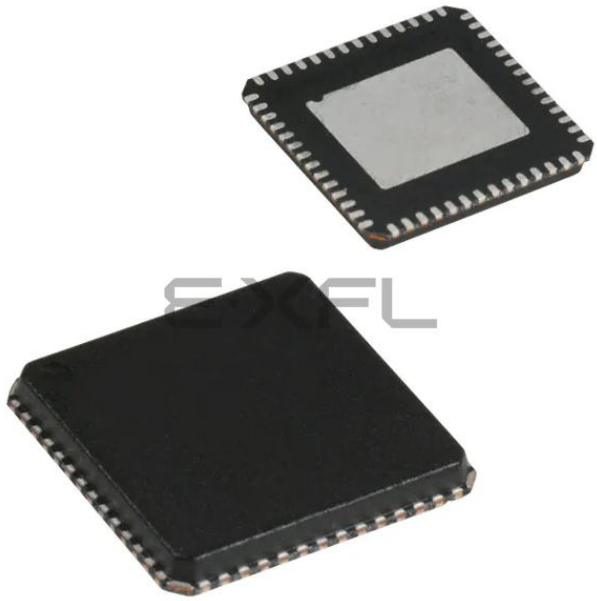


Welcome to E-XFL.COM

Embedded - Microcontrollers - Application Specific: Tailored Solutions for Precision and Performance

Embedded - Microcontrollers - Application Specific represents a category of microcontrollers designed with unique features and capabilities tailored to specific application needs. Unlike general-purpose microcontrollers, application-specific microcontrollers are optimized for particular tasks, offering enhanced performance, efficiency, and functionality to meet the demands of specialized applications.

What Are Embedded - Microcontrollers - Application Specific?

Application-specific microcontrollers are engineered to

Details	
Product Status	Obsolete
Applications	USB Microcontroller
Core Processor	8051
Program Memory Type	ROMless
Controller Series	CY7C680xx
RAM Size	16K x 8
Interface	I²C, USB, USART
Number of I/O	24
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/cy7c68013a-56lfxc

Contents

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.

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.

I²C Port Pins

The I²C pins SCL and SDA must have external 2.2-kΩ 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

I²C Interface Boot Load Access

At power-on reset, the I²C 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. I²C interface boot loads only occur after power-on reset.

I²C Interface General-Purpose Access

The 8051 can control peripherals connected to the I²C bus using the I2CTL and I2DAT registers. FX2LP provides I²C master control only; it is never an I²C 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 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-56PVXCT or 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

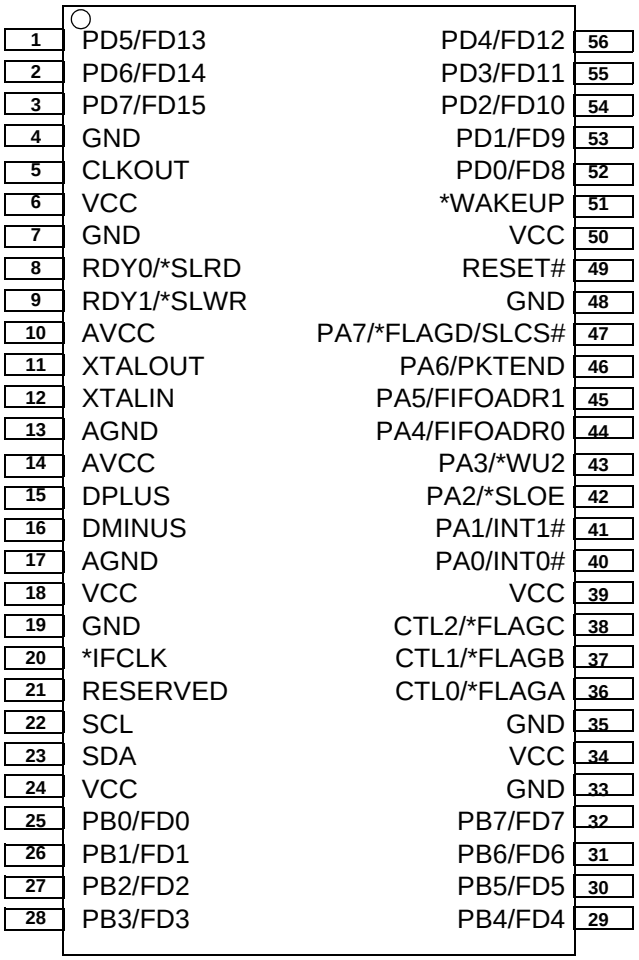
Table 10. CY7C68013A/14A and CY7C68015A/16A Pin Differences

CY7C68013A/CY7C68014A	CY7C68015A/CY7C68016A
IFCLK	PE0
CLKOUT	PE1

Note
10. This EEPROM does not have address pins.

