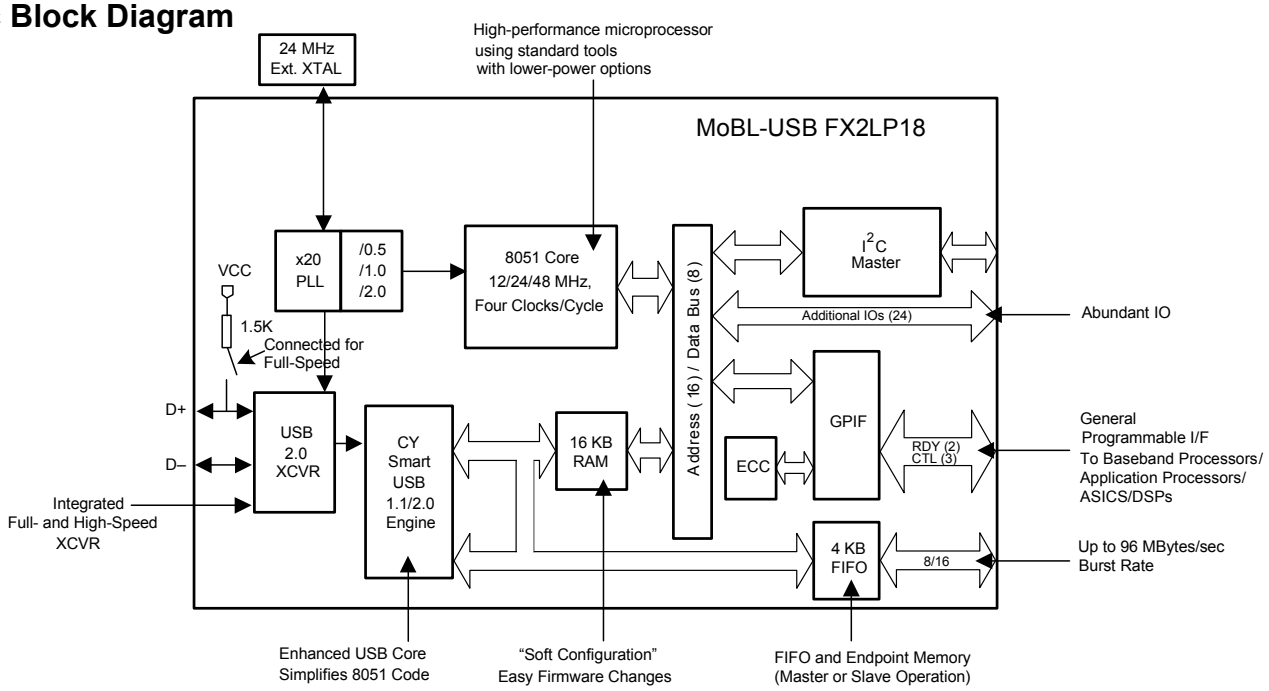
CY7C68053

MoBL-USB™ FX2LP18 USB Microcontroller

Features

- USB 2.0 9 V USB-IF high speed and full speed compliant (TID# 40000188)
- Single-chip integrated USB 2.0 transceiver, smart SIE, and enhanced 8051 microprocessor
- Ideal for mobile applications (cell phone, smart phones, PDAs, MP3 players)
  - Ultra low power
  - Suspend current: 20 µA (typical)
- Software: 8051 Code runs from:
  - Internal RAM, which is loaded from EEPROM
- 16 kBytes of on-chip code/data RAM
- Four programmable BULK/INTERRUPT/ISOCHRONOUS endpoints
  - Buffering options: double, triple, and quad
- Additional Programmable (BULK/INTERRUPT) 64-Byte Endpoint
- 8 or 16-Bit External Data Interface
- Smart Media Standard ECC Generation
- GPIF (General Programmable Interface)
  - Allows direct connection to most parallel interface
  - Programmable waveform descriptors and configuration registers to define waveforms
  - Supports multiple Ready and Control outputs
- Integrated, Industry Standard Enhanced 8051
  - 48 MHz, 24 MHz, or 12 MHz CPU operation
  - Four clocks per instruction cycle
  - Three counter/timers
  - Expanded interrupt system
  - Two data pointers
- 1.8 V Core Operation
- 1.8 V to 3.3 V I/O Operation
- Vectored USB Interrupts and GPIF/FIFO Interrupts
- Separate Data Buffers for Setup and Data Portions of a CONTROL Transfer
- Integrated I^2C Controller, runs at 100 or 400 kHz
- Four Integrated FIFOs
  - Integrated glue logic and FIFOs lower system cost
  - Automatic conversion to and from 16-bit buses
  - Master or slave operation
  - Uses external clock or asynchronous strobes
  - Easy interface to ASIC and DSP ICs
- Available in Industrial Temperature Grade
- Available in one Pb-free Package with up to 24 GPIOs
  - 56-pin VFBGA (24 GPIOs)

Logic Block Diagram



Contents

Functional Description

Cypress Semiconductor Corporation's MoBL-USB™ FX2LP18 (CY7C68053) is a low voltage (1.8 V) version of the EZ-USB® FX2LP (CY7C68013A), which is a highly integrated, low power USB 2.0 microcontroller. By integrating the USB 2.0 transceiver, serial interface engine (SIE), enhanced 8051 microcontroller, and a programmable peripheral interface in a single chip, Cypress has created a very cost effective solution that provides superior time-to-market advantages with low power to enable bus powered applications.

The ingenious architecture of MoBL-USB FX2LP18 results in data transfer rates of over 53 Mbytes per second, the maximum allowable USB 2.0 bandwidth, while still using a low cost 8051 microcontroller in a package as small as a 56VFBGA (5 mm x 5 mm). Because it incorporates the USB 2.0 transceiver, the MoBL-USB FX2LP18 is more economical, providing a smaller footprint solution than USB 2.0 SIE or external transceiver implementations. With MoBL-USB FX2LP18, the Cypress Smart SIE handles most of the USB 1.1 and 2.0 protocol in hardware, freeing the embedded microcontroller for application-specific functions and decreasing development time to ensure USB compatibility.

The General Programmable Interface (GPIF) and Master/Slave Endpoint FIFO (8 or 16-bit data bus) provide an easy and glueless interface to popular interfaces such as ATA, UTOPIA, EPP, PCMCIA, and most DSP/processors.

The MoBL-USB FX2LP18 is also referred to as FX2LP18 in this document.

Applications

There are a wide variety of applications for the MoBL-USB FX2LP18. It is used in cell phones, smart phones, PDAs, and MP3 players, to name a few.

The 'Reference Designs' section of the Cypress web site provides additional tools for typical USB 2.0 applications. Each reference design comes complete with firmware source and object code, schematics, and documentation. For more information, visit http://www.cypress.com.

Functional Overview

The functionality of this chip is described in the sections below.

USB Signaling Speed

FX2LP18 operates at two of the three rates defined in the USB Specification Revision 2.0, dated April 27, 2000.

- Full speed, with a signaling bit rate of 12 Mbps
- High speed, with a signaling bit rate of 480 Mbps

FX2LP18 does not support the low speed signaling mode of 1.5 Mbps.

8051 Microprocessor

The 8051 microprocessor embedded in the FX2LP18 family has 256 bytes of register RAM, an expanded interrupt system, and three timer/counters.

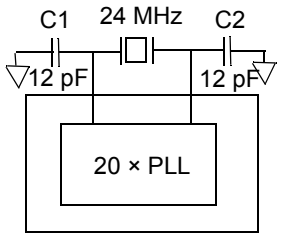
8051 Clock Frequency

FX2LP18 has an on-chip oscillator circuit that uses an external 24 MHz (±100-ppm) crystal with the following characteristics:

- Parallel resonant
- Fundamental mode
- 500 µW drive level
- 12 pF (5% tolerance) load capacitors

An on-chip PLL multiplies the 24 MHz oscillator up to 480 MHz, as required by the transceiver/PHY; internal counters divide it down for use as the 8051 clock. The default 8051 clock frequency is 12 MHz. The clock frequency of the 8051 can be changed by the 8051 through the CPUCS register, dynamically.

Figure 1. Crystal Configuration



12 pF capacitor values assumes a trace capacitance of 3 pF per side on a four-layer FR4 PCA

The CLKOUT pin, which can be tristated and inverted using internal control bits, outputs the 50% duty cycle 8051 clock, at the selected 8051 clock frequency — 48, 24, or 12 MHz.

Special Function Registers

Certain 8051 Special Function Register (SFR) addresses are populated to provide fast access to critical FX2LP18 functions. These SFR additions are shown in Table 1 on page 4. Bold type indicates non standard, enhanced 8051 registers. The two SFR rows that end with '0' and '8' contain bit-addressable registers. The four I/O ports A–D use the SFR addresses used in the standard 8051 for ports 0–3, which are not implemented in FX2LP18. Because of the faster and more efficient SFR addressing, the FX2LP18 I/O ports are not addressable in external RAM space (using the MOVX instruction).

Table 1. Special Function Registers

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

I²C™ Bus

FX2LP18 supports the I²C bus as a master only at 100 or 400 KHz. SCL and SDA pins have open-drain outputs and hysteresis inputs. These signals must be pulled up to either V_{CC} or V_{CC_IO}, even if no I²C device is connected. (Connecting to V_{CC_IO} may be more convenient.)

Buses

This 56-pin package has an 8- or 16-bit ‘FIFO’ bidirectional data bus, multiplexed on I/O ports B and D.

USB Boot Methods

During the power up sequence, internal logic checks the I²C port for the connection of an EEPROM whose first byte is 0xC2. If found, it boot-loads the EEPROM contents into internal RAM (0xC2 load). If no EEPROM is present, an external processor must emulate an I²C slave. The FX2LP18 does not enumerate using internally stored descriptors (for example, Cypress’s VID/PID/DID is not used for enumeration).[1]

ReNumeration™

Because the FX2LP18’s configuration is soft, one chip can take on the identities of multiple distinct USB devices.

When first plugged into USB, the FX2LP18 enumerates automatically and downloads firmware and USB descriptor tables over the USB cable. Next, the FX2LP18 enumerates again, this time as a device defined by the downloaded information. This patented two-step process, called ReNumeration™, happens instantly when the device is plugged in, with no hint that the initial download step has occurred.

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

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

Bus-Powered Applications

The FX2LP18 fully supports bus-powered designs by enumerating with less than 100 mA as required by the USB 2.0 specification.

Note

1. 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.

Interrupt System

The FX2LP18 interrupts are described in this section.

INT2 Interrupt Request and Enable Registers

FX2LP18 implements an autovector feature for INT2. There are 27 INT2 (USB) vectors. See the *MoBL-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 normally required to identify the individual USB interrupt source, the FX2LP18 provides a second level of interrupt vectoring, called 'Autovectoring.' When a USB interrupt is asserted, the FX2LP18 pushes the program counter onto its stack then jumps to address 0x0043, where it expects to find a 'jump' instruction to the USB interrupt service routine.

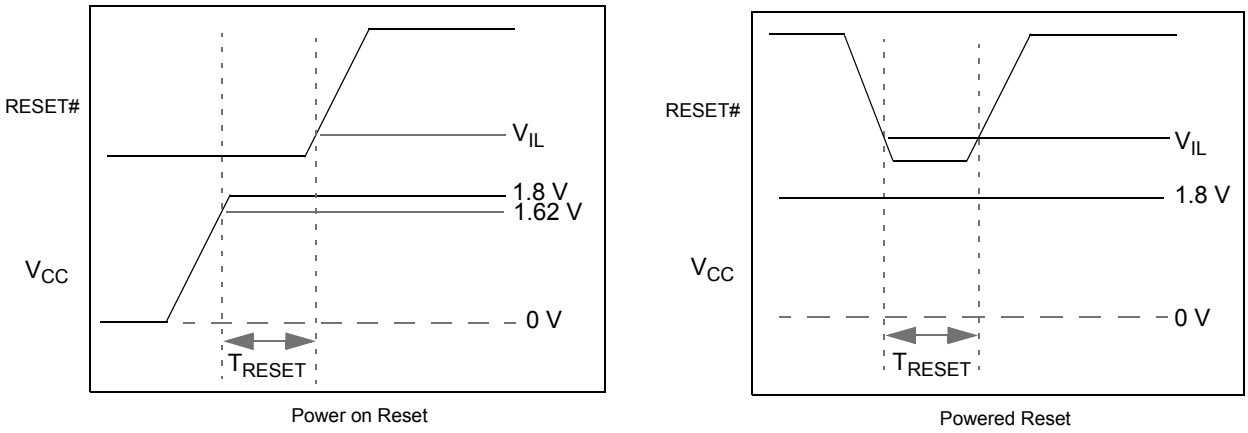
The FX2LP18 jump instruction is encoded as shown in Table 2 on page 5.

If Autovectoring is enabled (AV2EN = 1 in the INTSET-UP register), the FX2LP18 substitutes its INT2VEC byte. Therefore, if the high byte ('page') of a jump-table address is preloaded at location 0x0044, the automatically inserted INT2VEC byte at 0x0045 directs the jump to the correct address out of the 27 addresses within the page.

Table 2. INT2 USB Interrupts

Priority	INT2VEC Value	Source	Notes
1	00	SUDAV	Setup data available
2	04	SOF	Start of frame (or microframe)
3	08	SUTOK	Setup token received
4	0C	SUSPEND	USB suspend request
5	10	USB RESET	Bus reset
6	14	HISPEED	Entered high speed operation
7	18	EP0ACK	FX2LP18 ACK'd the control handshake
8	1C		Reserved
9	20	EP0-IN	EP0-IN ready to be loaded with data
10	24	EP0-OUT	EP0-OUT has USB data
11	28	EP1-IN	EP1-IN ready to be loaded with data
12	2C	EP1-OUT	EP1-OUT has USB data
13	30	EP2	IN: buffer available. OUT: buffer has data
14	34	EP4	IN: buffer available. OUT: buffer has data
15	38	EP6	IN: buffer available. OUT: buffer has data
16	3C	EP8	IN: buffer available. OUT: buffer has data
17	40	IBN	IN-Bulk-NAK (any IN endpoint)
18	44		Reserved
19	48	EP0PING	EP0 OUT was pinged and it NAK'd
20	4C	EP1PING	EP1 OUT was pinged and it NAK'd
21	50	EP2PING	EP2 OUT was pinged and it NAK'd
22	54	EP4PING	EP4 OUT was pinged and it NAK'd
23	58	EP6PING	EP6 OUT was pinged and it NAK'd
24	5C	EP8PING	EP8 OUT was pinged and it NAK'd
25	60	ERRLIMIT	Bus errors exceeded the programmed limit
26	64		
27	68		Reserved
28	6C		Reserved
29	70	EP2ISOERR	ISO EP2 OUT PID sequence error
30	74	EP4ISOERR	ISO EP4 OUT PID sequence error
31	78	EP6ISOERR	ISO EP6 OUT PID sequence error
32	7C	EP8ISOERR	ISO EP8 OUT PID sequence error

Figure 2. Reset Timing Plots



Reset and Wakeup

The reset and wakeup pins are described in detail in this section.

Reset Pin

The input pin, RESET#, resets the FX2LP18 when asserted. This pin has hysteresis and is active LOW. When a crystal is used with the CY7C68053, the reset period must allow for the stabilization of the crystal and the PLL. This reset period must be approximately 5 ms after VCC has reached 3.0 V. If the crystal input pin is driven by a clock signal the internal PLL stabilizes in 200 µs after V_{CC} has reached 3.0 V[2]. Figure 2 shows a power on reset condition and a reset applied during operation. A power on reset is defined as the time reset is asserted while power is being applied to the circuit. A powered reset is defined as a reset in which the FX2LP18 has previously been powered on and operating and the RESET# pin is asserted.

