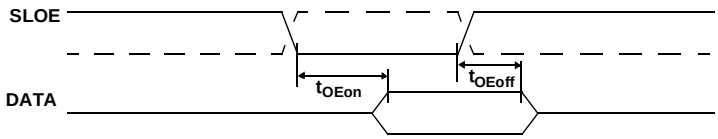

Figure 10-14. Slave FIFO Output Enable Timing Diagram[17]

Table 10-15. Slave FIFO Output Enable Parameters

Parameter	Description	Min.	Max.	Unit
t_{OEon}	SLOE Assert to FIFO DATA Output		10.5	ns
t_{OEoff}	SLOE Deassert to FIFO DATA Hold		10.5	ns

10.14 Slave FIFO Address to Flags/Data

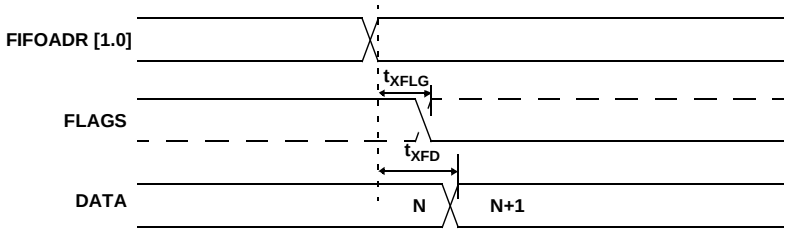

Figure 10-15. Slave FIFO Address to Flags/Data Timing Diagram[17]

Table 10-16. Slave FIFO Address to Flags/Data Parameters

Parameter	Description	Min.	Max.	Unit
t_{XFLG}	FIFOADR[1:0] to FLAGS Output Propagation Delay		10.7	ns
t_{XFD}	FIFOADR[1:0] to FIFODATA Output Propagation Delay		14.3	ns

10.15 Slave FIFO Synchronous Address

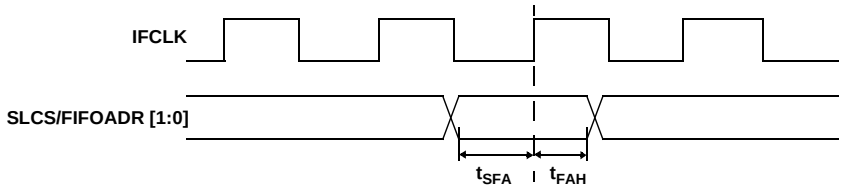

Figure 10-16. Slave FIFO Synchronous Address Timing Diagram

Table 10-17. Slave FIFO Synchronous Address Parameters [19]

Parameter	Description	Min.	Max.	Unit
t_{IFCLK}	Interface Clock Period	20.83	200	ns
t_{SFA}	FIFOADR[1:0] to Clock Setup Time	25		ns
t_{FAH}	Clock to FIFOADR[1:0] Hold Time	10		ns

10.16 Slave FIFO Asynchronous Address

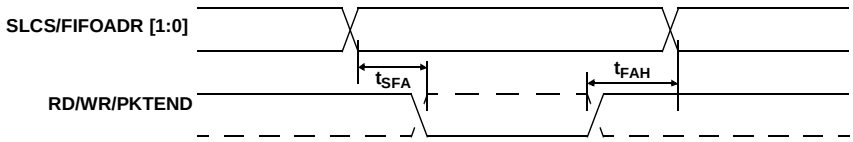

Figure 10-17. Slave FIFO Asynchronous Address Timing Diagram[17]

Table 10-18. Slave FIFO Asynchronous Address Parameters[20]

Parameter	Description	Min.	Max.	Unit
t_{SFA}	FIFOADR[1:0] to RD/WR/PKTEND Setup Time	10		ns
t_{FAH}	RD/WR/PKTEND to FIFOADR[1:0] Hold Time	10		ns