Figure 9. CY7C68013A/CY7C68014A 56-Pin SSOP Pin Assignment

CY7C68013A/CY7C68014A
56-pin SSOP



* denotes programmable polarity

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

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	Name	Type	Default	Reset[12]	Description
7	6	–	–	–	RDY3	Input	N/A	N/A	**RDY3** is a GPIF input signal.
8	7	–	–	–	RDY4	Input	N/A	N/A	**RDY4** is a GPIF input signal.
9	8	–	–	–	RDY5	Input	N/A	N/A	**RDY5** is a GPIF input signal.
69	54	36	29	7H	CTL0 or FLAGA	O/Z	H	L	Multiplexed pin whose function is selected by the following bits: IFCONFIG[1..0]. **CTL0** is a GPIF control output. **FLAGA** is a programmable slave-FIFO output status flag signal. Defaults to programmable for the FIFO selected by the FIFOADR[1:0] pins.
70	55	37	30	7G	CTL1 or FLAGB	O/Z	H	L	Multiplexed pin whose function is selected by the following bits: IFCONFIG[1..0]. **CTL1** is a GPIF control output. **FLAGB** is a programmable slave-FIFO output status flag signal. Defaults to FULL for the FIFO selected by the FIFOADR[1:0] pins.
71	56	38	31	8H	CTL2 or FLAGC	O/Z	H	L	Multiplexed pin whose function is selected by the following bits: IFCONFIG[1..0]. **CTL2** is a GPIF control output. **FLAGC** is a programmable slave-FIFO output status flag signal. Defaults to EMPTY for the FIFO selected by the FIFOADR[1:0] pins.
66	51	–	–	–	CTL3	O/Z	H	L	**CTL3** is a GPIF control output.
67	52	–	–	–	CTL4	Output	H	L	**CTL4** is a GPIF control output.
98	76	–	–	–	CTL5	Output	H	L	**CTL5** is a GPIF control output.
32	26	20	13	2G	IFCLK on CY7C68013A and CY7C68014A --- PE0 on CY7C68015A and CY7C68016A	I/O/Z --- - I/O/Z	Z --- I	Z --- Z	Interface Clock, used for synchronously clocking data into or out of the slave FIFOs. IFCLK also serves as a timing reference for all slave FIFO control signals and GPIF. When internal clocking is used (IFCONFIG.7 = 1) the IFCLK pin can be configured to output 30/48 MHz by bits IFCONFIG.5 and IFCONFIG.6. IFCLK may be inverted, whether internally or externally sourced, by setting the bit IFCONFIG.4 =1. --- **PE0** is a bidirectional I/O port pin.
28	22	–	–	–	INT4	Input	N/A	N/A	**INT4** is the 8051 INT4 interrupt request input signal. The INT4 pin is edge-sensitive, active HIGH.
106	84	–	–	–	INT5#	Input	N/A	N/A	**INT5#** is the 8051 INT5 interrupt request input signal. The INT5 pin is edge-sensitive, active LOW.
31	25	–	–	–	T2	Input	N/A	N/A	**T2** is the active HIGH T2 input signal to 8051 Timer2, which provides the input to Timer2 when C/T2 = 1. When C/T2 = 0, Timer2 does not use this pin.

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	rbbbbbbb
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	GPIFIRQ[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	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	x	x	x	x	x	x	x	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	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	PORTECFG	I/O PORTE Alternate Configuration	GPIFA8	T2EX	INT6	RXD1OUT	RXD0OUT	T2OUT	T1OUT	T0OUT	00000000	RW
E673	4	reserved											
E677	1	reserved											
E678	1	I^2CS	I²C 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	xxxxxxxx	RW
E67A	1	I^2CTL	I²C 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	xxxxxxxx	RW
E67C	1	XAUTODAT2	Autoptr2 MOVX access, when APTREN=1	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	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	WAKEUPCS	Wakeup Control & Status	WU2	WU	WU2POL	WUPOL	0	DPEN	WU2EN	WUEN	xx000101	bbbbrbbb
E683	1	TOGCTL	Toggle Control	Q	S	R	I/O	EP3	EP2	EP1	EP0	x0000000	rrrbbbbb
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	xxxxxxxx	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)	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	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	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	xxxxxxxx	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	xxxxxxxx	RW
E69A	2	reserved											
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 L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxxx	RW
E69E	2	reserved											

Table 12. FX2LP Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
xxxx		I²C Configuration Byte		0	DISCON	0	0	0	0	0	400KHZ	xxxxxxxx [16]	n/a
		Special Function Registers (SFRs)											
80	1	IOA[15]	Port A (bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
81	1	SP	Stack Pointer	D7	D6	D5	D4	D3	D2	D1	D0	00000111	RW
82	1	DPL0	Data Pointer 0 L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
83	1	DPH0	Data Pointer 0 H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
84	1	DPL1[15]	Data Pointer 1 L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
85	1	DPH1[15]	Data Pointer 1 H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
86	1	DPS[15]	Data Pointer 0/1 select	0	0	0	0	0	0	0	SEL	00000000	RW
87	1	PCON	Power Control	SMOD0	x	1	1	x	x	x	IDLE	00110000	RW
88	1	TCON	Timer/Counter Control (bit addressable)	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00000000	RW
89	1	TMOD	Timer/Counter Mode Control	GATE	CT	M1	M0	GATE	CT	M1	M0	00000000	RW
8A	1	TL0	Timer 0 reload L	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
8B	1	TL1	Timer 1 reload L	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
8C	1	TH0	Timer 0 reload H	D15	D14	D13	D12	D11	D10	D9	D8	00000000	RW
8D	1	TH1	Timer 1 reload H	D15	D14	D13	D12	D11	D10	D9	D8	00000000	RW
8E	1	CKCON[15]	Clock Control	x	x	T2M	T1M	T0M	MD2	MD1	MD0	00000001	RW
8F	1	reserved											
90	1	IOB[15]	Port B (bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
91	1	EXIF[15]	External Interrupt Flag(s)	IE5	IE4	I²CINT	USBNT	1	0	0	0	00001000	RW
92	1	MPAGE[15]	Upper Addr Byte of MOVX using @R0 / @R1	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
93	5	reserved											
98	1	SCON0	Serial Port 0 Control (bit addressable)	SM0_0	SM1_0	SM2_0	REN_0	TB8_0	RB8_0	TI_0	RI_0	00000000	RW
99	1	SBUF0	Serial Port 0 Data Buffer	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
9A	1	AUTOPTRH1[15]	Autopointer 1 Address H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
9B	1	AUTOPTRL1[15]	Autopointer 1 Address L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
9C	1	reserved											
9D	1	AUTOPTRH2[15]	Autopointer 2 Address H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
9E	1	AUTOPTRL2[15]	Autopointer 2 Address L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
9F	1	reserved											
A0	1	IOC[15]	Port C (bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
A1	1	INT2CLR[15]	Interrupt 2 clear	x	x	x	x	x	x	x	x	xxxxxxxx	W
A2	1	INT4CLR[15]	Interrupt 4 clear	x	x	x	x	x	x	x	x	xxxxxxxx	W
A3	5	reserved											
A8	1	IE	Interrupt Enable (bit addressable)	EA	ES1	ET2	ES0	ET1	EX1	ET0	EX0	00000000	RW
A9	1	reserved											
AA	1	EP2468STAT[15]	Endpoint 2,4,6,8 status flags	EP8F	EP8E	EP6F	EP6E	EP4F	EP4E	EP2F	EP2E	01011010	R
AB	1	EP24FIFOFLGS[15]	Endpoint 2,4 slave FIFO status flags	0	EP4PF	EP4EF	EP4FF	0	EP2PF	EP2EF	EP2FF	00100010	R
AC	1	EP68FIFOFLGS[15]	Endpoint 6,8 slave FIFO status flags	0	EP8PF	EP8EF	EP8FF	0	EP6PF	EP6EF	EP6FF	01100110	R
AD	2	reserved											
AF	1	AUTOPTRSETUP[15]	Autopointer 1&2 setup	0	0	0	0	0	APTR2INC	APTR1INC	APTREN	00000110	RW
B0	1	IOD[15]	Port D (bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
B1	1	IOE[15]	Port E (NOT bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
B2	1	OEA[15]	Port A Output Enable	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B3	1	OEB[15]	Port B Output Enable	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B4	1	OEC[15]	Port C Output Enable	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B5	1	OED[15]	Port D Output Enable	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B6	1	OEE[15]	Port E Output Enable	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B7	1	reserved											
B8	1	IP	Interrupt Priority (bit addressable)	1	PS1	PT2	PS0	PT1	PX1	PT0	PX0	10000000	RW
B9	1	reserved											
BA	1	EP01STAT[15]	Endpoint 0&1 Status	0	0	0	0	0	EP1INBSY	EP1OUTBSY	EP0BSY	00000000	R
BB	1	GPIFTRIG[15, 13]	Endpoint 2,4,6,8 GPIF slave FIFO Trigger	DONE	0	0	0	0	RW	EP1	EP0	10000xxx	brrrrbbb
BC	1	reserved											
BD	1	GPIFSGLDATH[15]	GPIF Data H (16-bit mode only)	D15	D14	D13	D12	D11	D10	D9	D8	xxxxxxxx	RW

