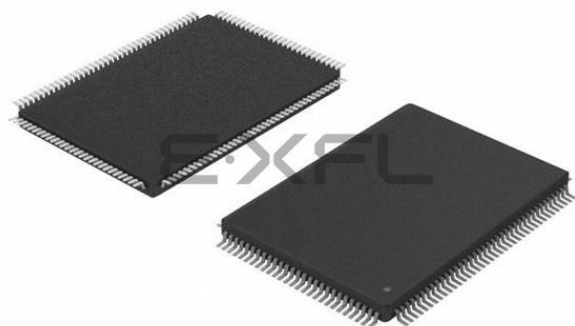




Welcome to [E-XFL.COM](https://www.e-xfl.com)

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	Active
Applications	USB Microcontroller
Core Processor	8051
Program Memory Type	ROMless
Controller Series	CY7C680xx
RAM Size	16K x 8
Interface	I ² C, USB, USART
Number of I/O	40
Voltage - Supply	3V ~ 3.6V
Operating Temperature	-40°C ~ 85°C
Mounting Type	Surface Mount
Package / Case	128-LQFP
Supplier Device Package	128-TQFP (14x20)
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy7c68013a-128axi

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 FX2LP™- 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.

Program/Data RAM

The FX2LP has 16 KB of internal program/data RAM, where PSEN#/RD# signals are internally ORed to enable the 8051 to access it as both program and data memory. No USB control registers appear in this space.

Two memory maps are shown in the following diagrams:

Figure 3 on page 10 shows the Internal Code Memory, EA = 0.

Figure 4 on page 11 shows the External Code Memory, EA = 1.

Internal Code Memory, EA = 0

This mode implements the internal 16 KB block of RAM (starting at 0) as combined code and data memory. When external RAM or ROM is added, the external read and write strobes are suppressed for memory spaces that exist inside the chip. This enables the user to connect a 64 KB memory without requiring address decodes to keep clear of internal memory spaces.

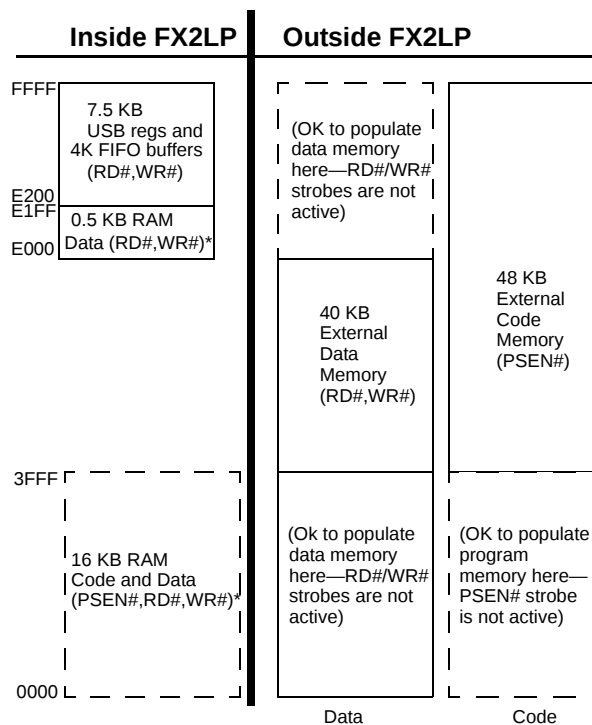
Only the internal 16 KB and scratch pad 0.5 KB RAM spaces have the following access:

- USB download
- USB upload
- Setup data pointer
- I²C interface boot load

External Code Memory, EA = 1

The bottom 16 KB of program memory is external and therefore the bottom 16 KB of internal RAM is accessible only as a data memory.

Figure 3. Internal Code Memory, EA = 0



*SUDPTR, USB upload/download, I²C interface boot access

Note

4. If the external clock is powered at the same time as the CY7C680xxA and has a stabilization wait period, it must be added to the 200 μ s.

Default Full-Speed Alternate Settings

Table 6. Default Full Speed Alternate Settings^[5, 6]

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×)

Default High Speed Alternate Settings

Table 7. Default Hi-Speed Alternate Settings^[5, 6]

Alternate Setting	0	1	2	3
ep0	64	64	64	64
ep1out	0	512 bulk ^[7]	64 int	64 int
ep1in	0	512 bulk ^[7]	64 int	64 int
ep2	0	512 bulk out (2×)	512 int out (2×)	512 iso out (2×)
ep4	0	512 bulk out (2×)	512 bulk out (2×)	512 bulk out (2×)
ep6	0	512 bulk in (2×)	512 int in (2×)	512 iso in (2×)
ep8	0	512 bulk in (2×)	512 bulk in (2×)	512 bulk in (2×)

External FIFO Interface

Architecture

The FX2LP 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).

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 and the slave FIFO interface for externally controlled transfers.

Master/Slave Control Signals

The FX2LP endpoint FIFOs are implemented as eight physically distinct 256×16 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 instantaneously, giving essentially zero transfer time between

“USB FIFOs” and “Slave FIFOs.” Because they are physically the same memory, no bytes are actually transferred between buffers.

At any time, some RAM blocks are filling/emptying with the USB data under SIE control, while other RAM blocks are available to the 8051, the I/O control unit, or both. The RAM blocks operates 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.

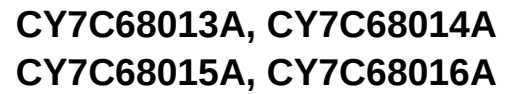
In Master (M) mode, the GPIF internally controls FIFOADR[1..0] to select a FIFO. The RDY pins (two in the 56-pin package, six in the 100-pin and 128-pin packages) can be used as flag inputs from an external FIFO or other logic if desired. The GPIF can be run from either an internally derived clock or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

Notes

5. “0” means “not implemented.”

6. “2×” means “double buffered.”

7. Even though these buffers are 64 bytes, they are reported as 512 for USB 2.0 compliance. The user must never transfer packets larger than 64 bytes to EP1.



Port	GPIF Master	Slave FIFO
0		
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		
16		
17		
18		
19		
20		
21		
22		
23		
24		
25		
26		
27		
28		
29		
30		
31		
32		
33		
34		
35		
36		
37		
38		
39		
40		
41		
42		
43		
44		
45		
46		
47		
48		
49		
50		
51		
52		
53		
54		
55		
56		
57		
58		
59		
60		
61		
62		
63		
64		
65		
66		
67		
68		
69		
70		
71		
72		
73		
74		
75		
76		
77		
78		
79		
80		
81		
82		
83		
84		
85		
86		
87		
88		
89		
90		
91		
92		
93		
94		
95		
96		
97		
98		
99		

