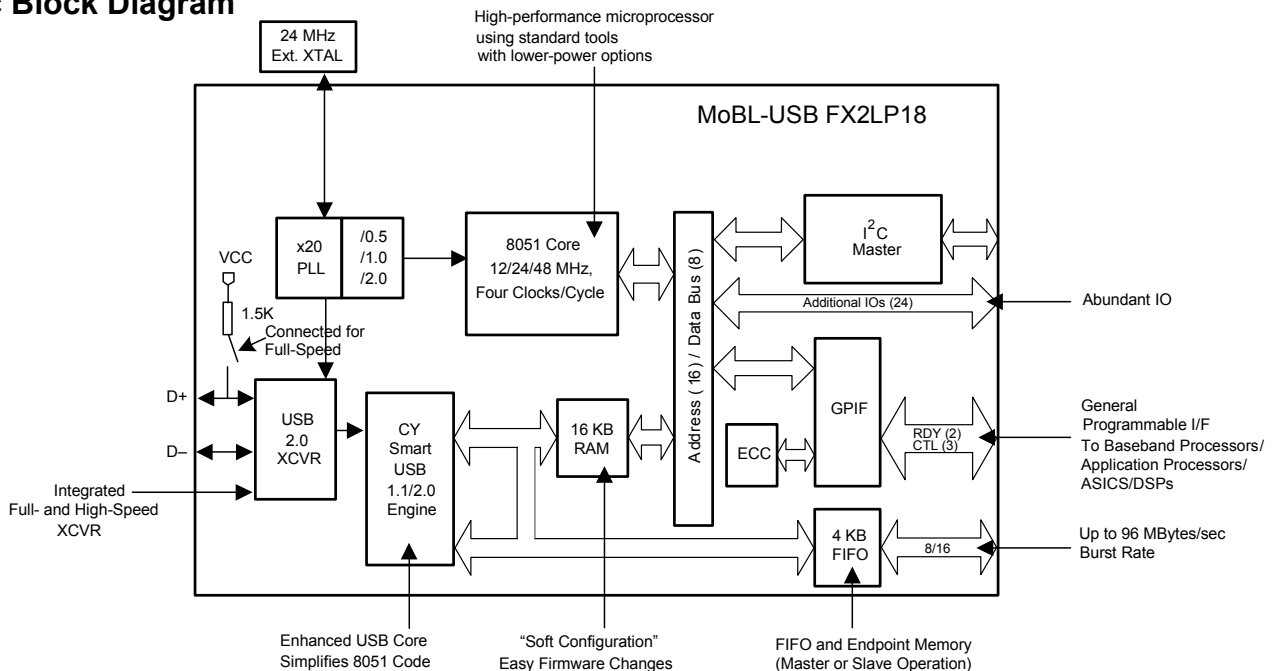


# MoBL-USB™ FX2LP18 USB Microcontroller

## 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  $\mu$ A (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/ISOCRONOUS 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<sup>2</sup>C 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



## More Information

Cypress provides a wealth of data at [www.cypress.com](http://www.cypress.com) to help you to select the right device for your design, and to help you to quickly and effectively integrate the device into your design. For a comprehensive list of resources, see the knowledge base article [AN65209 - Getting Started with FX2LP](#).

- Overview: [USB Portfolio](#), [USB Roadmap](#)
- USB 3.0 Product Selectors: [FX2LP](#), [AT2LP](#), [NX2LP-Flex](#), [SX2](#)
- Application notes: Cypress offers a large number of USB application notes covering a broad range of topics, from basic to advanced level. Recommended application notes for getting started with FX3 are:
  - [AN65209](#) - Getting Started with FX2LP
  - [AN15652](#) - Interfacing a Cypress MoBL-USB™ FX2LP18 with an Intel PXA27x Processor
  - [AN6076](#) - Differences between EZ-USB® FX2LP™ and MoBL-USB™ FX2LP18
  - For complete list of Application notes, [click here](#)
- Code Examples:
  - [USB Hi-Speed](#)
- Technical Reference Manual (TRM):
  - [MoBL-USB™ FX2LP18 Technical Reference Manual](#)
- Reference Designs:
  - [CY4661 - External USB Hard Disk Drives \(HDD\) with Fingerprint Authentication Security](#)
  - [FX2LP DMB-T/H TV Dongle Reference Design](#)
- Models: [IBIS](#)

### MoBL-USB FX2LP18 Development Kit

The [CY3687 MoBL-USB™ FX2LP18 Development Kit](#) is a complete development resource for FX2LP18. It provides a platform to develop and test custom projects using FX2LP18. The development kit contains collateral materials for the firmware, hardware, and software aspects of a design using FX2LP18.

### GPIF™ II Designer

FX2LP™ General Programmable Interface (GPIF) provides an independent hardware unit, which creates the data and control signals required by an external interface. FX2LP GPIF Designer allows users to create and modify GPIF waveform descriptors for EZ-USB FX2/ FX2LP family of chips using a graphical user interface. Extensive discussion of general GPIF discussion and programming using GPIF Designer is included in [FX2LP18 Technical Reference Manual](#) and GPIF Designer User Guide, distributed with GPIF Designer. [AN66806](#) - Getting Started with EZ-USB® FX2LP™ GPIF can be a good starting point.

**Contents**

|                                       |           |  |           |
|---------------------------------------|-----------|--|-----------|
| <b>Functional Description</b> .....   | <b>4</b>  | <b>AC Electrical Characteristics</b> .....           | <b>27</b> |
| <b>Applications</b> .....             | <b>4</b>  | USB Transceiver .....                                | 27        |
| <b>Functional Overview</b> .....      | <b>4</b>  | GPIF Synchronous Signals .....                       | 27        |
| USB Signaling Speed .....             | 4         | Slave FIFO Synchronous Read.....                     | 28        |
| 8051 Microprocessor .....             | 4         | Slave FIFO Asynchronous Read.....                    | 29        |
| I <sup>2</sup> C™ Bus .....           | 5         | Slave FIFO Synchronous Write.....                    | 30        |
| Buses .....                           | 5         | Slave FIFO Asynchronous Write .....                  | 31        |
| USB Boot Methods.....                 | 5         | Slave FIFO Synchronous Packet End Strobe .....       | 32        |
| ReNumeration™ .....                   | 5         | Slave FIFO Asynchronous Packet End Strobe .....      | 33        |
| Bus-Powered Applications .....        | 5         | Slave FIFO Output Enable .....                       | 33        |
| Interrupt System .....                | 6         | Slave FIFO Address to Flags/Data.....                | 34        |
| Reset and Wakeup .....                | 8         | Slave FIFO Synchronous Address .....                 | 34        |
| Program/Data RAM .....                | 9         | Slave FIFO Asynchronous Address .....                | 34        |
| Register Addresses .....              | 9         | Sequence Diagram.....                                | 35        |
| Endpoint RAM .....                    | 9         | <b>Ordering Information</b> .....                    | <b>39</b> |
| External FIFO Interface .....         | 11        | Ordering Code Definitions .....                      | 39        |
| GPIF .....                            | 11        | <b>Package Diagram</b> .....                         | <b>40</b> |
| ECC Generation <sup>[6]</sup> .....   | 12        | <b>PCB Layout Recommendations</b> .....              | <b>41</b> |
| USB Uploads and Downloads .....       | 12        | <b>Acronyms</b> .....                                | <b>42</b> |
| Autopointer Access .....              | 12        | <b>Document Conventions</b> .....                    | <b>42</b> |
| I <sup>2</sup> C Controller.....      | 12        | Units of Measure .....                               | 42        |
| <b>Pin Assignments</b> .....          | <b>13</b> | <b>Document History Page</b> .....                   | <b>43</b> |
| CY7C68053 Pin Descriptions .....      | 15        | <b>Sales, Solutions, and Legal Information</b> ..... | <b>44</b> |
| <b>Register Summary</b> .....         | <b>19</b> | Worldwide Sales and Design Support.....              | 44        |
| <b>Absolute Maximum Ratings</b> ..... | <b>26</b> | Products .....                                       | 44        |
| <b>Operating Conditions</b> .....     | <b>26</b> | PSoC® Solutions .....                                | 44        |
| <b>DC Characteristics</b> .....       | <b>26</b> | Cypress Developer Community.....                     | 44        |
|                                       |           | Technical Support .....                              | 44        |

## 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.

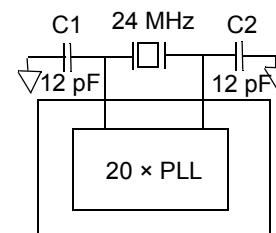
#### 8051 Clock Frequency

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

- Parallel resonant
- Fundamental mode
- 500  $\mu$ W 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 5](#). 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 |        |       |     |     |

**I<sup>2</sup>C™ Bus**

FX2LP18 supports the I<sup>2</sup>C 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<sub>CC</sub> or V<sub>CC\_IO</sub>, even if no I<sup>2</sup>C device is connected. (Connecting to V<sub>CC\_IO</sub> 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 I<sup>2</sup>C 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 I<sup>2</sup>C slave. The FX2LP18 does not enumerate using internally stored descriptors (for example, Cypress’s VID/PID/DID is not used for enumeration).<sup>[1]</sup>

**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 USB CS (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 I<sup>2</sup>C 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.

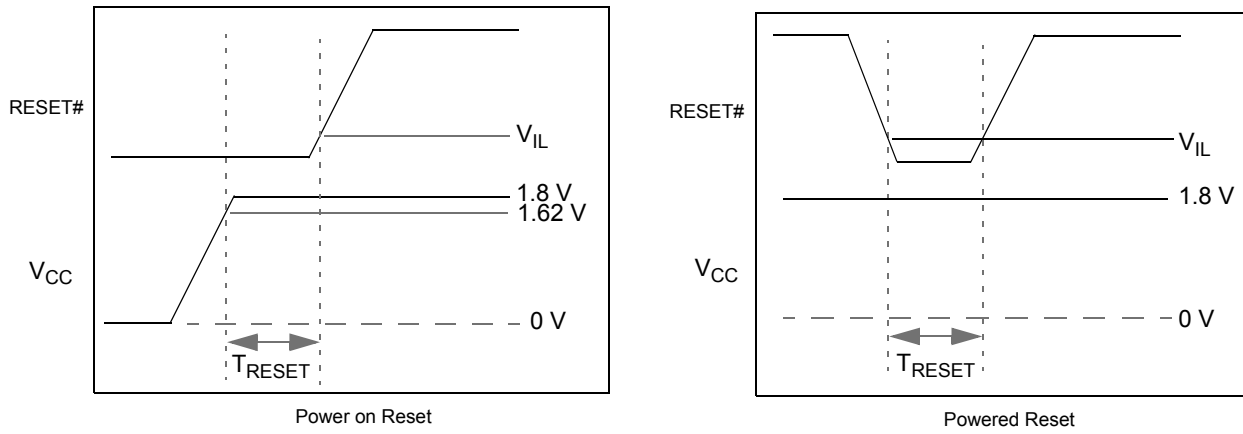
The FX2LP18 jump instruction is encoded, as shown in [Table 2](#).

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 VCC has reached 3.0 V<sup>[2]</sup>. 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 <sub>RESET</sub>            |
|------------------------------------|-------------------------------|
| 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.

#### Note

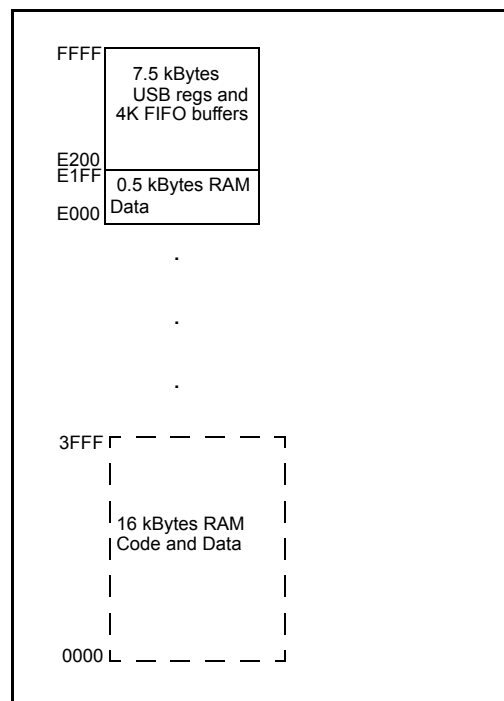
2. If the external clock is powered at the same time as the CY7C680xx and has a stabilization wait period, it must be added to the 200 μs.

#### Lowering Suspend Current

Good design practices for CMOS circuits dictate that any unused input pins must not be floating between V<sub>IL</sub> and V<sub>IH</sub>. 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<sub>CC\_IO</sub> 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<sub>CC\_IO</sub> or GND or driven by another chip.
- CTL0-2. If tri-stated via GPIFIDLECTL, these pins must be pulled to V<sub>CC\_IO</sub> or GND or driven by another chip.
- RESET#, WAKEUP#. These pins must be pulled to V<sub>CC\_IO</sub> or GND or driven by another chip during suspend.