Notes
15. SFRs not part of the standard 8051 architecture.
16. If no EEPROM is detected by the SIE then the default is 00000000.

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	xxxxxxxx	RW
BF	1	GPIFSGLDATL-NOX[15]	GPIF Data L w/ No Trigger	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	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	EI²C	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	PI²C	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

AC Electrical Characteristics

USB Transceiver

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

Program Memory Read

Figure 12. Program Memory Read Timing Diagram

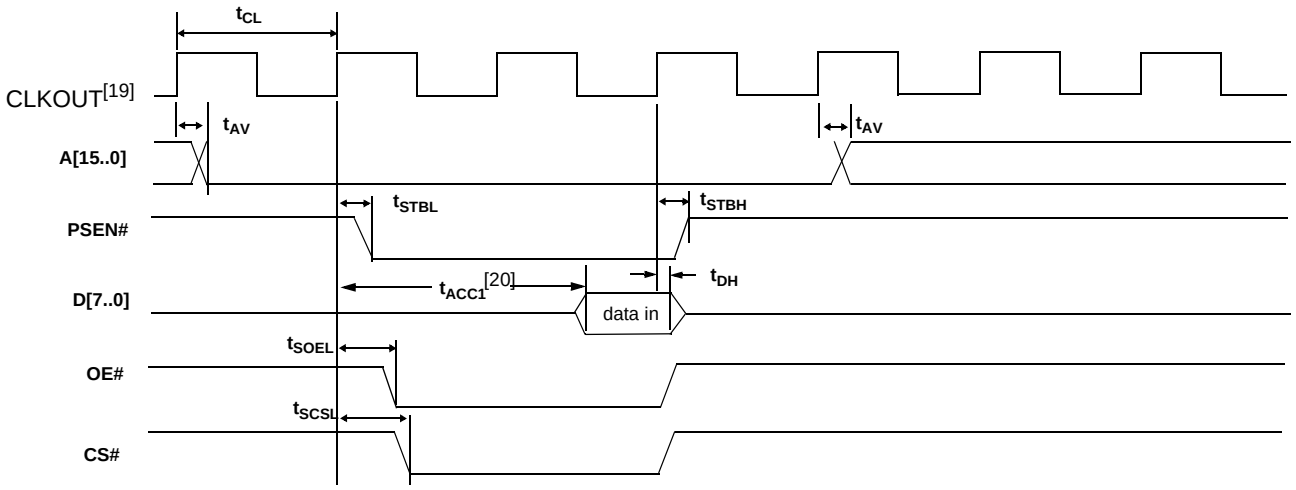


Table 15. Program Memory Read Parameters

Parameter	Description	Min	Typ	Max	Unit	Notes
t_{CL}	1/CLKOUT frequency	–	20.83	–	ns	48 MHz
		–	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.

Data Memory Read[21]