Cypress provides an application note which describes and recommends power on reset implementation, which can be found on the Cypress web site. For more information on reset implementation for the MoBL-USB family of products, visit the Cypress web site at http://www.cypress.com.

Table 3. Reset Timing Values

Condition	T_{RESET}
Power on reset with crystal	5 ms
Power on reset with external clock	200 µs + clock stability time
Powered reset	200 µs

Wakeup Pins

The 8051 puts itself and the rest of the chip into a power-down mode by setting PCON.0 = 1. This stops the oscillator and PLL. When WAKEUP is asserted by external logic, the oscillator restarts, after the PLL stabilizes, and then the 8051 receives a wakeup interrupt. This applies whether or not FX2LP18 is connected to the USB.

The FX2LP18 exits the power down (USB suspend) state using one of the following methods:

- USB bus activity (if D+/D– lines are left floating, noise on these lines may indicate activity to the FX2LP18 and initiate a wakeup)
- External logic asserts the WAKEUP pin
- External logic asserts the PA3/WU2 pin

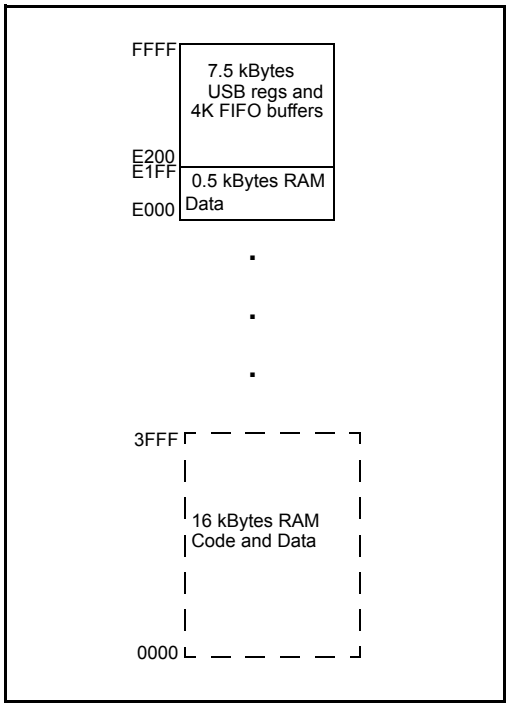
The second wakeup pin, WU2, can also be configured as a general purpose I/O pin. This allows a simple external R-C network to be used as a periodic wakeup source. Note that WAKEUP is active LOW by default.

Lowering Suspend Current

Good design practices for CMOS circuits dictate that any unused input pins must not be floating between V_{IL} and V_{IH}. Floating input pins will not damage the chip, but can substantially increase suspend current. To achieve the lowest suspend current, configure unused port pins as outputs. Connect unused input pins to ground. Some examples of pins that need attention during suspend are:

- Port pins. For Port A, B, D pins, take extra care in shared bus situations.
  - Connect completely unused pins to V_{CC_IO} or GND.
  - In a single-master system, the firmware must output enable all the port pins and drive them high or low, before FX2LP18 enters the suspend state.
  - In a multi-master system (FX2LP18 and another processor sharing a common data bus), when FX2LP18 is suspended, the external master must drive the pins high or low. The external master must not let the pins float.
- CLKOUT. If CLKOUT is not used, it must be tri-stated during normal operation, but driven during suspend.
- IFCLK, RDY0, RDY1. These pins must be pulled to V_{CC_IO} or GND or driven by another chip.
- CTL0-2. If tri-stated via GPIFIDLECTL, these pins must be pulled to V_{CC_IO} or GND or driven by another chip.
- RESET#, WAKEUP#. These pins must be pulled to V_{CC_IO} or GND or driven by another chip during suspend.

Figure 3. FX2LP18 Internal Code Memory



Note
2. If the external clock is powered at the same time as the CY7C680xx and has a stabilization wait period, it must be added to the 200 µs.

Program/Data RAM

This section describes the FX2LP18 RAM.

Size

The FX2LP18 has 16 kBytes of internal program/data RAM. No USB control registers appear in this space.

Memory maps are shown in Figure 3 and Figure 4.

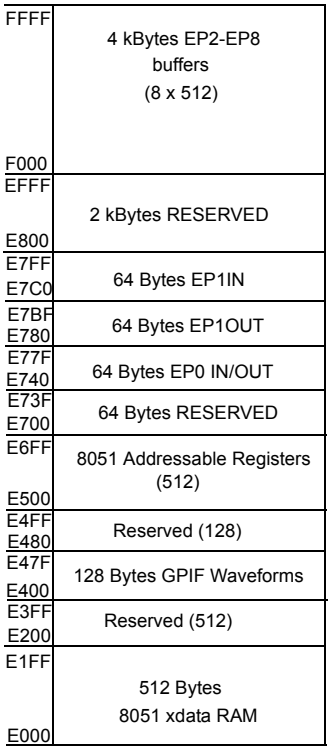
Internal Code Memory

This mode implements the internal 16-kByte block of RAM (starting at 0) as combined code and data memory. Only the **internal** 16 kBytes and **scratch pad** 0.5 kBytes RAM spaces have the following access:

- USB download
- USB upload
- Setup data pointer
- I^2C interface boot load

Register Addresses

Figure 4. Register Address Memory



Endpoint RAM

This section describes the FX2LP18 Endpoint RAM.

Size

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

Organization

- EP0
- Bidirectional endpoint zero, 64-byte buffer
- EP1IN, EP1OUT
- 64-byte buffers: bulk or interrupt
- EP2, 4, 6, 8
- Eight 512-byte buffers: bulk, interrupt, or isochronous. EP4 and EP8 can be double buffered, while EP2 and 6 can be double, triple, or quad buffered. For high speed endpoint configuration options, see Figure 5 on page 9.

Setup Data Buffer

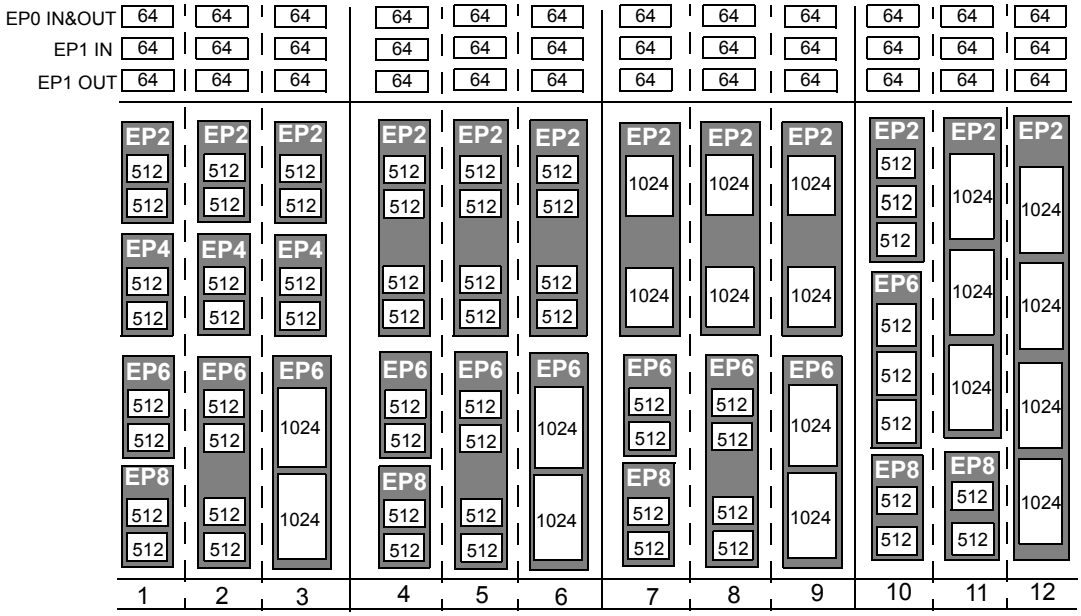
A separate 8-byte buffer at 0xE6B8-0xE6BF holds the setup data from a CONTROL transfer.

Endpoint Configurations (High Speed Mode)

Endpoints 0 and 1 are the same for every configuration. Endpoint 0 is the only CONTROL endpoint, and endpoint 1 can be either BULK or INTERRUPT. The endpoint buffers can be configured in any one of the 12 configurations shown in the vertical columns of Figure 5. When operating in full speed BULK mode only the first 64 bytes of each buffer are used. For example, in high speed the maximum packet size is 512 bytes, but in full speed it is 64 bytes. Even though a buffer is configured to be a 512 byte buffer, in full speed only the first 64 bytes are used. The unused endpoint buffer space is not available for other operations. An example endpoint configuration is:

EP2–1024 double buffered; EP6–512 quad buffered (column 8).

Figure 5. Endpoint Configuration



Default Full Speed Alternate Settings

Table 4. Default Full Speed Alternate Settings[3, 4]

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 5. Default High Speed Alternate Settings[3, 4]

Alternate Setting	0	1	2	3
ep0	64	64	64	64
ep1out	0	512 bulk[5]	64 int	64 int
ep1in	0	512 bulk[5]	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×)

Notes
3. '0' means 'not implemented.'
4. '2×' means 'double buffered.'
5. Even though these buffers are 64 bytes, they are reported as 512 for USB 2.0 compliance. Nnever transfer packets larger than 64 bytes to EP1.

External FIFO Interface

The architecture, control signals, and clock rates are presented in this section.

Architecture

The FX2LP18 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 or the slave FIFO interface for externally controlled transfers.

Master/Slave Control Signals

The FX2LP18 endpoint FIFOs are implemented as eight physically distinct 256x16 RAM blocks. The 8051/SIE can switch any of the RAM blocks between two domains, the USB (SIE) domain and the 8051-I/O Unit domain. This switching is instantaneous, giving 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 given time, some RAM blocks are filling and emptying with USB data under SIE control, while other RAM blocks are available to the 8051, the I/O control unit, or both. The RAM blocks operate as single port in the USB domain, and dual port in the 8051-I/O domain. The blocks can be configured as single, double, triple, or quad buffered as previously shown.

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

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

In Slave (S) mode, the FX2LP18 accepts either an internally derived clock or externally supplied clock (IFCLK, maximum frequency 48 MHz) and SLCS#, SLRD, SLWR, SLOE, PKTEND signals from external logic. When using an external IFCLK, the external clock must be present before switching to the external clock with the IFCLKSRC bit. Each endpoint can individually be selected for byte or word operation by an internal configuration bit, and a Slave FIFO Output Enable signal (SLOE) enables data of the selected width. External logic must insure that the output enable signal is inactive when writing data to a slave FIFO. The slave interface can also operate asynchronously, where the SLRD and SLWR signals act directly as strobes, rather than a clock qualifier as in synchronous mode. The signals SLRD, SLWR, SLOE, and PKTEND are gated by the signal SLCS#.

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. Another bit within the IFCONFIG register inverts the IFCLK signal whether internally or externally sourced.

GPIF

The GPIF is a flexible 8- or 16-bit parallel interface driven by a user programmable finite state machine. It allows the CY7C68053 to perform local bus mastering, and can implement a wide variety of protocols such as ATA interface, parallel printer port, and Utopia.

The GPIF has three programmable control outputs (CTL), and two general purpose ready inputs.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, and so on. A sequence of the GPIF vectors makes up a single waveform that is executed to perform the desired data move between the FX2LP18 and the external device.

Three Control OUT Signals

The 56-pin package brings out three of these signals, CTL0–CTL2. The 8051 programs the GPIF unit to define the CTL waveforms. CTLx waveform edges can be programmed to make transitions as fast as once per clock cycle (20.8 ns using a 48 MHz clock).

Two Ready IN Signals

The FX2LP18 package brings out all two Ready inputs (RDY0–RDY1). The 8051 programs the GPIF unit to test the RDY pins for GPIF branching.

Long Transfer Mode

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

ECC Generation[6]

The MoBL-USB can calculate Error Correcting Codes (ECCs) 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 1-bit error or detect any 2-bit error.

ECC Implementation

The two ECC configurations are selected by the ECCM bit.

ECCM = 0

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

This configuration writes any value to ECCRESET, then passes 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 is calculated over a 512-byte block of data.

This configuration writes any value to ECCRESET then passes 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 value in ECC1 does not change until ECCRESET is written again, even if more data is subsequently passed across the interface.

USB Uploads and Downloads

The core has the ability to directly edit the data contents of the internal 16-kByte RAM and of the internal 512-byte scratch pad RAM using a vendor-specific command. This capability is normally used when ‘soft’ downloading user code and is available only to and from internal RAM, only when the 8051 is held in reset. The available RAM spaces are 16 kBytes from 0x0000–0x3FFF (code/data) and 512 bytes from 0xE000–0xE1FF (scratch pad data RAM).[7]

Autopointer Access

FX2LP18 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. The autopointers are available in external FX2LP18 registers, under control of a mode bit (AUTOPTRSET-UP.0). Using the external FX2LP18 autopointer access (at 0xE67B – 0xE67C) allows the autopointer to access all RAM. Also, the autopointers can point to any FX2LP18 register or endpoint buffer space.

I²C Controller

FX2LP18 has one I²C port that is driven by two internal controllers. One controller automatically operates at boot time to load the VID/PID/DID, configuration byte, and firmware. The second controller is used by the 8051, once 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 2.2K ohm pull up resistors even if no EEPROM is connected to the FX2LP18. The value of the pull up resistors required may vary, depending on the combination of V_{CC_IO} and the supply used for the EEPROM. The pull up resistors used must be such that when the EEPROM pulls SDA low, the voltage level meets the V_{IL} specification of the FX2LP18. For example, if the EEPROM runs off a 3.3 V supply and V_{CC_IO} is 1.8 V, the pull up resistors recommended are 10K ohm. This requirement may also vary depending on the devices being run on the I²C pins. Refer to the I²C specifications for details.

External EEPROM device address pins must be configured properly. See Table 6 on page 11 for configuring the device address pins.

If no EEPROM is connected to the I²C port, EEPROM emulation is required by an external processor. This is because the FX2LP18 comes out of reset with the DISCON bit set, so the device will not enumerate without an EEPROM (C2 load) or EEPROM emulation.

Table 6. Strap Boot EEPROM Address Lines to these Values

Bytes	Example EEPROM	A2	A1	A0
16	24AA00[8]	N/A	N/A	N/A
128	24AA01	0	0	0
256	24AA02	0	0	0
4K	24AA32	0	0	1
8K	24AA64	0	0	1
16K	24AA128	0	0	1

I²C Interface Boot Load Access

At power on reset the I²C interface boot loader loads the VID/PID/DID and configuration bytes and up to 16 kBytes of program/data. The available RAM spaces are 16 kBytes from 0x0000–0x3FFF and 512 bytes from 0xE000–0xE1FF. The 8051 is 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. FX2LP18 provides I²C master control only, it is never an I²C slave.

Notes

6. To use the ECC logic, the GPIF or Slave FIFO interface must be configured for byte-wide operation.
7. After the data is downloaded from the host, a ‘loader’ can execute from internal RAM in order to transfer downloaded data to external memory.
8. This EEPROM does not have address pins.