**Figure 3. FX2LP18 Internal Code Memory**





**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<sup>2</sup>C interface boot load

**Register Addresses**

**Figure 4. Register Address Memory**

|      |                                    |
|------|------------------------------------|
| FFFF | 4 kBytes EP2-EP8 buffers (8 x 512) |
| F000 | 2 kBytes RESERVED                  |
| EFFF |                                    |
| E800 | 64 Bytes EP1IN                     |
| E7FF |                                    |
| E7C0 | 64 Bytes EP1OUT                    |
| E7BF |                                    |
| E780 | 64 Bytes EP0 IN/OUT                |
| E77F |                                    |
| E740 | 64 Bytes RESERVED                  |
| E73F |                                    |
| E700 | 8051 Addressable Registers (512)   |
| E6FF |                                    |
| E500 | Reserved (128)                     |
| E4FF |                                    |
| E480 | 128 Bytes GPIF Waveforms           |
| E47F |                                    |
| E400 | Reserved (512)                     |
| E3FF |                                    |
| E200 | 512 Bytes 8051 xdata RAM           |
| E1FF |                                    |
| E000 |                                    |

**Endpoint RAM**

This section describes the FX2LP18 Endpoint RAM.

*Size*

- 3 x 64 bytes (Endpoints 0, 1)
- 8 x 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](#).

*Setup Data Buffer*

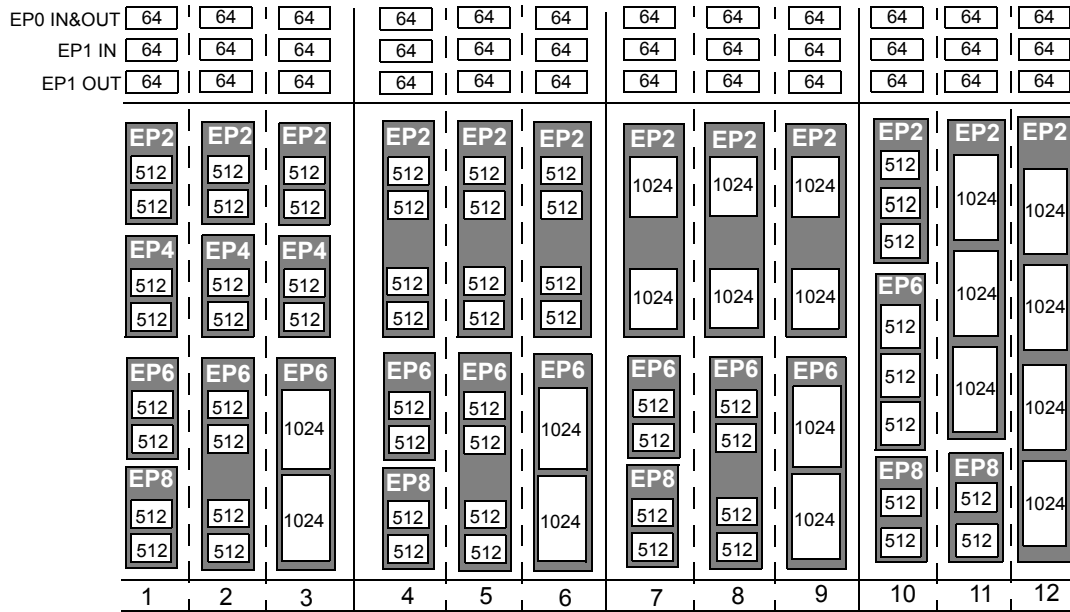
A separate 8-byte buffer at 0xE6B8-0xE6BF holds the setup data from a CONTROL transfer.

*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<sup>[3, 4]</sup>

| 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 (2x) | 64 int out (2x)  | 64 iso out (2x)  |
| ep4               | 0  | 64 bulk out (2x) | 64 bulk out (2x) | 64 bulk out (2x) |
| ep6               | 0  | 64 bulk in (2x)  | 64 int in (2x)   | 64 iso in (2x)   |
| ep8               | 0  | 64 bulk in (2x)  | 64 bulk in (2x)  | 64 bulk in (2x)  |

Default High Speed Alternate Settings

Table 5. Default High Speed Alternate Settings<sup>[3, 4]</sup>

| Alternate Setting | 0  | 1                       | 2                 | 3                 |
|-------------------|----|-------------------------|-------------------|-------------------|
| ep0               | 64 | 64                      | 64                | 64                |
| ep1out            | 0  | 512 bulk <sup>[5]</sup> | 64 int            | 64 int            |
| ep1in             | 0  | 512 bulk <sup>[5]</sup> | 64 int            | 64 int            |
| ep2               | 0  | 512 bulk out (2x)       | 512 int out (2x)  | 512 iso out (2x)  |
| ep4               | 0  | 512 bulk out (2x)       | 512 bulk out (2x) | 512 bulk out (2x) |
| ep6               | 0  | 512 bulk in (2x)        | 512 int in (2x)   | 512 iso in (2x)   |
| ep8               | 0  | 512 bulk in (2x)        | 512 bulk in (2x)  | 512 bulk in (2x)  |

Notes

3. '0' means 'not implemented.'
4. '2x' means 'double buffered.'
5. Even though these buffers are 64 bytes, they are reported as 512 for USB 2.0 compliance. Never 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<sup>[6]</sup>

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)<sup>[7]</sup>.

## 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 (AUTOPTRESET-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.

## I<sup>2</sup>C Controller

FX2LP18 has one I<sup>2</sup>C 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 I<sup>2</sup>C devices. The I<sup>2</sup>C port operates in master mode only.

### I<sup>2</sup>C Port Pins

The I<sup>2</sup>C 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<sub>CC\_IO</sub> 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<sub>IL</sub> specification of the FX2LP18. For example, if the EEPROM runs off a 3.3 V supply and V<sub>CC\_IO</sub> 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 I<sup>2</sup>C pins. Refer to the I<sup>2</sup>C specifications for details.

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

If no EEPROM is connected to the I<sup>2</sup>C 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 <sup>[8]</sup> | 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   |

### I<sup>2</sup>C Interface Boot Load Access

At power on reset the I<sup>2</sup>C 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. I<sup>2</sup>C interface boot loads only occur after power on reset.

### I<sup>2</sup>C Interface General Purpose Access

The 8051 can control peripherals connected to the I<sup>2</sup>C bus using the I2CTL and I2DAT registers. FX2LP18 provides I<sup>2</sup>C master control only, it is never an I<sup>2</sup>C slave.

## Notes

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

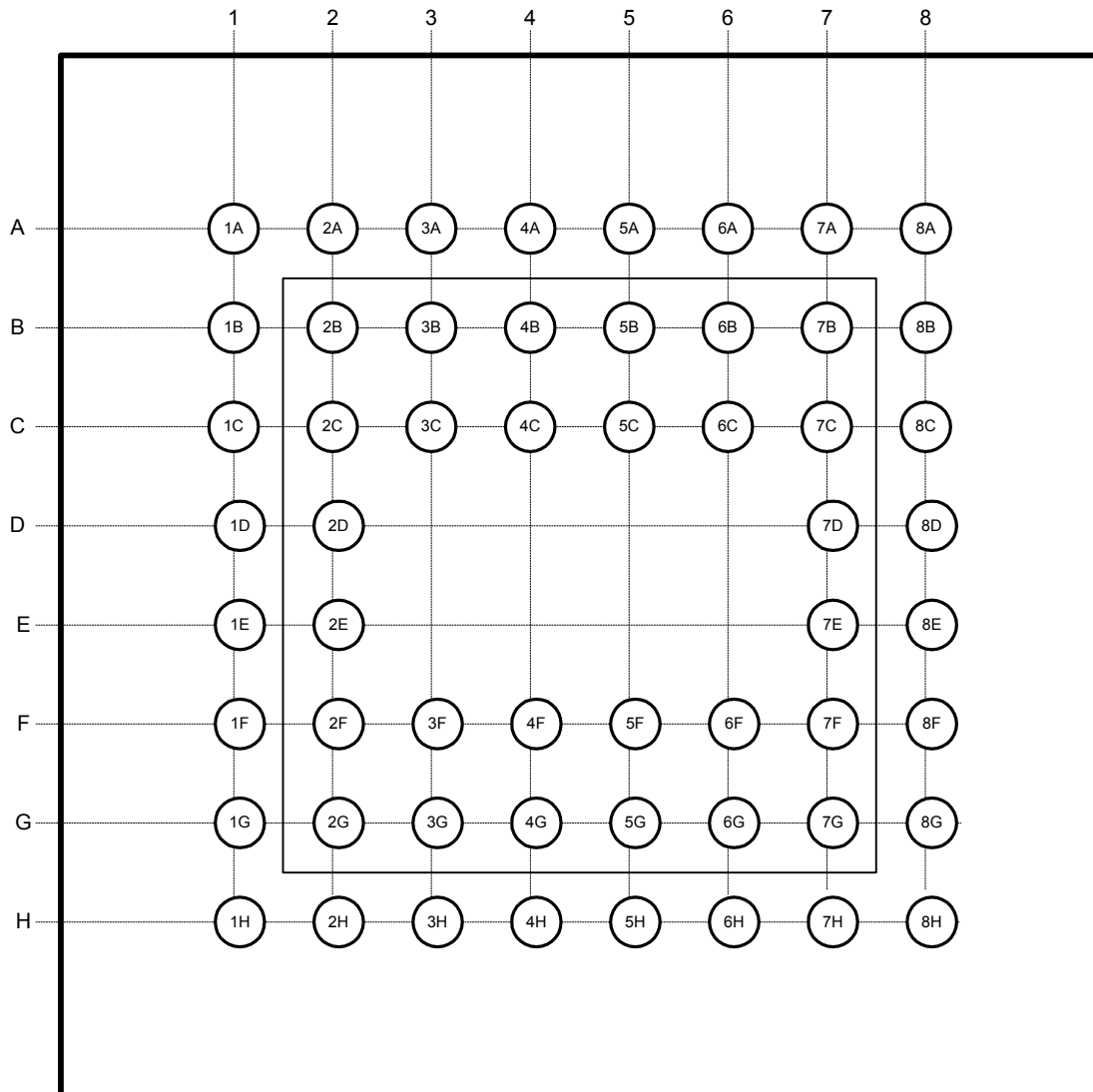
## Pin Assignments

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

|  | Port | GPIF Master | Slave FIFO |
|--|------|-------------|------------|
|  | PD7  | ↔ FD[15]    | ↔ FD[15]   |
|  | PD6  | ↔ FD[14]    | ↔ FD[14]   |
|  | PD5  | ↔ FD[13]    | ↔ FD[13]   |
|  | PD4  | ↔ FD[12]    | ↔ FD[12]   |
|  | PD3  | ↔ FD[11]    | ↔ FD[11]   |
|  | PD2  | ↔ FD[10]    | ↔ FD[10]   |
|  | PD1  | ↔ FD[9]     | ↔ FD[9]    |
|  | PD0  | ↔ FD[8]     | ↔ FD[8]    |
|  | PB7  | ↔ FD[7]     | ↔ FD[7]    |
|  | PB6  | ↔ FD[6]     | ↔ FD[6]    |
|  | PB5  | ↔ FD[5]     | ↔ FD[5]    |
|  | PB4  | RDY0        | ← FD[4]    |
|  | PB3  | RDY1        | ← FD[3]    |
|  | PB2  | →           | → FD[2]    |
|  | PB1  | CTL0        | → FD[1]    |
|  | PB0  | CTL1        | FD[0]      |
|  |      | CTL2        |            |
|  |      | INT0#/PA0   | ← SLRD     |
|  |      | INT1#/PA1   | ← SLWR     |
|  |      |             | ←          |