Figure 13. Data Memory Read Timing Diagram

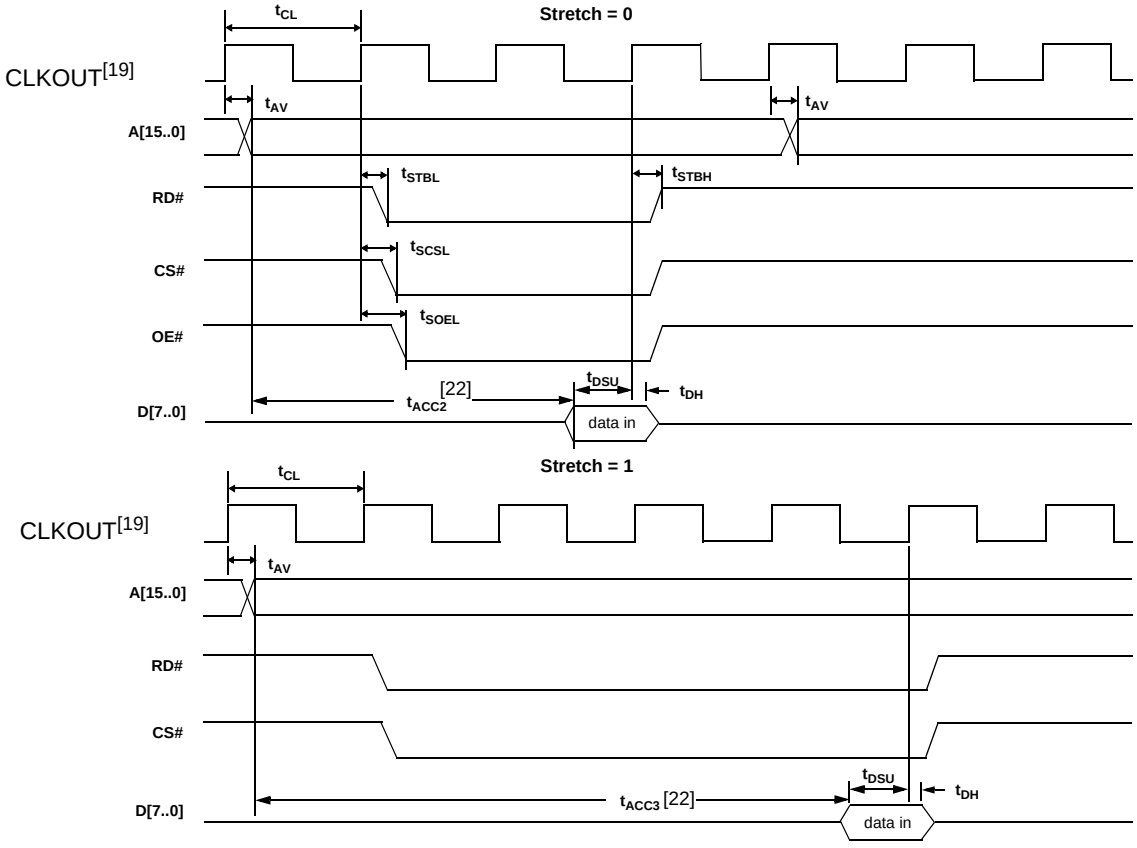


Table 16. Data Memory Read Parameters

Parameter	Description	Min	Typ	Max	Unit	Notes
t_{CL}	1/CLKOUT frequency	–	20.83	–	ns	48 MHz
		–	41.66	–	ns	24 MHz
		–	83.2	–	ns	12 MHz
t_{AV}	Delay from clock to valid address	–	–	10.7	ns	–
t_{STBL}	Clock to RD LOW	–	–	11	ns	–
t_{STBH}	Clock to RD HIGH	–	–	11	ns	–
t_{SCSL}	Clock to CS LOW	–	–	13	ns	–
t_{SOEL}	Clock to OE LOW	–	–	11.1	ns	–
t_{DSU}	Data setup to clock	9.6	–	–	ns	–
t_{DH}	Data hold time	0	–	–	ns	–

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

Notes

21. The stretch memory cycle feature enables EZ-USB firmware to adjust the speed of data memory accesses not the program memory accesses. Details including typical strobe width timings can be found in the section 12.1.2 of the Technical Reference Manual. The address cycle width can be interpreted from these.
22. t_{ACC2} and t_{ACC3} are computed from these parameters as follows:
t_{ACC2}(24 MHz) = 3*t_{CL} – t_{AV} –t_{DSU} = 106 ns
t_{ACC2}(48 MHz) = 3*t_{CL} – t_{AV} – t_{DSU} = 43 ns
t_{ACC3}(24 MHz) = 5*t_{CL} – t_{AV} –t_{DSU} = 190 ns
t_{ACC3}(48 MHz) = 5*t_{CL} – t_{AV} – t_{DSU} = 86 ns

PORTC Strobe Feature Timings

The RD# and WR# are present in the 100-pin version and the 128-pin package. In these 100-pin and 128-pin versions, an 8051 control bit can be set to pulse the RD# and WR# pins when the 8051 reads from or writes to PORTC. This feature is enabled by setting PORTCSTB bit in CPUCS register.

The RD# and WR# strobes are asserted for two CLKOUT cycles when PORTC is accessed.

The WR# strobe is asserted two clock cycles after PORTC is updated and is active for two clock cycles after that, as shown in Figure 16.

As for read, the value of PORTC three clock cycles before the assertion of RD# is the value that the 8051 reads in. The RD# is pulsed for two clock cycles after three clock cycles from the point when the 8051 has performed a read function on PORTC.

The RD# signal prompts the external logic to prepare the next data byte. Nothing gets sampled internally on assertion of the RD# signal itself; it is just a prefetch type signal to get the next data byte prepared. So, using it with that in mind easily meets the setup time to the next read.

The purpose of this pulsing of RD# is to allow the external peripheral to know that the 8051 is done reading PORTC and the data was latched into PORTC three CLKOUT cycles before asserting the RD# signal. After the RD# is pulsed, the external logic can update the data on PORTC.

Following is the timing diagram of the read and write strobing function on accessing PORTC. Refer to Data Memory Read[21] and Data Memory Write[23] for details on propagation delay of RD# and WR# signals.

Figure 16. WR# Strobe Function when PORTC is Accessed by 8051

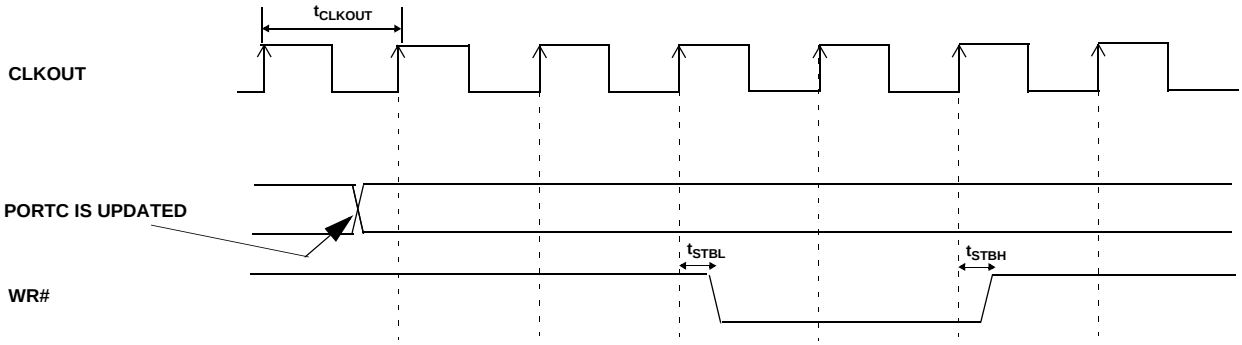
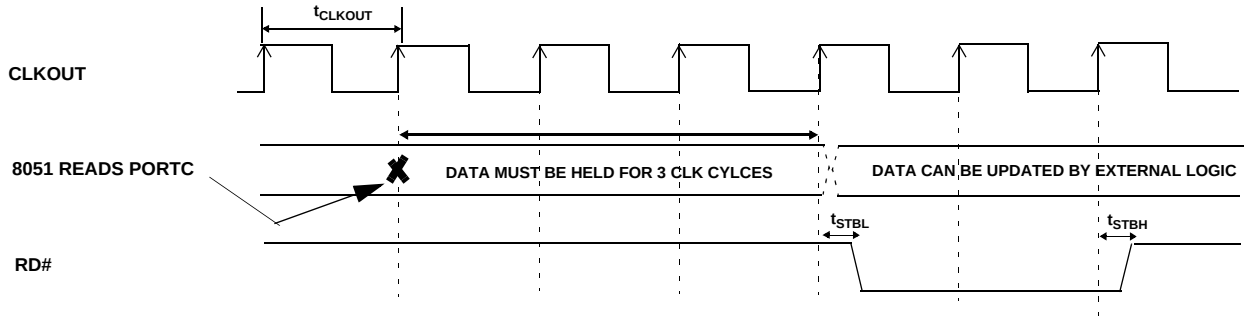