Pin Assignments

Figure 6 identifies all signals for the package. It is followed by the pin diagram.Three modes are available: Port, GPIF master, and Slave FIFO. These modes define the signals on the right edge of the diagram. The 8051 selects the interface mode using the IFCONFIG[1:0] register bits. Port mode is the power on default configuration.

Figure 6. Signals

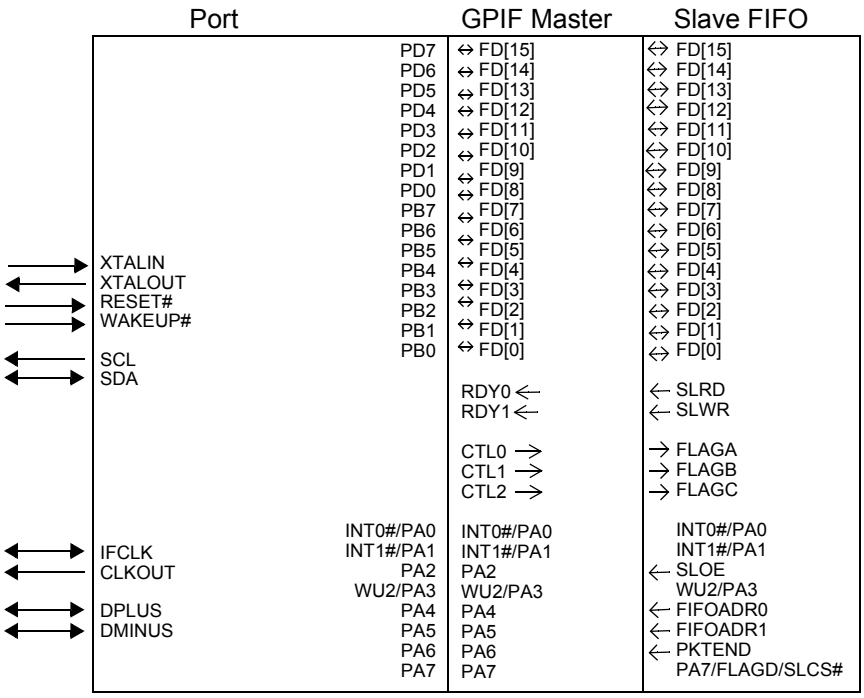
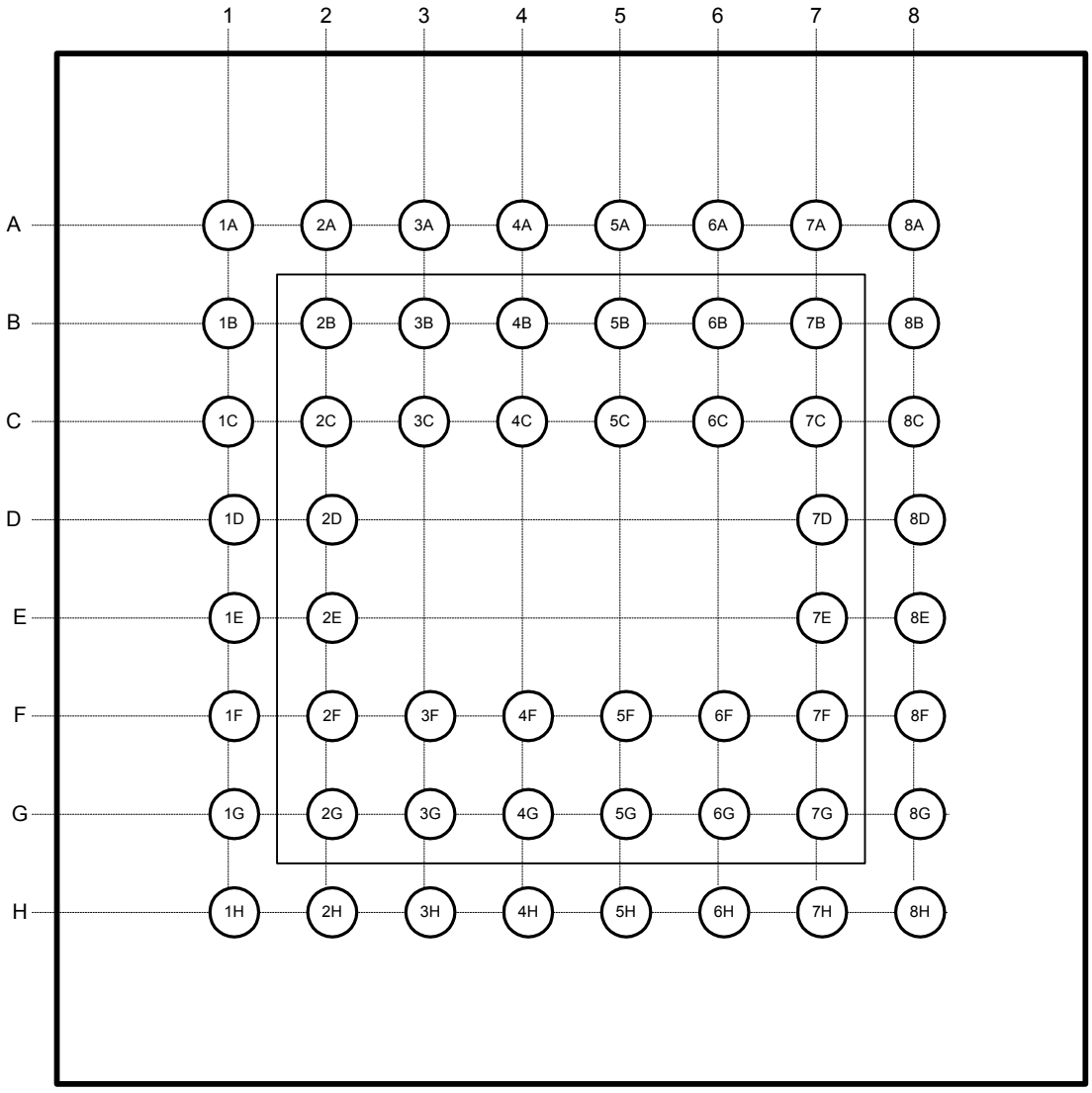


Figure 7. CY7C68053 56-pin VFBGA Pin Assignment - Top View



CY7C68053 Pin Descriptions

Table 7. FX2LP18 Pin Descriptions [9]

56 VFBGA	Name	Type	Default	Description
2D	AV_{CC}	Power	N/A	**Analog VCC**. Connect this pin to 3.3 V power source. This signal provides power to the analog section of the chip. Provide an appropriate bulk/bypass capacitance for this supply rail.
1D	AV_{CC}	Power	N/A	**Analog VCC**. Connect this pin to 3.3 V power source. This signal provides power to the analog section of the chip.
2F	AGND	Ground	N/A	**Analog Ground**. Connect this pin to ground with as short a path as possible.
1F	AGND	Ground	N/A	**Analog Ground**. Connect to this pin ground with as short a path as possible.
1E	DMINUS	I/O/Z	Z	**USB D– Signal**. Connect this pin to the USB D– signal.
2E	DPLUS	I/O/Z	Z	**USB D+ Signal**. Connect this pin to the USB D+ signal.
8B	RESET#	Input	N/A	**Active LOW Reset**. This pin resets the entire chip. See Reset and Wakeup on page 7 for details.
1C	XTALIN	Input	N/A	**Crystal Input**. Connect this signal to a 24 MHz parallel resonant, fundamental mode crystal and load capacitor to GND. It is also correct to drive XTALIN with an external 24 MHz square wave derived from another clock source.
2C	XTALOUT	Output	N/A	**Crystal Output**. Connect this signal to a 24 MHz parallel resonant, fundamental mode crystal and load capacitor to GND. If an external clock is used to drive XTALIN, leave this pin open.
2B	CLKOUT	O/Z	12 MHz	**CLKOUT.** 12, 24, or 48 MHz clock, phase locked to the 24 MHz input clock. The 8051 defaults to 12 MHz operation. The 8051 may tri-state this output by setting CPUCS.1 = 1.
Port A				
8G	PA0 or INT0#	I/O/Z	I (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).
6G	PA1 or INT1#	I/O/Z	I (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).
8F	PA2 or SLOE	I/O/Z	I (PA2)	Multiplexed pin whose function is selected by two bits: IFCONFIG[1:0]. **PA2** is a bidirectional I/O port pin. **SLOE** is an input-only output enable with programmable polarity (FIFOPINPOLAR.4) for the slave FIFO's connected to FD[7:0] or FD[15:0].
7F	PA3 or WU2	I/O/Z	I (PA3)	Multiplexed pin whose function is selected by: WAKEUP.7 and OEA.3 **PA3** is a bidirectional I/O port pin. **WU2** is an alternate source for **USB Wakeup,** enabled by WU2EN bit (WAKEUP.1) and polarity set by WU2POL (WAKEUP.4). If the 8051 is in suspend and WU2EN = 1, a transition on this pin starts up the oscillator and interrupts the 8051 to allow it to exit the suspend mode. Asserting this pin inhibits the chip from suspending, if WU2EN = 1.

Note

9. Do not leave unused inputs floating. Tie either HIGH or LOW as appropriate. Only pull outputs up or down to ensure signals at power up and in standby. Do not drive any pins while the device is powered down.

Table 7. FX2LP18 Pin Descriptions (continued)[9]

56 VFBGA	Name	Type	Default	Description
6F	PA4 or FIFOADR0	I/O/Z	I (PA4)	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **PA4** is a bidirectional I/O port pin. **FIFOADR0** is an input-only address select for the slave FIFOs connected to FD[7:0] or FD[15:0].
8C	PA5 or FIFOADR1	I/O/Z	I (PA5)	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **PA5** is a bidirectional I/O port pin. **FIFOADR1** is an input-only address select for the slave FIFOs connected to FD[7:0] or FD[15:0].
7C	PA6 or PKTEND	I/O/Z	I (PA6)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] bits. **PA6** is a bidirectional I/O port pin. **PKTEND** is an input that commits the FIFO packet data to the endpoint and whose polarity is programmable using FIFOPINPOLAR.5.
6C	PA7 or FLAGD or SLCS#	I/O/Z	I (PA7)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] and PORTACFG.7 bits. **PA7** is a bidirectional I/O port pin. **FLAGD** is a programmable slave FIFO output status flag signal. **SLCS#** gates all other slave FIFO enable/strobes
Port B				
3H	PB0 or FD[0]	I/O/Z	I (PB0)	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **PB0** is a bidirectional I/O port pin. **FD[0]** is the bidirectional FIFO/GPIF data bus.
4F	PB1 or FD[1]	I/O/Z	I (PB1)	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **PB1** is a bidirectional I/O port pin. **FD[1]** is the bidirectional FIFO/GPIF data bus.
4H	PB2 or FD[2]	I/O/Z	I (PB2)	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **PB2** is a bidirectional I/O port pin. **FD[2]** is the bidirectional FIFO/GPIF data bus.
4G	PB3 or FD[3]	I/O/Z	I (PB3)	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **PB3** is a bidirectional I/O port pin. **FD[3]** is the bidirectional FIFO/GPIF data bus.
5H	PB4 or FD[4]	I/O/Z	I (PB4)	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **PB4** is a bidirectional I/O port pin. **FD[4]** is the bidirectional FIFO/GPIF data bus.
5G	PB5 or FD[5]	I/O/Z	I (PB5)	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **PB5** is a bidirectional I/O port pin. **FD[5]** is the bidirectional FIFO/GPIF data bus.
5F	PB6 or FD[6]	I/O/Z	I (PB6)	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **PB6** is a bidirectional I/O port pin. **FD[6]** is the bidirectional FIFO/GPIF data bus.
6H	PB7 or FD[7]	I/O/Z	I (PB7)	Multiplexed pin whose function is selected IFCONFIG[1:0]. **PB7** is a bidirectional I/O port pin. **FD[7]** is the bidirectional FIFO/GPIF data bus.

Table 7. FX2LP18 Pin Descriptions (continued)[9]

56 VFBGA	Name	Type	Default	Description
PORT D				
8A	PD0 or FD[8]	I/O/Z	I (PD0)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFIFOCFG.0 (wordwide) bits. **FD[8]** is the bidirectional FIFO/GPIF data bus.
7A	PD1 or FD[9]	I/O/Z	I (PD1)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFIFOCFG.0 (wordwide) bits. **FD[9]** is the bidirectional FIFO/GPIF data bus.
6B	PD2 or FD[10]	I/O/Z	I (PD2)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFIFOCFG.0 (wordwide) bits. **FD[10]** is the bidirectional FIFO/GPIF data bus.
6A	PD3 or FD[11]	I/O/Z	I (PD3)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFIFOCFG.0 (wordwide) bits. **FD[11]** is the bidirectional FIFO/GPIF data bus.
3B	PD4 or FD[12]	I/O/Z	I (PD4)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFIFOCFG.0 (wordwide) bits. **FD[12]** is the bidirectional FIFO/GPIF data bus.
3A	PD5 or FD[13]	I/O/Z	I (PD5)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFIFOCFG.0 (wordwide) bits. **FD[13]** is the bidirectional FIFO/GPIF data bus.
3C	PD6 or FD[14]	I/O/Z	I (PD6)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFIFOCFG.0 (wordwide) bits. **FD[14]** is the bidirectional FIFO/GPIF data bus.
2A	PD7 or FD[15]	I/O/Z	I (PD7)	Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFIFOCFG.0 (wordwide) bits. **FD[15]** is the bidirectional FIFO/GPIF data bus.
1A	RDY0 or SLRD	Input	N/A	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **RDY0** is a GPIF input signal. **SLRD** is the input only read strobe with programmable polarity (FIFOPIN-POLAR.3) for the slave FIFOs connected to FD[7:0] or FD[15:0].
1B	RDY1 or SLWR	Input	N/A	Multiplexed pin whose function is selected by IFCONFIG[1:0]. **RDY1** is a GPIF input signal. **SLWR** is the input only write strobe with programmable polarity (FIFOPIN-POLAR.2) for the slave FIFOs connected to FD[7:0] or FD[15:0].
7H	CTL0 or FLAGA	O/Z	H	Multiplexed pin whose function is selected by 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.
7G	CTL1 or FLAGB	O/Z	H	Multiplexed pin whose function is selected by 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.
8H	CTL2 or FLAGC	O/Z	H	Multiplexed pin whose function is selected 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.
2G	IFCLK	I/O/Z	Z	Interface clock, used to synchronous clock 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 or 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.

Table 7. FX2LP18 Pin Descriptions (continued)[9]

56 VFBGA	Name	Type	Default	Description
7B	WAKEUP	Input	N/A	**USB Wakeup**. If the 8051 is in suspend, asserting this pin starts up the oscillator and interrupts the 8051 to allow it to exit the suspend mode. Holding WAKEUP asserted inhibits the MoBL-USB® chip from suspending. This pin has programmable polarity (WAKEUP.4).
3F	SCL	OD	Z	**Clock** for the I^2C interface. Connect to V_{CC_IO} or V_{CC} with a 2.2K–10K pull up resistor. (An I^2C peripheral is required.)
3G	SDA	OD	Z	**Data** for the I^2C interface. Connect to V_{CC_IO} or V_{CC} with a 2.2K–10K pull up resistor. (An I^2C peripheral is required.)
5A	V_{CC_IO}	Power	N/A	**VCC**. Connect this pin to 1.8V to 3.3 V power source. Provide the appropriate bulk and bypass capacitance for this supply rail.
5B	V_{CC_IO}	Power	N/A	**VCC**. Connect this pin to 1.8V to 3.3 V power source.
7E	V_{CC_IO}	Power	N/A	**VCC**. Connect this pin to 1.8 V to 3.3 V power source.
8E	V_{CC_IO}	Power	N/A	**VCC**. Connect this pin to 1.8V to 3.3 V power source.
5C	V_{CC_D}	Power	N/A	**VCC**. Connect this pin to 1.8V power source. (Supplies power to internal digital 1.8 V circuits.) Provide the appropriate bulk and bypass capacitance for this supply rail.
1G	V_{CC_A}	Power	N/A	**VCC**. Connect this pin to 1.8V power source. (Supplies power to internal analog 1.8 V circuits.)
1H	GND	Ground	N/A	**Ground**.
2H	GND	Ground	N/A	**Ground**.
4A	GND	Ground	N/A	**Ground**.
4B	GND	Ground	N/A	**Ground**.
4C	GND	Ground	N/A	**Ground**.
7D	GND	Ground	N/A	**Ground**.
8D	GND	Ground	N/A	**Ground**.

