E·XFL



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 <u>Embedded - Microcontrollers -</u> <u>Application Specific</u>?

Application enacific microcontrollars are analyzared to

Details

Product Status	Active
Applications	USB Microcontroller
Core Processor	8051
Program Memory Type	ROMIess
Controller Series	CY7C680xx
RAM Size	16K x 8
Interface	I ² C, USB, USART
Number of I/O	26
Voltage - Supply	3V ~ 3.6V
Operating Temperature	0°C ~ 70°C
Mounting Type	Surface Mount
Package / Case	56-VFQFN Exposed Pad
Supplier Device Package	56-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy7c68016a-56ltxc

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong



More Information

Cypress provides a wealth of data at www.cypress.com to help you to select the right device for your design, and to help you to quickly and effectively integrate the device into your design. For a comprehensive list of resources, see the application note AN65209 - Getting Started with FX2LP.

- Overview: USB Portfolio, USB Roadmap
- USB 2.0 Product Selectors: FX2LP, AT2LP, NX2LP-Flex, SX2
- Application notes: Cypress offers a large number of USB application notes covering a broad range of topics, from basic to advanced level. Recommended application notes for getting started with FX2LP are:
 - AN65209 Getting Started with FX2LP
 - □ AN15456 Guide to Successful EZ-USB[®] FX2LP[™] and EZ-USB FX1[™] Hardware Design and Debug
 - □ AN50963 EZ-USB[®] FX1[™]/FX2LP[™] Boot Options
 - □ AN66806 EZ-USB[®] FX2LP[™] GPIF Design Guide
 - □ AN61345 Implementing an FX2LPTM- FPGA Interface
 - □ AN57322 Interfacing SRAM with FX2LP over GPIF
 - AN4053 Streaming Data through Isochronous/Bulk Endpoints on EZ-USB[®] FX2 and EZUSB FX2LP
 - □ AN63787 EZ-USB[®] FX2LP[™] GPIF and Slave FIFO Configuration Examples using 8-bit Asynchronous Interface

For complete list of Application notes, click here.

- Code Examples: USB Hi-Speed
- Technical Reference Manual (TRM):
 EZ-USB FX2LP Technical Reference Manual
- Reference Designs:
 - CY4661 External USB Hard Disk Drives (HDD) with Fingerprint Authentication Security
 - FX2LP DMB-T/H TV Dongle reference design
- Models: IBIS

EZ-USB FX2LP Development Kit

The CY3684 EZ-USB FX2LP Development Kit is a complete development resource for FX2LP. It provides a platform to develop and test custom projects using FX2LP. The development kit contains collateral materials for the firmware, hardware, and software aspects of a design using FX2LP.

GPIF™ Designer

FX2LP[™] General Programmable Interface (GPIF) provides an independent hardware unit, which creates the data and control signals required by an external interface. FX2LP GPIF Designer allows users to create and modify GPIF waveform descriptors for EZ-USB FX2/ FX2LP family of chips using a graphical user interface. Extensive discussion of general GPIF discussion and programming using GPIF Designer is included in *FX2LP Technical Reference Manual* and *GPIF Designer User Guide*, distributed with GPIF Designer. *AN66806* - *Getting Started with EZ-USB*[®] *FX2LP*[™] *GPIF* can be a good starting point.



Table 1. Special Function Registers

x	8x	9x	Ax	Bx	Сх	Dx	Ex	Fx
0	IOA	IOB	IOC	IOD	SCON1	PSW	ACC	В
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	-	_	-
В	TL1	AUTOPTRL1	EP24FIFOFLGS	GPIFTRIG	RCAP2H	-	_	-
С	TH0	reserved	EP68FIFOFLGS		TL2	-	_	-
D	TH1	AUTOPTRH2	-	GPIFSGLDATH	TH2	_	_	_
E	CKCON	AUTOPTRL2	-	GPIFSGLDATLX	-	_	_	_
F	_	reserved	AUTOPTRSET-UP	GPIFSGLDATLNOX	_	_	_	_

USB Boot Methods

During the power-up sequence, internal logic checks the l^2C 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, FX2LP enumerates using internally stored descriptors. The default ID values for FX2LP are VID/PID/DID (0x04B4, 0x8613, 0xAxxx where xxx = Chip revision).^[2]

Table 2. Default ID Values for FX2LP

Default VID/PID/DID							
Vendor ID	0x04B4	Cypress Semiconductor					
Product ID	0x8613	EZ-USB FX2LP					
Device release	0xAnnn	Depends on chip revision (nnn = chip revision where first silicon = 001)					

ReNumeration

Because the FX2LP's configuration is soft, one chip can take on the identities of multiple distinct USB devices.

When first plugged into USB, the FX2LP enumerates automatically and downloads firmware and USB descriptor tables over the USB cable. Next, the FX2LP 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, without a 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 handles device requests over endpoint zero: if RENUM = 0, the Default USB Device handles device requests; if RENUM = 1, the firmware services the requests.

Bus-Powered Applications

The FX2LP fully supports bus-powered designs by enumerating with less than 100 mA as required by the USB 2.0 specification.

Interrupt System

INT2 Interrupt Request and Enable Registers

FX2LP 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.