Figure 17. RD# Strobe Function when PORTC is Accessed by 8051



GPIF Synchronous Signals

Figure 18. GPIF Synchronous Signals Timing Diagram[24]

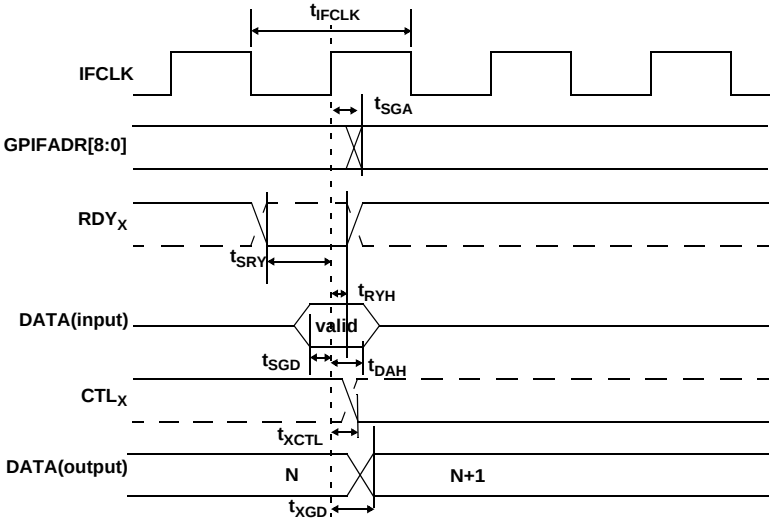


Table 18. GPIF Synchronous Signals Parameters with Internally Sourced IFCLK[24, 25]

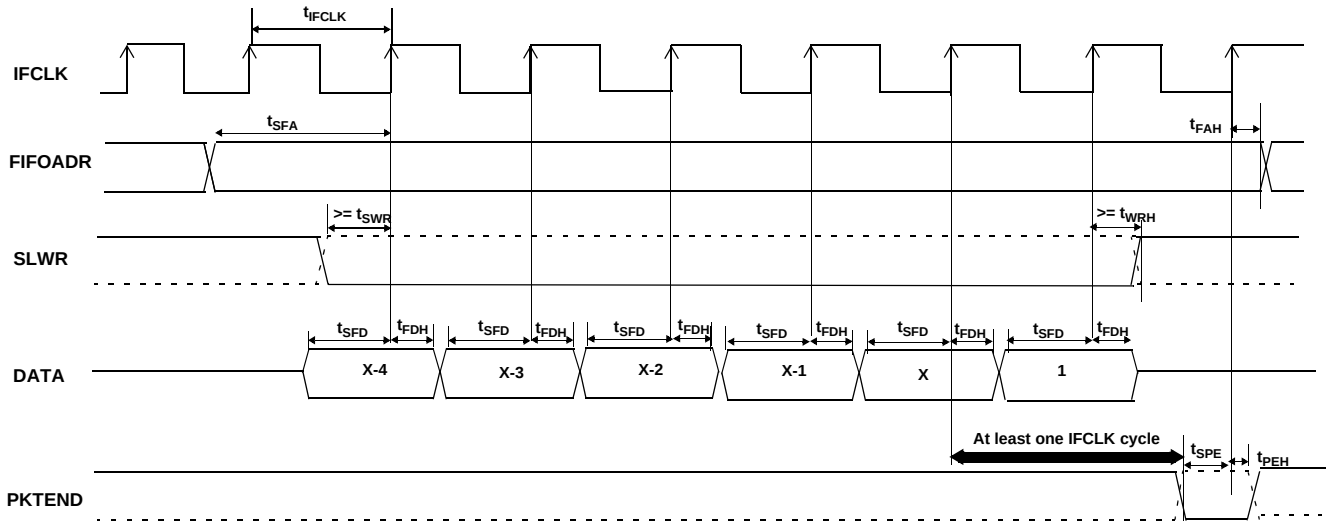
Parameter	Description	Min	Max	Typ		Unit
				Min	Max	
t_{IFCLK}	IFCLK Period	20.83	–	–	–	ns
t_{SRY}	RDY_X to clock setup time	8.9	–	–	–	ns
t_{RYH}	Clock to RDY_X	0	–	–	–	ns
t_{SGD}	GPIF data to clock setup time	9.2	–	–	–	ns
t_{DAH}	GPIF data hold time	0	–	–	–	ns
t_{SGA}	Clock to GPIF address propagation 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]

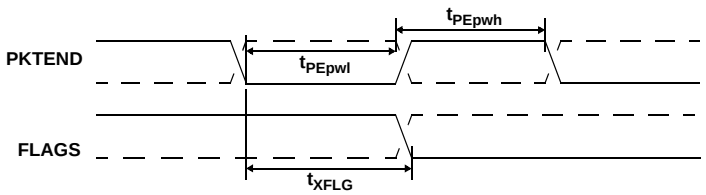


Table 28. Slave FIFO Asynchronous Packet End Strobe Parameters[27]