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



**CY7C68053 Pin Descriptions**
**Table 7. FX2LP18 Pin Descriptions<sup>[9]</sup>**

| 56 VFBGA      | Name             | Type   | Default    | Description   |
|---------------|------------------|--------|------------|---|
| 2D            | AV <sub>CC</sub> | Power  | N/A        | <b>Analog VCC.</b> 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 <sub>CC</sub> | Power  | N/A        | <b>Analog VCC.</b> 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        | <b>Analog Ground.</b> Connect this pin to ground with as short a path as possible.  |
| 1F            | AGND             | Ground | N/A        | <b>Analog Ground.</b> Connect to this pin ground with as short a path as possible.  |
| 1E            | DMINUS           | I/O/Z  | Z          | <b>USB D– Signal.</b> Connect this pin to the USB D– signal.  |
| 2E            | DPLUS            | I/O/Z  | Z          | <b>USB D+ Signal.</b> Connect this pin to the USB D+ signal.  |
| 8B            | RESET#           | Input  | N/A        | <b>Active LOW Reset.</b> This pin resets the entire chip. See <a href="#">Reset and Wakeup on page 8</a> for details.   |
| 1C            | XTALIN           | Input  | N/A        | <b>Crystal Input.</b> 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        | <b>Crystal Output.</b> 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     | <b>CLKOUT.</b> 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.   |
| <b>Port A</b> |                  |        |            |   |
| 8G            | PA0 or INT0#     | I/O/Z  | I<br>(PA0) | Multiplexed pin whose function is selected by PORTACFG.0<br><b>PA0</b> is a bidirectional I/O port pin.<br><b>INT0#</b> 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<br>(PA1) | Multiplexed pin whose function is selected by PORTACFG.1<br><b>PA1</b> is a bidirectional I/O port pin.<br><b>INT1#</b> 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<br>(PA2) | Multiplexed pin whose function is selected by two bits: IFCONFIG[1:0].<br><b>PA2</b> is a bidirectional I/O port pin.<br><b>SLOE</b> is an input-only output enable with programmable polarity (FIFOPIN-POLAR.4) for the slave FIFO's connected to FD[7:0] or FD[15:0].   |
| 7F            | PA3 or WU2       | I/O/Z  | I<br>(PA3) | Multiplexed pin whose function is selected by: WAKEUP.7 and OEA.3<br><b>PA3</b> is a bidirectional I/O port pin.<br><b>WU2</b> is an alternate source for <b>USB Wakeup</b> , 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<sup>[9]</sup> (continued)

| 56 VFBGA      | Name                  | Type  | Default    | Description   |
|---------------|-----------------------|-------|------------|---|
| 6F            | PA4 or FIFOADR0       | I/O/Z | I<br>(PA4) | Multiplexed pin whose function is selected by IFCONFIG[1:0].<br><b>PA4</b> is a bidirectional I/O port pin.<br><b>FIFOADR0</b> 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<br>(PA5) | Multiplexed pin whose function is selected by IFCONFIG[1:0].<br><b>PA5</b> is a bidirectional I/O port pin.<br><b>FIFOADR1</b> 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<br>(PA6) | Multiplexed pin whose function is selected by the IFCONFIG[1:0] bits.<br><b>PA6</b> is a bidirectional I/O port pin.<br><b>PKTEND</b> 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<br>(PA7) | Multiplexed pin whose function is selected by the IFCONFIG[1:0] and PORTACFG.7 bits.<br><b>PA7</b> is a bidirectional I/O port pin.<br><b>FLAGD</b> is a programmable slave FIFO output status flag signal.<br><b>SLCS#</b> gates all other slave FIFO enable/strobes |
| <b>Port B</b> |                       |       |            |   |
| 3H            | PB0 or FD[0]          | I/O/Z | I<br>(PB0) | Multiplexed pin whose function is selected by IFCONFIG[1:0].<br><b>PB0</b> is a bidirectional I/O port pin.<br><b>FD[0]</b> is the bidirectional FIFO/GPIF data bus.  |
| 4F            | PB1 or FD[1]          | I/O/Z | I<br>(PB1) | Multiplexed pin whose function is selected by IFCONFIG[1:0].<br><b>PB1</b> is a bidirectional I/O port pin.<br><b>FD[1]</b> is the bidirectional FIFO/GPIF data bus.  |
| 4H            | PB2 or FD[2]          | I/O/Z | I<br>(PB2) | Multiplexed pin whose function is selected by IFCONFIG[1:0].<br><b>PB2</b> is a bidirectional I/O port pin.<br><b>FD[2]</b> is the bidirectional FIFO/GPIF data bus.  |
| 4G            | PB3 or FD[3]          | I/O/Z | I<br>(PB3) | Multiplexed pin whose function is selected by IFCONFIG[1:0].<br><b>PB3</b> is a bidirectional I/O port pin.<br><b>FD[3]</b> is the bidirectional FIFO/GPIF data bus.  |
| 5H            | PB4 or FD[4]          | I/O/Z | I<br>(PB4) | Multiplexed pin whose function is selected by IFCONFIG[1:0].<br><b>PB4</b> is a bidirectional I/O port pin.<br><b>FD[4]</b> is the bidirectional FIFO/GPIF data bus.  |
| 5G            | PB5 or FD[5]          | I/O/Z | I<br>(PB5) | Multiplexed pin whose function is selected by IFCONFIG[1:0].<br><b>PB5</b> is a bidirectional I/O port pin.<br><b>FD[5]</b> is the bidirectional FIFO/GPIF data bus.  |
| 5F            | PB6 or FD[6]          | I/O/Z | I<br>(PB6) | Multiplexed pin whose function is selected by IFCONFIG[1:0].<br><b>PB6</b> is a bidirectional I/O port pin.<br><b>FD[6]</b> is the bidirectional FIFO/GPIF data bus.  |
| 6H            | PB7 or FD[7]          | I/O/Z | I<br>(PB7) | Multiplexed pin whose function is selected IFCONFIG[1:0].<br><b>PB7</b> is a bidirectional I/O port pin.<br><b>FD[7]</b> is the bidirectional FIFO/GPIF data bus.   |



**Table 7. FX2LP18 Pin Descriptions<sup>[9]</sup>** (continued)

| 56 VFBGA      | Name          | Type  | Default    | Description  |
|---------------|---------------|-------|------------|--|
| <b>PORT D</b> |               |       |            |  |
| 8A            | PD0 or FD[8]  | I/O/Z | I<br>(PD0) | Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFI-FOCFG.0 (wordwide) bits.<br><b>FD[8]</b> is the bidirectional FIFO/GPIF data bus.  |
| 7A            | PD1 or FD[9]  | I/O/Z | I<br>(PD1) | Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFI-FOCFG.0 (wordwide) bits.<br><b>FD[9]</b> is the bidirectional FIFO/GPIF data bus.  |
| 6B            | PD2 or FD[10] | I/O/Z | I<br>(PD2) | Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFI-FOCFG.0 (wordwide) bits.<br><b>FD[10]</b> is the bidirectional FIFO/GPIF data bus.   |
| 6A            | PD3 or FD[11] | I/O/Z | I<br>(PD3) | Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFI-FOCFG.0 (wordwide) bits.<br><b>FD[11]</b> is the bidirectional FIFO/GPIF data bus.   |
| 3B            | PD4 or FD[12] | I/O/Z | I<br>(PD4) | Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFI-FOCFG.0 (wordwide) bits.<br><b>FD[12]</b> is the bidirectional FIFO/GPIF data bus.   |
| 3A            | PD5 or FD[13] | I/O/Z | I<br>(PD5) | Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFI-FOCFG.0 (wordwide) bits.<br><b>FD[13]</b> is the bidirectional FIFO/GPIF data bus.   |
| 3C            | PD6 or FD[14] | I/O/Z | I<br>(PD6) | Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFI-FOCFG.0 (wordwide) bits.<br><b>FD[14]</b> is the bidirectional FIFO/GPIF data bus.   |
| 2A            | PD7 or FD[15] | I/O/Z | I<br>(PD7) | Multiplexed pin whose function is selected by the IFCONFIG[1:0] and EPxFI-FOCFG.0 (wordwide) bits.<br><b>FD[15]</b> is the bidirectional FIFO/GPIF data bus.   |
| 1A            | RDY0 or SLRD  | Input | N/A        | Multiplexed pin whose function is selected by IFCONFIG[1:0].<br><b>RDY0</b> is a GPIF input signal.<br><b>SLRD</b> 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].<br><b>RDY1</b> is a GPIF input signal.<br><b>SLWR</b> 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].<br><b>CTL0</b> is a GPIF control output.<br><b>FLAGA</b> is a programmable slave FIFO output status flag signal.<br>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].<br><b>CTL1</b> is a GPIF control output.<br><b>FLAGB</b> is a programmable slave FIFO output status flag signal.<br>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].<br><b>CTL2</b> is a GPIF control output.<br><b>FLAGC</b> is a programmable slave FIFO output status flag signal.<br>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<sup>[9]</sup> (continued)

| 56 VFBGA | Name               | Type   | Default | Description   |
|----------|--------------------|--------|---------|---|
| 7B       | WAKEUP             | Input  | N/A     | <b>USB Wakeup.</b> 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 <sup>®</sup> chip from suspending. This pin has programmable polarity (WAKEUP.4). |
| 3F       | SCL                | OD     | Z       | <b>Clock</b> for the I <sup>2</sup> C interface. Connect to V <sub>CC_IO</sub> or V <sub>CC</sub> with a 2.2K–10K pull up resistor. (An I <sup>2</sup> C peripheral is required.)   |
| 3G       | SDA                | OD     | Z       | <b>Data</b> for the I <sup>2</sup> C interface. Connect to V <sub>CC_IO</sub> or V <sub>CC</sub> with a 2.2K–10K pull up resistor. (An I <sup>2</sup> C peripheral is required.)  |
| 5A       | V <sub>CC_IO</sub> | Power  | N/A     | <b>VCC.</b> 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 <sub>CC_IO</sub> | Power  | N/A     | <b>VCC.</b> Connect this pin to 1.8V to 3.3 V power source.   |
| 7E       | V <sub>CC_IO</sub> | Power  | N/A     | <b>VCC.</b> Connect this pin to 1.8 V to 3.3 V power source.  |
| 8E       | V <sub>CC_IO</sub> | Power  | N/A     | <b>VCC.</b> Connect this pin to 1.8V to 3.3 V power source.   |
| 5C       | V <sub>CC_D</sub>  | Power  | N/A     | <b>VCC.</b> 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 <sub>CC_A</sub>  | Power  | N/A     | <b>VCC.</b> Connect this pin to 1.8V power source. (Supplies power to internal analog 1.8 V circuits.)  |
| 1H       | GND                | Ground | N/A     | <b>Ground</b>   |
| 2H       | GND                | Ground | N/A     | <b>Ground</b>   |
| 4A       | GND                | Ground | N/A     | <b>Ground</b>   |
| 4B       | GND                | Ground | N/A     | <b>Ground</b>   |
| 4C       | GND                | Ground | N/A     | <b>Ground</b>   |
| 7D       | GND                | Ground | N/A     | <b>Ground</b>   |
| 8D       | GND                | Ground | N/A     | <b>Ground</b>   |