USB Interrupt Autovectors

The main USB interrupt is shared by 27 interrupt sources. To save the code and processing time that is required to identify the individual USB interrupt source, the FX2LP provides a second level of interrupt vectoring, called Autovectoring. When a USB interrupt is asserted, the FX2LP pushes the program counter to its stack, and then jumps to the address 0x0043 where it expects to find a "jump" instruction to the USB interrupt service routine.

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.



In the Slave (S) mode, FX2LP 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) that enables data of the selected width. External logic must ensure 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 signal SLCS#.

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 inverts the IFCLK signal whether internally or externally sourced.

GPIF

The GPIF is a flexible 8-bit or 16-bit parallel interface driven by a user-programmable finite state machine. It enables the CY7C68013A/15A 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 is executed to perform the desired data move between the FX2LP and the external device.

Six Control OUT Signals

The 100-pin 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).

Six Ready IN Signals

The 100-pin 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.

Nine GPIF Address OUT Signals

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

Long Transfer Mode

In the master mode, the 8051 appropriately sets GPIF transaction count registers (GPIFTCB3, GPIFTCB2, GPIFTCB1, or GPIFTCB0) for unattended transfers of up to 2³² 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.

ECC Generation^[8]

The EZ-USB 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.

ECC Implementation

The two ECC configurations are selected by the ECCM bit:

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 is calculated and stored in ECC1. The ECC for the next 256 bytes is stored in ECC2. After the second ECC is calculated, the values in the ECCx registers do not change until ECCRESET is written again, even if more data is subsequently passed across the interface.

ECCM = 1

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

Write any value to ECCRESET then pass data across the GPIF or Slave FIFO interface. The ECC for the first 512 bytes of data is calculated and stored in ECC1; ECC2 is unused. After the ECC is calculated, the values in ECC1 do not change even if more data is subsequently passed across the interface, till ECCRESET is written again.

USB Uploads and Downloads

The core has the ability to directly edit the data contents of the internal 16-KB RAM and of the internal 512-byte scratch pad RAM via a vendor-specific command. This capability is normally used when soft downloading the user code and is available only to and from the internal RAM, only when the 8051 is held in reset. The available RAM spaces are 16 KB from 0x0000–0x3FFF (code/data) and 512 bytes from 0xE000–0xE1FF (scratch pad data RAM)^[9].

Notes

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

^{9.} After the data is downloaded from the host, a "loader" can execute from internal RAM to transfer downloaded data to external memory.



Autopointer Access

FX2LP provides two identical autopointers. They are similar to the internal 8051 data pointers but with an additional feature: they can optionally increment after every memory access. This capability is available to and from both internal and external RAM. Autopointers are available in external FX2LP registers under the control of a mode bit (AUTOPTRSET-UP.0). Using the external FX2LP autopointer access (at 0xE67B – 0xE67C) enables the autopointer to access all internal and external RAM to the part.

Also, autopointers can point to any FX2LP register or endpoint buffer space. When the autopointer access to external memory is enabled, locations 0xE67B and 0xE67C in XDATA and code space cannot be used.

I²C Controller

FX2LP has one I²C port that is driven by two internal controllers, the one that automatically operates at boot time to load VID/PID/DID and configuration information, and another that the 8051 uses when running to control external I²C devices. The I²C port operates in master mode only.

²C Port Pins

The I²C pins SCL and SDA must have external 2.2-k Ω pull-up resistors even if no EEPROM is connected to the FX2LP. External EEPROM device address pins must be configured properly. See Table 8 for configuring the device address pins.

Table 8. Strap Boot EEPROM Address Lines to These Values

Bytes	Example EEPROM	A2	A1	A0
16	24LC00 ^[10]	N/A	N/A	N/A
128	24LC01	0	0	0
256	24LC02	0	0	0
4K	24LC32	0	0	1
8K	24LC64	0	0	1
16K	24LC128	0	0	1

P^2C Interface Boot Load Access

At power-on reset, the I^2C interface boot loader loads the VID/PID/DID configuration bytes and up to 16 KB of program/data. The available RAM spaces are 16 KB from 0x0000–0x3FFF and 512 bytes from 0xE000–0xE1FF. The 8051 is in reset. I^2C interface boot loads only occur after power-on reset.

PC Interface General-Purpose Access

The 8051 can control peripherals connected to the I^2C bus using the I2CTL and I2DAT registers. FX2LP provides I^2C master control only; it is never an I^2C slave.

Compatible with Previous Generation EZ-USB FX2

The EZ-USB FX2LP is form-, fit-, and with minor exceptions, functionally-compatible with its predecessor, the EZ-USB FX2.

This makes for an easy transition for designers wanting to upgrade their systems from the FX2 to the FX2LP. The pinout and package selection are identical and a vast majority of firmware previously developed for the FX2 functions in the FX2LP.

For designers migrating from the FX2 to the FX2LP, a change in the bill of material and review of the memory allocation (due to increased internal memory) is required. For more information about migrating from EZ-USB FX2 to EZ-USB FX2LP, see the application note titled *Migrating from EZ-USB FX2 to EZ-USB FX2 to EZ-USB FX2LP* available in the Cypress web site.