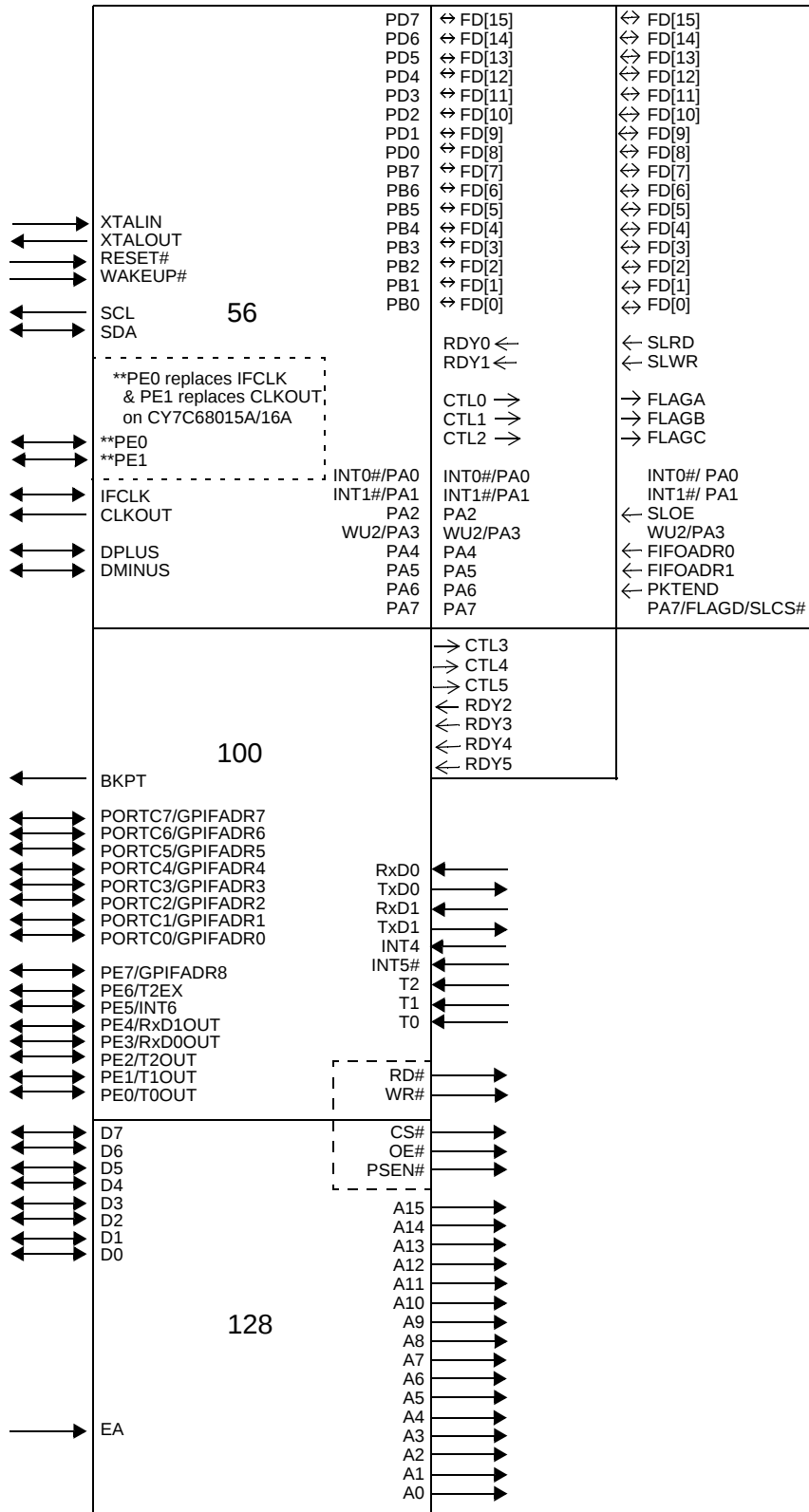
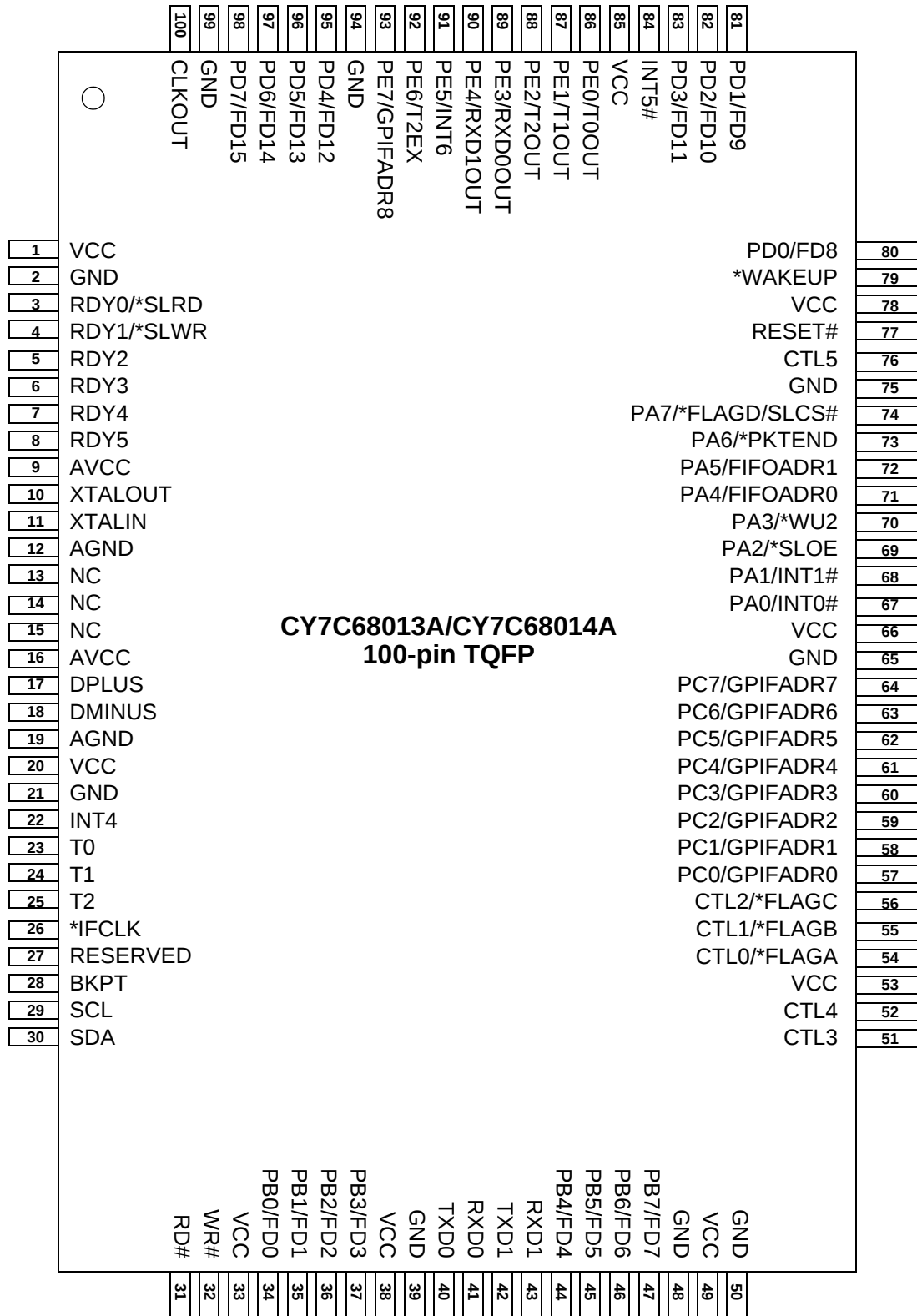
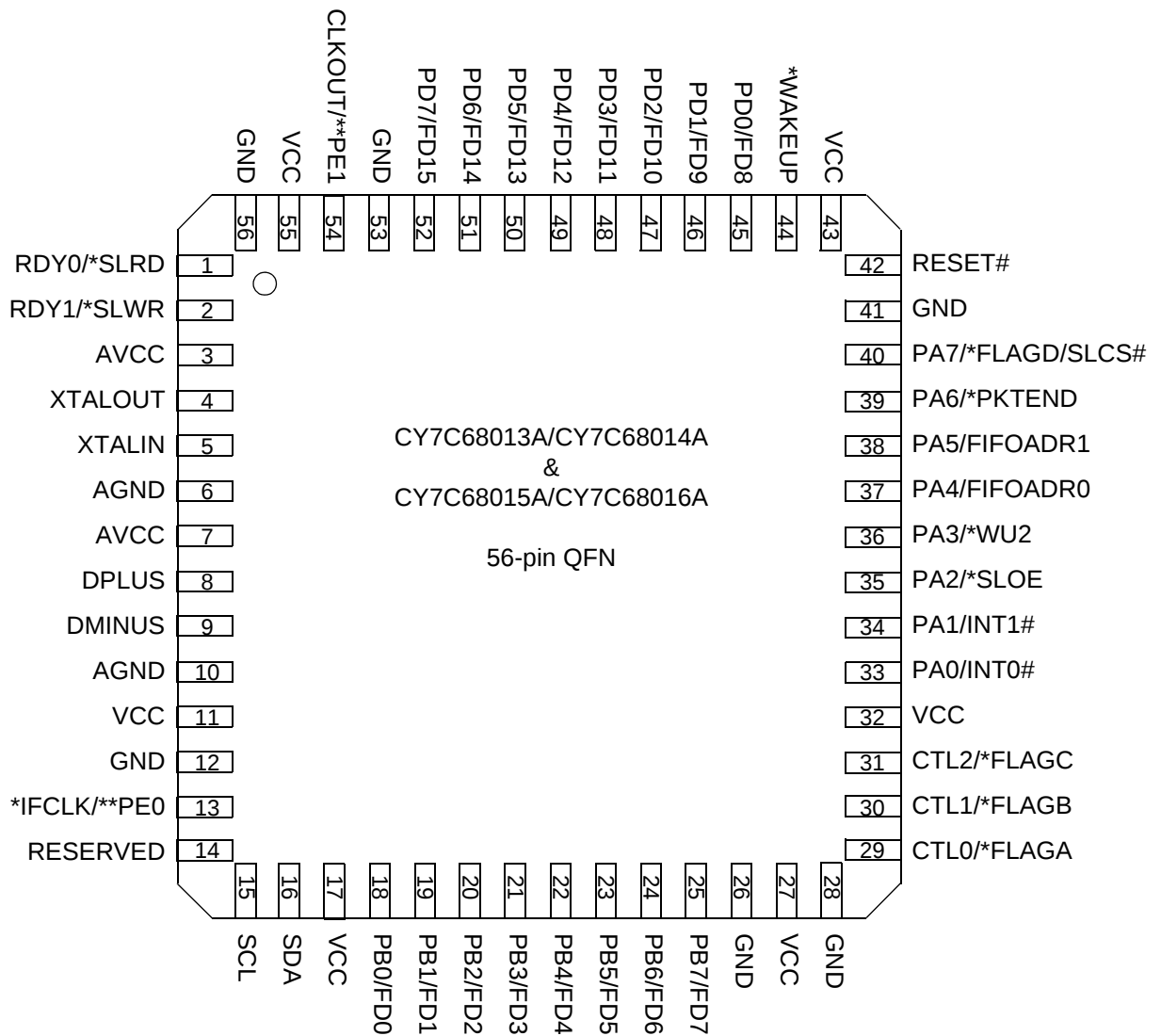