## 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        | xxxxxxx       | RW       |
| E480                          | 128  | Reserved                      |  |          |          |          |                     |          |           |           |           |               |          |
| <b>GENERAL CONFIGURATION</b>  |      |                               |  |          |          |          |                     |          |           |           |           |               |          |
| 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      | rrbbbbb  |
| E601                          | 1    | IFCONFIG                      | Interface Configuration (Ports, GPIF, Slave FIFOs) | IFCLKSRC | 3048MHZ  | IFCLKOE  | IFCLKPOL            | ASYNC    | GSTATE    | IFCFG1    | IFCFG0    | 10000000      | RW       |
| E602                          | 1    | PINFLAGSAB <sup>[10]</sup>    | Slave FIFO FLAGA and FLAGB pin configuration       | FLAGB3   | FLAGB2   | FLAGB1   | FLAGB0              | FLAGA3   | FLAGA2    | FLAGA1    | FLAGA0    | 00000000      | RW       |
| E603                          | 1    | PINFLAGSCD <sup>[10]</sup>    | Slave FIFO FLAGC and FLAGD pin configuration       | FLAGD3   | FLAGD2   | FLAGD1   | FLAGD0              | FLAGC3   | FLAGC2    | FLAGC1    | FLAGC0    | 00000000      | RW       |
| E604                          | 1    | FIFORESET <sup>[10]</sup>     | Restore FIFOs to default state                     | NAKALL   | 0        | 0        | 0                   | EP3      | EP2       | EP1       | EP0       | xxxxxxx       | W        |
| E605                          | 1    | BREAKPT                       | Breakpoint control                                 | 0        | 0        | 0        | 0                   | BREAK    | BPPULSE   | BPEN      | 0         | 00000000      | rrrrbbbr |
| E606                          | 1    | BPADDRH                       | Breakpoint address H                               | A15      | A14      | A13      | A12                 | A11      | A10       | A9        | A8        | xxxxxxx       | RW       |
| E607                          | 1    | BPADDRL                       | Breakpoint address L                               | A7       | A6       | A5       | A4                  | A3       | A2        | A1        | A0        | xxxxxxx       | RW       |
| E608                          | 1    | Reserved                      | Reserved   | 0        | 0        | 0        | 0                   | 0        | 0         | 0         | 0         | 00000000      | rrrrbb   |
| E609                          | 1    | FIFOPINPOLAR <sup>[10]</sup>  | Slave FIFO interface pins polarity                 | 0        | 0        | PKTEND   | SLOE                | SLRD     | SLWR      | EF        | FF        | 00000000      | rrbbbbb  |
| E60A                          | 1    | REVID                         | Chip revision                                      | rv7      | rv6      | rv5      | rv4                 | rv3      | rv2       | rv1       | rv0       | RevA 00000001 | R        |
| E60B                          | 1    | REVCTL <sup>[10]</sup>        | Chip revision control                              | 0        | 0        | 0        | 0                   | 0        | 0         | dyn_out   | enh_pkt   | 00000000      | rrrrbb   |
| <b>UDMA</b>                   |      |                               |  |          |          |          |                     |          |           |           |           |               |          |
| E60C                          | 1    | GPIFHOLDAMOUNT                | MSTB hold time (for UDMA)                          | 0        | 0        | 0        | 0                   | 0        | 0         | HOLDTIME1 | HOLDTIME0 | 00000000      | rrrrbb   |
|                               | 3    | Reserved                      |  |          |          |          |                     |          |           |           |           |               |          |
| <b>ENDPOINT CONFIGURATION</b> |      |                               |  |          |          |          |                     |          |           |           |           |               |          |
| E610                          | 1    | EP1OUTCFG                     | Endpoint 1-OUT configuration                       | VALID    | 0        | TYPE1    | TYPE0               | 0        | 0         | 0         | 0         | 10100000      | brbrrrr  |
| E611                          | 1    | EP1INCFG                      | Endpoint 1-IN configuration                        | VALID    | 0        | TYPE1    | TYPE0               | 0        | 0         | 0         | 0         | 10100000      | brbrrrr  |
| 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      | bbbrrrr  |
| 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      | bbbrrrr  |
|                               | 2    | Reserved                      |  |          |          |          |                     |          |           |           |           |               |          |
| E618                          | 1    | EP2FIFOCFG <sup>[10]</sup>    | Endpoint 2/Slave FIFO configuration                | 0        | INFM1    | OEP1     | AUTOOUT             | AUTOIN   | ZEROLENIN | 0         | WORDWIDE  | 00000101      | rbbbbbrb |
| E619                          | 1    | EP4FIFOCFG <sup>[10]</sup>    | Endpoint 4/Slave FIFO configuration                | 0        | INFM1    | OEP1     | AUTOOUT             | AUTOIN   | ZEROLENIN | 0         | WORDWIDE  | 00000101      | rbbbbbrb |
| E61A                          | 1    | EP6FIFOCFG <sup>[10]</sup>    | Endpoint 6/Slave FIFO configuration                | 0        | INFM1    | OEP1     | AUTOOUT             | AUTOIN   | ZEROLENIN | 0         | WORDWIDE  | 00000101      | rbbbbbrb |
| E61B                          | 1    | EP8FIFOCFG <sup>[10]</sup>    | Endpoint 8/Slave FIFO configuration                | 0        | INFM1    | OEP1     | AUTOOUT             | AUTOIN   | ZEROLENIN | 0         | WORDWIDE  | 00000101      | rbbbbbrb |
| E61C                          | 4    | Reserved                      |  |          |          |          |                     |          |           |           |           |               |          |
| E620                          | 1    | EP2AUTOINLENH <sup>[10]</sup> | Endpoint 2 AUTOIN packet length H                  | 0        | 0        | 0        | 0                   | 0        | PL10      | PL9       | PL8       | 00000010      | rrrrbb   |
| E621                          | 1    | EP2AUTOINLENL <sup>[10]</sup> | Endpoint 2 AUTOIN packet length L                  | PL7      | PL6      | PL5      | PL4                 | PL3      | PL2       | PL1       | PL0       | 00000000      | RW       |
| E622                          | 1    | EP4AUTOINLENH <sup>[10]</sup> | Endpoint 4 AUTOIN packet length H                  | 0        | 0        | 0        | 0                   | 0        | PL9       | PL8       | PL7       | 00000010      | rrrrbb   |
| E623                          | 1    | EP4AUTOINLENL <sup>[10]</sup> | Endpoint 4 AUTOIN packet length L                  | PL7      | PL6      | PL5      | PL4                 | PL3      | PL2       | PL1       | PL0       | 00000000      | RW       |
| E624                          | 1    | EP6AUTOINLENH <sup>[10]</sup> | Endpoint 6 AUTOIN packet length H                  | 0        | 0        | 0        | 0                   | 0        | PL10      | PL9       | PL8       | 00000010      | rrrrbb   |
| E625                          | 1    | EP6AUTOINLENL <sup>[10]</sup> | Endpoint 6 AUTOIN packet length L                  | PL7      | PL6      | PL5      | PL4                 | PL3      | PL2       | PL1       | PL0       | 00000000      | RW       |
| E626                          | 1    | EP8AUTOINLENH <sup>[10]</sup> | Endpoint 8 AUTOIN packet length H                  | 0        | 0        | 0        | 0                   | 0        | PL9       | PL8       | PL7       | 00000010      | rrrrbb   |
| E627                          | 1    | EP8AUTOINLENL <sup>[10]</sup> | Endpoint 8 AUTOIN packet length L                  | PL7      | PL6      | PL5      | PL4                 | PL3      | PL2       | PL1       | PL0       | 00000000      | RW       |
| E628                          | 1    | ECCCFG                        | ECC Configuration                                  | 0        | 0        | 0        | 0                   | 0        | 0         | 0         | ECCM      | 00000000      | rrrrrb   |
| E629                          | 1    | ECCRESET                      | ECC Reset  | x        | x        | x        | x                   | x        | x         | x         | x         | 00000000      | W        |
| E62A                          | 1    | ECC1B0                        | ECC1 Byte 0 address                                | LINE15   | LINE14   | LINE13   | LINE12              | LINE11   | LINE10    | LINE9     | LINE8     | 00000000      | R        |
| E62B                          | 1    | ECC1B1                        | ECC1 Byte 1 address                                | LINE7    | LINE6    | LINE5    | LINE4               | LINE3    | LINE2     | LINE1     | LINE0     | 00000000      | R        |

**Table 8. FX2LP18 Register Summary** (continued)

| Hex | Size | Name | Description | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 | Default | Access |
|-----|------|------|-------------|----|----|----|----|----|----|----|----|---------|--------|
|-----|------|------|-------------|----|----|----|----|----|----|----|----|---------|--------|