Register Summary

FX2LP18 register bit definitions are described in the *MoBL-USB FX2LP18 TRM* in greater detail.

Table 8. FX2LP18 Register Summary

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
		GPIF Waveform Memories											
E400	128	WAVEDATA	GPIF Waveform descriptor 0, 1, 2, 3 data	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E480	128	Reserved											
		GENERAL CONFIGURATION											
E50D		GPCR2	General Purpose Configuration Register 2	Reserved	Reserved	Reserved	FULL_SPEED_ONLY	Reserved	Reserved	Reserved	Reserved	00000000	R
E600	1	CPUCS	CPU Control and 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[10]	Slave FIFO FLAGA and FLAGB pin configuration	FLAGB3	FLAGB2	FLAGB1	FLAGB0	FLAGA3	FLAGA2	FLAGA1	FLAGA0	00000000	RW
E603	1	PINFLAGSCD[10]	Slave FIFO FLAGC and FLAGD pin configuration	FLAGD3	FLAGD2	FLAGD1	FLAGD0	FLAGC3	FLAGC2	FLAGC1	FLAGC0	00000000	RW
E604	1	FIFORESET[10]	Restore FIFOs to default state	NAKALL	0	0	0	EP3	EP2	EP1	EP0	xxxxxxxx	W
E605	1	BREAKPT	Breakpoint control	0	0	0	0	BREAK	BPPULSE	BPEN	0	00000000	rrrrbbbr
E606	1	BPADDRH	Breakpoint address H	A15	A14	A13	A12	A11	A10	A9	A8	xxxxxxxx	RW
E607	1	BPADDRL	Breakpoint address L	A7	A6	A5	A4	A3	A2	A1	A0	xxxxxxxx	RW
E608	1	Reserved	Reserved	0	0	0	0	0	0	0	0	00000000	rrrrrrbb
E609	1	FIFOPINPOLAR[10]	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[10]	Chip revision control	0	0	0	0	0	0	dyn_out	enh_pkt	00000000	rrrrrrbb
		UDMA											
E60C	1	GPIFHOLDAMOUNT	MSTB hold time (for UDMA)	0	0	0	0	0	0	HOLDTIME1	HOLDTIME0	00000000	rrrrrrbb
	3	Reserved											
		ENDPOINT CONFIGURATION											
E610	1	EP1OUTCFG	Endpoint 1-OUT configuration	VALID	0	TYPE1	TYPE0	0	0	0	0	10100000	brbbrrrr
E611	1	EP1INCFG	Endpoint 1-IN configuration	VALID	0	TYPE1	TYPE0	0	0	0	0	10100000	brbbrrrr
E612	1	EP2CFG	Endpoint 2 configuration	VALID	DIR	TYPE1	TYPE0	SIZE	0	BUF1	BUF0	10100010	bbbbbrbb
E613	1	EP4CFG	Endpoint 4 configuration	VALID	DIR	TYPE1	TYPE0	0	0	0	0	10100000	bbbbrrrr
E614	1	EP6CFG	Endpoint 6 configuration	VALID	DIR	TYPE1	TYPE0	SIZE	0	BUF1	BUF0	11100010	bbbbbrbb
E615	1	EP8CFG	Endpoint 8 configuration	VALID	DIR	TYPE1	TYPE0	0	0	0	0	11100000	bbbbrrrr
	2	Reserved											
E618	1	EP2FIFOCFG[10]	Endpoint 2/Slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E619	1	EP4FIFOCFG[10]	Endpoint 4/Slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61A	1	EP6FIFOCFG[10]	Endpoint 6/Slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61B	1	EP8FIFOCFG[10]	Endpoint 8/Slave FIFO configuration	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61C	4	Reserved											
E620	1	EP2AUTOINLENH[10]	Endpoint 2 AUTOIN packet length H	0	0	0	0	0	PL10	PL9	PL8	00000010	rrrrrbbb
E621	1	EP2AUTOINLENL[10]	Endpoint 2 AUTOIN packet length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E622	1	EP4AUTOINLENH[10]	Endpoint 4 AUTOIN packet length H	0	0	0	0	0	0	PL9	PL8	00000010	rrrrrrbb
E623	1	EP4AUTOINLENL[10]	Endpoint 4 AUTOIN packet length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E624	1	EP6AUTOINLENH[10]	Endpoint 6 AUTOIN packet length H	0	0	0	0	0	PL10	PL9	PL8	00000010	rrrrrbbb
E625	1	EP6AUTOINLENL[10]	Endpoint 6 AUTOIN packet length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E626	1	EP8AUTOINLENH[10]	Endpoint 8 AUTOIN packet length H	0	0	0	0	0	0	PL9	PL8	00000010	rrrrrrbb
E627	1	EP8AUTOINLENL[10]	Endpoint 8 AUTOIN packet length L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E628	1	ECCCFG	ECC Configuration	0	0	0	0	0	0	0	ECCM	00000000	rrrrrrrb
E629	1	ECCRESET	ECC Reset	x	x	x	x	x	x	x	x	00000000	W
E62A	1	ECC1B0	ECC1 Byte 0 address	LINE15	LINE14	LINE13	LINE12	LINE11	LINE10	LINE9	LINE8	00000000	R
E62B	1	ECC1B1	ECC1 Byte 1 address	LINE7	LINE6	LINE5	LINE4	LINE3	LINE2	LINE1	LINE0	00000000	R

Note

10. Read and writes to these registers may require synchronization delay, see *MoBL-USB FX2LP18 Technical Reference Manual* for ‘Synchronization Delay.’

Table 8. FX2LP18 Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
E62C	1	ECC1B2	ECC1 Byte 2 address	COL5	COL4	COL3	COL2	COL1	COL0	LINE17	LINE16	00000000	R
E62D	1	ECC2B0	ECC2 Byte 0 address	LINE15	LINE14	LINE13	LINE12	LINE11	LINE10	LINE9	LINE8	00000000	R
E62E	1	ECC2B1	ECC2 Byte 1 address	LINE7	LINE6	LINE5	LINE4	LINE3	LINE2	LINE1	LINE0	00000000	R
E62F	1	ECC2B2	ECC2 Byte 2 address	COL5	COL4	COL3	COL2	COL1	COL0	0	0	00000000	R
E630 H.S.	1	EP2FIFOPFH[10]	Endpoint 2/Slave FIFO programmable flag H	DECIS	PKTSTAT	IN:PKTS[2] OUT:PFC12	IN:PKTS[1] OUT:PFC11	IN:PKTS[0] OUT:PFC10	0	PFC9	PFC8	10001000	bbbbbrbb
E630 F.S.	1	EP2FIFOPFH[10]	Endpoint 2/Slave FIFO programmable flag H	DECIS	PKTSTAT	OUT:PFC12	OUT:PFC11	OUT:PFC10	0	PFC9	IN:PKTS[2] OUT:PFC8	10001000	bbbbbrbb
E631 H.S.	1	EP2FIFOPFL[10]	Endpoint 2/Slave FIFO programmable flag L	PFC7	PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E631 F.S	1	EP2FIFOPFL[10]	Endpoint 2/Slave FIFO programmable flag L	IN:PKTS[1] OUT:PFC7	IN:PKTS[0] OUT:PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E632 H.S.	1	EP4FIFOPFH[10]	Endpoint 4/Slave FIFO programmable flag H	DECIS	PKTSTAT	0	IN: PKTS[1] OUT:PFC10	IN: PKTS[0] OUT:PFC9	0	0	PFC8	10001000	bbrbbrrb
E632 F.S	1	EP4FIFOPFH[10]	Endpoint 4/Slave FIFO programmable flag H	DECIS	PKTSTAT	0	OUT:PFC10	OUT:PFC9	0	0	PFC8	10001000	bbrbbrrb
E633 H.S.	1	EP4FIFOPFL[10]	Endpoint 4/Slave FIFO programmable flag L	PFC7	PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E633 F.S	1	EP4FIFOPFL[10]	Endpoint 4/Slave FIFO programmable flag L	IN: PKTS[1] OUT:PFC7	IN: PKTS[0] OUT:PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E634 H.S.	1	EP6FIFOPFH[10]	Endpoint 6/Slave FIFO programmable flag H	DECIS	PKTSTAT	IN:PKTS[2] OUT:PFC12	IN:PKTS[1] OUT:PFC11	IN:PKTS[0] OUT:PFC10	0	PFC9	PFC8	00001000	bbbbbrbb
E634 F.S	1	EP6FIFOPFH[10]	Endpoint 6/Slave FIFO programmable flag H	DECIS	PKTSTAT	OUT:PFC12	OUT:PFC11	OUT:PFC10	0	PFC9	IN:PKTS[2] OUT:PFC8	00001000	bbbbbrbb
E635 H.S.	1	EP6FIFOPFL[10]	Endpoint 6/Slave FIFO programmable flag L	PFC7	PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E635 F.S	1	EP6FIFOPFL[10]	Endpoint 6/Slave FIFO programmable flag L	IN:PKTS[1] OUT:PFC7	IN:PKTS[0] OUT:PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E636 H.S.	1	EP8FIFOPFH[10]	Endpoint 8/Slave FIFO programmable flag H	DECIS	PKTSTAT	0	IN: PKTS[1] OUT:PFC10	IN: PKTS[0] OUT:PFC9	0	0	PFC8	00001000	bbrbbrrb
E636 F.S	1	EP8FIFOPFH[10]	Endpoint 8/Slave FIFO programmable flag H	DECIS	PKTSTAT	0	OUT:PFC10	OUT:PFC9	0	0	PFC8	00001000	bbrbbrrb
E637 H.S.	1	EP8FIFOPFL[10]	Endpoint 8/Slave FIFO programmable flag L	PFC7	PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E637 F.S	1	EP8FIFOPFL[10]	Endpoint 8/Slave FIFO programmable flag L	IN: PKTS[1] OUT:PFC7	IN: PKTS[0] OUT:PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
	8	Reserved											
E640	1	EP2ISOINPKTS	EP2 (if ISO) IN packets per frame (1-3)	AADJ	0	0	0	0	0	INPPF1	INPPF0	00000001	brrrrrbb
E641	1	EP4ISOINPKTS	EP4 (if ISO) IN packets per frame (1-3)	AADJ	0	0	0	0	0	INPPF1	INPPF0	00000001	brrrrrrr
E642	1	EP6ISOINPKTS	EP6 (if ISO) IN packets per frame (1-3)	AADJ	0	0	0	0	0	INPPF1	INPPF0	00000001	brrrrrbb
E643	1	EP8ISOINPKTS	EP8 (if ISO) IN packets per frame (1-3)	AADJ	0	0	0	0	0	INPPF1	INPPF0	00000001	brrrrrrr
E644	4	Reserved											
E648	1	INPKTEND[10]	Force IN packet end	Skip	0	0	0	EP3	EP2	EP1	EP0	xxxxxxxx	W
E649	7	OUTPKTEND[10]	Force OUT packet end	Skip	0	0	0	EP3	EP2	EP1	EP0	xxxxxxxx	W
		INTERRUPTS											
E650	1	EP2FIFOIE[10]	Endpoint 2 Slave FIFO flag interrupt enable	0	0	0	0	EDGEPF	PF	EF	FF	00000000	RW
E651	1	EP2FIFOIRQ[10,11]	Endpoint 2 Slave FIFO flag interrupt request	0	0	0	0	0	PF	EF	FF	00000000	rrrrrbbb
E652	1	EP4FIFOIE[10]	Endpoint 4 Slave FIFO flag interrupt enable	0	0	0	0	EDGEPF	PF	EF	FF	00000000	RW
E653	1	EP4FIFOIRQ[10,11]	Endpoint 4 Slave FIFO flag interrupt request	0	0	0	0	0	PF	EF	FF	00000000	rrrrrbbb
E654	1	EP6FIFOIE[10]	Endpoint 6 Slave FIFO flag interrupt enable	0	0	0	0	EDGEPF	PF	EF	FF	00000000	RW
E655	1	EP6FIFOIRQ[10,11]	Endpoint 6 Slave FIFO flag interrupt request	0	0	0	0	0	PF	EF	FF	00000000	rrrrrbbb
E656	1	EP8FIFOIE[10]	Endpoint 8 Slave FIFO flag interrupt enable	0	0	0	0	EDGEPF	PF	EF	FF	00000000	RW
E657	1	EP8FIFOIRQ[10,11]	Endpoint 8 Slave FIFO flag interrupt request	0	0	0	0	0	PF	EF	FF	00000000	rrrrrbbb
E658	1	IBNIE	IN-BULK-NAK interrupt enable	0	0	EP8	EP6	EP4	EP2	EP1	EP0	00000000	RW
E659	1	IBNIRQ[11]	IN-BULK-NAK interrupt request	0	0	EP8	EP6	EP4	EP2	EP1	EP0	00xxxxxx	rrbbbbbb
E65A	1	NAKIE	Endpoint Ping-NAK/IBN interrupt enable	EP8	EP6	EP4	EP2	EP1	EP0	0	IBN	00000000	RW
E65B	1	NAKIRQ[11]	Endpoint Ping-NAK/IBN interrupt request	EP8	EP6	EP4	EP2	EP1	EP0	0	IBN	xxxxxx0x	bbbbbbrb
E65C	1	USBIE	USB interrupt enables	0	EP0ACK	HSGRANT	URES	SUSP	SUTOK	SOF	SUDAV	00000000	RW
E65D	1	USBIRQ[11]	USB interrupt requests	0	EP0ACK	HSGRANT	URES	SUSP	SUTOK	SOF	SUDAV	0xxxxxxx	rbbbbbbb

Note
11. The register can only be reset, it cannot be set.