Figure 8. CY7C68013A/CY7C68014A 100-Pin TQFP Pin Assignment



* denotes programmable polarity

Figure 10. CY7C68013A/14A/15A/16A 56-Pin QFN Pin Assignment



* denotes programmable polarity
** denotes CY7C68015A/CY7C68016A pinout

Table 11. FX2LP Pin Descriptions^[11] (continued)

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	Name	Type	Default	Reset ^[12]	Description
34	28	—	—		BKPT	Output	L	L	Breakpoint. This pin goes active (HIGH) when the 8051 address bus matches the BPADDRH/L registers and breakpoints are enabled in the BREAKPT register (BPEN = 1). If the BPPULSE bit in the BREAKPT register is HIGH, this signal pulses HIGH for eight 12-/24-/48-MHz clocks. If the BPPULSE bit is LOW, the signal remains HIGH until the 8051 clears the BREAK bit (by writing 1 to it) in the BREAKPT register.
99	77	49	42	8B	RESET#	Input	N/A	N/A	Active LOW Reset. Resets the entire chip. See section "Reset and Wakeup" on page 9 for more details.
35	—	—	—	—	EA	Input	N/A	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	12	5	1C	XTALIN	Input	N/A	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 3.3-V square wave.
11	10	11	4	2C	XTALOUT	Output	N/A	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	5	54	2B	CLKOUT on CY7C68013A and CY7C68014A ----- PE1 on CY7C68015A and CY7C68016A	O/Z ----- - I/O/Z	12 MHz ----- I	Clock Driven ----- Z	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. ----- PE1 is a bidirectional I/O port pin.
Port A									
82	67	40	33	8G	PA0 or INT0#	I/O/Z	I (PA0)	Z (PA0)	Multiplexed pin whose function is selected by PORTACFG.0 PA0 is a bidirectional I/O 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	41	34	6G	PA1 or INT1#	I/O/Z	I (PA1)	Z (PA1)	Multiplexed pin whose function is selected by: PORTACFG.1 PA1 is a bidirectional I/O port pin. INT1# is the active-LOW 8051 INT1 interrupt input signal, which is either edge-triggered (IT1 = 1) or level-triggered (IT1 = 0).

Table 11. FX2LP Pin Descriptions^[11] (continued)

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	Name	Type	Default	Reset ^[12]	Description
30	24	—	—	—	T1	Input	N/A	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	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	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	L	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	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	L	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	H	CS# is the active LOW chip select for external memory.
41	32	—	—	—	WR#	Output	H	H	WR# is the active LOW write strobe output for external memory.
40	31	—	—	—	RD#	Output	H	H	RD# is the active LOW read strobe output for external memory.
38		—	—	—	OE#	Output	H	H	OE# is the active LOW output enable for external memory.
33	27	21	14	2H	Reserved	Input	N/A	N/A	Reserved. Connect to ground.
101	79	51	44	7B	WAKEUP	Input	N/A	N/A	USB Wakeup. If the 8051 is in suspend, asserting this pin starts up the oscillator and interrupts the 8051 to enable it to exit the suspend mode. Holding WAKEUP asserted inhibits the EZ-USB chip from suspending. This pin has programmable polarity (WAKEUP.4).
36	29	22	15	3F	SCL	OD	Z	Z (if booting is done)	Clock for the I ² C interface. Connect to VCC with a 2.2-kΩ resistor, even if no I ² C peripheral is attached.
37	30	23	16	3G	SDA	OD	Z	Z (if booting is done)	Data for I ² C compatible interface. Connect to VCC with a 2.2-kΩ resistor, even if no I²C compatible peripheral is attached.
2	1	6	55	5A	VCC	Power	N/A	N/A	VCC. Connect to the 3.3-V power source.
26	20	18	11	1G	VCC	Power	N/A	N/A	VCC. Connect to the 3.3-V power source.
43	33	24	17	7E	VCC	Power	N/A	N/A	VCC. Connect to the 3.3-V power source.
48	38	—	—	—	VCC	Power	N/A	N/A	VCC. Connect to 3.3-V power source.
64	49	34	27	8E	VCC	Power	N/A	N/A	VCC. Connect to the 3.3-V power source.
68	53	—	—	—	VCC	Power	N/A	N/A	VCC. Connect to the 3.3-V power source.
81	66	39	32	5C	VCC	Power	N/A	N/A	VCC. Connect to the 3.3-V power source.