10.17.2 Single and Burst Synchronous Write

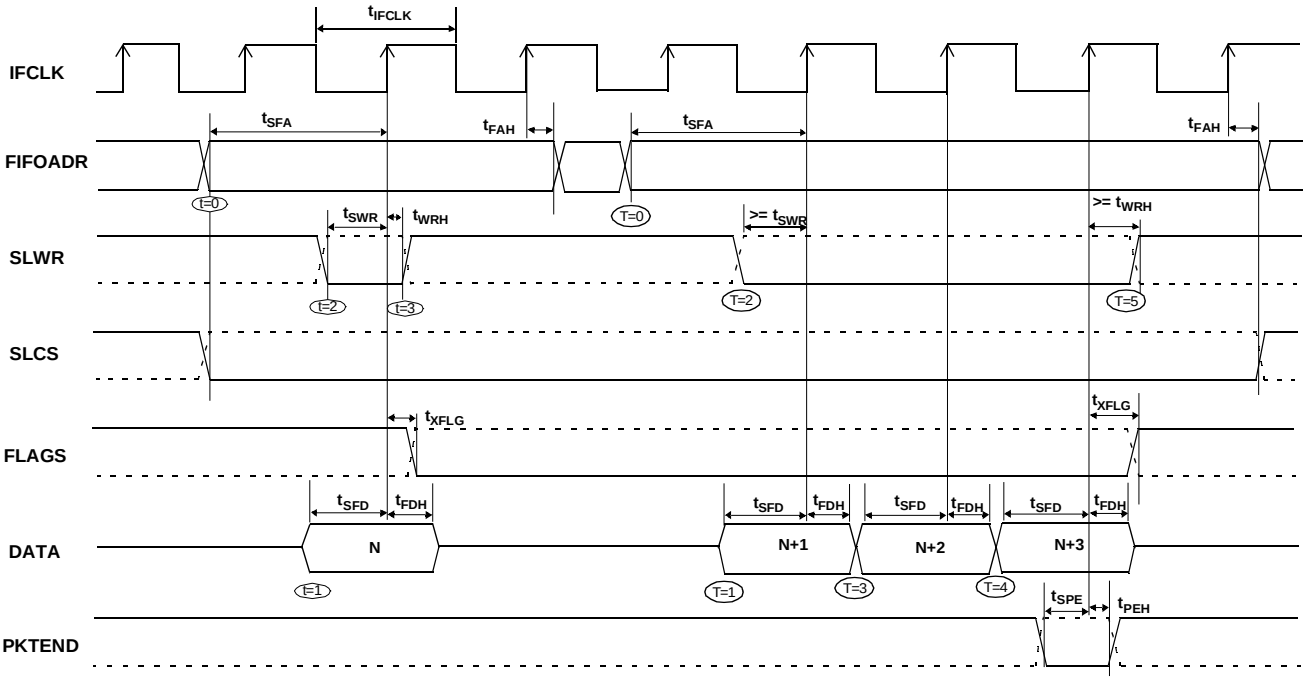

Figure 10-20. Slave FIFO Synchronous Write Sequence and Timing Diagram[17]

The *Figure 10-20* shows the timing relationship of the SLAVE FIFO signals during a synchronous write using IFCLK as the synchronizing clock. The diagram illustrates a single write followed by burst write of 3 bytes and committing all 4 bytes as a short packet using the PKTEND pin.

- At t = 0 the FIFO address is stable and the signal SLCS is asserted. (SLCS may be tied low in some applications) Note: t_{SFA} has a minimum of 25 ns. This means when IFCLK is running at 48 MHz, the FIFO address setup time is more than one IFCLK cycle.
- At t = 1, the external master/peripheral must outputs the data value onto the data bus with a minimum set up time of t_{SFD} before the rising edge of IFCLK.
- At t = 2, SLWR is asserted. The SLWR must meet the setup time of t_{SWR} (time from asserting the SLWR signal to the rising edge of IFCLK) and maintain a minimum hold time of t_{WRH} (time from the IFCLK edge to the deassertion of the SLWR signal). If SLCS signal is used, it must be asserted with SLWR or before SLWR is asserted. (i.e., the SLCS and SLWR signals must both be asserted to start a valid write condition).
- While the SLWR is asserted, data is written to the FIFO and on the rising edge of the IFCLK, the FIFO pointer is incremented. The FIFO flag will also be updated after a delay of t_{XFLG} from the rising edge of the clock.