Table 8. FX2LP18 Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
E65E	1	EPIE	Endpoint interrupt enables	EP8	EP6	EP4	EP2	EP1OUT	EP1IN	EP0OUT	EP0IN	00000000	RW
E65F	1	EPIRQ[11]	Endpoint interrupt requests	EP8	EP6	EP4	EP2	EP1OUT	EP1IN	EP0OUT	EP0IN	0	RW
E660	1	GPIFIE[10]	GPIF interrupt enable	0	0	0	0	0	0	GPIFWF	GPIFDONE	00000000	RW
E661	1	GPIFIRQ[10]	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[11]	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	Reserved		1	0	0	0	0	0	0	0	10000000	R
E668	1	INTSET-UP	Interrupt 2 and 4 setup	0	0	0	0	AV2EN	0	Reserved	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	I2CS	I²C bus control and 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	I2CTL	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[10]	UDMA CRC MSB	CRC15	CRC14	CRC13	CRC12	CRC11	CRC10	CRC9	CRC8	01001010	RW
E67E	1	UDMACRCL[10]	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 and 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 and 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[10]	Endpoint 0 byte count H	(BC15)	(BC14)	(BC13)	(BC12)	(BC11)	(BC10)	(BC9)	(BC8)	xxxxxxxx	RW
E68B	1	EP0BCL[10]	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[10]	Endpoint 2 byte count H	0	0	0	0	BC10	BC9	BC8	00000xxx	RW
E691	1	EP2BCL[10]	Endpoint 2 byte count L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxxx	RW
E692	2	Reserved											
E694	1	EP4BCH[10]	Endpoint 4 byte count H	0	0	0	0	0	0	BC9	BC8	000000xx	RW
E695	1	EP4BCL[10]	Endpoint 4 byte count L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxxx	RW
E696	2	Reserved											
E698	1	EP6BCH[10]	Endpoint 6 byte count H	0	0	0	0	0	BC10	BC9	BC8	00000xxx	RW
E699	1	EP6BCL[10]	Endpoint 6 byte count L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxxx	RW
E69A	2	Reserved											
E69C	1	EP8BCH[10]	Endpoint 8 byte count H	0	0	0	0	0	0	BC9	BC8	000000xx	RW
E69D	1	EP8BCL[10]	Endpoint 8 byte count L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxxx	RW
E69E	2	Reserved											
E6A0	1	EP0CS	Endpoint 0 control and status	HSNAK	0	0	0	0	0	BUSY	STALL	10000000	bbbbbbrb
E6A1	1	EP1OUTCS	Endpoint 1 OUT control and status	0	0	0	0	0	0	BUSY	STALL	00000000	bbbbbbrb

Table 8. FX2LP18 Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
E6A2	1	EP1INCS	Endpoint 1 IN control and status	0	0	0	0	0	0	BUSY	STALL	00000000	bbbbbbrb
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	xxxxxxxx	RW
E6B4	1	SUDPTRL	Setup data pointer low address byte	A7	A6	A5	A4	A3	A2	A1	0	xxxxxxx0	bbbbbbbr
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	xxxxxxxx	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	0	0	0	CTL2	CTL1	CTL0	11111111	RW
E6C3	1	GPIFCTLCFG	CTL drive type	TRICTL	0	0	0	0	CTL2	CTL1	CTL0	00000000	RW
E6C4	1	Reserved										00000000	
E6C5	1	Reserved										00000000	
		FLOWSTATE											
E6C6	1	FLOWSTATE	Flowstate enable and selector	FSE	0	0	0	0	FS2	FS1	FS0	00000000	brrrrbbb
E6C7	1	FLOWLOGIC	Flowstate logic	LFUNC1	LFUNC0	TERMA2	TERMA1	TERMA0	TERMB2	TERMB1	TERMB0	00000000	RW
E6C8	1	FLOWEQ0CTL	CTL-pin states in flow state (when Logic = 0)	CTL0E3	CTL0E2	CTL0E1	CTL0E0	0	CTL2	CTL1	CTL0	00000000	RW
E6C9	1	FLOWEQ1CTL	CTL-pin states in flow state (when Logic = 1)	CTL0E3	CTL0E2	CTL0E1	CTL0E0	0	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	RDYASYNC	CTLTOGL	SUSTAIN	0	MSTB2	MSTB1	MSTB0	00100000	RW
E6CC	1	FLOWSTBEDGE	Flowstate rising/falling edge configuration	0	0	0	0	0	0	FALLING	RISING	00000001	rrrrrrbb
E6CD	1	FLOWSTBPERIOD	Master strobe half period	D7	D6	D5	D4	D3	D2	D1	D0	00000010	RW
E6CE	1	GPIFTCB3[10]	GPIF transaction count Byte 3	TC31	TC30	TC29	TC28	TC27	TC26	TC25	TC24	00000000	RW
E6CF	1	GPIFTCB2[10]	GPIF transaction count Byte 2	TC23	TC22	TC21	TC20	TC19	TC18	TC17	TC16	00000000	RW
E6D0	1	GPIFTCB1[10]	GPIF transaction count Byte 1	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8	00000000	RW
E6D1	1	GPIFTCB0[10]	GPIF transaction count Byte 0	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0	00000001	RW
	2	Reserved										00000000	RW
		Reserved											

Table 8. FX2LP18 Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
		Reserved											
E6D2	1	EP2GPIFFLGSEL[10]	Endpoint 2 GPIF flag select	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6D3	1	EP2GPIFPFSTOP	Endpoint 2 GPIF stop transaction on program flag	0	0	0	0	0	0	0	FIFO2FLAG	00000000	RW
E6D4	1	EP2GPIFTRIG[10]	Endpoint 2 GPIF trigger	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	Reserved											
		Reserved											
		Reserved											
E6DA	1	EP4GPIFFLGSEL[10]	Endpoint 4 GPIF flag select	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6DB	1	EP4GPIFPFSTOP	Endpoint 4 GPIF stop transaction on GPIF flag	0	0	0	0	0	0	0	FIFO4FLAG	00000000	RW
E6DC	1	EP4GPIFTRIG[10]	Endpoint 4 GPIF trigger	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	Reserved											
		Reserved											
		Reserved											
E6E2	1	EP6GPIFFLGSEL[10]	Endpoint 6 GPIF flag select	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6E3	1	EP6GPIFPFSTOP	Endpoint 6 GPIF stop transaction on program flag	0	0	0	0	0	0	0	FIFO6FLAG	00000000	RW
E6E4	1	EP6GPIFTRIG[10]	Endpoint 6 GPIF trigger	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	Reserved											
		Reserved											
		Reserved											
E6EA	1	EP8GPIFFLGSEL[10]	Endpoint 8 GPIF flag select	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6EB	1	EP8GPIFPFSTOP	Endpoint 8 GPIF stop transaction on program flag	0	0	0	0	0	0	0	FIFO8FLAG	00000000	RW
E6EC	1	EP8GPIFTRIG[10]	Endpoint 8 GPIF trigger	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	Reserved											
E6F0	1	XGPIFSGLDATH	GPIF Data H (16-bit mode only)	D15	D14	D13	D12	D11	D10	D9	D8	xxxxxxxx	RW
E6F1	1	XGPIFSGLDATLX	Read/Write GPIF Data L and trigger transaction	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E6F2	1	XGPIFSGLDATL-NOX	Read GPIF Data L, no transaction trigger	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	R
E6F3	1	GPIFREADYCFG	Internal RDY, sync/async, RDY pin states	INTRDY	SAS	TCXRDY5	0	0	0	0	0	00000000	bbbrrrrr
E6F4	1	GPIFREADYSTAT	GPIF ready status	0	0	0	0	0	0	RDY1	RDY0	00xxxxxx	R
E6F5	1	GPIFABORT	Abort GPIF waveforms	x	x	x	x	x	x	x	x	xxxxxxxx	W
E6F6	2	Reserved											
		ENDPOINT BUFFERS											
E740	64	EP0BUF	EP0-IN/-OUT buffer	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E780	64	EP10UTBUF	EP1-OUT buffer	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E7C0	64	EP1INBUF	EP1-IN buffer	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
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	xxxxxxxx	RW
F400	512	EP4FIFOBUF	512 byte EP 4/Slave FIFO buffer (IN or OUT)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	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	xxxxxxxx	RW
FC00	512	EP8FIFOBUF	512 byte EP 8/Slave FIFO buffer (IN or OUT)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
FE00	512	Reserved											
xxxx		I²C Configuration Byte		0	DISCON	0	0	0	0	0	400KHZ	xxxxxxxx [12]	n/a

Note
12. If no EEPROM is detected by the SIE then the default is 00000000.

Table 8. FX2LP18 Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
		Special Function Registers (SFRs)											
80	1	IOA[13]	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[13]	Data Pointer 1 L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
85	1	DPH1[13]	Data Pointer 1 H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
86	1	DPS[13]	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[13]	Clock control	x	x	T2M	T1M	T0M	MD2	MD1	MD0	00000001	RW
8F	1	Reserved											
90	1	IOB[13]	Port B (bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
91	1	EXIF[13]	External interrupt flags	IE5	IE4	I²CINT	USBNT	1	0	0	0	00001000	RW
92	1	MPAGE[13]	Upper address 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[13]	Autopointer 1 address H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
9B	1	AUTOPTRL1[13]	Autopointer 1 address L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
9C	1	Reserved											
9D	1	AUTOPTRH2[13]	Autopointer 2 address H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
9E	1	AUTOPTRL2[13]	Autopointer 2 address L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
9F	1	Reserved											
A0	1	IOC[13]	Port C (bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
A1	1	INT2CLR[13]	Interrupt 2 Clear	x	x	x	x	x	x	x	x	xxxxxxxx	W
A2	1	Reserved		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[13]	Endpoint 2,4,6,8 status flags	EP8F	EP8E	EP6F	EP6E	EP4F	EP4E	EP2F	EP2E	01011010	R
AB	1	EP24FIFOFLGS[13]	Endpoint 2,4 Slave FIFO status flags	0	EP4PF	EP4EF	EP4FF	0	EP2PF	EP2EF	EP2FF	00100010	R
AC	1	EP68FIFOFLGS[13]	Endpoint 6,8 Slave FIFO status flags	0	EP8PF	EP8EF	EP8FF	0	EP6PF	EP6EF	EP6FF	01100110	R
AD	2	Reserved											
AF	1	AUTOPTRSETUP[13]	Autopointer 1 and 2 Setup	0	0	0	0	0	APTR2INC	APTR1INC	APTREN	00000110	RW
B0	1	IOD[13]	Port D (bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
B1	1	IOE[13]	Port E (NOT bit addressable)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
B2	1	OEA[13]	Port A Output Enable	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B3	1	OEB[13]	Port B Output Enable	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B4	1	OEC[13]	Port C Output Enable	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B5	1	OED[13]	Port D Output Enable	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B6	1	OEE[13]	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[13]	Endpoint 0 and 1 Status	0	0	0	0	0	EP1INBSY	EP1OUTBSY	EP0BSY	00000000	R
BB	1	GPIFTRIG[13, 10]	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[13]	GPIF Data H (16-bit mode only)	D15	D14	D13	D12	D11	D10	D9	D8	xxxxxxxx	RW
BE	1	GPIFSGLDATLX[13]	GPIF Data L w/trigger	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
BF	1	GPIFSGLDATL-NOX[13]	GPIF Data L w/no trigger	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	R

Note
13. SFRs not part of the standard 8051 architecture.

Table 8. FX2LP18 Register Summary (continued)

Hex	Size	Name	Description	b7	b6	b5	b4	b3	b2	b1	b0	Default	Access
C0	1	SCON1[13]	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[13]	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[13]	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[13]	External Interrupt Enables	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[13]	External Interrupt Priority control	1	1	1	PX6	PX5	PX4	PI²C	PUSB	11100000	RW
F9	7	Reserved											

Ledgend
R = All bits read only
W = All bits write only
r = Read-only bit
w = Write-only bit
b = Both read/write bit

Absolute Maximum Ratings

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

Storage temperature ..... –65 °C to +150 °C

Ambient temperature with power supplied

Industrial ..... –40 °C to +85 °C

Supply voltage to ground potential

For 3.3 V power domain .....–0.5 V to +4.0 V

For 1.8 V power domain .....–0.5 V to +2.0 V

DC input voltage to any input pin

For pins under 3.3 V power domain ..... 3.6 V[14]

For pins under 1.8 V to 3.3 V Power Domain (GPIOs) .....1.89 V to 3.6 V[14]

(The GPIOs are not over voltage tolerant, except the SCL and SDA pins, which are 3.3 V tolerant)

DC voltage applied to outputs in high Z State ..... –0.5 V to VCC +0.5 V

Maximum power dissipation

From AVcc Supply ..... 90 mW

From I/O supply ..... 36 mW

From core supply ..... 95 mW

Static discharge voltage .....>2000 V

(I2C SCL and SDA pins only ..... >1500 V)

Maximum output current, per I/O port ..... 10 mA

Operating Conditions

T_A (Ambient temperature under bias)
industrial ..... –40 °C to +85 °C

Supply voltage

3.3 V Power Supply .....3.0 V to 3.6 V

1.8 V Power Supply .....1.71 V to1.89 V

Ground Voltage ..... 0 V

F_{OSC} (Oscillator or Crystal Frequency).... 24 MHz ± 100 ppm
..... Parallel Resonant
.....500 µW drive level
..... Load capacitors 12 pF

DC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
AV_{CC}	3.3 V supply (to oscillator and PHY)		3.00	3.3	3.60	V
V_{CC_IO}	1.8 V to 3.3 V supply (to I/O)		1.71	1.8	3.60	V
V_{CC_A}	1.8 V supply to analog core		1.71	1.8	1.89	V
V_{CC_D}	1.8 V supply to digital core		1.71	1.8	1.89	V
V_{IH}	Input HIGH voltage		$0.6*V_{CC\_IO}$		$V_{CC\_IO+10\%}$	V
V_{IL}	Input LOW voltage		0		$0.3*V_{CC\_IO}$	V
V_{IH_X}	Crystal input HIGH voltage		2.0		3.60	V
V_{IL_X}	Crystal input LOW voltage		–0.5		0.8	V
	Hysteresis		50			mV
I_I	Input leakage current	$0< V_{IN} < V_{CC\_IO}$			±10	µA
V_{OH}	Output voltage HIGH	I_{OUT} = 4 mA	V_{CC_IO} – 0.4			V
V_{OL}	Output LOW voltage	I_{OUT} = –4 mA			0.4	V
I_{OH}	Output current HIGH				4	mA
I_{OL}	Output current LOW				4	mA
C_{IN}	Input pin capacitance	Except D+/D–			10	pF
		D+/D–			15	pF
I_{SUSP}	Suspend current	Connected		220	380[15]	µA
		Disconnected		20	150[15]	µA

Notes
14. Do not power I/O when chip power is OFF.
15. Measured at maximum V_{CC}, 25°C.

DC Characteristics (continued)

Parameter	Description	Conditions	Min	Typ	Max	Unit
I_{CC_AVcc}	Supply current (AV_{CC})	8051 running, connected to USB HS		15	25	mA
		8051 running, connected to USB FS		10	20	mA
I_{CC_IO}	Supply current (V_{CC_IO})	8051 running, connected to USB HS		3	10	mA
		8051 running, connected to USB FS		1	5	mA
I_{CC_CORE}	Supply current (V_{CC_CORE})	8051 running, connected to USB HS		32	50	mA
		8051 running, connected to USB FS		24	40	mA
T_{RESET}	Reset time after valid power	V_{CC} min = 3.0 V	5.0			ms
	Pin reset after powered on		200			µs

AC Electrical Characteristics

USB Transceiver

USB 2.0-compliant in full and high speed modes.