Table 9. Part Number Conversion Table

EZ-USB FX2 Part Number	EZ-USB FX2LP Part Number	Package Description
CY7C68013-56PVC	CY7C68013A-56PVXC or CY7C68014A-56PVXC	56-pin SSOP
CY7C68013-56PVCT	CY7C68013A-56PVXCTor CY7C68014A-56PVXCT	56-pin SSOP – Tape and Reel
CY7C68013-56LFC	CY7C68013A-56LFXC or CY7C68014A-56LFXC	56-pin QFN
CY7C68013-100AC	CY7C68013A-100AXC or CY7C68014A-100AXC	100-pin TQFP
CY7C68013-128AC	CY7C68013A-128AXC or CY7C68014A-128AXC	128-pin TQFP

CY7C68013A/14A and CY7C68015A/16A Differences

CY7C68013A is identical to CY7C68014A in form, fit, and functionality. CY7C68015A is identical to CY7C68016A in form, fit, and functionality. CY7C68014A and CY7C68016A have a lower suspend current than CY7C68013A and CY7C68015A respectively and are ideal for power-sensitive battery applications.

CY7C68015A and CY7C68016A are available in 56-pin QFN package only. Two additional GPIO signals are available on the CY7C68015A and CY7C68016A to provide more flexibility when neither IFCLK or CLKOUT are needed in the 56-pin package.

USB developers wanting to convert their FX2 56-pin application to a bus-powered system directly benefit from these additional signals. The two GPIOs give developers the signals they need for the power-control circuitry of their bus-powered application without pushing them to a high-pincount version of FX2LP.

The CY7C68015A is only available in the 56-pin QFN package

Table 10. CY7C68013A/14A and CY7C68015A/16A Pin Differences

CY7C68013A/CY7C68014A	CY7C68015A/CY7C68016A		
IFCLK	PE0		
CLKOUT	PE1		





* denotes programmable polarity



CY7C68013A, CY7C68014A CY7C68015A, CY7C68016A



Figure 11. CY7C68013A 56-pin VFBGA Pin Assignment – Top View



CY7C68013A/15A Pin Descriptions

Table 11. FX2LP Pin Descriptions^[11]

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	Name	Туре	Default	Reset ^[12]	Description
10	9	10	3	2D	AVCC	Power	N/A	N/A	Analog VCC . Connect this pin to the 3.3 V power source. This signal provides power to the analog section of the chip.
17	16	14	7	1D	AVCC	Power	N/A	N/A	Analog VCC . Connect this pin to the 3.3 V power source. This signal provides power to the analog section of the chip.
13	12	13	6	2F	AGND	Ground	N/A	N/A	Analog Ground . Connect to ground with as short a path as possible.
20	19	17	10	1F	AGND	Ground	N/A	N/A	Analog Ground . Connect to ground with as short a path as possible.
19	18	16	9	1E	DMINUS	I/O/Z	Z	N/A	USB D- Signal. Connect to the USB D- signal.
18	17	15	8	2E	DPLUS	I/O/Z	Z	N/A	USB D+ Signal. Connect to the USB D+ signal.
94	-	-	—	-	A0	Output	L	L	
95	-	-	—	-	A1	Output	L	L	
96	-	-	—	_	A2	Output	L	L	
97	_	_	_	-	A3	Output	L	L	
117	-	-	—	_	A4	Output	L	L	
118	_	-	—	-	A5	Output	L	L	
119	_	_	_	-	A6	Output	L	L	
120	_	-	—	-	A7	Output	L	L	8051 Address Bus . This bus is driven at all times.
126	_	-	—	-	A8	Output	L	L	reflects the internal address.
127	_	_	_	-	A9	Output	L	L	
128	—	-	—	-	A10	Output	L	L	
21	_	-	—	-	A11	Output	L	L	
22	_	-	—	-	A12	Output	L	L	
23	_	-	—	-	A13	Output	L	L	
24	_	-	—	-	A14	Output	L	L	
25	_	-	—	-	A15	Output	L	L	
59	—	-	—	-	D0	I/O/Z	Z	Z	
60	—	-	—	-	D1	I/O/Z	Z	Z	
61	_	-	—	-	D2	I/O/Z	Z	Z	8051 Data Bus. This bidirectional bus is
62	_	-	—	-	D3	I/O/Z	Z	Z	and output for bus writes. The data bus is used for
63	-	-	—	-	D4	I/O/Z	Z	Z	external 8051 program and data memory. The data
86	_	-	—	-	D5	I/O/Z	Z	Z	driven LOW in suspend.
87	_	-	—	-	D6	I/O/Z	Z	Z	
88	-	-	—	-	D7	I/O/Z	Z	Z	
39	_	_	_	_	PSEN#	Output	н	н	Program Store Enable . This active LOW signal indicates an 8051 code fetch from external memory. It is active for program memory fetches from 0x4000–0xFFFF when the EA pin is LOW, or from 0x0000–0xFFFF when the EA pin is HIGH.

 Notes

 11. Unused inputs must not be left floating. Tie either HIGH or LOW as appropriate. Outputs should only be pulled up or down to ensure signals at power up and in standby. Note also that no pins should be driven while the device is powered down.

 12. The Reset column indicates the state of signals during reset (RESET# asserted) or during Power on Reset (POR).



Register Summary

FX2LP register bit definitions are described in the FX2LP TRM in greater detail.