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

Figure 26. Slave FIFO Output Enable Timing Diagram[24]

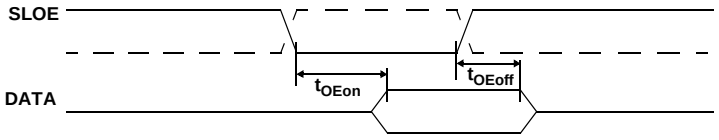


Table 29. Slave FIFO Output Enable Parameters

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

Figure 27. Slave FIFO Address to Flags/Data Timing Diagram[24]

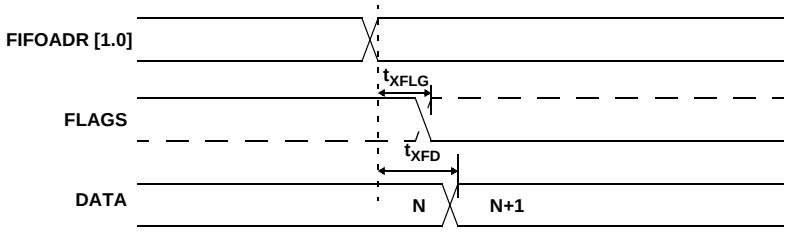


Table 30. Slave FIFO Address to Flags/Data Parameters

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]

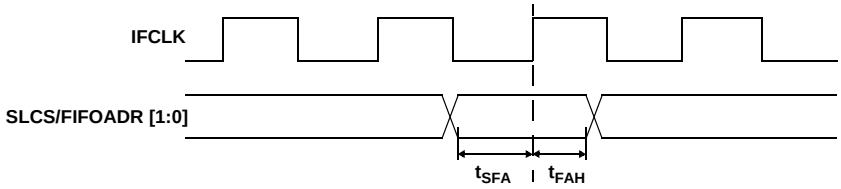


Table 31. Slave FIFO Synchronous Address Parameters[25]

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]

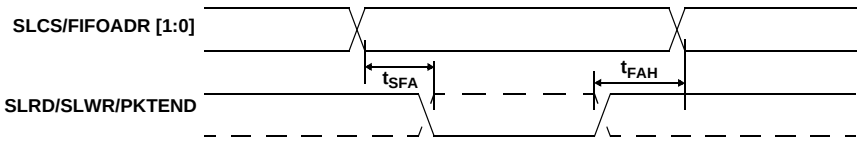


Table 32. Slave FIFO Asynchronous Address Parameters[27]

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

Single and Burst Synchronous Write

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

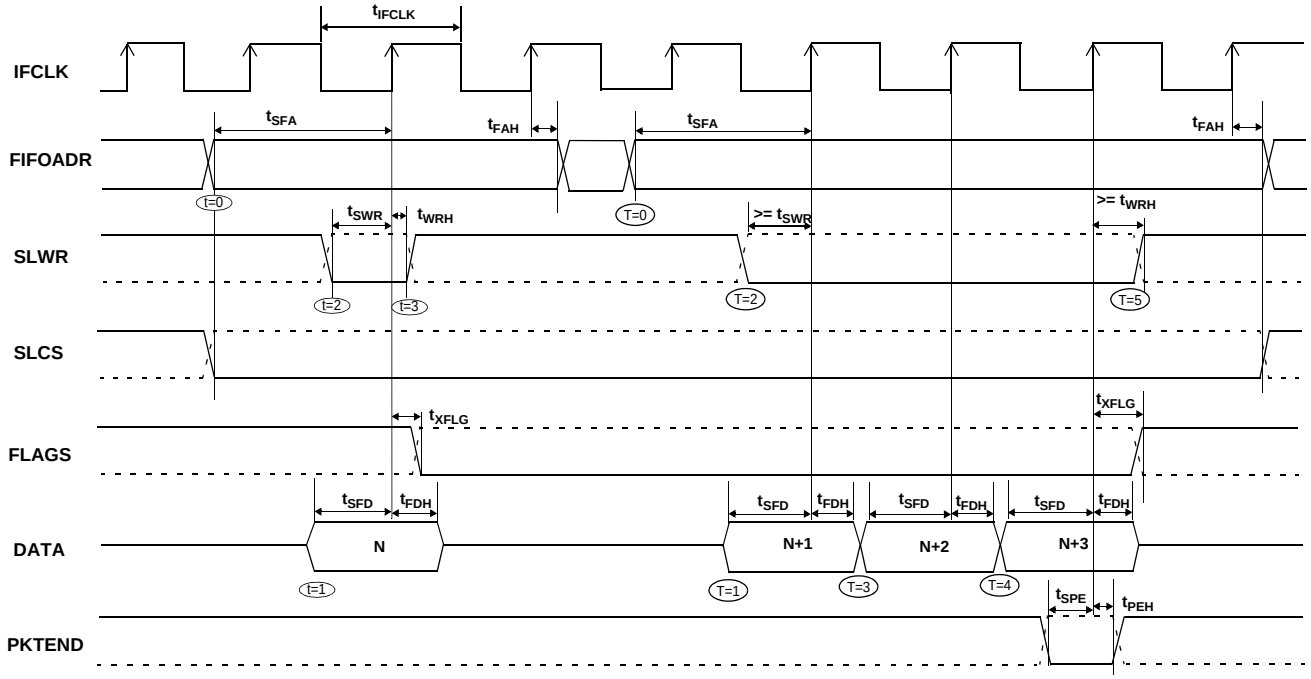


Figure 32 shows the timing relationship of the SLAVE FIFO signals during a synchronous write using IFCLK as the synchronizing clock. The diagram illustrates a single write followed by burst write of three bytes and committing all four bytes as a short packet using the PKTEND pin.