GPIF Synchronous Signals

Figure 8. GPIF Synchronous Signals Timing Diagram[16]

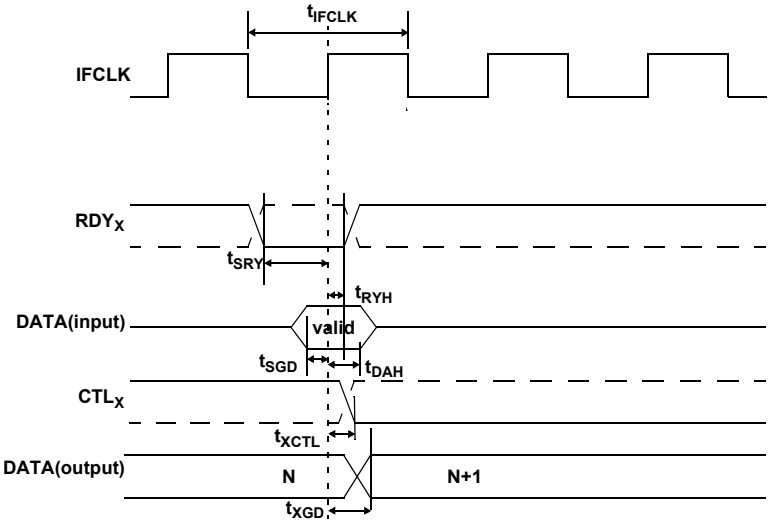


Table 9. GPIF Synchronous Signals Parameters with Internally Sourced IFCLK[16,17]

Parameter	Description	Min	Max	Unit
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_{XGD}	Clock to GPIF data output propagation delay		11	ns
t_{XCTL}	Clock to CTL_X output propagation delay		6.7	ns

Notes
16. Dashed lines denote signals with programmable polarity.
17. GPIF asynchronous RDY_x signals have a minimum setup time of 50 ns when using internal 48 MHz IFCLK.

Table 10. GPIF Synchronous Signals Parameters with Externally Sourced IFCLK[18]

Parameter	Description	Min	Max	Unit
t_{IFCLK}	IFCLK period[19]	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_{XGD}	Clock to GPIF data output propagation delay		15	ns
t_{XCTL}	Clock to CTL_X output propagation delay		13.06	ns

Slave FIFO Synchronous Read

Figure 9. Slave FIFO Synchronous Read Timing Diagram[20]

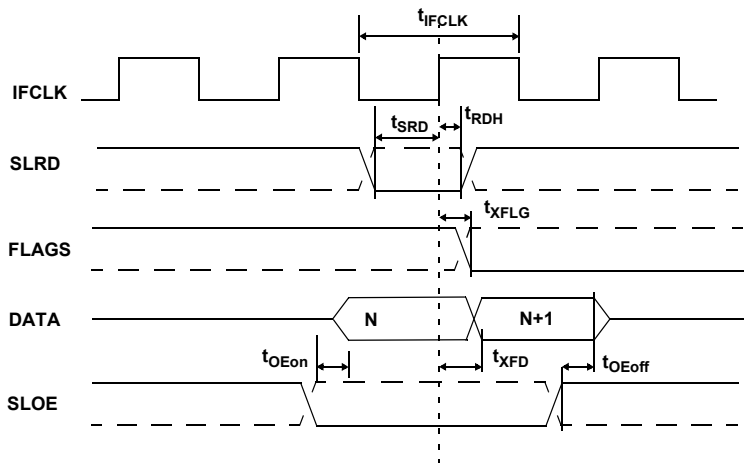


Table 11. Slave FIFO Synchronous Read Parameters with Internally Sourced IFCLK[18]

Parameter	Description	Min	Max	Unit
t_{IFCLK}	IFCLK period	20.83		ns
t_{SRD}	SLRD to clock setup time	18.7		ns
t_{RDH}	Clock to SLRD hold time	0		ns
t_{OEon}	SLOE turn-on to FIFO data valid		10.5	ns
t_{OEoff}	SLOE turn-off to FIFO data hold	2.15	10.5	ns
t_{XFLG}	Clock to FLAGS output propagation delay		9.5	ns
t_{XFD}	Clock to FIFO data output propagation delay		11	ns

Notes
18. GPIF asynchronous RDY_x signals have a minimum setup time of 50 ns when using internal 48 MHz IFCLK.
19. IFCLK must not exceed 48 MHz.
20. Dashed lines denote signals with programmable polarity.

Table 12. Slave FIFO Synchronous Read Parameters with Externally Sourced IFCLK[21]

Parameter	Description	Min	Max	Unit
t_{IFCLK}	IFCLK period	20.83	200	ns
t_{SRD}	SLRD to clock setup time	12.7		ns
t_{RDH}	Clock to SLRD hold time	3.7		ns
t_{OEon}	SLOE turn-on to FIFO data valid		10.5	ns
t_{OEoff}	SLOE turn-off to FIFO data hold	2.15	10.5	ns
t_{XFLG}	Clock to FLAGS output propagation delay		13.5	ns
t_{XFD}	Clock to FIFO data output propagation delay		17.31	ns

Slave FIFO Asynchronous Read

Figure 10. Slave FIFO Asynchronous Read Timing Diagram[22]

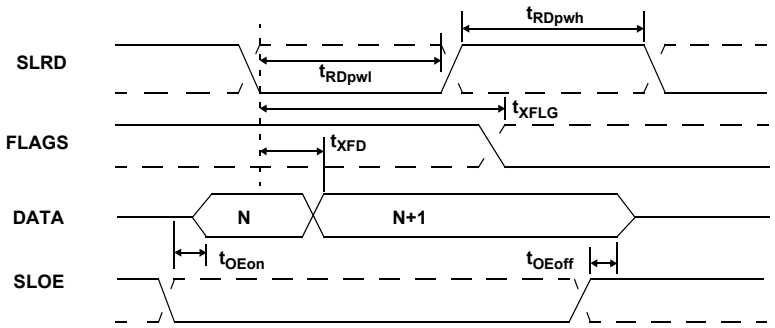


Table 13. Slave FIFO Asynchronous Read Parameters[23]

Parameter	Description	Min	Max	Unit
t_{RDpwl}	SLRD pulse width LOW	50		ns
t_{RDpwh}	SLRD pulse width HIGH	50		ns
t_{XFLG}	SLRD to FLAGS output propagation delay		70	ns
t_{XFD}	SLRD to FIFO data output propagation delay		15	ns
t_{OEon}	SLOE turn-on to FIFO data valid		10.5	ns
t_{OEoff}	SLOE turn-off to FIFO data hold	2.15	10.5	ns

Notes
21. GPIF asynchronous RDY_x signals have a minimum setup time of 50 ns when using internal 48 MHz IFCLK.
22. Dashed lines denote signals with programmable polarity.
23. Slave FIFO asynchronous parameter values use internal IFCLK setting at 48 MHz.

Slave FIFO Synchronous Write

Figure 11. Slave FIFO Synchronous Write Timing Diagram[24]

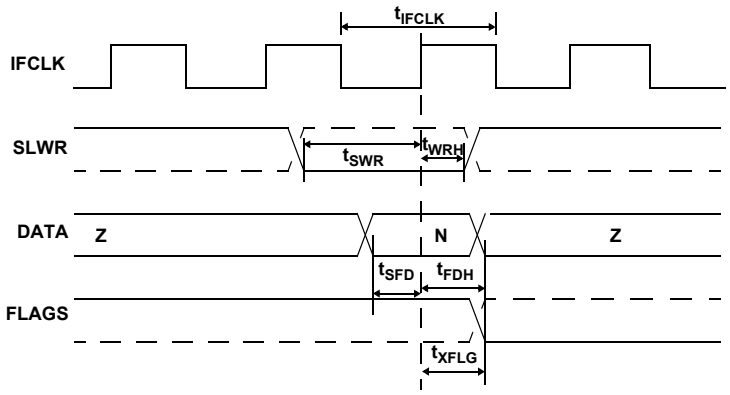


Table 14. Slave FIFO Synchronous Write Parameters with Internally Sourced IFCLK[25]

Parameter	Description	Min	Max	Unit
t_{IFCLK}	IFCLK period	20.83		ns
t_{SWR}	SLWR to clock setup time	18.1		ns
t_{WRH}	Clock to SLWR hold time	0		ns
t_{SFD}	FIFO data to clock setup time	10.64		ns
t_{FDH}	Clock to FIFO data hold time	0		ns
t_{XFLG}	Clock to FLAGS output propagation time		9.5	ns

Table 15. Slave FIFO Synchronous Write Parameters with Externally Sourced IFCLK[26]

Parameter	Description	Min	Max	Unit
t_{IFCLK}	IFCLK period	20.83	200	ns
t_{SWR}	SLWR to clock setup time	12.1		ns
t_{WRH}	Clock to SLWR hold time	3.6		ns
t_{SFD}	FIFO data to clock setup time	3.2		ns
t_{FDH}	Clock to FIFO data hold time	4.5		ns
t_{XFLG}	Clock to FLAGS output propagation time		13.5	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 internal 48 MHz IFCLK.
26. Read and writes to these registers may require synchronization delay, see *MoBL-USB FX2LP18 Technical Reference Manual* for ‘Synchronization Delay.’

Slave FIFO Asynchronous Write

Figure 12. Slave FIFO Asynchronous Write Timing Diagram[27]

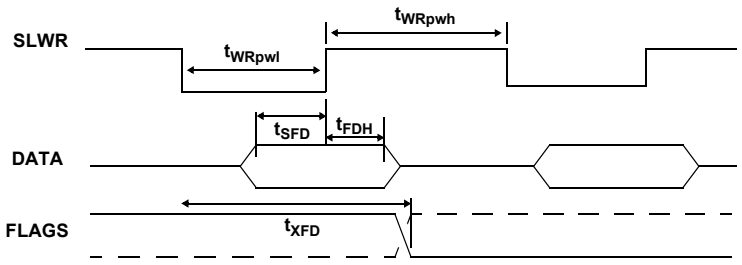


Table 16. Slave FIFO Asynchronous Write Parameters with Internally Sourced IFCLK[28]

Parameter	Description	Min	Max	Unit
t_{WRpwl}	SLWR pulse LOW	50		ns
t_{WRpwh}	SLWR pulse HIGH	50		ns
t_{SFD}	SLWR to FIFO data setup time	10		ns
t_{FDH}	FIFO data to SLWR hold time	10		ns
t_{XFD}	SLWR to FLAGS output propagation delay		70	ns

Notes
27. Dashed lines denote signals with programmable polarity.
28. Slave FIFO asynchronous parameter values use internal IFCLK setting at 48 MHz.

Slave FIFO Synchronous Packet End Strobe

Figure 13. Slave FIFO Synchronous Packet End Strobe Timing Diagram[29]

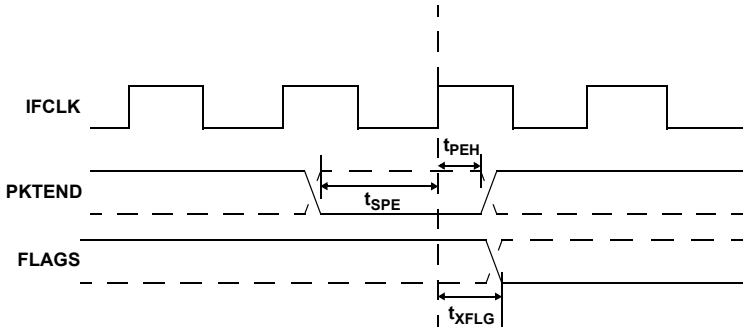


Table 17. Slave FIFO Synchronous Packet End Strobe Parameters with Internally Sourced IFCLK[30]

Parameter	Description	Min	Max	Unit
t_{IFCLK}	IFCLK period	20.83		ns
t_{SPE}	PKTEND to clock setup time	14.6		ns
t_{PEH}	Clock to PKTEND hold time	0		ns
t_{XFLG}	Clock to FLAGS output propagation delay		9.5	ns

Table 18. Slave FIFO Synchronous Packet End Strobe Parameters with Externally Sourced IFCLK[30]

Parameter	Description	Min	Max	Unit
t_{IFCLK}	IFCLK period	20.83	200	ns
t_{SPE}	PKTEND to clock setup time	8.6		ns
t_{PEH}	Clock to PKTEND hold time	3.04		ns
t_{XFLG}	Clock to FLAGS output propagation delay		13.5	ns

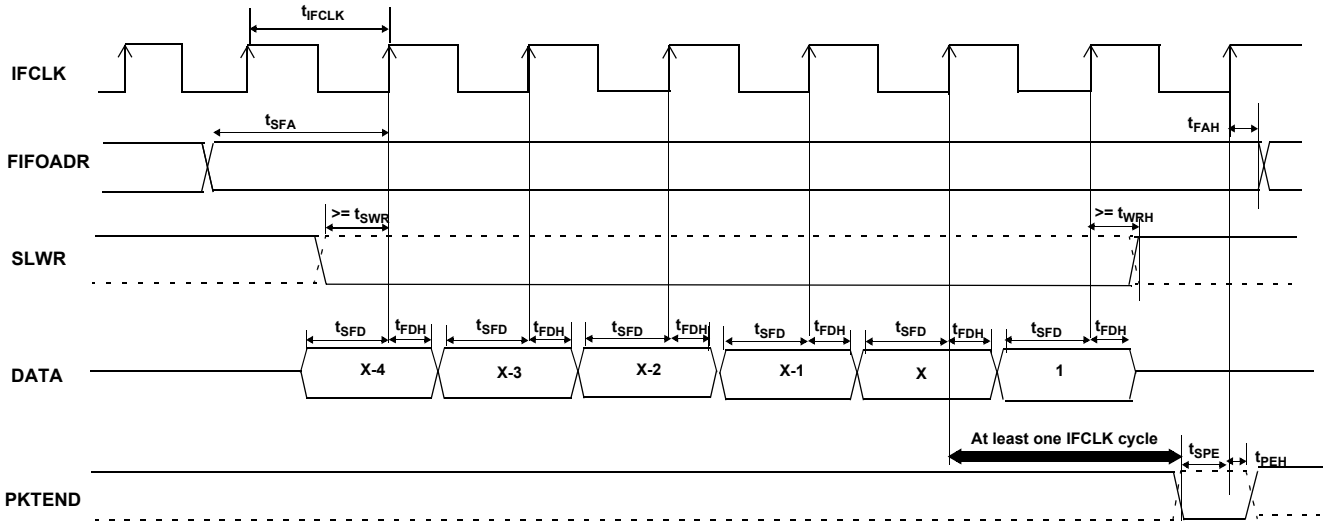
There is no specific timing requirement to be met for asserting the PKTEND pin with regards to asserting SLWR. PKTEND can be asserted with the last data value clocked into the FIFOs or thereafter. The only consideration is that the setup time t_{SPE} and the hold time t_{PEH} must be met.

Although there are no specific timing requirements 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. There is an additional timing requirement to be met when the FIFO is configured to operate in auto mode and you want to send two packets back to back: a full packet (full defined as the number of bytes in the FIFO meeting the level set in AUTOINLEN register) committed automatically followed by a short one byte/word packet committed manually using the PKTEND pin. In this scenario, make sure to assert PKTEND at least one clock cycle after the rising edge that caused the last byte/word to be clocked into the previous auto committed packet. Figure 14 shows this scenario. X is the value the AUTOINLEN register is set to when the IN endpoint is configured to be in auto mode.

Figure 14 shows a scenario where two packets are committed. The first packet is committed automatically when the number of bytes in the FIFO reaches X (value set in AUTOINLEN register) and the second one byte/word short packet is committed manually using PKTEND. Note that there is at least one IFCLK cycle timing between the assertion of PKTEND and clocking of the last byte of the previous packet (causing the packet to be committed automatically). Failing to adhere to this timing, results in the FX2LP18 failing to send the one byte/word short packet.

Notes
29. Dashed lines denote signals with programmable polarity.
30. Read and writes to these registers may require synchronization delay, see *MoBL-USB FX2LP18 Technical Reference Manual* for ‘Synchronization Delay.’