Table 12. FX2LP Register Summary

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
		GPIF Waveform Memo	ories										
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 CONFIGUI	RATION										
E50D		GPCR2	General Purpose Configu- ration Register 2	reserved	reserved	reserved	FULL_SPEE D_ONLY	reserved	reserved	reserved	reserved	00000000	R
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 ^[13]	Slave FIFO FLAGA and FLAGB Pin Configuration	FLAGB3	FLAGB2	FLAGB1	FLAGB0	FLAGA3	FLAGA2	FLAGA1	FLAGA0	00000000	RW
E603	1	PINFLAGSCD ^[13]	Slave FIFO FLAGC and FLAGD Pin Configuration	FLAGD3	FLAGD2	FLAGD1	FLAGD0	FLAGC3	FLAGC2	FLAGC1	FLAGC0	00000000	RW
E604	1	FIFORESET ^[13]	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	xxxxxxx	RW
E607	1	BPADDRL	Breakpoint Address L	A7	A6	A5	A4	A3	A2	A1	A0	xxxxxxx	RW
E608	1	UART230	230 Kbaud internally generated ref. clock	0	0	0	0	0	0	230UART1	230UART0	00000000	rrrrrbb
E609	1	FIFOPINPOLAR ^[13]	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 ^[13]	Chip Revision Control	0	0	0	0	0	0	dyn_out	enh_pkt	00000000	rrrrrbb
		UDMA											
E60C	1	GPIFHOLDAMOUNT	MSTB Hold Time (for UDMA)	0	0	0	0	0	0	HOLDTIME1	HOLDTIME0	00000000	rrrrrbb
	3	reserved											
		ENDPOINT CONFIGU	IRATION										
E610	1	EP1OUTCFG	Endpoint 1-OUT	VALID	0	TYPE1	TYPE0	0	0	0	0	10100000	brbbrrrr
E611	1	EP1INCFG	Configuration Endpoint 1-IN	VALID	0	TYPE1	TYPE0	0	0	0	0	10100000	brbbrrrr
5040		500050	Configuration		DID			0175	0	DUEA	DUEO	40400040	
E612	1	EP2CFG	Endpoint 2 Configuration	VALID	DIR	TYPE1	TYPE0	SIZE	0	BUF1	BUFU	10100010	daradada
E613	1	EP4CFG	Endpoint 4 Configuration	VALID	DIR	TYPE1	TYPE0	0	0	0	0	10100000	pppprrrr
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 ^[13]	Endpoint 2 / slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E619	1	EP4FIFOCFG ^[13]	Endpoint 4 / slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61A	1	EP6FIFOCFG ^[13]	Endpoint 6 / slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61B	1	EP8FIFOCFG ^[13]	Endpoint 8 / slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61C	4	reserved											
E620	1	EP2AUTOINLENH ^{[13}	Endpoint 2 AUTOIN Packet Length H	0	0	0	0	0	PL10	PL9	PL8	00000010	rrrrbbb
E621	1	EP2AUTOINLENL ^[13]	Endpoint 2 AUTOIN Packet Length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E622	1	EP4AUTOINLENH ^[13]	Endpoint 4 AUTOIN Packet Length H	0	0	0	0	0	0	PL9	PL8	00000010	rrrrrbb
E623	1	EP4AUTOINLENL ^[13]	Endpoint 4 AUTOIN Packet Length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E624	1	EP6AUTOINLENH ^[13]	Endpoint 6 AUTOIN Packet Length H	0	0	0	0	0	PL10	PL9	PL8	00000010	rrrrbbb
E625	1	EP6AUTOINLENL ^[13]	Endpoint 6 AUTOIN Packet Length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E626	1	EP8AUTOINLENH ^[13]	Endpoint 8 AUTOIN Packet Length H	0	0	0	0	0	0	PL9	PL8	00000010	rrrrrbb
E627	1	EP8AUTOINLENL ^[13]	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	rrrrrb
E629	1	ECCRESET	ECC Reset	x	x	x	х	х	х	х	х	00000000	W
E62A	1	ECC1B0	ECC1 Byte 0 Address	LINE15	LINE14	LINE13	LINE12	LINE11	LINE10	LINE9	LINE8	00000000	R

Note

13. Read and writes to these registers may require synchronization delay; see Technical Reference Manual for "Synchronization Delay."