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 Memories													
E400	128	WAVEDATA	GPIF Waveform Descriptor 0, 1, 2, 3 data	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
E480	128	reserved											
GENERAL CONFIGURATION													
E50D		GPCR2	General Purpose Configuration Register 2	reserved	reserved	reserved	FULL_SPEED_ONLY	reserved	reserved	reserved	reserved	00000000	R
E600	1	CPUCS	CPU Control & Status	0	0	PORTCSTB	CLKSPD1	CLKSPD0	CLKINV	CLKOE	8051RES	00000010	rbbbbbr
E601	1	IFCONFIG	Interface Configuration (Ports, GPIF, slave FIFOs)	IFCLKSRC	3048MHZ	IFCLKOE	IFCLKPOL	ASYN	GSTATE	IFCFG1	IFCFG0	10000000	RW
E602	1	PINFLAGSA ^[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	xxxxxxx	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	rrrrrb
E609	1	FIFOPINPOLAR ^[13]	Slave FIFO Interface pins polarity	0	0	PKTEND	SLOE	SLRD	SLWR	EF	FF	00000000	rbbbbbbb
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	rrrrrb
UDMA													
E60C	1	GPIFHOLDAMOUNT	MSTB Hold Time (for UDMA)	0	0	0	0	0	0	HOLDTIME1	HOLDTIME0	00000000	rrrrrb
	3	reserved											
ENDPOINT CONFIGURATION													
E610	1	EP1OUTCFG	Endpoint 1-OUT Configuration	VALID	0	TYPE1	TYPE0	0	0	0	0	10100000	rbbrrrr
E611	1	EP1INCFG	Endpoint 1-IN Configuration	VALID	0	TYPE1	TYPE0	0	0	0	0	10100000	rbbrrrr
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	bbbbrrr
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	bbbbrrr
	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	rrrrrb
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	rrrrrb
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	rrrrrb
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	rrrrrb
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	x	x	x	x	x	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	EP0ACK	HSGRANT	URES	SUSP	SUTOK	SOF	SUDAV	0xxxxxxx	rrrrrrrr
E65E	1	EPIE	Endpoint Interrupt Enables	EP8	EP6	EP4	EP2	EP1OUT	EP1IN	EP0OUT	EP0IN	00000000	RW
E65F	1	EPIRQ ^[14]	Endpoint Interrupt Requests	EP8	EP6	EP4	EP2	EP1OUT	EP1IN	EP0OUT	EP0IN	0	RW
E660	1	GPIFIE ^[13]	GPIF Interrupt Enable	0	0	0	0	0	0	GPIFWF	GPIFDONE	00000000	RW
E661	1	GPIFIQ ^[13]	GPIF Interrupt Request	0	0	0	0	0	0	GPIFWF	GPIFDONE	000000xx	RW
E662	1	USBERRIE	USB Error Interrupt Enables	ISOEP8	ISOEP6	ISOEP4	ISOEP2	0	0	0	ERRLIMIT	00000000	RW
E663	1	USBERRIRQ ^[14]	USB Error Interrupt Requests	ISOEP8	ISOEP6	ISOEP4	ISOEP2	0	0	0	ERRLIMIT	0000000x	bbbbrrrr
E664	1	ERRCNTLIM	USB Error counter and limit	EC3	EC2	EC1	EC0	LIMIT3	LIMIT2	LIMIT1	LIMIT0	xxxx0100	rrrrrrrr
E665	1	CLRERRCNT	Clear Error Counter EC3:0	x	x	x	x	x	x	x	x	xxxxxxx	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	I4V3	I4V2	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	PORTTECFG	I/O PORTE Alternate Configuration	GPIFA8	T2EX	INT6	RXD1OUT	RXD0OUT	T2OUT	T1OUT	T0OUT	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	bbbbrrrr
E679	1	I2DAT	I ² C 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	Autopr1 MOVX access, when APTREN=1	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
E67C	1	XAUTODAT2	Autopr2 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	QSIGNA2	QSIGNA1	QSIGNA0	00000000	bbbbrrrr
USB CONTROL													
E680	1	USBCS	USB Control & Status	HSM	0	0	0	DISCON	NOSYNOSF	RENUM	SIGRSUME	x0000000	rrrrrrrr
E681	1	SUSPEND	Put chip into suspend	x	x	x	x	x	x	x	x	xxxxxxx	W
E682	1	WAKEUPCS	Wakeup Control & Status	WU2	WU	WU2POL	WUPOL	0	DPEN	WU2EN	WUEN	xx000101	bbbbrrrr
E683	1	TOGCTL	Toggle Control	Q	S	R	I/O	EP3	EP2	EP1	EP0	x0000000	rrrrrrrr
E684	1	USBFRAMEH	USB Frame count H	0	0	0	0	0	FC10	FC9	FC8	00000xxx	R
E685	1	USBFRAMEL	USB Frame count L	FC7	FC6	FC5	FC4	FC3	FC2	FC1	FC0	xxxxxxx	R
E686	1	MICROFRAME	Microframe count, 0-7	0	0	0	0	0	MF2	MF1	MF0	00000xxx	R
E687	1	FNADDR	USB Function address	0	FA6	FA5	FA4	FA3	FA2	FA1	FA0	0xxxxxxx	R
E688	2	reserved											
ENDPOINTS													
E68A	1	EP0BCH ^[13]	Endpoint 0 Byte Count H	(BC15)	(BC14)	(BC13)	(BC12)	(BC11)	(BC10)	(BC9)	(BC8)	xxxxxxx	RW
E68B	1	EP0BCL ^[13]	Endpoint 0 Byte Count L	(BC7)	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxx	RW
E68C	1	reserved											
E68D	1	EP1OUTBC	Endpoint 1 OUT Byte Count	0	BC6	BC5	BC4	BC3	BC2	BC1	BC0	0xxxxxxx	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	xxxxxxx	RW
E692	2	reserved											
E694	1	EP4BCH ^[13]	Endpoint 4 Byte Count H	0	0	0	0	0	0	BC9	BC8	00000xxx	RW
E695	1	EP4BCL ^[13]	Endpoint 4 Byte Count L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxx	RW
E696	2	reserved											
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	xxxxxxx	RW
E69A	2	reserved											
E69C	1	EP8BCH ^[13]	Endpoint 8 Byte Count H	0	0	0	0	0	0	BC9	BC8	00000xxx	RW
E69D	1	EP8BCL ^[13]	Endpoint 8 Byte Count L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxx	RW
E69E	2	reserved											