Figure 14. Slave FIFO Synchronous Write Sequence and Timing Diagram[31]



Slave FIFO Asynchronous Packet End Strobe

Figure 15. Slave FIFO Asynchronous Packet End Strobe Timing Diagram[31]

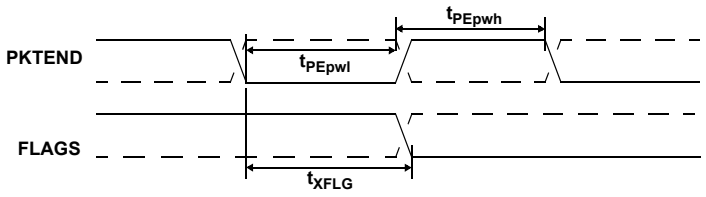


Table 19. Slave FIFO Asynchronous Packet End Strobe Parameters[32]

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

Notes
31. Dashed lines denote signals with programmable polarity.
32. Slave FIFO asynchronous parameter values use internal IFCLK setting at 48 MHz.

Slave FIFO Output Enable

Figure 16. Slave FIFO Output Enable Timing Diagram[33]

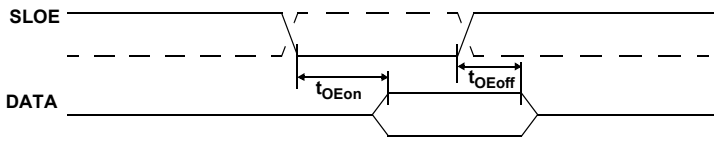


Table 20. Slave FIFO Output Enable Parameters

Parameter	Description	Min	Max	Unit
t_{OEon}	SLOE assert to FIFO data output		10.5	ns
t_{OEoff}	SLOE deassert to FIFO data hold	2.15	10.5	ns

Slave FIFO Address to Flags/Data

Figure 17. Slave FIFO Address to Flags/Data Timing Diagram[33]

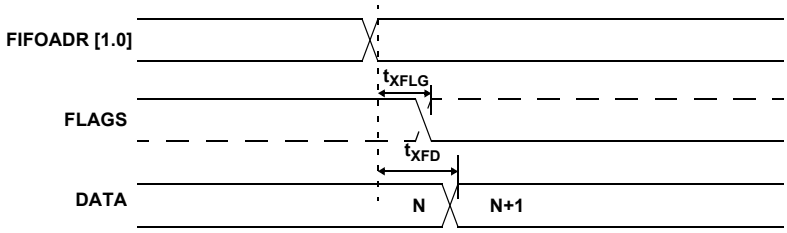


Table 21. Slave FIFO Address to Flags/Data Parameters

Parameter	Description	Min	Max	Unit
t_{XFLG}	FIFOADR[1:0] to flags output propagation delay		10.7	ns
t_{XFD}	FIFOADR[1:0] to FIFO data output propagation delay		14.3	ns

Note
33. Dashed lines denote signals with programmable polarity.

Slave FIFO Synchronous Address

Figure 18. Slave FIFO Synchronous Address Timing Diagram[34]

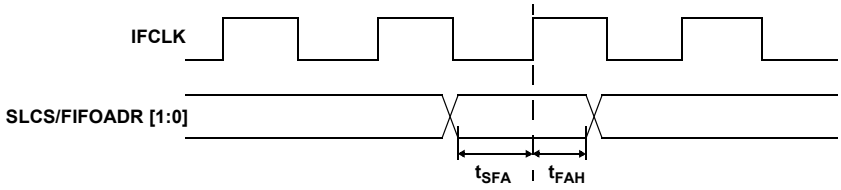
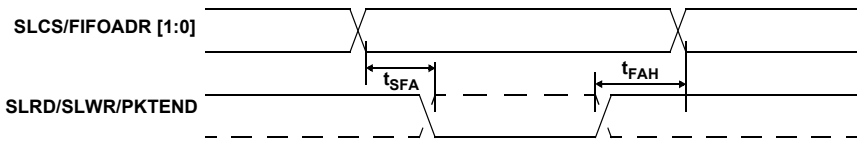


Table 22. Slave FIFO Synchronous Address Parameters[35]

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 19. Slave FIFO Asynchronous Address Timing Diagram[34]



Slave FIFO Asynchronous Address Parameters[36]

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

Notes
34. Dashed lines denote signals with programmable polarity.
35. Read and writes to these registers may require synchronization delay, see *MoBL-USB FX2LP18 Technical Reference Manual* for 'Synchronization Delay.'
36. Slave FIFO asynchronous parameter values use internal IFCLK setting at 48 MHz.

Sequence Diagram

Various sequence diagrams and examples are presented in this section.

Single and Burst Synchronous Read Example

Figure 20. Slave FIFO Synchronous Read Sequence and Timing Diagram[37]

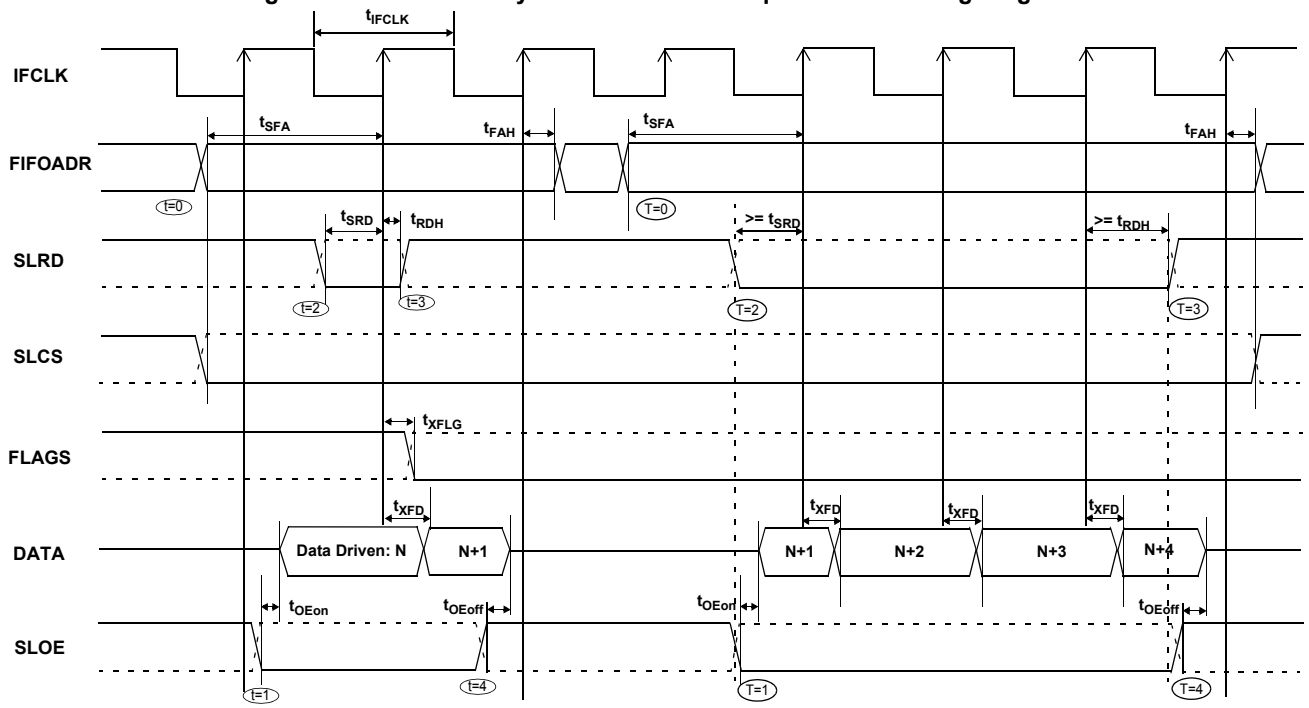


Figure 21. Slave FIFO Synchronous Sequence of Events Diagram

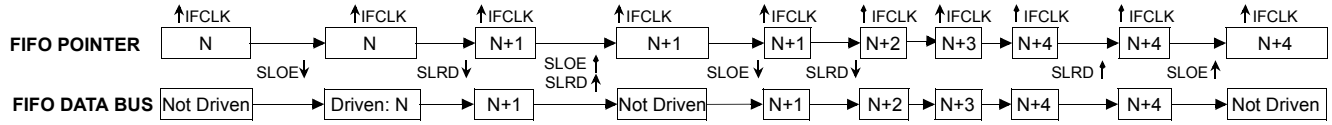


Figure 20 shows the timing relationship of the SLAVE FIFO signals during a synchronous FIFO read using IFCLK as the synchronizing clock. The diagram illustrates a single read followed by a burst read.

- At t = 0 the FIFO address is stable and the signal SLCS is asserted (SLCS may be tied low in some applications). **Note** 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 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 (that is, 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 is shown for a burst read and is 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 increments 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.

Note
37. Dashed lines denote signals with programmable polarity.

Single and Burst Synchronous Write

Figure 22. Slave FIFO Synchronous Write Sequence and Timing Diagram[38]

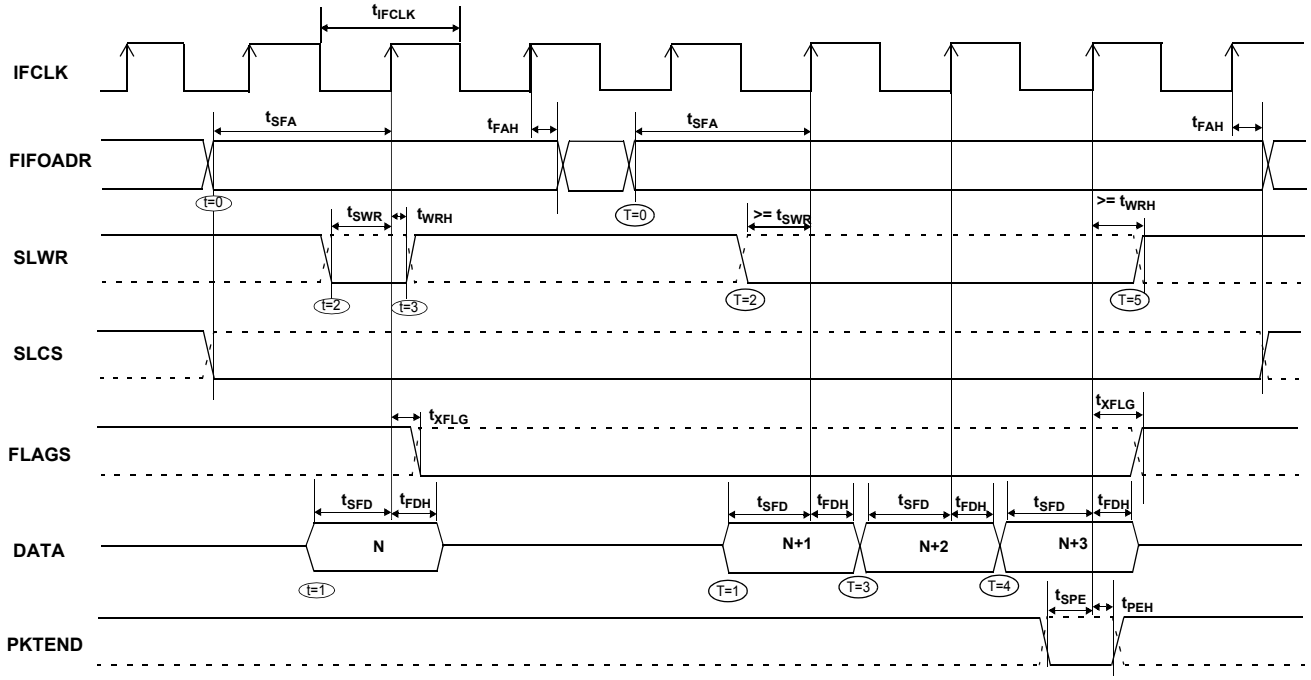


Figure 22 shows the timing relationship of the SLAVE FIFO signals during a synchronous write using IFCLK as the synchronizing clock. The diagram illustrates a single write followed by burst write of 3 bytes and committing all 4 bytes as a short packet using the PKTEND pin.

- At t = 0 the FIFO address is stable and the signal SLCS is asserted. (SLCS may be tied low in some applications) **Note** t_{SFA} has a minimum of 25 ns. This means that 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 setup 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 before SLWR is asserted. (That is, 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 is also shown for a burst write and is 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, when 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 22, when 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 needs to be met for asserting the 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 22, 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 must be held constant during the PKTEND assertion.

Although there are no specific timing requirements 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 you want to send two packets: a full packet (full defined as the number of bytes in the FIFO meeting the level set in AUTOINLEN register) committed automatically followed by a short one byte/word packet committed manually using the PKTEND pin. In this case, the external master must make sure to assert the PKTEND pin at least one clock cycle after the rising edge that caused the last byte/word to be clocked into the previous auto committed packet (the packet with the number of bytes equal to what is set in the AUTOINLEN register). Refer to Figure 14 on page 32 for further details about this timing.

Note
38. Dashed lines denote signals with programmable polarity.

Sequence Diagram of a Single and Burst Asynchronous Read

Figure 23. Slave FIFO Asynchronous Read Sequence and Timing Diagram[39]

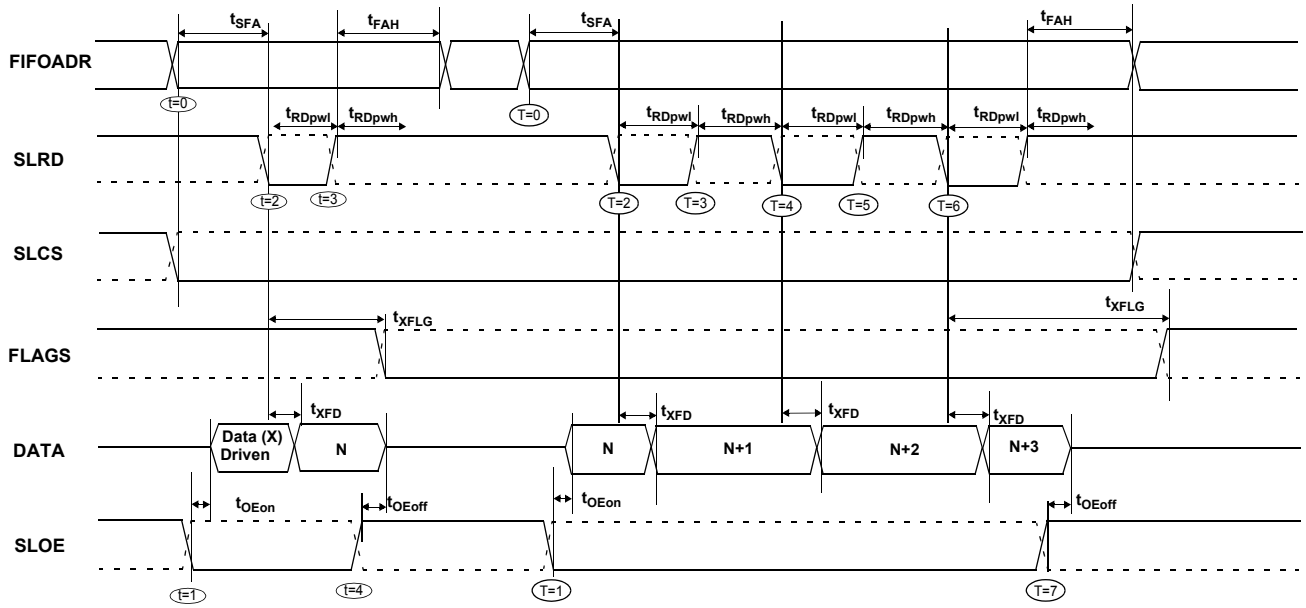


Figure 24. Slave FIFO Asynchronous Read Sequence of Events Diagram

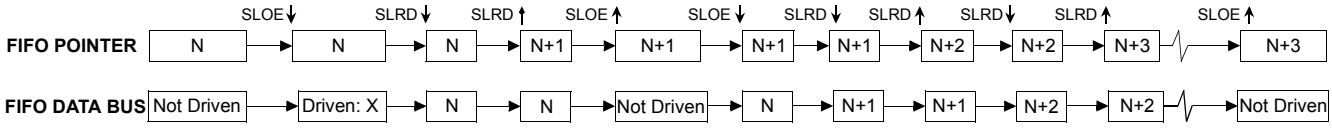


Figure 23 illustrates the timing relationship of the SLAVE FIFO signals during an asynchronous FIFO read. It shows a single read followed by a burst read.