Table 12. FX2LP Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
E65D	1	USBIRQ ^[14]	USB Interrupt Requests	0	EPOACK	HSGRANT	URES	SUSP	SUTOK	SOF	SUDAV	0xxxxxxx	rbbbbbbb
E65E	1	EPIE	Endpoint Interrupt	EP8	EP6	EP4	EP2	EP1OUT	EP1IN	EP0OUT	EPOIN	00000000	RW
E65F	1	EPIRQ ^[14]	Endpoint Interrupt	EP8	EP6	EP4	EP2	EP1OUT	EP1IN	EP0OUT	EP0IN	0	RW
EGGO	1		CBIE Interrupt Enchlo	0	0	0	0	0	0			0000000	DW/
E000	4		CDIF Interrupt Degruget	0	0	0	0	0	0	GEIEWE	GFIFDONE	00000000	
E001	4		GPIF Interrupt Request					0	0	GPIFWF	GPIFDONE	00000000	RW
E002	1	USBERRIE	Enables	ISUEP8	ISOEP6	ISOEP4	ISOEP2	0	0	0	ERRLINIT	0000000	RVV
E663	1	USBERRIRQ ^[14]	USB Error Interrupt Requests	ISOEP8	ISOEP6	ISOEP4	ISOEP2	0	0	0	ERRLIMIT	0000000x	bbbbrrrb
E664	1	ERRCNTLIM	USB Error counter and limit	EC3	EC2	EC1	EC0	LIMIT3	LIMIT2	LIMIT1	LIMIT0	xxxx0100	rrrrbbbb
E665	1	CLRERRCNT	Clear Error Counter EC3:0	х	x	x	x	х	x	х	x	XXXXXXXX	W
E666	1	INT2IVEC	Interrupt 2 (USB) Autovector	0	I2V4	I2V3	I2V2	I2V1	I2V0	0	0	00000000	R
E667	1	INT4IVEC	Interrupt 4 (slave FIFO & GPIF) Autovector	1	0	14V3	14V2	I4V1	I4V0	0	0	10000000	R
E668	1	INTSET-UP	Interrupt 2&4 setup	0	0	0	0	AV2EN	0	INT4SRC	AV4EN	00000000	RW
E669	7	reserved											
		INPUT / OUTPUT											
E670	1	PORTACFG	I/O PORTA Alternate Configuration	FLAGD	SLCS	0	0	0	0	INT1	INT0	00000000	RW
E671	1	PORTCCFG	I/O PORTC Alternate Configuration	GPIFA7	GPIFA6	GPIFA5	GPIFA4	GPIFA3	GPIFA2	GPIFA1	GPIFA0	00000000	RW
E672	1	PORTECFG	I/O PORTE Alternate Configuration	GPIFA8	T2EX	INT6	RXD1OUT	RXD0OUT	T2OUT	T1OUT	TOOUT	00000000	RW
E673	4	reserved											
E677	1	reserved											
E678	1	I ² CS	I ² C Bus Control & Status	START	STOP	LASTRD	ID1	ID0	BERR	ACK	DONE	000xx000	bbbrrrrr
E679	1	I2DAT	I²C Bus Data	d7	d6	d5	d4	d3	d2	d1	d0	xxxxxxx	RW
E67A	1	I ² CTL	I ² C Bus Control	0	0	0	0	0	0	STOPIE	400KHZ	00000000	RW
E67B	1	XAUTODAT1	Autoptr1 MOVX access, when APTREN=1	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
E67C	1	XAUTODAT2	Autoptr2 MOVX access, when APTREN=1	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
		UDMA CRC											
E67D	1	UDMACRCH ^[13]	UDMA CRC MSB	CRC15	CRC14	CRC13	CRC12	CRC11	CRC10	CRC9	CRC8	01001010	RW
E67E	1	UDMACRCL ^[13]	UDMA CRC LSB	CRC7	CRC6	CRC5	CRC4	CRC3	CRC2	CRC1	CRC0	10111010	RW
E67F	1	UDMACRC- QUALIFIER	UDMA CRC Qualifier	QENABLE	0	0	0	QSTATE	QSIGNAL2	QSIGNAL1	QSIGNAL0	00000000	brrrbbbb
		USB CONTROL											
E680	1	USBCS	USB Control & Status	HSM	0	0	0	DISCON	NOSYNSOF	RENUM	SIGRSUME	x0000000	rrrrbbbb
E681	1	SUSPEND	Put chip into suspend	x	x	x	x	x	x	x	x	XXXXXXXX	W
E682	1		Wakeup Control & Status		WU	WI I2POI	WUPOL	0		WI12EN	WUEN	xx000101	hhhhrhhh
E683	1	TOGCTI	Toggle Control	0	s	R	1/0	EP3	EP2	FP1	EP0	x0000101	rrrhhhhhh
E684	1	USBERAMEH	USB Frame count H	0	0	0	0	0	EC10	FC9	EC8	000000000	R
E685	1		USB Frame count I	0 FC7	5 FC6	5 EC5	G EC4	EC3	FC2	FC1	FCO	*****	R
E686	1		Microframe count 0-7	0	0	0	0	0	ME2	ME1	MEO	00000vvv	D
E607	1		USB Eurotion address	0	5 EAG	EAE		5 EA2			EAO	00000	D
E699	2	reserved	COD F UNCTION AUGIESS	0	1 70	1.43		1.75	1.72	101		~~~~~	r.
L000	2	leselved											
		ENDPOINTS											
E68A	1	EP0BCH ^[13]	Endpoint 0 Byte Count H	(BC15)	(BC14)	(BC13)	(BC12)	(BC11)	(BC10)	(BC9)	(BC8)	XXXXXXXX	RW
E68B	1	EP0BCL ^[13]	Endpoint 0 Byte Count L	(BC7)	BC6	BC5	BC4	BC3	BC2	BC1	BC0	XXXXXXXX	RW
E68C	1	reserved											
E68D	1	EP1OUTBC	Endpoint 1 OUT Byte Count	0	BC6	BC5	BC4	BC3	BC2	BC1	BC0	0xxxxxx	RW
E68E	1	reserved											
E68F	1	EP1INBC	Endpoint 1 IN Byte Count	0	BC6	BC5	BC4	BC3	BC2	BC1	BC0	0xxxxxxx	RW
E690	1	EP2BCH ^[13]	Endpoint 2 Byte Count H	0	0	0	0	0	BC10	BC9	BC8	00000xxx	RW
E691	1	EP2BCL ^[13]	Endpoint 2 Byte Count L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	XXXXXXXX	RW
E692	2	reserved											
E694	1	EP4BCH ^[13]	Endpoint 4 Byte Count H	0	0	0	0	0	0	BC9	BC8	000000xx	RW
E695	1	EP4BCL ^[13]	Endpoint 4 Byte Count L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxx	RW
E696	2	reserved	, , ,									<u> </u>	
E698	1	EP6BCH ^[13]	Endpoint 6 Byte Count H	0	0	0	0	0	BC10	BC9	BC8	00000xxx	RW
E699	1	EP6BCL ^[13]	Endpoint 6 Byte Count L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	XXXXXXXX	RW
E69A	2	reserved	, , ,									<u> </u>	
E69C	1	EP8BCH ^[13]	Endpoint 8 Byte Count H	0	0	0	0	0	0	BC9	BC8	000000xx	RW
E69D	1	EP8BCL ^[13]	Endpoint 8 Byte Count I	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	XXXXXXXX	RW
E69E	2	reserved	, , ,									<u> </u>	