Table 12. FX2LP Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
E6A0	1	EP0CS	Endpoint 0 Control and Status	HSNAK	0	0	0	0	0	BUSY	STALL	10000000	bbbbbbbrb
E6A1	1	EP1OUTCS	Endpoint 1 OUT Control and Status	0	0	0	0	0	0	BUSY	STALL	00000000	bbbbbbbrb
E6A2	1	EP1INCS	Endpoint 1 IN Control and Status	0	0	0	0	0	0	BUSY	STALL	00000000	bbbbbbbrb
E6A3	1	EP2CS	Endpoint 2 Control and Status	0	NPAK2	NPAK1	NPAK0	FULL	EMPTY	0	STALL	00101000	rrrrrrrb
E6A4	1	EP4CS	Endpoint 4 Control and Status	0	0	NPAK1	NPAK0	FULL	EMPTY	0	STALL	00101000	rrrrrrrb
E6A5	1	EP6CS	Endpoint 6 Control and Status	0	NPAK2	NPAK1	NPAK0	FULL	EMPTY	0	STALL	00000100	rrrrrrrb
E6A6	1	EP8CS	Endpoint 8 Control and Status	0	0	NPAK1	NPAK0	FULL	EMPTY	0	STALL	00000100	rrrrrrrb
E6A7	1	EP2FIFOFLGS	Endpoint 2 slave FIFO Flags	0	0	0	0	0	PF	EF	FF	00000010	R
E6A8	1	EP4FIFOFLGS	Endpoint 4 slave FIFO Flags	0	0	0	0	0	PF	EF	FF	00000010	R
E6A9	1	EP6FIFOFLGS	Endpoint 6 slave FIFO Flags	0	0	0	0	0	PF	EF	FF	00000110	R
E6AA	1	EP8FIFOFLGS	Endpoint 8 slave FIFO Flags	0	0	0	0	0	PF	EF	FF	00000110	R
E6AB	1	EP2FIFOBCH	Endpoint 2 slave FIFO total byte count H	0	0	0	BC12	BC11	BC10	BC9	BC8	00000000	R
E6AC	1	EP2FIFOBCL	Endpoint 2 slave FIFO total byte count L	BC7	BC6	BC5	BC4	BC3	BC2	BC1	BC0	00000000	R
E6AD	1	EP4FIFOBCH	Endpoint 4 slave FIFO total byte count H	0	0	0	0	0	BC10	BC9	BC8	00000000	R
E6AE	1	EP4FIFOBCL	Endpoint 4 slave FIFO total byte count L	BC7	BC6	BC5	BC4	BC3	BC2	BC1	BC0	00000000	R
E6AF	1	EP6FIFOBCH	Endpoint 6 slave FIFO total byte count H	0	0	0	0	BC11	BC10	BC9	BC8	00000000	R
E6B0	1	EP6FIFOBCL	Endpoint 6 slave FIFO total byte count L	BC7	BC6	BC5	BC4	BC3	BC2	BC1	BC0	00000000	R
E6B1	1	EP8FIFOBCH	Endpoint 8 slave FIFO total byte count H	0	0	0	0	0	BC10	BC9	BC8	00000000	R
E6B2	1	EP8FIFOBCL	Endpoint 8 slave FIFO total byte count L	BC7	BC6	BC5	BC4	BC3	BC2	BC1	BC0	00000000	R
E6B3	1	SUDPTRH	Setup Data Pointer high address byte	A15	A14	A13	A12	A11	A10	A9	A8	xxxxxxx	RW
E6B4	1	SUDPTL	Setup Data Pointer low address byte	A7	A6	A5	A4	A3	A2	A1	0	xxxxxxx0	bbbbbbbrb
E6B5	1	SUDPTRCTL	Setup Data Pointer Auto Mode	0	0	0	0	0	0	0	SDPAUTO	00000001	RW
	2	reserved											
E6B8	8	SET-UPDAT	8 bytes of setup data	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	R
			SET-UPDAT[0] = bmRequestType										
			SET-UPDAT[1] = bmRequest										
			SET-UPDAT[2:3] = wValue										
			SET-UPDAT[4:5] = wIndex										
			SET-UPDAT[6:7] = wLength										
		GPIF											
E6C0	1	GPIFWFSELECT	Waveform Selector	SINGLEWR1	SINGLEWR0	SINGLERD1	SINGLERD0	FIFOWR1	FIFOWR0	FIFORD1	FIFORD0	11100100	RW
E6C1	1	GPIFIDLECS	GPIF Done, GPIF IDLE drive mode	DONE	0	0	0	0	0	0	IDLEDRV	10000000	RW
E6C2	1	GPIFIDLECTL	Inactive Bus, CTL states	0	0	CTL5	CTL4	CTL3	CTL2	CTL1	CTL0	11111111	RW
E6C3	1	GPIFCTLCFG	CTL Drive Type	TRICTL	0	CTL5	CTL4	CTL3	CTL2	CTL1	CTL0	00000000	RW
E6C4	1	GPIFADRH ^[13]	GPIF Address H	0	0	0	0	0	0	0	GPIFA8	00000000	RW
E6C5	1	GPIFADRL ^[13]	GPIF Address L	GPIFA7	GPIFA6	GPIFA5	GPIFA4	GPIFA3	GPIFA2	GPIFA1	GPIFA0	00000000	RW
		FLOWSTATE											
E6C6	1	FLOWSTATE	Flowstate Enable and Selector	FSE	0	0	0	0	FS2	FS1	FS0	00000000	brrrrrbbb
E6C7	1	FLOWLOGIC	Flowstate Logic	LFUNC1	LFUNC0	TERMA2	TERMA1	TERMA0	TERMB2	TERMB1	TERMB0	00000000	RW
E6C8	1	FLOWEQ0CTL	CTL-Pin States in Flowstate (when Logic = 0)	CTL0E3	CTL0E2	CTL0E1/CTL5	CTL0E0/CTL4	CTL3	CTL2	CTL1	CTL0	00000000	RW
E6C9	1	FLOWEQ1CTL	CTL-Pin States in Flowstate (when Logic = 1)	CTL0E3	CTL0E2	CTL0E1/CTL5	CTL0E0/CTL4	CTL3	CTL2	CTL1	CTL0	00000000	RW
E6CA	1	FLOWHOLDOFF	Holdoff Configuration	HOPERIOD3	HOPERIOD2	HOPERIOD1	HOPERIOD0	HOSTATE	HOCTL2	HOCTL1	HOCTL0	00010010	RW
E6CB	1	FLOWSTB	Flowstate Strobe Configuration	SLAVE	RDYASYN	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	rrrrrrbbb
E6CD	1	FLOWSTBPERIOD	Master-Strobe Half-Period	D7	D6	D5	D4	D3	D2	D1	D0	00000010	RW
E6CE	1	GPIFTCB3 ^[13]	GPIF Transaction Count Byte 3	TC31	TC30	TC29	TC28	TC27	TC26	TC25	TC24	00000000	RW