The same sequence of events are also shown for a burst write and are marked with the time indicators of T = 0 through 5. **Note:** For the burst mode, SLWR and SLCS are left asserted for the entire duration of writing all the required data values. In this burst write mode, once the SLWR is asserted, the data on the FIFO data bus is written to the FIFO on every rising edge of IFCLK. The FIFO pointer is updated on each rising edge of IFCLK. In *Figure 10-20*, once the four bytes are written to the FIFO, SLWR is de-asserted. The short 4-byte packet can be committed to the host by asserting the PKTEND signal.

There is no specific timing requirement that needs to be met for asserting PKTEND signal with regards to asserting the SLWR signal. PKTEND can be asserted with the last data value or thereafter. The only consideration is the setup time t_{SPE} and the hold time t_{PEH} must be met. In the scenario of *Figure 10-20*, the number of data values committed includes the last value written to the FIFO. In this example, both the data value and the PKTEND signal are clocked on the same rising edge of IFCLK. PKTEND can be asserted in subsequent clock cycles. The FIFOADDR lines should be held constant during the PKTEND assertion.

Although there are no specific timing requirement for asserting PKTEND, there is a specific corner case condition that needs attention while using the PKTEND to commit a one byte/word packet. Additional timing requirements exists when the FIFO is configured to operate in auto mode and it is desired to send two packets: a full packet (full defined as the number of bytes in the FIFO meeting the level set in AUTOINLEN register) committed automatically followed by a short one byte/word packet committed manually using the PKTEND pin. In this case, the external master must make sure to assert the PKTEND pin at least one clock cycle after the rising edge that caused the last byte/word to be clocked into the previous auto committed packet (the packet with the number of bytes equal to what is set in the AUTOINLEN register). Refer to section 10-10 for further details on this timing.

10.17.4 Sequence Diagram of a Single and Burst Asynchronous Write

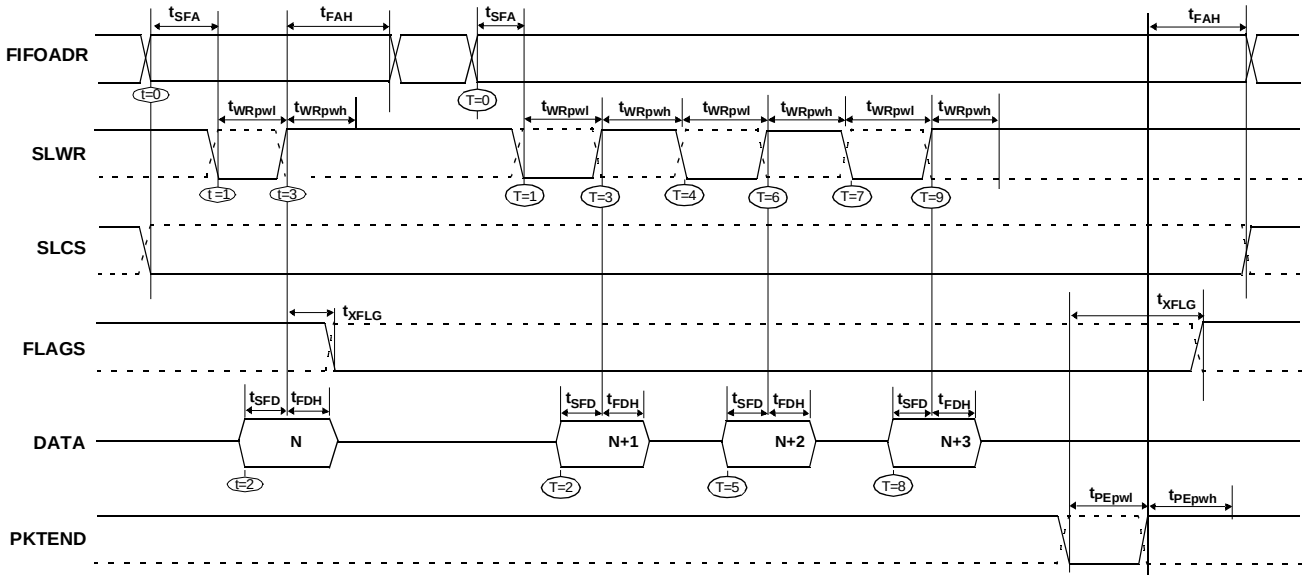

Figure 10-23. Slave FIFO Asynchronous Write Sequence and Timing Diagram[17]

Figure 10-23 diagrams the timing relationship of the SLAVE FIFO write in an asynchronous mode. The diagram shows a single write followed by a burst write of 3 bytes and committing the 4-byte-short packet using PKTEND.