Absolute Maximum Ratings

Exceeding maximum ratings may shorten the useful life of the device. User guidelines are not tested.

Storage temperature
Ambient temperature with power supplied (commercial)0 °C to +70 °C
Ambient temperature with power supplied (industrial)40 °C to + 105 °C
Supply voltage to ground potential–0.5 V to +4.0 V
DC input voltage to any input pin ^[17] 5.25 V
DC voltage applied to outputs in high Z state –0.5 V to V_{CC} + 0.5 V
Power dissipation
Static discharge voltage>2000 V
Static discharge voltage>2000 V Max output current, per I/O port

Operating Conditions

T _A (ambient temperature under bias) Commercial	0 °C to +70 °C
T _A (ambient temperature under bias) Industrial	–40 °C to +105 °C
Supply voltage	+3.00 V to +3.60 V
Ground voltage	0 V
F _{OSC} (oscillator or crystal frequency)	24 MHz ± 100 ppm, parallel resonant

Thermal Characteristics

The following table displays the thermal characteristics of various packages:

Table 13. Thermal Characteristics

Package	Ambient Temperature (°C)	θJc Junction to Case Thermal Resistance (°C/W)	θJa Junction to Ambient Thermal Resistance (°C/W)
56 SSOP	70	24.4	47.7
100 TQFP	70	11.9	45.9
128 TQFP	70	15.5	43.2
56 QFN	70	10.6	25.2
56 VFBGA	70	30.9	58.6

The junction temperature θ_j , can be calculated using the following equation: θ_j = P* θ_{Ja} + θ_a Where,

P = Power

 θ_{Ja} = Junction to ambient temperature ($\theta_{Jc} + \theta_{Ca}$)

 θ_a = Ambient temperature (70 °C)

The case temperature θ_c , can be calculated using the following equation: θ_c = P* θ_{Ca} + θ_a where,

P = Power

 θ_{Ca} = Case to ambient temperature

 θ_a = Ambient temperature (70 °C)



DC Characteristics

Table 14. DC Characteristics

Parameter	Description	Conditions	Min	Тур	Max	Unit
VCC	Supply voltage	-	3.00	3.3	3.60	V
VCC Ramp Up	0 to 3.3 V	-	200	_	-	μS
V _{IH}	Input HIGH voltage	-	2	_	5.25	V
V _{IL}	Input LOW voltage	-	-0.5	-	0.8	V
V _{IH_X}	Crystal input HIGH voltage	-	2	_	5.25	V
V _{IL_X}	Crystal input LOW voltage	-	-0.5	-	0.8	V
I _I	Input leakage current	0< V _{IN} < V _{CC}	-	-	±10	μΑ
V _{OH}	Output voltage HIGH	I _{OUT} = 4 mA	2.4	_	-	V
V _{OL}	Output LOW voltage	I _{OUT} = -4 mA	_	_	0.4	V
I _{ОН}	Output current HIGH	-	-	-	4	mA
I _{OL}	Output current LOW	-	_	_	4	mA
C		Except D+/D-	-	-	10	pF
CIN		D+/D-	_	-	15	pF
	Suspend current	Connected	_	300	380 ^[18]	μΑ
1	CY7C68014/CY7C68016	Disconnected	-	100	150 ^[18]	μΑ
SUSP	Suspend current	Connected	-	0.5	1.2 ^[18]	mA
	CY7C68013/CY7C68015	Disconnected	_	0.3	1.0 ^[18]	mA
	Supply ourrent	8051 running, connected to USB HS	_	50	85	mA
CC		8051 running, connected to USB FS	_	35	65	mA
т	Reset time after valid power	$V_{}$ min = 3.0 V	5.0	—	-	ms
'RESET	Pin reset after powered on		200	_	-	μS

USB Transceiver

USB 2.0 compliant in Full Speed and Hi-Speed modes.



AC Electrical Characteristics

USB Transceiver

USB 2.0 compliant in Full-Speed and Hi-Speed modes.

Program Memory Read





Table 15.	Program	Memory Read	Parameters
-----------	---------	--------------------	------------

Parameter	Description	Min	Тур	Max	Unit	Notes
		-	20.83	_	ns	48 MHz
t _{CL}	1/CLKOUT frequency	_	41.66	_	ns	24 MHz
		-	83.2	_	ns	12 MHz
t _{AV}	Delay from clock to valid address	0	_	10.7	ns	-
t _{STBL}	Clock to PSEN LOW	0	_	8	ns	_
t _{STBH}	Clock to PSEN HIGH	0	_	8	ns	-
t _{SOEL}	Clock to OE LOW	_	_	11.1	ns	-
t _{SCSL}	Clock to CS LOW	-	-	13	ns	_
t _{DSU}	Data setup to clock	9.6	_	_	ns	-
t _{DH}	Data hold time	0	_	_	ns	_

Notes

19. CLKOUT is shown with positive polarity.

20. t_{ACC1} is computed from these parameters as follows: t_{ACC1}(24 MHz) = $3^{*}t_{CL} - t_{AV} - t_{DSU} = 106$ ns. t_{ACC1}(48 MHz) = $3^{*}t_{CL} - t_{AV} - t_{DSU} = 43$ ns.