Table 12. FX2LP Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
E6CF	1	GPIFTCB2 ^[13]	GPIF Transaction Count Byte 2	TC23	TC22	TC21	TC20	TC19	TC18	TC17	TC16	00000000	RW
E6D0	1	GPIFTCB1 ^[13]	GPIF Transaction Count Byte 1	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8	00000000	RW
E6D1	1	GPIFTCB0 ^[13]	GPIF Transaction Count Byte 0	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0	00000001	RW
	2	reserved										00000000	RW
		reserved											
		reserved											
E6D2	1	EP2GPIFFLGSEL ^[13]	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 ^[13]	Endpoint 2 GPIF Trigger	x	x	x	x	x	x	x	x	xxxxxxx	W
	3	reserved											
		reserved											
		reserved											
E6DA	1	EP4GPIFFLGSEL ^[13]	Endpoint 4 GPIF Flag select	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6DB	1	EP4GPIFPFSTOP	Endpoint 4 GPIF stop transaction on prog. flag	0	0	0	0	0	0	0	FIFO4FLAG	00000000	RW
E6DC	1	EP4GPIFTRIG ^[13]	Endpoint 4 GPIF Trigger	x	x	x	x	x	x	x	x	xxxxxxx	W
	3	reserved											
		reserved											
		reserved											
E6E2	1	EP6GPIFFLGSEL ^[13]	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 ^[13]	Endpoint 6 GPIF Trigger	x	x	x	x	x	x	x	x	xxxxxxx	W
	3	reserved											
		reserved											
		reserved											
E6EA	1	EP8GPIFFLGSEL ^[13]	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 ^[13]	Endpoint 8 GPIF Trigger	x	x	x	x	x	x	x	x	xxxxxxx	W
	3	reserved											
E6F0	1	XGPIFSGLDATH	GPIF Data H (16-bit mode only)	D15	D14	D13	D12	D11	D10	D9	D8	xxxxxxx	RW
E6F1	1	XGPIFSGLDATLX	Read/Write GPIF Data L & trigger transaction	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
E6F2	1	XGPIFSGLDATLNOX	Read GPIF Data L, no transaction trigger	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	R
E6F3	1	GPIFREADYCFG	Internal RDY, Sync/Async, RDY pin states	INTRDY	SAS	TCXRDY5	0	0	0	0	0	00000000	bbbbrrrr
E6F4	1	GPIFREADYSTAT	GPIF Ready Status	0	0	RDY5	RDY4	RDY3	RDY2	RDY1	RDY0	00xxxxx	R
E6F5	1	GPIFABORT	Abort GPIF Waveforms	x	x	x	x	x	x	x	x	xxxxxxx	W
E6F6	2	reserved											
		ENDPOINT BUFFERS											
E740	64	EP0BUF	EP0-IN/-OUT buffer	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
E780	64	EP10UTBUF	EP1-OUT buffer	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
E7C0	64	EP1INBUF	EP1-IN buffer	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
E800	2048	reserved											RW
F000	1024	EP2FIFOBUF	512/1024 byte EP 2 / slave FIFO buffer (IN or OUT)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
F400	512	EP4FIFOBUF	512 byte EP 4 / slave FIFO buffer (IN or OUT)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
F600	512	reserved											
F800	1024	EP6FIFOBUF	512/1024 byte EP 6 / slave FIFO buffer (IN or OUT)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
FC00	512	EP8FIFOBUF	512 byte EP 8 / slave FIFO buffer (IN or OUT)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
FE00	512	reserved											

Table 12. FX2LP Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
BE	1	GPIFSGLDATLX ^[15]	GPIF Data L w/ Trigger	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
BF	1	GPIFSGLDATL- NOX ^[15]	GPIF Data L w/ No Trigger	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	R
C0	1	SCON1 ^[15]	Serial Port 1 Control (bit addressable)	SM0_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1	00000000	RW
C1	1	SBUF1 ^[15]	Serial Port 1 Data Buffer	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
C2	6	reserved											
C8	1	T2CON	Timer/Counter 2 Control (bit addressable)	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CT2	CPRL2	00000000	RW
C9	1	reserved											
CA	1	RCAP2L	Capture for Timer 2, auto-reload, up-counter	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
CB	1	RCAP2H	Capture for Timer 2, auto-reload, up-counter	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
CC	1	TL2	Timer 2 reload L	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
CD	1	TH2	Timer 2 reload H	D15	D14	D13	D12	D11	D10	D9	D8	00000000	RW
CE	2	reserved											
D0	1	PSW	Program Status Word (bit addressable)	CY	AC	F0	RS1	RS0	OV	F1	P	00000000	RW
D1	7	reserved											
D8	1	EICON ^[15]	External Interrupt Control	SMOD1	1	ERESI	RESI	INT6	0	0	0	01000000	RW
D9	7	reserved											
E0	1	ACC	Accumulator (bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
E1	7	reserved											
E8	1	EIE ^[15]	External Interrupt Enable(s)	1	1	1	EX6	EX5	EX4	E ² C	EUSB	11100000	RW
E9	7	reserved											
F0	1	B	B (bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
F1	7	reserved											
F8	1	EIP ^[15]	External Interrupt Priority Control	1	1	1	PX6	PX5	PX4	P ² C	PUSB	11100000	RW
F9	7	reserved											

R = all bits read-only
W = all bits write-only
r = read-only bit
w = write-only bit
b = both read/write bit

Sequence Diagram

Single and Burst Synchronous Read Example

Figure 30. Slave FIFO Synchronous Read Sequence and Timing Diagram^[24]