- 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 previous data; it is data that was in the FIFO from a prior read cycle.
- At t = 2, SLRD is asserted. The SLRD must meet the minimum active pulse of t_{RDpwl} and minimum inactive pulse width of t_{RDpwh}. If SLCS is used then, SLCS must be asserted before SLRD is asserted (that is, 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 23, data N is the first valid data read from the FIFO. For data to appear on the data bus during the read cycle (for example, 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 burst read mode, during SLOE assertion, the data bus is in a driven state and outputs the previous data. Once 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.

Note
39. Dashed lines denote signals with programmable polarity.

Sequence Diagram of a Single and Burst Asynchronous Write

Figure 25. Slave FIFO Asynchronous Write Sequence and Timing Diagram[40]

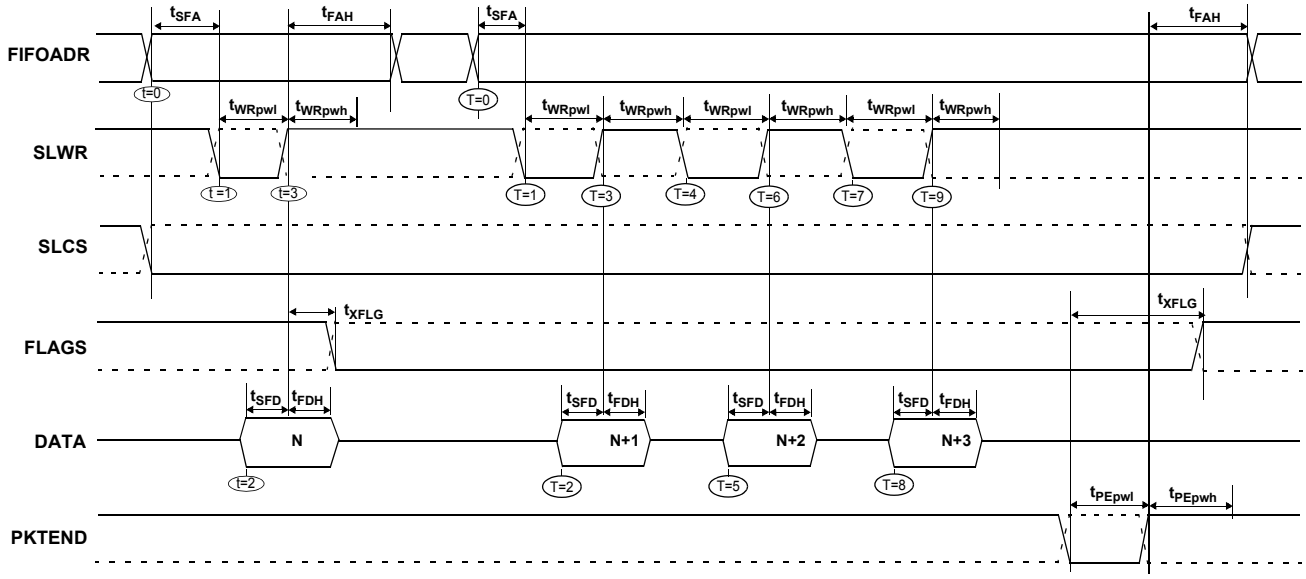


Figure 25 illustrates the timing relationship of the SLAVE FIFO write in an asynchronous mode. The diagram shows a single write followed by a burst write of 3 bytes and committing the 4-byte-short packet using PKTEND.

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

The same sequence of events is shown for a burst write and is indicated by the timing marks of T = 0 through 5.

Note In the burst write mode, once SLWR is deasserted, the data is written to the FIFO and then the FIFO pointer is incremented to the next byte in the FIFO. The FIFO pointer is post incremented.

In Figure 25 when the four bytes are written to the FIFO and SLWR is deasserted, the short 4-byte packet can be committed to the host using the PKTEND. The external device must be designed to not assert SLWR and the PKTEND signal at the same time. It must be designed to assert the PKTEND after SLWR is deasserted and meet the minimum deasserted pulse width. The FIFOADDR lines are to be held constant during the PKTEND assertion.

Note
40. Dashed lines denote signals with programmable polarity.

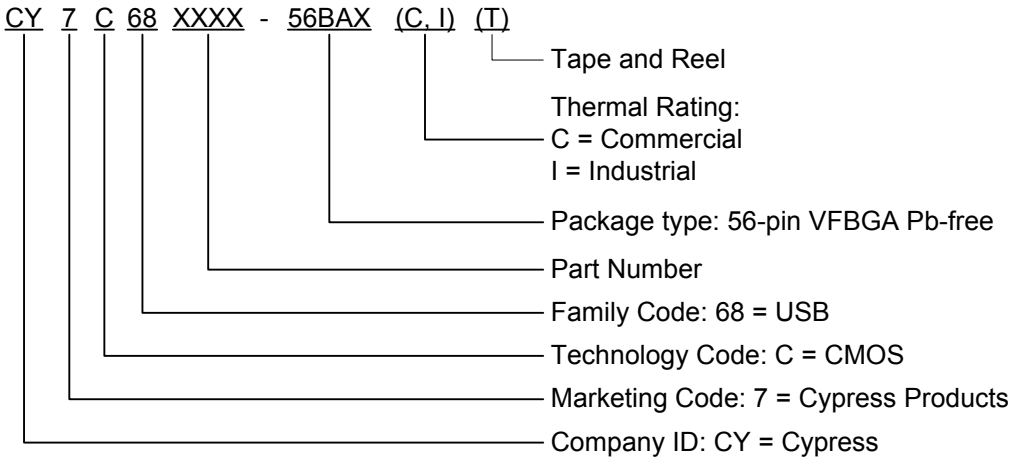
Ordering Information

Table 23 lists the key package features and ordering codes. The table contains only the parts that are currently available. If you do not see what you are looking for, contact your local sales representative. For more information, visit the Cypress website at www.cypress.com and refer to the product summary page at http://www.cypress.com/products.

Table 23. Key Features and Ordering Information

Ordering Code	Package Type	RAM Size	Number of Prog I/Os	8051 Address/Data Buses
CY7C68053-56BAXI	56-ball VFBGA (Pb-free)	16 K	24	–
Development Tool Kit				
CY3687	MoBL-USB FX2LP18 Development Kit			

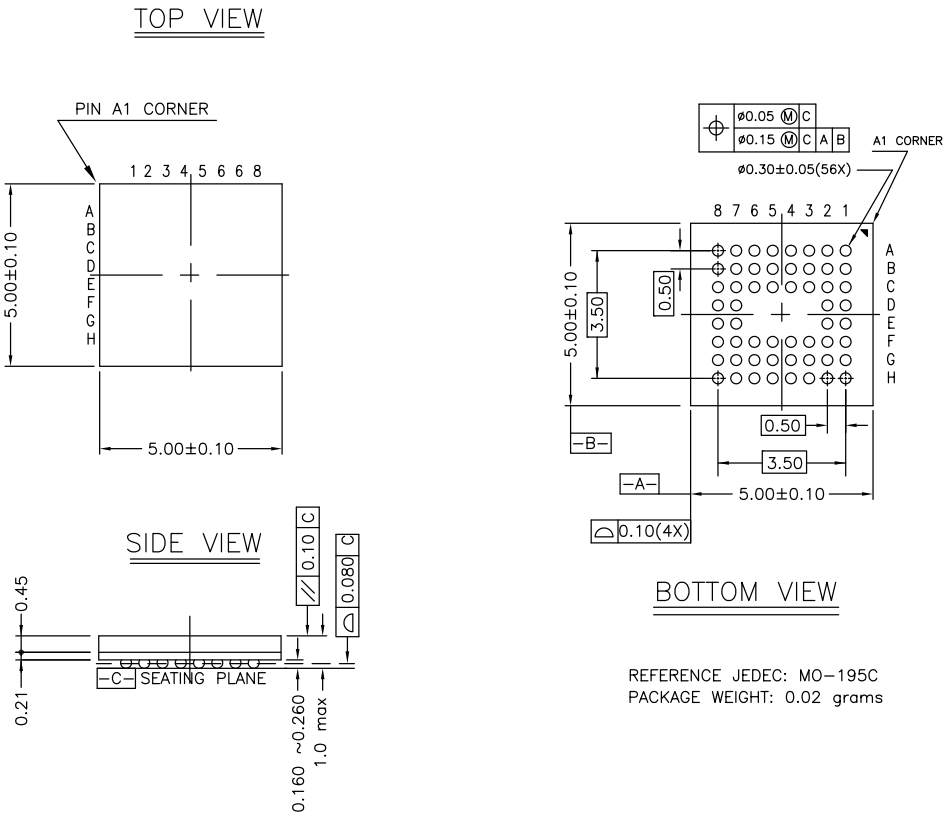
Ordering Code Definitions



Package Diagram

The FX2LP18 is available in a 56-ball VFBGA package.

Figure 26. 56-ball VFBGA (5 × 5 × 1.0 mm) 0.50 Pitch, 0.30 Ball BZ56 Package Outline, 001-03901



PCB Layout Recommendations

The following recommendations must be followed to ensure reliable high performance operation.

- At least a four-layer impedance controlled board is 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 to within specifications.
- Minimize stubs to minimize reflected signals.
- Connections between the USB connector shell and signal ground must be done near the USB connector.
- Bypass or flyback caps on VBus, near connector, are recommended.
- DPLUS and DMINUS trace lengths must be kept 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 be split under these traces.
- It is preferable to have no vias placed on the DPLUS or DMINUS trace routing.
- Isolate the DPLUS and DMINUS traces from all other signal traces by no less than 10 mm.

Acronyms

Table 24. Acronyms Used in this Document

Acronym	Description
ATA	Advanced Technology Attachment
ASIC	Application Specific Integrated Circuit
CPU	Central Processing Unit
DID	Device Identifier
DSP	Digital Signal Processor
ECC	Error Correction Code
EEPROM	Electrically Erasable Programmable Read Only Memory
EPP	Enhanced Parallel Port
FIFO	First In First Out
GPIF	General Programmable Interface
GPIO	General Purpose Input/Output
I/O	Input/Output
I2C	Inter-Integrated Circuit
PDA	Personal Digital Assistant
PLL	Phase-Locked Loop
PID	Product Identifier
RAM	Random Access Memory
SIE	Serial Interface Engine
SOF	Start Of Frame
USB	Universal Serial Bus
VID	Vendor Identifier
VFBGA	Very Fine-Pitch Ball Grid Array
UTOPIA	Universal Test and Operations Physical-Layer

Document Conventions

Units of Measure

Table 25. Units of Measure

Symbol	Unit of Measure
KHz	kilohertz
Mbytes	megabyte
MHz	megahertz
µA	microampere
µs	microsecond
µW	microwatt
mA	milliampere
mW	milliwatt
ns	nanosecond
ppm	parts per million
pF	picofarad
V	volt

Document History Page

Document Title: CY7C68053, MoBL-USB™ FX2LP18 USB Microcontroller Document Number: 001-06120				
Revision	**ECN**	**Orig. of Change**	**Submission Date**	**Description of Change**
**	430449	OSG	03/03/06	New data sheet
*A	434754	OSG	03/24/06	In Section 3.3, stated that SCL and SDA pins can be connected to V_{CC} or V_{CC_IO} Chnged sections 3.5, 3.18.1 and pin descriptions of SCL, SDA to indicate that since DISCON=1 after reset, an EEPROM or EEPROM emulation is required on the I^2C interface In pin description table, renamed pin 2H (Reserved) to Ground In Section 6, added statement “The GPIO’s are not over voltage tolerant, except the SCL and SDA pins, which are 3.3 V tolerant“ In Section 8,added a footnote to the DC char table stating that AVcc can be floated in low power mode In Section 8, changed V_{IH}max in DC char table from 3.6 V to V_{CC_IO} + 10%
*B	465471	OSG	See ECN	Changed the recommendation for the pull up resistors on I^2C Split Icc into 4 different values, corresponding to the different voltage supplies Changed Isus typical to 20uA and 220uA Added section 3.9.3 on suspend current considerations
*C	484726	ARI	See ECN	Removed all references the part number CY7C68055. Corrected the bullet in Features to state that 24 GPIO’s are available. Added the Test ID (TID#) to the Features on the front page. Made changes to the block diagram on the first page (this is now a Visio drawing instead of a Framemaker drawing). Corrected the Ambient Temperature with Power Supplied. Moved figure titles to meet the new template. Checked grammar. Took out 9-bit address bus from the block diagram on the first page. Corrected Figure 4.1
*D	492009	OSG	See ECN	Added Icc data in DC Characteristics and Maximum Power dissipation
*E	500408	OSG	See ECN	Changed ESD spec to 1500 V
*F	502115	OSG	See ECN	Changed ESD spec to 2000 V and 1500 V only for SCL and SDA pins. Added min spec for t_{OEoff} Changed Icc and power dissipation numbers
*G	1128404	OSG / ARI	See ECN	Removed SLCS from figure in Section 9.6 Slave FIFO Asynchronous Write Changed SLWR Pulse HIGH parameter to 50ns Section 9.13.1 9 V Removed the indication that SLCS and SLRD can be asserted together Section 9.13.3 - Removed the indication that SLCS and SLRD can be asserted together Implemented the latest template.
*H	1349903	AESA	See ECN	Section 7 - Changed -0°C to -40°C
*I	2728476	ODC	07/02/09	Deleted Note on AVcc parameter in DC Characteristics table
*J	3072698	ODC	10/27/10	Template update and styles update. Included Table of Contents, Ordering Code Definitions, Acronyms, and Units of measure. Updated package diagram revision from *B to *C.
*K	4212673	DBIR	12/06/2013	Updated Package Diagram: spec 001-03901 – Changed revision from *C to *E. Updated in new template. Completing Sunset Review.

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

Automotive	cypress.com/go/automotive
Clocks & Buffers	cypress.com/go/clocks
Interface	cypress.com/go/interface
Lighting & Power Control	cypress.com/go/powerpsoc cypress.com/go/plc
Memory	cypress.com/go/memory
PSoC	cypress.com/go/psoc
Touch Sensing	cypress.com/go/touch
USB Controllers	cypress.com/go/USB
Wireless/RF	cypress.com/go/wireless

PSoC® Solutions

psoc.cypress.com/solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

Cypress Developer Community

Community | Forums | Blogs | Video | Training

Technical Support

cypress.com/go/support