GPIF Synchronous Signals



Table 18. GPIF Synchronous Signals Parameters with Internally Sourced $\mbox{IFCLK}^{[24,\ 25]}$

Parameter	Description	Min	Max	Ту	Unit	
i arameter	Description		Max	Min	Max	Onne
t _{IFCLK}	IFCLK Period	20.83	-	-	-	ns
t _{SRY}	RDY _X to clock setup time	8.9	-	-	-	ns
t _{RYH}	Clock to RDY _X	0	-	-	-	ns
t _{SGD}	GPIF data to clock setup time	9.2	-	-	-	ns
t _{DAH}	GPIF data hold time	0	-	-	-	ns
t _{SGA}	Clock to GPIF address propagation delay	-	7.5	-	-	ns
t _{XGD}	Clock to GPIF data output propagation delay	-	11	-	-	ns
t _{XCTL}	Clock to CTL _X output propagation delay	-	6.7	-	-	ns
t _{IFCLKR}	IFCLK rise time	-	-	-	900	ps
t _{IFCLKF}	IFCLK fall time	-	-	-	900	ps
t _{IFCLKOD}	IFCLK output duty cycle	-	_	49	51	%
t _{IFCLKJ}	IFCLK jitter peak to peak	-	_	-	300	ps

Table 19. GPIF Synchronous Signals Parameters with Externally Sourced IFCLK^[25]

Parameter	Description	Min	Max	Unit
t _{IFCLK}	IFCLK period ^[26]	20.83	200	ns
t _{SRY}	RDY _X to clock setup time	2.9	_	ns
t _{RYH}	Clock to RDY _X	3.7	_	ns
t _{SGD}	GPIF data to clock setup time	3.2	_	ns
t _{DAH}	GPIF data hold time	4.5	_	ns
t _{SGA}	Clock to GPIF address propagation delay	-	11.5	ns
t _{XGD}	Clock to GPIF data output propagation delay	-	15	ns
t _{XCTL}	Clock to CTL _X output propagation delay	-	10.7	ns

Notes

24. Dashed lines denote signals with programmable polarity.
 25. GPIF asynchronous RDY_x signals have a minimum setup time of 50 ns when using the internal 48-MHz IFCLK.
 26. IFCLK must not exceed 48 MHz.





Figure 24. Slave FIFO Synchronous Write Sequence and Timing Diagram^[24]

Slave FIFO Asynchronous Packet End Strobe

Figure 25. Slave FIFO Asynchronous Packet End Strobe Timing Diagram^[24]



Table 28. Slave FIFO Asynchronous Packet End Strobe Parameters^[27]

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

Slave FIFO Output Enable





Table 29. 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



Slave FIFO Address to Flags/Data



Table 30. 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

Slave FIFO Synchronous Address

Figure 28. Slave FIFO Synchronous Address Timing Diagram^[24]



Table 31. Slave FIFO Synchronous Address Parameters^[25]

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

Slave FIFO Asynchronous Address

Figure 29. Slave FIFO Asynchronous Address Timing Diagram^[24]



Table 32. Slave FIFO Asynchronous Address Parameters^[27]

Parameter	Description	Min	Max	Unit
t _{SFA}	FIFOADR[1:0] to SLRD/SLWR/PKTEND setup time	10	-	ns
t _{FAH}	RD/WR/PKTEND to FIFOADR[1:0] hold time	10	_	ns



Sequence Diagram

Single and Burst Synchronous Read Example







	∱ IFCLK		∱ IFCLK		∱ IFCLK		IFCLK	1	IFCLK	ſ	IFCLK	∱ IFCLK	∱ IFCL	K	FIFCLK		∱ IFCLK
FIFO POINTER	N		Ν	•	N+1		N+1		N+1		N+2	▶ N+3	► N+4		N+4		N+4
		SLOE	:	SLRD♥		SLOE 🕈 SLRD 🕈		SLOE 🕴	SL	RD			5	SLRD 🕇	S	SLOE 🕇	
FIFO DATA BUS	Not Driven		Driven: N		N+1	•	Not Driven		N+1		N+2	► N+3	► N+4		N+4		Not Driven

Figure 30 on page 52 shows the timing relationship of the SLAVE FIFO signals during a synchronous FIFO read using IFCLK as the synchronizing clock. The diagram illustrates a single read followed by a burst read.

- At t = 0, the FIFO address is stable and the signal SLCS is asserted (SLCS may be tied LOW in some applications). Note that t_{SFA} has a minimum of 25 ns. This means that when IFCLK is running at 48 MHz, the FIFO address setup time is more than one IFCLK cycle.
- At t = 1, SLOE is asserted. SLOE is an output enable only, whose sole function is to drive the data bus. The data that is driven on the bus is the data that the internal FIFO pointer is currently pointing to. In this example it is the first data value in the FIFO. Note: the data is prefetched and is driven on the bus when SLOE is asserted.
- At t = 2, SLRD is asserted. SLRD must meet the setup time of t_{SRD} (time from asserting the SLRD signal to the rising edge of the IFCLK) and maintain a minimum hold time of t_{RDH} (time from the IFCLK edge to the deassertion of the SLRD signal).