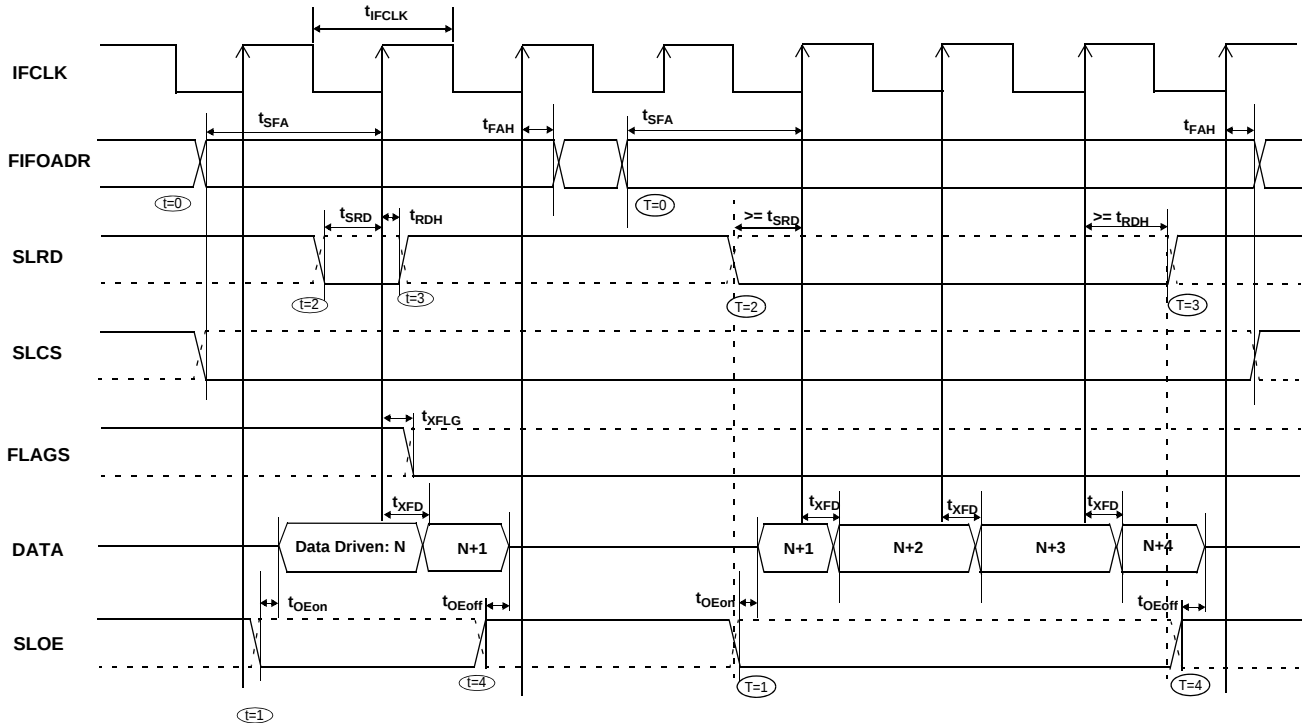


Figure 31. Slave FIFO Synchronous Sequence of Events Diagram

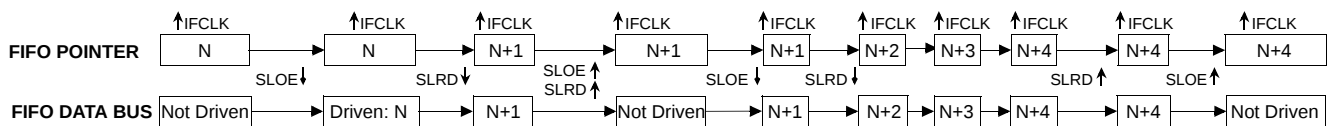


Figure 30 on page 52 shows the timing relationship of the SLAVE FIFO signals during a synchronous FIFO read using IFCLK as the synchronizing clock. The diagram illustrates a single read followed by a burst read.

- 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]

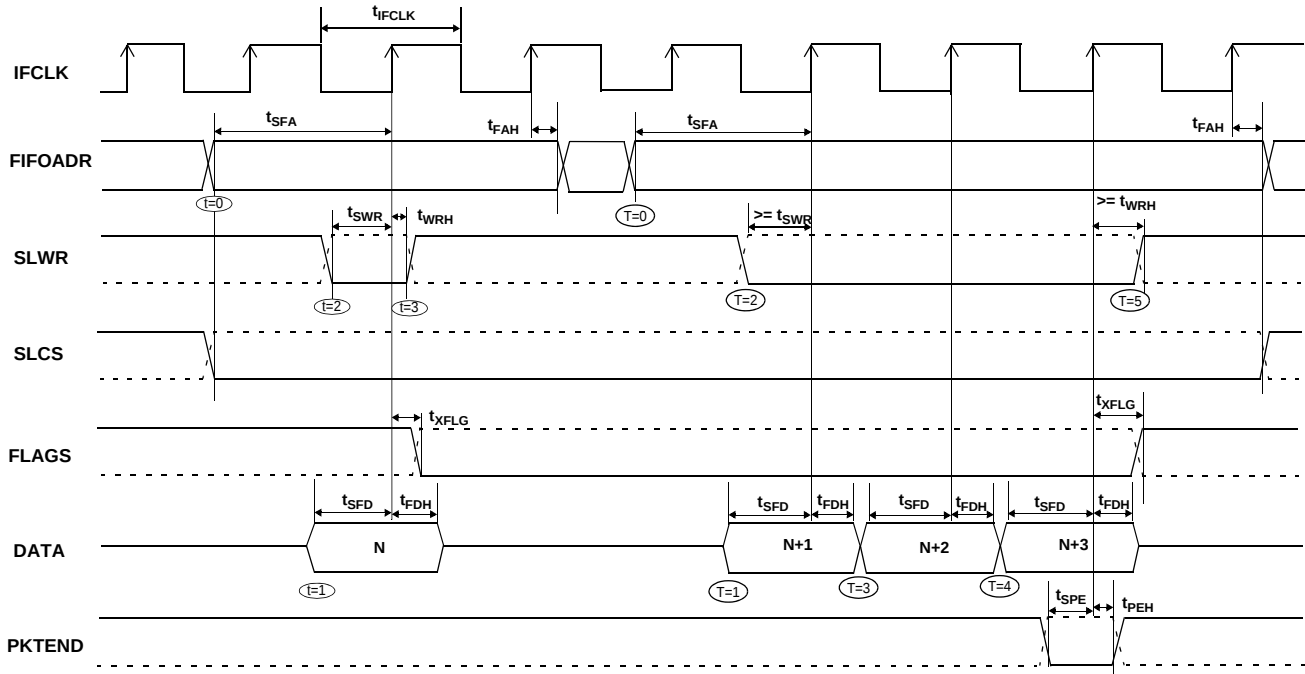


Figure 32 shows the timing relationship of the SLAVE FIFO signals during a synchronous write using IFCLK as the synchronizing clock. The diagram illustrates a single write followed by burst write of three bytes and committing all four bytes as a short packet using the PKTEND pin.

- 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 output 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 pin.

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.

Figure 39. 128-Pin Thin Plastic Quad Flatpack (14 × 20 × 1.4 mm) A128 (51-85101)