- 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]

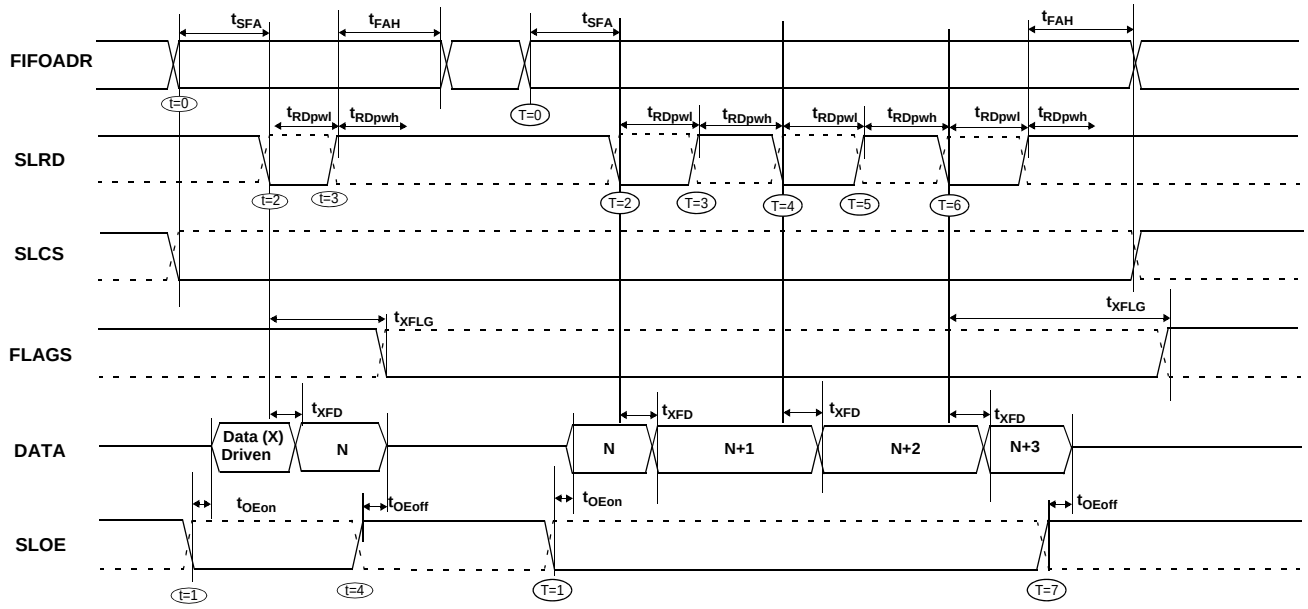


Figure 34. Slave FIFO Asynchronous Read Sequence of Events Diagram

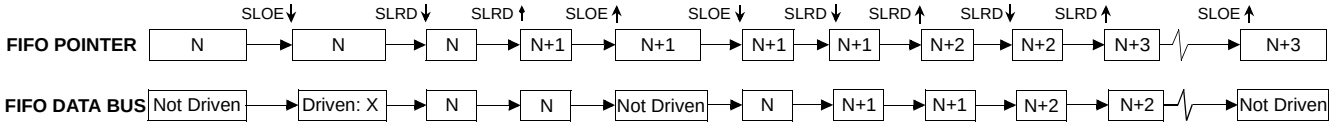


Figure 33 shows the timing relationship of the SLAVE FIFO signals during an asynchronous FIFO read. It shows a single read followed by a burst read.

- 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 anearlier 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.

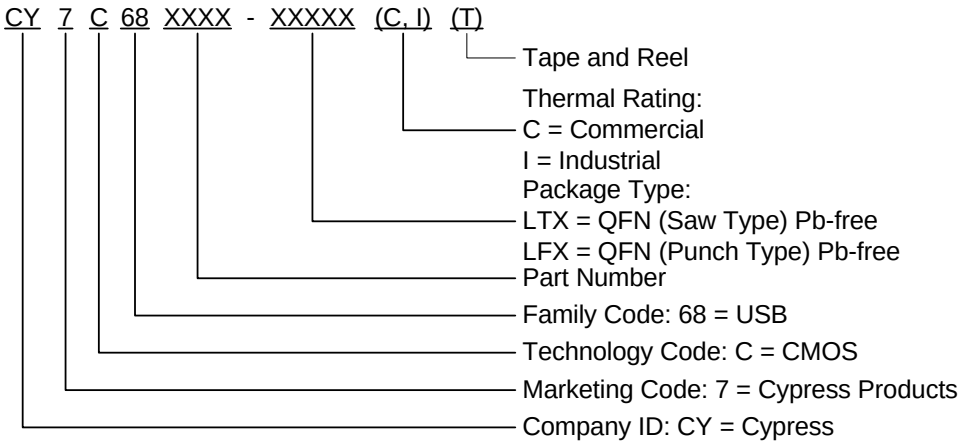
Note In the burst read mode, during SLOE is asserted, the data bus is in a driven state and outputs the previous data. After SLRD is asserted, the data from the FIFO is driven on the data bus (SLOE must also be asserted) and then the FIFO pointer is incremented.