If the SLCS signal is used, it must be asserted before SLRD is asserted (The SLCS and SLRD signals must both be asserted to start a valid read condition).

The FIFO pointer is updated on the rising edge of the IFCLK, while SLRD is asserted. This starts the propagation of data from the newly addressed location to the data bus. After a propagation delay of t_{XFD} (measured from the rising edge of IFCLK) the new data value is present. N is the first data value read from the FIFO. To have data on the FIFO data bus, SLOE MUST also be asserted.

The same sequence of events are shown for a burst read and are marked with the time indicators of T = 0 through 5.

Note For the burst mode, the SLRD and SLOE are left asserted during the entire duration of the read. In the burst read mode, when SLOE is asserted, data indexed by the FIFO pointer is on the data bus. During the first read cycle, on the rising edge of the clock, the FIFO pointer is updated and incremented to point to address N+1. For each subsequent rising edge of IFCLK, while the SLRD is asserted, the FIFO pointer is incremented and the next data value is placed on the data bus.



Single and Burst Synchronous Write



Figure 32. Slave FIFO Synchronous Write Sequence and Timing Diagram^[24]

Figure 32 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 three bytes and committing all four 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 that 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 the SLCS signal is used, it must be asserted with SLWR or before SLWR is asserted (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 is also 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, after 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 32, after the four bytes are written to the FIFO, SLWR is deasserted. The short 4 byte packet can be committed to the host by asserting the PKTEND signal.

There is no specific timing requirement that should 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 requirement is that the setup time t_{SPE} and the hold time t_{PEH} must be met. In the scenario of Figure 32, 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 also 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 the PKTEND assertion, there is a specific corner-case condition that needs attention while using the PKTEND to commit a one byte/word packet. Additional timing requirements exist 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 the AUTOINLEN register) committed automatically followed by a short one byte or word packet committed manually using the PKTEND pin.

In this case, the external master must ensure to assert the PKTEND pin at least one clock cycle after the rising edge that caused the last byte or word that needs 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 Figure 24 on page 50 for further details on this timing.



Sequence Diagram of a Single and Burst Asynchronous Read



Figure 33. Slave FIFO Asynchronous Read Sequence and Timing Diagram^[24]

Figure 34. Slave FIFO Asynchronous Read Sequence of Events Diagram



Figure 33 shows the timing relationship of the SLAVE FIFO signals during an asynchronous FIFO read. It shows a single read followed by a burst read.

- At t = 0, the FIFO address is stable and the SLCS signal is asserted.
- At t = 1, SLOE is asserted. This results in the data bus being driven. The data that is driven on to the bus is the previous data, the data that was in the FIFO from an earlier read cycle.
- At t = 2, SLRD is asserted. The SLRD must meet the minimum active pulse of t_{RDpwl} and minimum de-active pulse width of t_{RDpwh}. If SLCS is used, then SLCS must be asserted before SLRD is asserted (The SLCS and SLRD signals must both be asserted to start a valid read condition.)
- The data that is driven, after asserting SLRD, is the updated data from the FIFO. This data is valid after a propagation delay of t_{XFD} from the activating edge of SLRD. In Figure 33, data N is the first valid data read from the FIFO. For data to appear on the data bus during the read cycle (SLRD is asserted), SLOE must be in an asserted state. SLRD and SLOE can also be tied together.

The same sequence of events is also shown for a burst read marked with T = 0 through 5.

Note In the burst read mode, during SLOE is asserted, the data bus is in a driven state and outputs the previous data. After SLRD is asserted, the data from the FIFO is driven on the data bus (SLOE must also be asserted) and then the FIFO pointer is incremented.



Package Diagrams

The FX2LP is available in five packages:

- 56-pin SSOP
- 56-pin QFN
- 100-pin TQFP
- 128-pin TQFP
- 56-ball VFBGA



Figure 36. 56-Pin Shrunk Small Outline Package O56 (51-85062)



PCB Layout Recommendations

Follow these recommendations to ensure reliable high performance operation: $\ensuremath{^{[29]}}$

- Four-layer, impedance-controlled boards are required to maintain signal quality.
- Specify impedance targets (ask your board vendor what they can achieve).
- To control impedance, maintain trace widths and trace spacing.
- Minimize stubs to minimize reflected signals.
- Connections between the USB connector shell and signal ground must be near the USB connector.

- Bypass and flyback caps on VBUS, near connector, are recommended.
- DPLUS and DMINUS trace lengths should be kept to within 2 mm of each other in length, with preferred length of 20 to 30 mm.
- Maintain a solid ground plane under the DPLUS and DMINUS traces. Do not allow the plane to split under these traces.
- Do not place vias on the DPLUS or DMINUS trace routing.
- Isolate the DPLUS and DMINUS traces from all other signal traces by no less than 10 mm.

Note

^{29.} Source for recommendations: *EZ-USB FX2™PCB Design Recommendations*, http://www.cypress.com and *High Speed USB Platform Design Guidelines*, http://www.usb.org/developers/docs/hs_usb_pdg_r1_0.pdf.