- At t = 0 the FIFO address is applied, insuring that it meets the setup time of t_{SFA}. If SLCS is used, it must also be asserted (SLCS may be tied low in some applications).
- At t = 1 SLWR is asserted. SLWR must meet the minimum active pulse of t_{WRpwl} and minimum de-active pulse width of t_{WRpwh}. If the SLCS is used, it must be in asserted with SLWR or before SLWR is asserted.
- At t = 2, data must be present on the bus t_{SFD} before the deasserting edge of SLWR.
- At t = 3, deasserting SLWR will cause the data to be written from the data bus to the FIFO and then increments the FIFO pointer. The FIFO flag is also updated after t_{XFLG} from the de-asserting edge of SLWR.

The same sequence of events are shown for a burst write and is indicated by the timing marks of T = 0 through 5. **Note:** In the burst write mode, once SLWR is deasserted, the data is written to the FIFO and then the FIFO pointer is incremented to the next byte in the FIFO. The FIFO pointer is post incremented.

In *Figure 10-23*, once the four bytes are written to the FIFO and SLWR is deasserted, the short 4-byte packet can be committed to the host using the PKTEND. The external device should be designed to not assert SLWR and the PKTEND signal at the same time. It should be designed to assert the PKTEND after SLWR is deasserted and met the minimum deasserted pulse width. The FIFOADDR lines are to be held constant during the PKTEND assertion.

11.0 Ordering Information

Table 11-1. Ordering Information

Ordering Code	Package Type	RAM Size	# Prog I/Os	8051 Address /Data Busses
Ideal for battery powered applications				
CY7C64714-128AXC	128 TQFP – Lead-Free	16K	40	16/8 bit
CY7C64714-100AXC	100 TQFP – Lead-Free	16K	40	–
CY7C64714-56LFXC	56 QFN – Lead-Free	16K	24	–
Ideal for non-battery powered applications				
CY7C64713-128AXC	128 TQFP - Lead-Free	16K	40	16/8 bit
CY7C64713-100AXC	100 TQFP - Lead-Free	16K	40	-
CY7C64713-56LFXC	56 QFN - Lead-Free	16K	24	-
CY3674	EZ-USB FX1 Development Kit			

12.0 Package Diagrams

The FX1 is available in three packages:

- 56-pin QFN
- 100-pin TQFP
- 128-pin TQFP

Package Diagrams

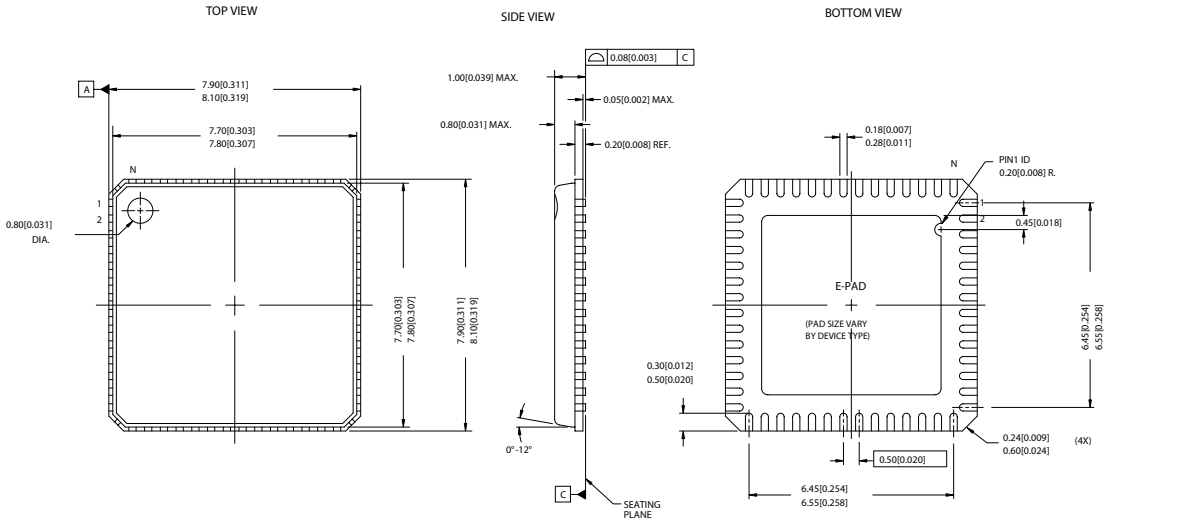

Figure 12-1. 56-Lead QFN 8 x 8 mm LF56A

Package Diagrams (continued)