Ordering Information

Table 33. Ordering Information

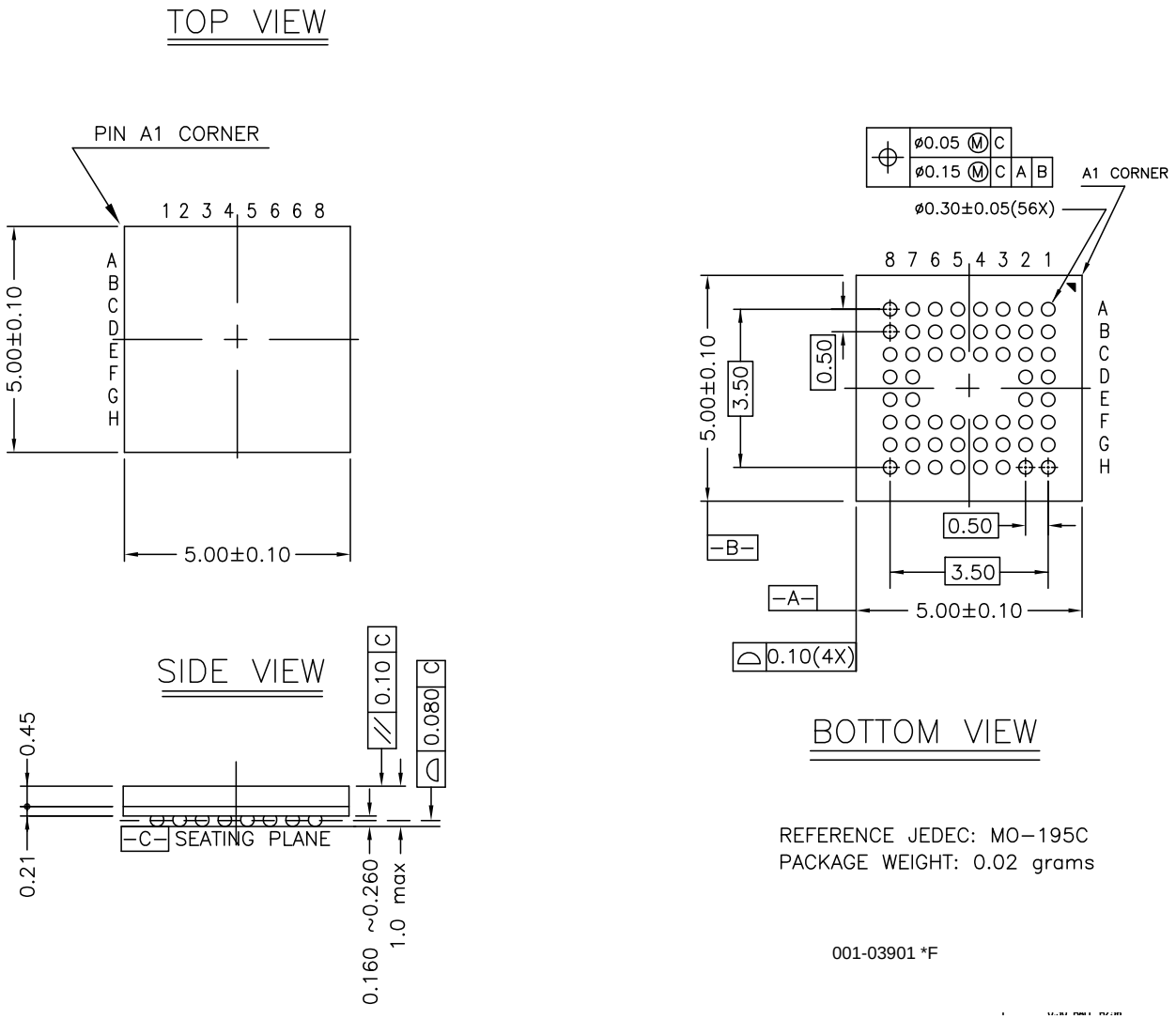
Ordering Code	Package Type	RAM Size	# Prog I/Os	8051 Address /Data Bus	Serial Debug[28]
Ideal for Battery Powered Applications					
CY7C68014A-128AXC	128 TQFP – Pb-free	16 K	40	16-/8-bit	Y
CY7C68014A-100AXC	100 TQFP – Pb-free	16 K	40	–	Y
CY7C68014A-56PVXC	56 SSOP – Pb-free	16 K	24	–	N
CY7C68014A-56LTXC	56 QFN - Pb-free	16 K	24	–	N
CY7C68016A-56LTXC	56 QFN - Pb-free	16 K	26	–	N
CY7C68016A-56LTXCT	56 QFN - Pb-free	16 K	26	–	N
Ideal for Non Battery Powered Applications					
CY7C68013A-128AXC	128 TQFP – Pb-free	16 K	40	16-/8-bit	Y
CY7C68013A-128AXI	128 TQFP – Pb-free (Industrial)	16 K	40	16-/8-bit	Y
CY7C68013A-100AXC	100 TQFP – Pb-free	16 K	40	–	Y
CY7C68013A-100AXI	100 TQFP – Pb-free (Industrial)	16 K	40	–	Y
CY7C68013A-56PVXC	56 SSOP – Pb-free	16 K	24	–	N
CY7C68013A-56PVXCT	56 SSOP – Pb-free	16 K	24	–	N
CY7C68013A-56PVXI	56 SSOP – Pb-free (Industrial)	16 K	24	–	N
CY7C68013A-56BAXC	56 VFBGA – Pb-free	16 K	24	–	N
CY7C68013A-56BAXCT	56 VFBGA – Pb-free	16 K	24	–	N
CY7C68013A-56LTXC	56 QFN – Pb-free	16 K	24	–	N
CY7C68013A-56LTXCT	56 QFN – Pb-free	16 K	24	–	N
CY7C68013A-56LTXI	56 QFN – Pb-free (Industrial)	16 K	24	–	N
CY7C68015A-56LTXC	56 QFN – Pb-free	16 K	26	–	N
Development Tool Kit					
CY3684	EZ-USB FX2LP development kit				
Reference Design Kit					
CY4611B	USB 2.0 to ATA/ATAPI reference design using EZ-USB FX2LP				

Ordering Code Definitions



Note
28. As UART is not available in the 56-pin package of CY7C68013A, serial port debugging using Keil Monitor is not possible.

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



Document History Page (continued)

Document Title: CY7C68013A, CY7C68014A, CY7C68015A, CY7C68016A, EZ-USB® FX2LP™ USB Microcontroller High-Speed USB Peripheral Controller Document Number: 38-08032				
Rev.	**ECN No.**	**Orig. of Change**	**Submission Date**	**Description of Change**
*W	3998554	GAYA	07/19/2013	Added Errata footnote (Note 3). Updated Functional Overview: Updated Interrupt System: Updated FIFO/GPIF Interrupt (INT4): Added Note 3 and referred the same note in “Endpoint 2 empty flag” in Table 4. Updated Package Diagrams: spec 51-85062 – Changed revision from *E to *F. spec 001-53450 – Changed revision from *B to *C. Added Errata. Updated in new template.
*X	4617527	GAYA	01/15/2015	Updated Figure 13 Added a note to sections Data Memory Read[21] and Data Memory Write[23] sections Updated template to include the More Information section Updated Figure 37, Figure 38, Figure 39 Updated Table 11 with Reset state information for pins Sunset Review
*Y	5317277	ODC	06/28/2016	Updated CY Logo and Sales Disclaimer.

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