**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 | 1    | EP2FIFOPFH <sup>[10]</sup>    | Endpoint 2/Slave FIFO programmable flag H    | DECIS                  | PKTSTAT                | IN:PKTS[2]<br>OUT:PFC12 | IN:PKTS[1]<br>OUT:PFC11 | IN:PKTS[0]<br>OUT:PFC10 | 0      | PFC9   | PFC8                   | 10001000 | bbbbbrbb |
| E630 | 1    | EP2FIFOPFH <sup>[10]</sup>    | Endpoint 2/Slave FIFO programmable flag H    | DECIS                  | PKTSTAT                | OUT:PFC12               | OUT:PFC11               | OUT:PFC10               | 0      | PFC9   | IN:PKTS[2]<br>OUT:PFC8 | 10001000 | bbbbbrbb |
| E631 | 1    | EP2FIFOPFL <sup>[10]</sup>    | Endpoint 2/Slave FIFO programmable flag L    | PFC7                   | PFC6                   | PFC5                    | PFC4                    | PFC3                    | PFC2   | PFC1   | PFC0                   | 00000000 | RW       |
| E631 | 1    | EP2FIFOPFL <sup>[10]</sup>    | Endpoint 2/Slave FIFO programmable flag L    | IN:PKTS[1]<br>OUT:PFC7 | IN:PKTS[0]<br>OUT:PFC6 | PFC5                    | PFC4                    | PFC3                    | PFC2   | PFC1   | PFC0                   | 00000000 | RW       |
| E632 | 1    | EP4FIFOPFH <sup>[10]</sup>    | Endpoint 4/Slave FIFO programmable flag H    | DECIS                  | PKTSTAT                | 0                       | IN:PKTS[1]<br>OUT:PFC10 | IN:PKTS[0]<br>OUT:PFC9  | 0      | 0      | PFC8                   | 10001000 | bbrbbrbb |
| E632 | 1    | EP4FIFOPFH <sup>[10]</sup>    | Endpoint 4/Slave FIFO programmable flag H    | DECIS                  | PKTSTAT                | 0                       | OUT:PFC10               | OUT:PFC9                | 0      | 0      | PFC8                   | 10001000 | bbrbbrbb |
| E633 | 1    | EP4FIFOPFL <sup>[10]</sup>    | Endpoint 4/Slave FIFO programmable flag L    | PFC7                   | PFC6                   | PFC5                    | PFC4                    | PFC3                    | PFC2   | PFC1   | PFC0                   | 00000000 | RW       |
| E633 | 1    | EP4FIFOPFL <sup>[10]</sup>    | Endpoint 4/Slave FIFO programmable flag L    | IN:PKTS[1]<br>OUT:PFC7 | IN:PKTS[0]<br>OUT:PFC6 | PFC5                    | PFC4                    | PFC3                    | PFC2   | PFC1   | PFC0                   | 00000000 | RW       |
| E634 | 1    | EP6FIFOPFH <sup>[10]</sup>    | Endpoint 6/Slave FIFO programmable flag H    | DECIS                  | PKTSTAT                | IN:PKTS[2]<br>OUT:PFC12 | IN:PKTS[1]<br>OUT:PFC11 | IN:PKTS[0]<br>OUT:PFC10 | 0      | PFC9   | PFC8                   | 00001000 | bbbbbrbb |
| E634 | 1    | EP6FIFOPFH <sup>[10]</sup>    | Endpoint 6/Slave FIFO programmable flag H    | DECIS                  | PKTSTAT                | OUT:PFC12               | OUT:PFC11               | OUT:PFC10               | 0      | PFC9   | IN:PKTS[2]<br>OUT:PFC8 | 00001000 | bbbbbrbb |
| E635 | 1    | EP6FIFOPFL <sup>[10]</sup>    | Endpoint 6/Slave FIFO programmable flag L    | PFC7                   | PFC6                   | PFC5                    | PFC4                    | PFC3                    | PFC2   | PFC1   | PFC0                   | 00000000 | RW       |
| E635 | 1    | EP6FIFOPFL <sup>[10]</sup>    | Endpoint 6/Slave FIFO programmable flag L    | IN:PKTS[1]<br>OUT:PFC7 | IN:PKTS[0]<br>OUT:PFC6 | PFC5                    | PFC4                    | PFC3                    | PFC2   | PFC1   | PFC0                   | 00000000 | RW       |
| E636 | 1    | EP8FIFOPFH <sup>[10]</sup>    | Endpoint 8/Slave FIFO programmable flag H    | DECIS                  | PKTSTAT                | 0                       | IN:PKTS[1]<br>OUT:PFC10 | IN:PKTS[0]<br>OUT:PFC9  | 0      | 0      | PFC8                   | 00001000 | bbrbbrbb |
| E636 | 1    | EP8FIFOPFH <sup>[10]</sup>    | Endpoint 8/Slave FIFO programmable flag H    | DECIS                  | PKTSTAT                | 0                       | OUT:PFC10               | OUT:PFC9                | 0      | 0      | PFC8                   | 00001000 | bbrbbrbb |
| E637 | 1    | EP8FIFOPFL <sup>[10]</sup>    | Endpoint 8/Slave FIFO programmable flag L    | PFC7                   | PFC6                   | PFC5                    | PFC4                    | PFC3                    | PFC2   | PFC1   | PFC0                   | 00000000 | RW       |
| E637 | 1    | EP8FIFOPFL <sup>[10]</sup>    | Endpoint 8/Slave FIFO programmable flag L    | IN:PKTS[1]<br>OUT:PFC7 | IN:PKTS[0]<br>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 <sup>[10]</sup>      | Force IN packet end                          | Skip                   | 0                      | 0                       | 0                       | EP3                     | EP2    | EP1    | EP0                    | xxxxxxx  | W        |
| E649 | 7    | OUTPKTEND <sup>[10]</sup>     | Force OUT packet end                         | Skip                   | 0                      | 0                       | 0                       | EP3                     | EP2    | EP1    | EP0                    | xxxxxxx  | W        |
|      |      | <b>INTERRUPTS</b>             |  |                        |                        |                         |                         |                         |        |        |                        |          |          |
| E650 | 1    | EP2FIFOIE <sup>[10]</sup>     | Endpoint 2 Slave FIFO flag interrupt enable  | 0                      | 0                      | 0                       | 0                       | EDGE PF                 | PF     | EF     | FF                     | 00000000 | RW       |
| E651 | 1    | EP2FIFOIRQ <sup>[10,11]</sup> | Endpoint 2 Slave FIFO flag interrupt request | 0                      | 0                      | 0                       | 0                       | 0                       | PF     | EF     | FF                     | 00000000 | rrrrrbbb |
| E652 | 1    | EP4FIFOIE <sup>[10]</sup>     | Endpoint 4 Slave FIFO flag interrupt enable  | 0                      | 0                      | 0                       | 0                       | EDGE PF                 | PF     | EF     | FF                     | 00000000 | RW       |
| E653 | 1    | EP4FIFOIRQ <sup>[10,11]</sup> | Endpoint 4 Slave FIFO flag interrupt request | 0                      | 0                      | 0                       | 0                       | 0                       | PF     | EF     | FF                     | 00000000 | rrrrrbbb |
| E654 | 1    | EP6FIFOIE <sup>[10]</sup>     | Endpoint 6 Slave FIFO flag interrupt enable  | 0                      | 0                      | 0                       | 0                       | EDGE PF                 | PF     | EF     | FF                     | 00000000 | RW       |
| E655 | 1    | EP6FIFOIRQ <sup>[10,11]</sup> | Endpoint 6 Slave FIFO flag interrupt request | 0                      | 0                      | 0                       | 0                       | 0                       | PF     | EF     | FF                     | 00000000 | rrrrrbbb |
| E656 | 1    | EP8FIFOIE <sup>[10]</sup>     | Endpoint 8 Slave FIFO flag interrupt enable  | 0                      | 0                      | 0                       | 0                       | EDGE PF                 | PF     | EF     | FF                     | 00000000 | RW       |
| E657 | 1    | EP8FIFOIRQ <sup>[10,11]</sup> | 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 <sup>[11]</sup>        | IN-BULK-NAK interrupt request                | 0                      | 0                      | EP8                     | EP6                     | EP4                     | EP2    | EP1    | EP0                    | 00xxxxx  | rrbbbbb  |
| E65A | 1    | NAKIE                         | Endpoint Ping-NAK/IBN interrupt enable       | EP8                    | EP6                    | EP4                     | EP2                     | EP1                     | EP0    | 0      | IBN                    | 00000000 | RW       |
| E65B | 1    | NAKIRQ <sup>[11]</sup>        | Endpoint Ping-NAK/IBN interrupt request      | EP8                    | EP6                    | EP4                     | EP2                     | EP1                     | EP0    | 0      | IBN                    | xxxxxx0x | bbbbbrbb |
| E65C | 1    | USBIE                         | USB interrupt enables                        | 0                      | EP0ACK                 | HSGRANT                 | URES                    | SUSP                    | SUTOK  | SOF    | SUDAV                  | 00000000 | RW       |
| E65D | 1    | USBIRQ <sup>[11]</sup>        | USB interrupt requests                       | 0                      | EP0ACK                 | HSGRANT                 | URES                    | SUSP                    | SUTOK  | SOF    | SUDAV                  | 0xxxxxx  | 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 <sup>[11]</sup>     | Endpoint interrupt requests             | EP8      | EP6    | EP4    | EP2     | EP1OUT  | EP1IN     | EP0OUT    | EP0IN     | 0        | RW        |
| E660                | 1    | GPIFIE <sup>[10]</sup>    | GPIF interrupt enable                   | 0        | 0      | 0      | 0       | 0       | 0         | GPIFWF    | GPIFDONE  | 00000000 | RW        |
| E661                | 1    | GPIFIRQ <sup>[10]</sup>   | 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 <sup>[11]</sup> | 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         | xxxxxxx  | 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                  |   |          |        |        |         |         |           |           |           |          |           |
| <b>INPUT/OUTPUT</b> |      |                           |   |          |        |        |         |         |           |           |           |          |           |
| 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                      | I <sup>2</sup> C bus control and status | START    | STOP   | LASTRD | ID1     | ID0     | BERR      | ACK       | DONE      | 000xx000 | bbbbrrrr  |
| E679                | 1    | I2DAT                     | I <sup>2</sup> C bus data               | d7       | d6     | d5     | d4      | d3      | d2        | d1        | d0        | xxxxxxx  | RW        |
| E67A                | 1    | I2CTL                     | I <sup>2</sup> C bus control            | 0        | 0      | 0      | 0       | 0       | 0         | STOPIE    | 400KHZ    | 00000000 | RW        |
| E67B                | 1    | XAUTODAT1                 | Autopr1 MOVX access, when APTREN = 1    | D7       | D6     | D5     | D4      | D3      | D2        | D1        | D0        | xxxxxxx  | RW        |
| E67C                | 1    | XAUTODAT2                 | Autopr2 MOVX access, when APTREN = 1    | D7       | D6     | D5     | D4      | D3      | D2        | D1        | D0        | xxxxxxx  | RW        |
| <b>UDMA CRC</b>     |      |                           |   |          |        |        |         |         |           |           |           |          |           |
| E67D                | 1    | UDMACRCH <sup>[10]</sup>  | UDMA CRC MSB                            | CRC15    | CRC14  | CRC13  | CRC12   | CRC11   | CRC10     | CRC9      | CRC8      | 01001010 | RW        |
| E67E                | 1    | UDMACRCL <sup>[10]</sup>  | 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  | Q SIGNAL2 | Q SIGNAL1 | Q SIGNAL0 | 00000000 | bbbbbbbb  |
| <b>USB CONTROL</b>  |      |                           |   |          |        |        |         |         |           |           |           |          |           |
| E680                | 1    | USBCS                     | USB control and status                  | HSM      | 0      | 0      | 0       | DISCON  | NOSYNOSOF | RENUM     | SIGRSUME  | x0000000 | rrrrbbbb  |
| E681                | 1    | SUSPEND                   | Put chip into suspend                   | x        | x      | x      | x       | x       | x         | x         | x         | xxxxxxx  | W         |
| E682                | 1    | WAKEUPCS                  | Wakeup control and status               | WU2      | WU     | WU2POL | WUPOL   | 0       | DPEN      | WU2EN     | WUEN      | xx000101 | bbbbrrbbb |
| E683                | 1    | TOGCTL                    | Toggle control                          | Q        | S      | R      | I/O     | EP3     | EP2       | EP1       | EP0       | x0000000 | rrrrbbbb  |
| E684                | 1    | USBFRAMEH                 | USB frame count H                       | 0        | 0      | 0      | 0       | 0       | FC10      | FC9       | FC8       | 00000xxx | R         |
| E685                | 1    | USBFRAMEL                 | USB frame count L                       | FC7      | FC6    | FC5    | FC4     | FC3     | FC2       | FC1       | FC0       | xxxxxxx  | R         |
| E686                | 1    | MICROFRAME                | Microframe count, 0-7                   | 0        | 0      | 0      | 0       | 0       | MF2       | MF1       | MF0       | 00000xxx | R         |
| E687                | 1    | FNADDR                    | USB function address                    | 0        | FA6    | FA5    | FA4     | FA3     | FA2       | FA1       | FA0       | 0xxxxxxx | R         |
| E688                | 2    | Reserved                  |   |          |        |        |         |         |           |           |           |          |           |
| <b>ENDPOINTS</b>    |      |                           |   |          |        |        |         |         |           |           |           |          |           |
| E68A                | 1    | EP0BCH <sup>[10]</sup>    | Endpoint 0 byte count H                 | (BC15)   | (BC14) | (BC13) | (BC12)  | (BC11)  | (BC10)    | (BC9)     | (BC8)     | xxxxxxx  | RW        |
| E68B                | 1    | EP0BCL <sup>[10]</sup>    | Endpoint 0 byte count L                 | (BC7)    | BC6    | BC5    | BC4     | BC3     | BC2       | BC1       | BC0       | xxxxxxx  | RW        |
| E68C                | 1    | Reserved                  |   |          |        |        |         |         |           |           |           |          |           |
| E68D                | 1    | EP1OUTBC                  | Endpoint 1 OUT byte count               | 0        | BC6    | BC5    | BC4     | BC3     | BC2       | BC1       | BC0       | 0xxxxxxx | RW        |
| E68E                | 1    | Reserved                  |   |          |        |        |         |         |           |           |           |          |           |
| E68F                | 1    | EP1INBC                   | Endpoint 1 IN byte count                | 0        | BC6    | BC5    | BC4     | BC3     | BC2       | BC1       | BC0       | 0xxxxxxx | RW        |
| E690                | 1    | EP2BCH <sup>[10]</sup>    | Endpoint 2 byte count H                 | 0        | 0      | 0      | 0       | 0       | BC10      | BC9       | BC8       | 00000xxx | RW        |
| E691                | 1    | EP2BCL <sup>[10]</sup>    | Endpoint 2 byte count L                 | BC7/SKIP | BC6    | BC5    | BC4     | BC3     | BC2       | BC1       | BC0       | xxxxxxx  | RW        |
| E692                | 2    | Reserved                  |   |          |        |        |         |         |           |           |           |          |           |
| E694                | 1    | EP4BCH <sup>[10]</sup>    | Endpoint 4 byte count H                 | 0        | 0      | 0      | 0       | 0       | 0         | BC9       | BC8       | 000000xx | RW        |
| E695                | 1    | EP4BCL <sup>[10]</sup>    | Endpoint 4 byte count L                 | BC7/SKIP | BC6    | BC5    | BC4     | BC3     | BC2       | BC1       | BC0       | xxxxxxx  | RW        |
| E696                | 2    | Reserved                  |   |          |        |        |         |         |           |           |           |          |           |
| E698                | 1    | EP6BCH <sup>[10]</sup>    | Endpoint 6 byte count H                 | 0        | 0      | 0      | 0       | 0       | BC10      | BC9       | BC8       | 00000xxx | RW        |
| E699                | 1    | EP6BCL <sup>[10]</sup>    | Endpoint 6 byte count L                 | BC7/SKIP | BC6    | BC5    | BC4     | BC3     | BC2       | BC1       | BC0       | xxxxxxx  | RW        |
| E69A                | 2    | Reserved                  |   |          |        |        |         |         |           |           |           |          |           |
| E69C                | 1    | EP8BCH <sup>[10]</sup>    | Endpoint 8 byte count H                 | 0        | 0      | 0      | 0       | 0       | 0         | BC9       | BC8       | 000000xx | RW        |
| E69D                | 1    | EP8BCL <sup>[10]</sup>    | Endpoint 8 byte count L                 | BC7/SKIP | BC6    | BC5    | BC4     | BC3     | BC2       | BC1       | BC0       | xxxxxxx  | RW        |
| E69E                | 2    | Reserved                  |   |          |        |        |         |         |           |           |           |          |           |
| E6A0                | 1    | EP0CS                     | Endpoint 0 control and status           | HSNAK    | 0      | 0      | 0       | 0       | 0         | BUSY      | STALL     | 10000000 | bbbbbbrrb |
| E6A1                | 1    | EP1OUTCS                  | Endpoint 1 OUT control and status       | 0        | 0      | 0      | 0       | 0       | 0         | BUSY      | STALL     | 00000000 | bbbbbbrrb |