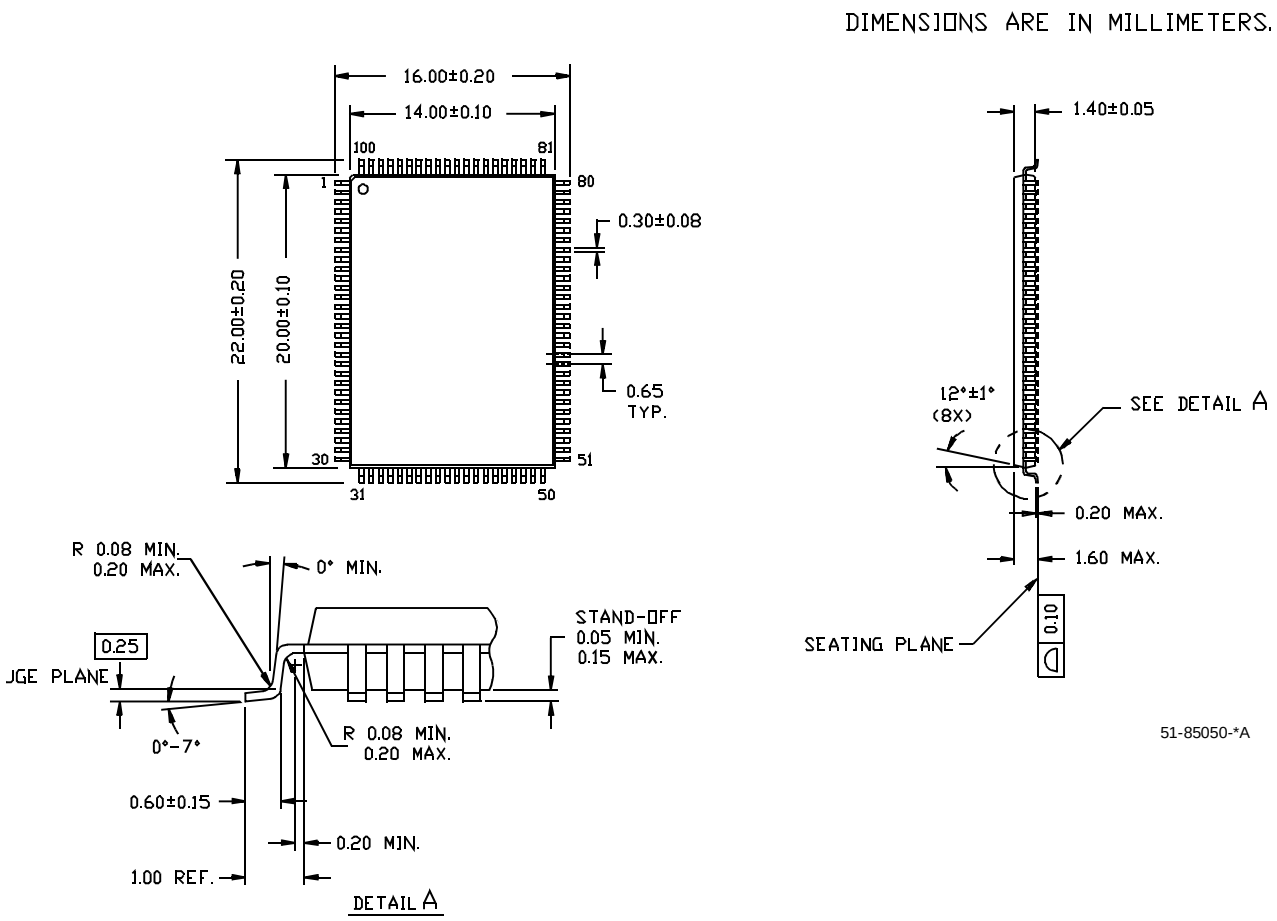

Figure 12-2. 100-Pin Thin Plastic Quad Flatpack (14 x 20 x 1.4 mm) A101

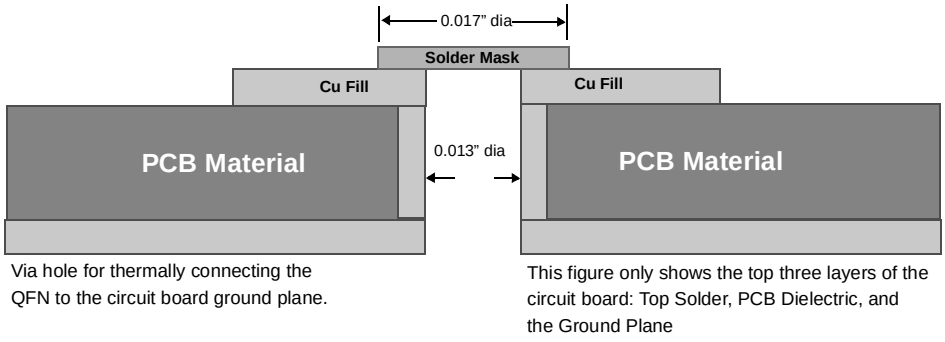

Figure 13-1. Cross-section of the Area Underneath the QFN Package

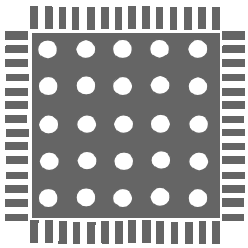

Figure 13-2. Plot of the Solder Mask (White Area)

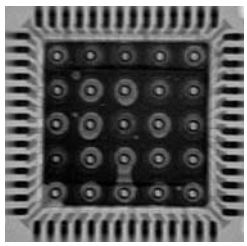

Figure 13-3. X-ray Image of the Assembly

Purchase of I^2C components from Cypress, or one of its sublicensed Associated Companies, conveys a license under the Philips I^2C Patent Rights to use these components in an I^2C system, provided that the system conforms to the I^2C Standard Specification as defined by Philips. EZ-USB FX1, EZ-USB FX2LP, EZ-USB FX2, and ReNumeration are trademarks, and EZ-USB is a registered trademark, of Cypress Semiconductor. All product and company names mentioned in this document are the trademarks of their respective holders.

Document History Page

Document Title: CY7C64713/4 EZ-USB FX1™ USB Microcontroller Full-Speed USB Peripheral Controller Document Number: 38-08039				
REV.	**ECN NO.**	**Issue Date**	**Orig. of Change**	**Description of Change**