128 Lead Thin Plastic Quad Flatpack 14 X 20 X 1.4mm – A128

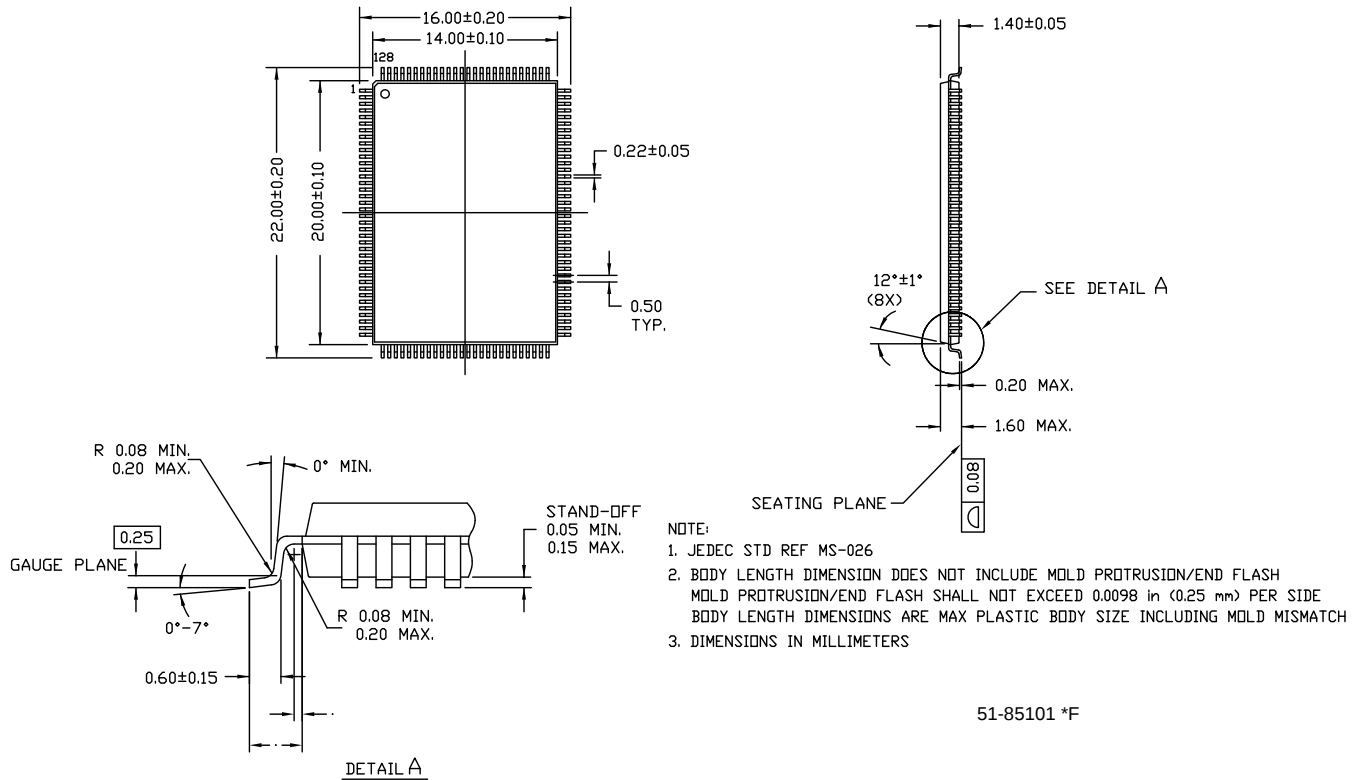
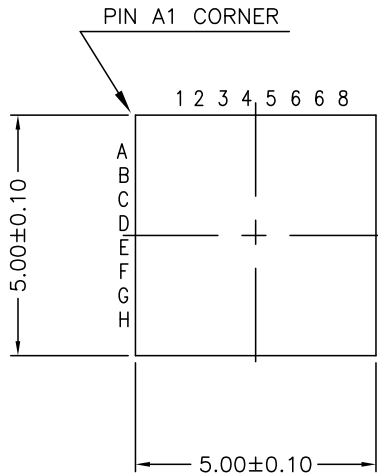
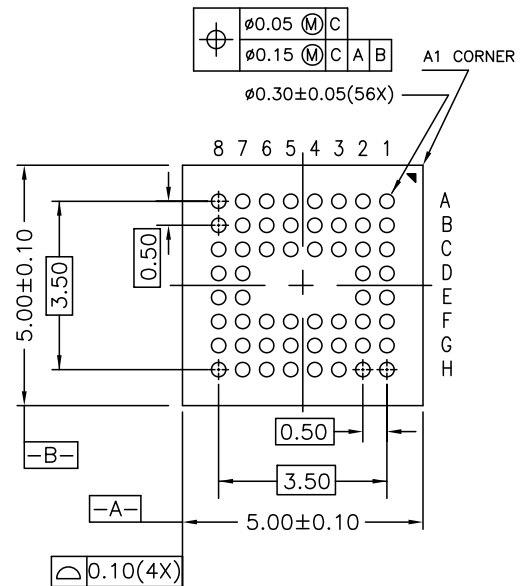
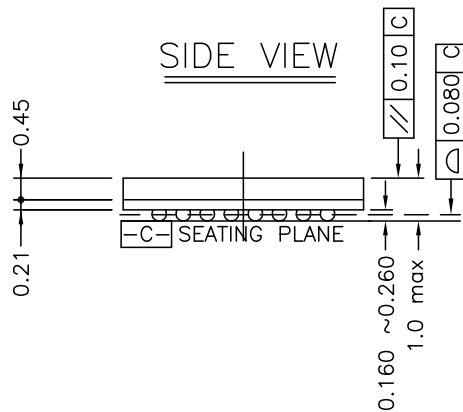


Figure 40. 56-Pin VFBGA (5 × 5 × 1.0 mm) 0.50 Pitch, 0.30 Ball BZ56 (001-03901)

TOP VIEW



SIDE VIEW



BOTTOM VIEW

REFERENCE JEDEC: MO-195C
PACKAGE WEIGHT: 0.02 grams

001-03901 *F

PCB Layout Recommendations

Follow these recommendations to ensure reliable high performance operation:^[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.

Errata

This section describes the errata for the EZ-USB® FX2LP™ CY7C68013A/14A/15A/16A Rev. B silicon. Details include errata trigger conditions, scope of impact, available workaround, and silicon revision applicability.

Contact your local Cypress Sales Representative if you have questions.

Part Numbers Affected

Part Number	Package Type	Operating Range
CY7C68013A	All	Commercial
CY7C68014A	All	Commercial
CY7C68015A	All	Commercial
CY7C68016A	All	Commercial

CY7C68013A/14A/15A/16A Qualification Status

In production

CY7C68013A/14A/15A/16A Errata Summary

This table defines the errata for available CY7C68013A/14A/15A/16A family devices. An "X" indicates that the errata pertain to the selected device.

Items	CY7C68013A/14A/15A/16A	Silicon Revision	Fix Status
[1.]. Empty Flag Assertion	X	B	No silicon fix planned currently. Use the workaround.

1. Empty Flag Assertion

■ Problem Definition

In Slave FIFO Asynchronous Word Wide mode, if a single word data is transferred from the USB host to EP2, configured as OUT Endpoint (EP) in the first transaction, then the Empty flag behaves incorrectly. This does not happen if the data size is more than one word in the first transaction.

■ Parameters Affected

NA

■ Trigger Condition(S)

In Slave FIFO Asynchronous Word Wide Mode, after firmware boot and initialization, EP2 OUT endpoint empty flag indicates the status as 'Empty'. When data is received in EP2, the status changes to 'Not-Empty'. However, if data transferred to EP2 is a single word, then asserting SLRD with FIFOADR pointing to any other endpoint changes 'Not-Empty' status to 'Empty' for EP2 even though there is a word data (or it is untouched). This is noticed only when the single word is sent as the first transaction and not if it follows a multi-word packet as the first transaction.

■ Scope of Impact

External interface does not see data available in EP2 OUT endpoint and can end up waiting for data to be read.

■ Workaround

One of the following workarounds can be used:

- Send a pulse signal to the SLWR pin, with FIFOADR pins pointing to an endpoint other than EP2, after firmware initialization and before or after transferring the data to EP2 from the host
- Set the length of the first data to EP2 to be more than a word
- Prioritize EP2 read from the Master for multiple OUT EPs and single word write to EP2
- Write to an IN EP, if any, from the Master before reading from other OUT EPs (other than EP2) from the Master.

■ Fix Status

There is no silicon fix planned for this currently; use the workarounds provided.

Sales, Solutions, and Legal Information

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at [Cypress Locations](#).

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Lighting & Power Control	cypress.com/powerpsoc
Memory	cypress.com/memory
PSoC	cypress.com/psoc
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless/RF	cypress.com/wireless

PSoC® Solutions

[PSoC 1](#) | [PSoC 3](#) | [PSoC 4](#) | [PSoC 5LP](#)

Cypress Developer Community

[Forums](#) | [Projects](#) | [Video](#) | [Blogs](#) | [Training](#) | [Components](#)

Technical Support

cypress.com/support

© Cypress Semiconductor Corporation, 2003-2016. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.