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 | bbbbbbbr |
| E6A3 | 1    | EP2CS                    | Endpoint 2 control and status                 | 0         | NPAK2     | NPAK1     | NPAK0     | FULL    | EMPTY   | 0       | STALL   | 00101000 | rrrrrrrb |
| E6A4 | 1    | EP4CS                    | Endpoint 4 control and status                 | 0         | 0         | NPAK1     | NPAK0     | FULL    | EMPTY   | 0       | STALL   | 00101000 | rrrrrrrb |
| E6A5 | 1    | EP6CS                    | Endpoint 6 control and status                 | 0         | NPAK2     | NPAK1     | NPAK0     | FULL    | EMPTY   | 0       | STALL   | 00000100 | rrrrrrrb |
| E6A6 | 1    | EP8CS                    | Endpoint 8 control and status                 | 0         | 0         | NPAK1     | NPAK0     | FULL    | EMPTY   | 0       | STALL   | 00000100 | rrrrrrrb |
| E6A7 | 1    | EP2FIFOFLGS              | Endpoint 2 Slave FIFO flags                   | 0         | 0         | 0         | 0         | 0       | PF      | EF      | FF      | 00000010 | R        |
| E6A8 | 1    | EP4FIFOFLGS              | Endpoint 4 Slave FIFO flags                   | 0         | 0         | 0         | 0         | 0       | PF      | EF      | FF      | 00000010 | R        |
| E6A9 | 1    | EP6FIFOFLGS              | Endpoint 6 Slave FIFO flags                   | 0         | 0         | 0         | 0         | 0       | PF      | EF      | FF      | 00000110 | R        |
| E6AA | 1    | EP8FIFOFLGS              | Endpoint 8 Slave FIFO flags                   | 0         | 0         | 0         | 0         | 0       | PF      | EF      | FF      | 00000110 | R        |
| E6AB | 1    | EP2FIFOBCH               | Endpoint 2 Slave FIFO total byte count H      | 0         | 0         | 0         | BC12      | BC11    | BC10    | BC9     | BC8     | 00000000 | R        |
| E6AC | 1    | EP2FIFOBCL               | Endpoint 2 Slave FIFO total byte count L      | BC7       | BC6       | BC5       | BC4       | BC3     | BC2     | BC1     | BC0     | 00000000 | R        |
| E6AD | 1    | EP4FIFOBCH               | Endpoint 4 Slave FIFO total byte count H      | 0         | 0         | 0         | 0         | 0       | BC10    | BC9     | BC8     | 00000000 | R        |
| E6AE | 1    | EP4FIFOBCL               | Endpoint 4 Slave FIFO total byte count L      | BC7       | BC6       | BC5       | BC4       | BC3     | BC2     | BC1     | BC0     | 00000000 | R        |
| E6AF | 1    | EP6FIFOBCH               | Endpoint 6 Slave FIFO total byte count H      | 0         | 0         | 0         | 0         | BC11    | BC10    | BC9     | BC8     | 00000000 | R        |
| E6B0 | 1    | EP6FIFOBCL               | Endpoint 6 Slave FIFO total byte count L      | BC7       | BC6       | BC5       | BC4       | BC3     | BC2     | BC1     | BC0     | 00000000 | R        |
| E6B1 | 1    | EP8FIFOBCH               | Endpoint 8 Slave FIFO total byte count H      | 0         | 0         | 0         | 0         | 0       | BC10    | BC9     | BC8     | 00000000 | R        |
| E6B2 | 1    | EP8FIFOBCL               | Endpoint 8 Slave FIFO total byte count L      | BC7       | BC6       | BC5       | BC4       | BC3     | BC2     | BC1     | BC0     | 00000000 | R        |
| E6B3 | 1    | SUDPTRH                  | Setup data pointer high address byte          | A15       | A14       | A13       | A12       | A11     | A10     | A9      | A8      | xxxxxxx  | RW       |
| E6B4 | 1    | SUDPTRL                  | Setup data pointer low address byte           | A7        | A6        | A5        | A4        | A3      | A2      | A1      | 0       | xxxxxxx  | bbbbbbbr |
| E6B5 | 1    | SUDPTRCTL                | Setup data pointer auto mode                  | 0         | 0         | 0         | 0         | 0       | 0       | 0       | SDPAUTO | 00000001 | RW       |
| E6B8 | 8    | SET-UPDAT                | 8 bytes of setup data                         | D7        | D6        | D5        | D4        | D3      | D2      | D1      | D0      | xxxxxxx  | R        |
|      |      |                          | SET-UPDAT[0] = bmRequestType                  |           |           |           |           |         |         |         |         |          |          |
|      |      |                          | SET-UPDAT[1] = bmRequest                      |           |           |           |           |         |         |         |         |          |          |
|      |      |                          | SET-UPDAT[2:3] = wValue                       |           |           |           |           |         |         |         |         |          |          |
|      |      |                          | SET-UPDAT[4:5] = wIndex                       |           |           |           |           |         |         |         |         |          |          |
|      |      |                          | SET-UPDAT[6:7] = wLength                      |           |           |           |           |         |         |         |         |          |          |
|      |      |                          | <b>GPIF</b>                                   |           |           |           |           |         |         |         |         |          |          |
| 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 |          |
|      |      |                          | <b>FLOWSTATE</b>                              |           |           |           |           |         |         |         |         |          |          |
| E6C6 | 1    | FLOWSTATE                | Flowstate enable and selector                 | FSE       | 0         | 0         | 0         | 0       | FS2     | FS1     | FS0     | 00000000 | brrrrbbr |
| 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       | FALLING | RISING  |         | 00000001 | rrrrrbbr |
| E6CD | 1    | FLOWSTBPERIOD            | Master strobe half period                     | D7        | D6        | D5        | D4        | D3      | D2      | D1      | D0      | 00000010 | RW       |
| E6CE | 1    | GPIFTCB3 <sup>[10]</sup> | GPIF transaction count Byte 3                 | TC31      | TC30      | TC29      | TC28      | TC27    | TC26    | TC25    | TC24    | 00000000 | RW       |
| E6CF | 1    | GPIFTCB2 <sup>[10]</sup> | GPIF transaction count Byte 2                 | TC23      | TC22      | TC21      | TC20      | TC19    | TC18    | TC17    | TC16    | 00000000 | RW       |
| E6D0 | 1    | GPIFTCB1 <sup>[10]</sup> | GPIF transaction count Byte 1                 | TC15      | TC14      | TC13      | TC12      | TC11    | TC10    | TC9     | TC8     | 00000000 | RW       |
| E6D1 | 1    | GPIFTCB0 <sup>[10]</sup> | 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 <sup>[10]</sup>       | Endpoint 2 GPIF flag select                      | 0      | 0      | 0       | 0   | 0   | 0   | FS1  | FS0       | 00000000        | RW      |
| E6D3 | 1    | EP2GPIPFSTOP                        | Endpoint 2 GPIF stop transaction on program flag | 0      | 0      | 0       | 0   | 0   | 0   | 0    | FIFO2FLAG | 00000000        | RW      |
| E6D4 | 1    | EP2GPIFTRIG <sup>[10]</sup>         | Endpoint 2 GPIF trigger                          | x      | x      | x       | x   | x   | x   | x    | x         | xxxxxxx         | W       |
|      | 3    | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
|      |      | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
|      |      | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
| E6DA | 1    | EP4GPIFFLGSEL <sup>[10]</sup>       | Endpoint 4 GPIF flag select                      | 0      | 0      | 0       | 0   | 0   | 0   | FS1  | FS0       | 00000000        | RW      |
| E6DB | 1    | EP4GPIPFSTOP                        | Endpoint 4 GPIF stop transaction on GPIF flag    | 0      | 0      | 0       | 0   | 0   | 0   | 0    | FIFO4FLAG | 00000000        | RW      |
| E6DC | 1    | EP4GPIFTRIG <sup>[10]</sup>         | Endpoint 4 GPIF trigger                          | x      | x      | x       | x   | x   | x   | x    | x         | xxxxxxx         | W       |
|      | 3    | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
|      |      | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
|      |      | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
| E6E2 | 1    | EP6GPIFFLGSEL <sup>[10]</sup>       | Endpoint 6 GPIF flag select                      | 0      | 0      | 0       | 0   | 0   | 0   | FS1  | FS0       | 00000000        | RW      |
| E6E3 | 1    | EP6GPIPFSTOP                        | Endpoint 6 GPIF stop transaction on program flag | 0      | 0      | 0       | 0   | 0   | 0   | 0    | FIFO6FLAG | 00000000        | RW      |
| E6E4 | 1    | EP6GPIFTRIG <sup>[10]</sup>         | Endpoint 6 GPIF trigger                          | x      | x      | x       | x   | x   | x   | x    | x         | xxxxxxx         | W       |
|      | 3    | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
|      |      | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
|      |      | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
| E6EA | 1    | EP8GPIFFLGSEL <sup>[10]</sup>       | Endpoint 8 GPIF flag select                      | 0      | 0      | 0       | 0   | 0   | 0   | FS1  | FS0       | 00000000        | RW      |
| E6EB | 1    | EP8GPIPFSTOP                        | Endpoint 8 GPIF stop transaction on program flag | 0      | 0      | 0       | 0   | 0   | 0   | 0    | FIFO8FLAG | 00000000        | RW      |
| E6EC | 1    | EP8GPIFTRIG <sup>[10]</sup>         | Endpoint 8 GPIF trigger                          | x      | x      | x       | x   | x   | x   | x    | x         | xxxxxxx         | W       |
|      | 3    | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
| E6F0 | 1    | XGPIFSGLDATH                        | GPIF Data H (16-bit mode only)                   | D15    | D14    | D13     | D12 | D11 | D10 | D9   | D8        | xxxxxxx         | RW      |
| E6F1 | 1    | XGPIFSGLDATLX                       | Read/Write GPIF Data L and trigger transaction   | D7     | D6     | D5      | D4  | D3  | D2  | D1   | D0        | xxxxxxx         | RW      |
| E6F2 | 1    | XGPIFSGLDATL-NOX                    | Read GPIF Data L, no transaction trigger         | D7     | D6     | D5      | D4  | D3  | D2  | D1   | D0        | xxxxxxx         | R       |
| E6F3 | 1    | GPIFREADYCFG                        | Internal RDY, sync/async, RDY pin states         | INTRDY | SAS    | TCXRDY5 | 0   | 0   | 0   | 0    | 0         | 00000000        | bbrrrrr |
| 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         | xxxxxxx         | W       |
| E6F6 | 2    | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
|      |      | <b>ENDPOINT BUFFERS</b>             |  |        |        |         |     |     |     |      |           |                 |         |
| E740 | 64   | EP0BUF                              | EP0-IN/-OUT buffer                               | D7     | D6     | D5      | D4  | D3  | D2  | D1   | D0        | xxxxxxx         | RW      |
| E780 | 64   | EP10UTBUF                           | EP1-OUT buffer                                   | D7     | D6     | D5      | D4  | D3  | D2  | D1   | D0        | xxxxxxx         | RW      |
| E7C0 | 64   | EP1INBUF                            | EP1-IN buffer                                    | D7     | D6     | D5      | D4  | D3  | D2  | D1   | D0        | xxxxxxx         | RW      |
| E800 | 2048 | Reserved                            |  |        |        |         |     |     |     |      |           |                 | RW      |
| F000 | 1024 | EP2FIFOBUF                          | 512/1024-byte EP 2/Slave FIFO buffer (IN or OUT) | D7     | D6     | D5      | D4  | D3  | D2  | D1   | D0        | xxxxxxx         | RW      |
| F400 | 512  | EP4FIFOBUF                          | 512 byte EP 4/Slave FIFO buffer (IN or OUT)      | D7     | D6     | D5      | D4  | D3  | D2  | D1   | D0        | xxxxxxx         | RW      |
| F600 | 512  | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
| F800 | 1024 | EP6FIFOBUF                          | 512/1024-byte EP 6/Slave FIFO buffer (IN or OUT) | D7     | D6     | D5      | D4  | D3  | D2  | D1   | D0        | xxxxxxx         | RW      |
| FC00 | 512  | EP8FIFOBUF                          | 512 byte EP 8/Slave FIFO buffer (IN or OUT)      | D7     | D6     | D5      | D4  | D3  | D2  | D1   | D0        | xxxxxxx         | RW      |
| FE00 | 512  | Reserved                            |  |        |        |         |     |     |     |      |           |                 |         |
| xxxx |      | I <sup>2</sup> C Configuration Byte |  | 0      | DISCON | 0       | 0   | 0   | 0   | 0    | 400KHZ    | xxxxxxx<br>[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   |    |
|--|------|---------------------------------|--|-------|-------|---------------------|-------|-------|----------|-----------|--------|----------|----------|----|
| <b>Special Function Registers (SFRs)</b> |      |                                 |  |       |       |                     |       |       |          |           |        |          |          |    |
| 80                                       | 1    | IOA <sup>[13]</sup>             | Port A (bit addressable)                 | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | xxxxxxx  | RW       |    |
| 81                                       | 1    | SP                              | Stack Pointer                            | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | 0000111  | 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 <sup>[13]</sup>            | Data Pointer 1 L                         | A7    | A6    | A5                  | A4    | A3    | A2       | A1        | A0     | 00000000 | RW       |    |
| 85                                       | 1    | DPH1 <sup>[13]</sup>            | Data Pointer 1 H                         | A15   | A14   | A13                 | A12   | A11   | A10      | A9        | A8     | 00000000 | RW       |    |
| 86                                       | 1    | DPS <sup>[13]</sup>             | Data Pointer 0/1 select                  | 0     | 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 <sup>[13]</sup>           | Clock control                            | x     | x     | T2M                 | T1M   | T0M   | MD2      | MD1       | MD0    | 00000001 | RW       |    |
| 8F                                       | 1    | Reserved                        |  |       |       |                     |       |       |          |           |        |          |          |    |
| 90                                       | 1    | IOB <sup>[13]</sup>             | Port B (bit addressable)                 | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | xxxxxxx  | RW       |    |
| 91                                       | 1    | EXIF <sup>[13]</sup>            | External interrupt flags                 | IE5   | IE4   | I <sup>2</sup> CINT | USBNT | 1     | 0        | 0         | 0      | 00001000 | RW       |    |
| 92                                       | 1    | MPAGE <sup>[13]</sup>           | 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 <sup>[13]</sup>       | Autopointer 1 address H                  | A15   | A14   | A13                 | A12   | A11   | A10      | A9        | A8     | 00000000 | RW       |    |
| 9B                                       | 1    | AUTOPTL1 <sup>[13]</sup>        | Autopointer 1 address L                  | A7    | A6    | A5                  | A4    | A3    | A2       | A1        | A0     | 00000000 | RW       |    |
| 9C                                       | 1    | Reserved                        |  |       |       |                     |       |       |          |           |        |          |          |    |
| 9D                                       | 1    | AUTOPTRH2 <sup>[13]</sup>       | Autopointer 2 address H                  | A15   | A14   | A13                 | A12   | A11   | A10      | A9        | A8     | 00000000 | RW       |    |
| 9E                                       | 1    | AUTOPTL2 <sup>[13]</sup>        | Autopointer 2 address L                  | A7    | A6    | A5                  | A4    | A3    | A2       | A1        | A0     | 00000000 | RW       |    |
| 9F                                       | 1    | Reserved                        |  |       |       |                     |       |       |          |           |        |          |          |    |
| A0                                       | 1    | IOC <sup>[13]</sup>             | Port C (bit addressable)                 | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | xxxxxxx  | RW       |    |
| A1                                       | 1    | INT2CLR <sup>[13]</sup>         | Interrupt 2 Clear                        | x     | x     | x                   | x     | x     | x        | x         | x      | xxxxxxx  | W        |    |
| A2                                       | 1    | Reserved                        |  | x     | x     | x                   | x     | x     | x        | x         | x      | xxxxxxx  | 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 <sup>[13]</sup>      | Endpoint 2,4,6,8 status flags            | EP8F  | EP8E  | EP6F                | EP6E  | EP4F  | EP4E     | EP2F      | EP2E   | 01011010 | R        |    |
| AB                                       | 1    | EP24FIFOFLGS <sup>[13]</sup>    | Endpoint 2,4 Slave FIFO status flags     | 0     | EP4PF | EP4EF               | EP4FF | 0     | EP2PF    | EP2EF     | EP2FF  | 00100010 | R        |    |
| AC                                       | 1    | EP68FIFOFLGS <sup>[13]</sup>    | Endpoint 6,8 Slave FIFO status flags     | 0     | EP8PF | EP8EF               | EP8FF | 0     | EP6PF    | EP6EF     | EP6FF  | 01100110 | R        |    |
| AD                                       | 2    | Reserved                        |  |       |       |                     |       |       |          |           |        |          |          |    |
| AF                                       | 1    | AUTOPTRSETUP <sup>[13]</sup>    | Autopointer 1 and 2 Setup                | 0     | 0     | 0                   | 0     | 0     | APTR2INC | APTR1INC  | APTREN | 00000110 | RW       |    |
| B0                                       | 1    | IOD <sup>[13]</sup>             | Port D (bit addressable)                 | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | xxxxxxx  | RW       |    |
| B1                                       | 1    | IOE <sup>[13]</sup>             | Port E (NOT bit addressable)             | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | xxxxxxx  | RW       |    |
| B2                                       | 1    | OEA <sup>[13]</sup>             | Port A Output Enable                     | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | 00000000 | RW       |    |
| B3                                       | 1    | OEB <sup>[13]</sup>             | Port B Output Enable                     | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | 00000000 | RW       |    |
| B4                                       | 1    | OEC <sup>[13]</sup>             | Port C Output Enable                     | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | 00000000 | RW       |    |
| B5                                       | 1    | OED <sup>[13]</sup>             | Port D Output Enable                     | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | 00000000 | RW       |    |
| B6                                       | 1    | OEE <sup>[13]</sup>             | 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 <sup>[13]</sup>        | Endpoint 0 and 1 Status                  | 0     | 0     | 0                   | 0     | 0     | EP1INBSY | EP1OUTBSY | EP0BSY | 00000000 | R        |    |
| BB                                       | 1    | GPIFTRIG <sup>[13, 10]</sup>    | 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 <sup>[13]</sup>     | GPIF Data H (16-bit mode only)           | D15   | D14   | D13                 | D12   | D11   | D10      | D9        | D8     | xxxxxxx  | RW       |    |
| BE                                       | 1    | GPIFSGLDATLX <sup>[13]</sup>    | GPIF Data L w/trigger                    | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | xxxxxxx  | RW       |    |
| BF                                       | 1    | GPIFSGLDATL-NOX <sup>[13]</sup> | GPIF Data L w/no trigger                 | D7    | D6    | D5                  | D4    | D3    | D2       | D1        | D0     | xxxxxxx  | 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 <sup>[13]</sup> | 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 <sup>[13]</sup> | 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 <sup>[13]</sup> | 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 <sup>[13]</sup>   | External Interrupt Enables                   | 1     | 1     | 1     | EX6   | EX5   | EX4   | EI <sup>2</sup> C | EUSB  | 11100000 | RW     |
| E9  | 7    | Reserved              |  |       |       |       |       |       |       |                   |       |          |        |
| F0  | 1    | B                     | B (bit addressable)                          | D7    | D6    | D5    | D4    | D3    | D2    | D1                | D0    | 00000000 | RW     |
| F1  | 7    | Reserved              |  |       |       |       |       |       |       |                   |       |          |        |
| F8  | 1    | EIP <sup>[13]</sup>   | External Interrupt Priority control          | 1     | 1     | 1     | PX6   | PX5   | PX4   | PI <sup>2</sup> C | 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<sup>[14]</sup>