**	132091	02/10/04	KKU	New Data Sheet
*A	230709	SEE ECN	KKU	Changed Lead free Marketing part numbers in *Table 11-1* according to spec change in 28-00054.
*B	307474	SEE ECN	BHA	Changed default PID in Table 4-2. Updated register table. Removed word compatible where associated with I2C. Changed Set-up to Setup. Added Power Dissipation. Changed Vcc from ± 10% to ± 5% Added values for V_{IH_X}, V_{IL_X} Added values for I_{CC} Added values for I_{SUSP} Removed $I_{UNCONFIGURED}$ from table 9-1 Changed PKTEND to FLAGS output propagation delay (asynchronous interface) in Table 10-14 from a maximum value of 70 ns to 115 ns. Removed 56 SSOP and added 56 QFN package Provided additional timing restrictions and requirement regarding the use of PKTEND pin to commit a short one byte/word packet subsequent to committing a packet automatically (when in auto mode). Added part number CY7C64714 ideal for battery powered applications. Changed Supply Voltage in section 8 to read +3.15V to +3.45V Added Min Vcc Ramp Up time (0 to 3.3v) Removed Preliminary
*C	392702	SEE ECN	BHA	Corrected signal name for pin 54 in Figure 5-4. Added information on the AUTOPTR1/AUTOPTR2 address timing with regards to data memory read/write timing diagram. Removed TBD in Table 10-6. Added Section 10-5.