For pins under 1.8 V to 3.3 V Power Domain

(GPIOs) ..... 1.89 V to 3.6 V<sup>[14]</sup>

(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<sub>A</sub> (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 to 1.89 V

Ground Voltage ..... 0 V

F<sub>OSC</sub> (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 <sub>CC</sub>   | 3.3 V supply (to oscillator and PHY) | —  | 3.00                     | 3.3 | 3.60                     | V    |
| V <sub>CC_IO</sub> | 1.8 V to 3.3 V supply (to I/O)       | —  | 1.71                     | 1.8 | 3.60                     | V    |
| V <sub>CC_A</sub>  | 1.8 V supply to analog core          | —  | 1.71                     | 1.8 | 1.89                     | V    |
| V <sub>CC_D</sub>  | 1.8 V supply to digital core         | —  | 1.71                     | 1.8 | 1.89                     | V    |
| V <sub>IH</sub>    | Input HIGH voltage                   | —  | 0.6 * V <sub>CC_IO</sub> | —   | V <sub>CC_IO</sub> +10%  | V    |
| V <sub>IL</sub>    | Input LOW voltage                    | —  | 0                        | —   | 0.3 * V <sub>CC_IO</sub> | V    |
| V <sub>IH_X</sub>  | Crystal input HIGH voltage           | —  | 2.0                      | —   | 3.60                     | V    |
| V <sub>IL_X</sub>  | Crystal input LOW voltage            | —  | -0.5                     | —   | 0.8                      | V    |
|                    | Hysteresis                           | —  | 50                       | —   | —                        | mV   |
| I <sub>I</sub>     | Input leakage current                | 0 < V <sub>IN</sub> < V <sub>CC_IO</sub> | —                        | —   | ±10                      | μA   |
| V <sub>OH</sub>    | Output voltage HIGH                  | I <sub>OUT</sub> = 4 mA                  | V <sub>CC_IO</sub> - 0.4 | —   | —                        | V    |
| V <sub>OL</sub>    | Output voltage LOW                   | I <sub>OUT</sub> = -4 mA                 | —                        | —   | 0.4                      | V    |
| I <sub>OH</sub>    | Output current HIGH                  | —  | —                        | —   | 4                        | mA   |
| I <sub>OL</sub>    | Output current LOW                   | —  | —                        | —   | 4                        | mA   |
| C <sub>IN</sub>    | Input pin capacitance                | Except D+/D-                             | —                        | —   | 10                       | pF   |
|                    |                                      | D+/D-                                    | —                        | —   | 15                       | pF   |
| I <sub>SUSP</sub>  | Suspend current                      | Connected                                | —                        | 220 | 380 <sup>[15]</sup>      | μA   |
|                    |                                      | Disconnected                             | —                        | 20  | 150 <sup>[15]</sup>      | μA   |

### Notes

14. Do not power I/O when chip power is OFF.

15. Measured at maximum V<sub>CC</sub>, 25°C.

**DC Characteristics** (continued)

| Parameter            | Description                            | Conditions                        | Min | Typ | Max | Unit |
|----------------------|--|-----------------------------------|-----|-----|-----|------|
| I <sub>CC_AVcc</sub> | Supply current (AV <sub>CC</sub> )     | 8051 running, connected to USB HS | –   | 15  | 25  | mA   |
|                      |  | 8051 running, connected to USB FS | –   | 10  | 20  | mA   |
| I <sub>CC_IO</sub>   | Supply current (V <sub>CC_IO</sub> )   | 8051 running, connected to USB HS | –   | 3   | 10  | mA   |
|                      |  | 8051 running, connected to USB FS | –   | 1   | 5   | mA   |
| I <sub>CC_CORE</sub> | Supply current (V <sub>CC_CORE</sub> ) | 8051 running, connected to USB HS | –   | 32  | 50  | mA   |
|                      |  | 8051 running, connected to USB FS | –   | 24  | 40  | mA   |
| T <sub>RESET</sub>   | Reset time after valid power           | V <sub>CC</sub> 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<sup>[16]</sup>

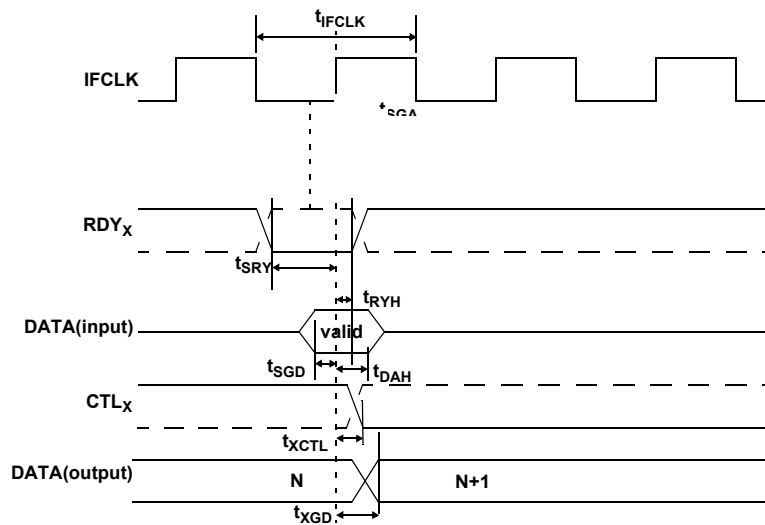


Table 9. GPIF Synchronous Signals Parameters with Internally Sourced IFCLK<sup>[16,17]</sup>

| Parameter          | Description  | Min   | Max | Unit |
|--------------------|--|-------|-----|------|
| t <sub>IFCLK</sub> | IFCLK period                                       | 20.83 | –   | ns   |
| t <sub>SRY</sub>   | RDY <sub>x</sub> to clock setup time               | 8.9   | –   | ns   |
| t <sub>RYH</sub>   | Clock to RDY <sub>x</sub>                          | 0     | –   | ns   |
| t <sub>SGD</sub>   | GPIF data to clock setup time                      | 9.2   | –   | ns   |
| t <sub>DAH</sub>   | GPIF data hold time                                | 0     | –   | ns   |
| t <sub>XGD</sub>   | Clock to GPIF data output propagation delay        | –     | 11  | ns   |
| t <sub>XCTL</sub>  | Clock to CTL <sub>x</sub> output propagation delay | –     | 6.7 | ns   |

**Notes**

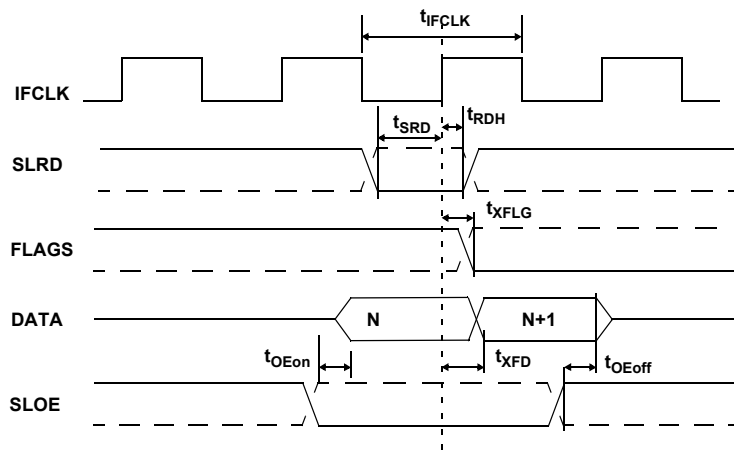
- 16. Dashed lines denote signals with programmable polarity.
- 17. GPIF asynchronous RDY<sub>x</sub> signals have a minimum setup time of 50 ns when using internal 48 MHz IFCLK.

**Table 10. GPIF Synchronous Signals Parameters with Externally Sourced IFCLK<sup>[18]</sup>**

| Parameter   | Description  | Min   | Max   | Unit |
|-------------|--|-------|-------|------|
| $t_{IFCLK}$ | IFCLK period <sup>[19]</sup>                       | 20.83 | 200   | ns   |
| $t_{SRD}$   | RDY <sub>X</sub> to clock setup time               | 2.9   | —     | ns   |
| $t_{RYH}$   | Clock to RDY <sub>X</sub>                          | 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 <sub>X</sub> output propagation delay | —     | 13.06 | ns   |

**Slave FIFO Synchronous Read**

**Figure 9. Slave FIFO Synchronous Read Timing Diagram<sup>[20]</sup>**



**Table 11. Slave FIFO Synchronous Read Parameters with Internally Sourced IFCLK<sup>[18]</sup>**

| 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**

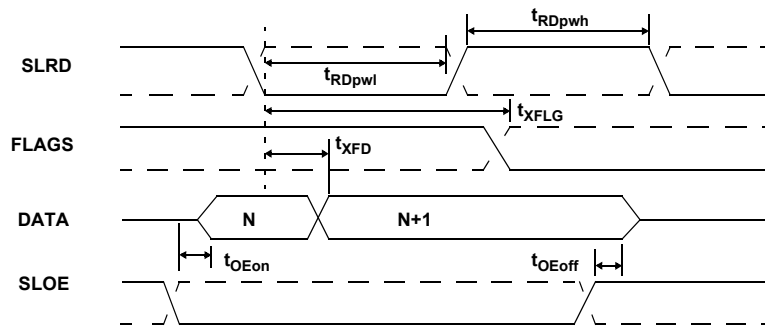
- 18. GPIF asynchronous RDY<sub>X</sub> signals have a minimum setup time of 50 ns when using internal 48 MHz IFCLK.
- 19. IFCLK must not exceed 48 MHz.
- 20. Dashed lines denote signals with programmable polarity.

**Table 12. Slave FIFO Synchronous Read Parameters with Externally Sourced IFCLK<sup>[21]</sup>**

| 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<sup>[22]</sup>**



**Table 13. Slave FIFO Asynchronous Read Parameters<sup>[23]</sup>**

| 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<sup>[24]</sup>

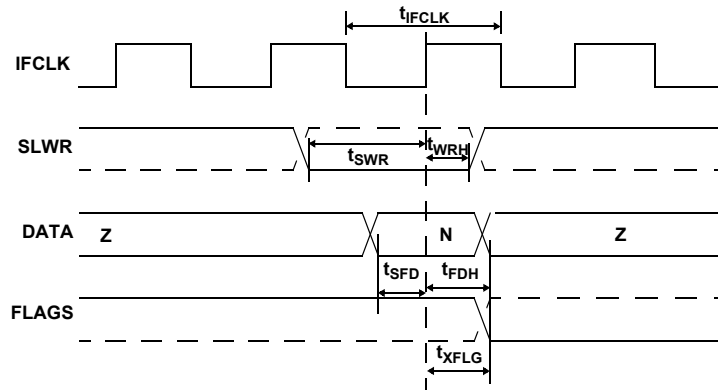


Table 14. Slave FIFO Synchronous Write Parameters with Internally Sourced IFCLK<sup>[25]</sup>

| 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<sup>[26]</sup>

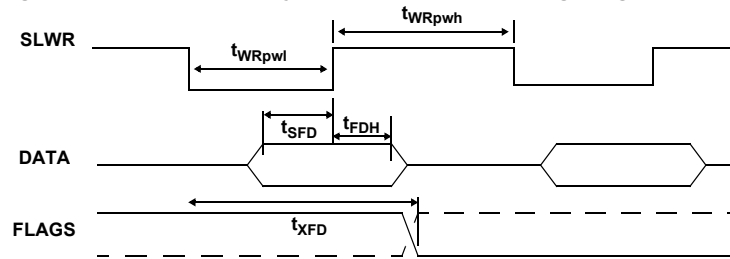
| Parameter   | Description                            | Min   | Max  | Unit |
|-------------|--|-------|------|------|
| $t_{IFCLK}$ | IFCLK period                           | 20.83 | 200  | ns   |
| $t_{SWR}$   | SLWR to clock setup time               | 12.1  | —    | ns   |
| $t_{WRH}$   | Clock to 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<sub>x</sub> 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<sup>[27]</sup>**

**Table 16. Slave FIFO Asynchronous Write Parameters with Internally Sourced IFCLK<sup>[28]</sup>**

| 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<sup>[29]</sup>

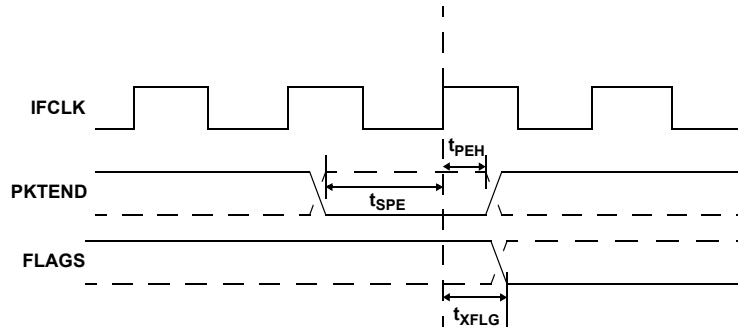


Table 17. Slave FIFO Synchronous Packet End Strobe Parameters with Internally Sourced IFCLK<sup>[30]</sup>

| 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<sup>[30]</sup>

| 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.

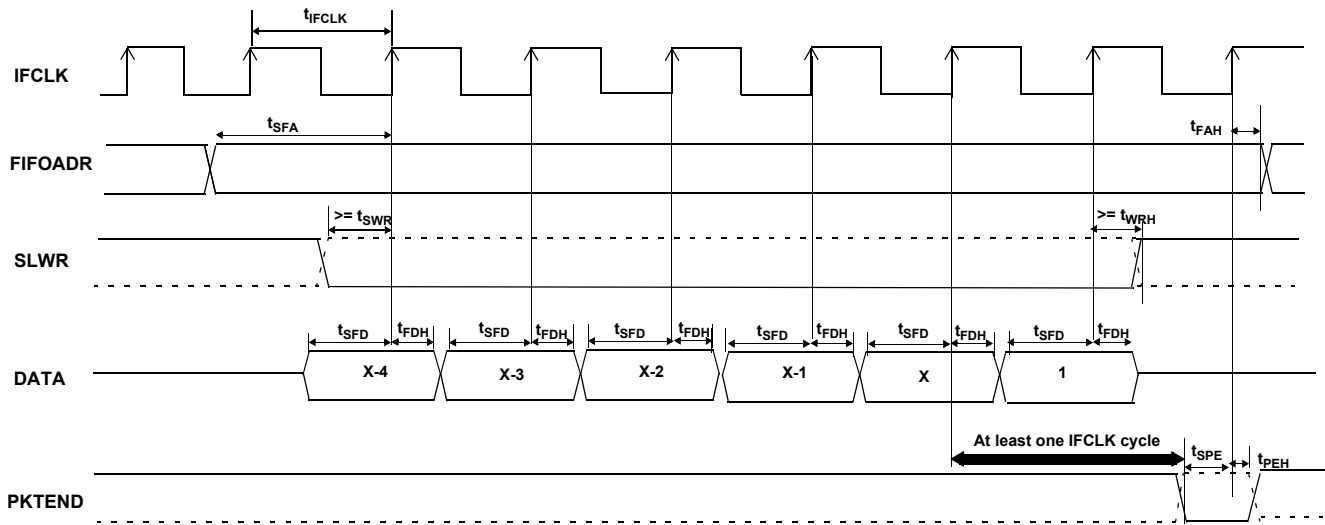
Figure 14 shows a scenario where two packets are committed. The first packet is committed automatically when the number of bytes in the FIFO reaches X (value set in AUTOINLEN register) and the second one byte/word short packet is committed manually using PKTEND. Note that there is at least one IFCLK 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<sup>[31]</sup>



Slave FIFO Asynchronous Packet End Strobe

Figure 15. Slave FIFO Asynchronous Packet End Strobe Timing Diagram<sup>[31]</sup>

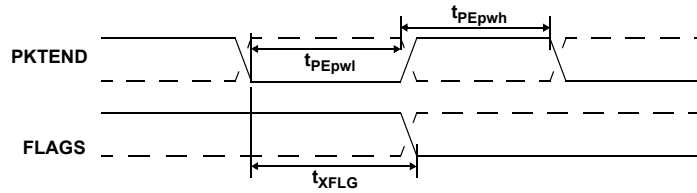


Table 19. Slave FIFO Asynchronous Packet End Strobe Parameters<sup>[32]</sup>

| Parameter | Description                              | Min | Max | Unit |
|-----------|--|-----|-----|------|
| tPEpwl    | PKTEND pulse width LOW                   | 50  | —   | ns   |
| tPEpwh    | PKTEND pulse width HIGH                  | 50  | —   | ns   |
| tXFLG     | PKTEND to FLAGS output propagation delay | —   | 115 | ns   |

Slave FIFO Output Enable

Figure 16. Slave FIFO Output Enable Timing Diagram<sup>[33]</sup>

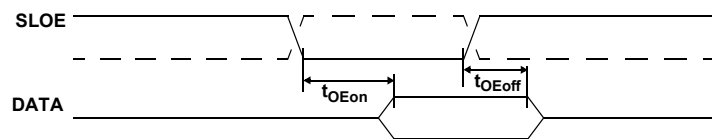


Table 20. Slave FIFO Output Enable Parameters

| Parameter | Description                     | Min  | Max  | Unit |
|-----------|---------------------------------|------|------|------|
| tOEon     | SLOE assert to FIFO data output | —    | 10.5 | ns   |
| tOEoff    | SLOE deassert to FIFO data hold | 2.15 | 10.5 | 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 Address to Flags/Data

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

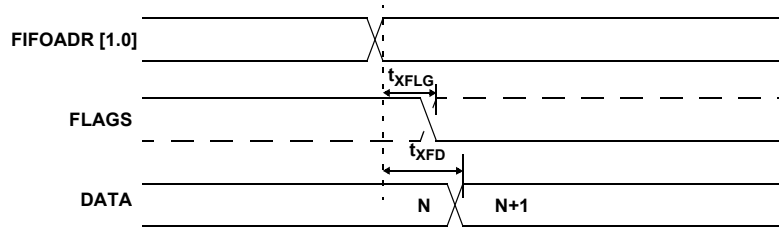


Table 21. Slave FIFO Address to Flags/Data Parameters

| 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   |

Slave FIFO Synchronous Address

Figure 18. Slave FIFO Synchronous Address Timing Diagram<sup>[33]</sup>

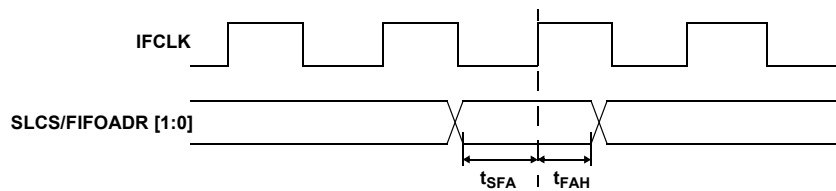
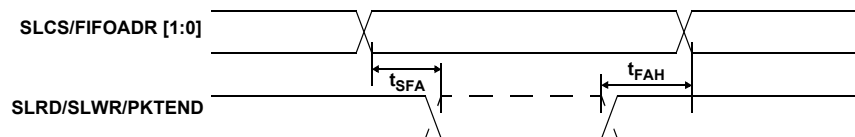


Table 22. Slave FIFO Synchronous Address Parameters<sup>[34]</sup>

| 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<sup>[33]</sup>



Slave FIFO Asynchronous Address Parameters<sup>[35]</sup>

| 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   |

Note

33. Dashed lines denote signals with programmable polarity.

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

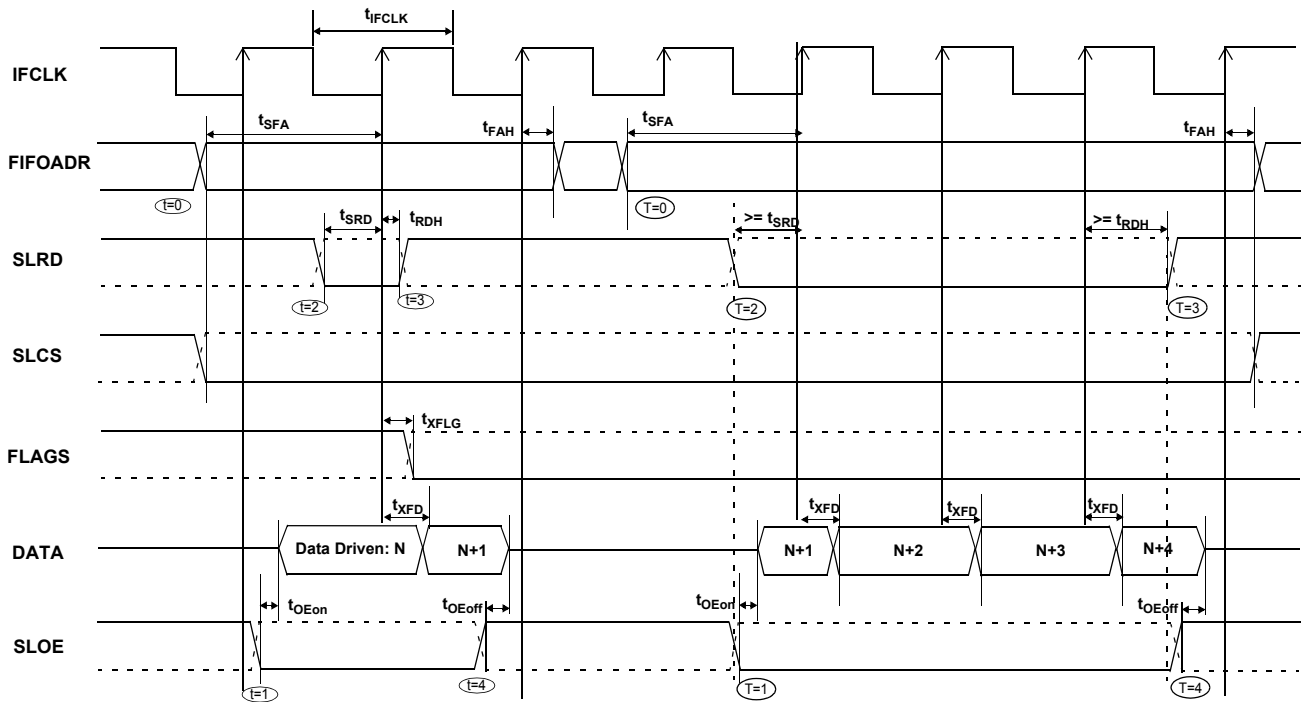
35. Slave FIFO asynchronous parameter values use internal IFCLK setting at 48 MHz.

**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<sup>[36]</sup>**



**Figure 21. Slave FIFO Synchronous Sequence of Events Diagram**

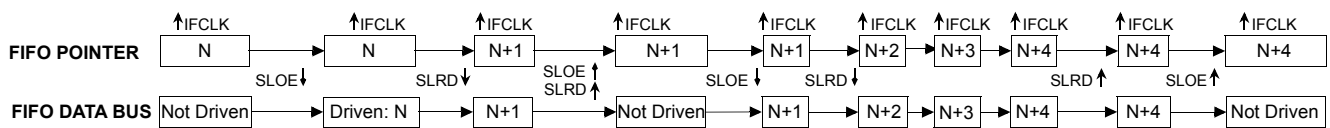


Figure 20 shows the timing relationship of the SLAVE FIFO signals during a synchronous FIFO read using IFCLK as the synchronizing clock. The diagram illustrates a single read followed by a burst read.

- 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.

**Notes**

36. Dashed lines denote signals with programmable polarity.

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.

Single and Burst Synchronous Write

Figure 22. Slave FIFO Synchronous Write Sequence and Timing Diagram<sup>[37]</sup>

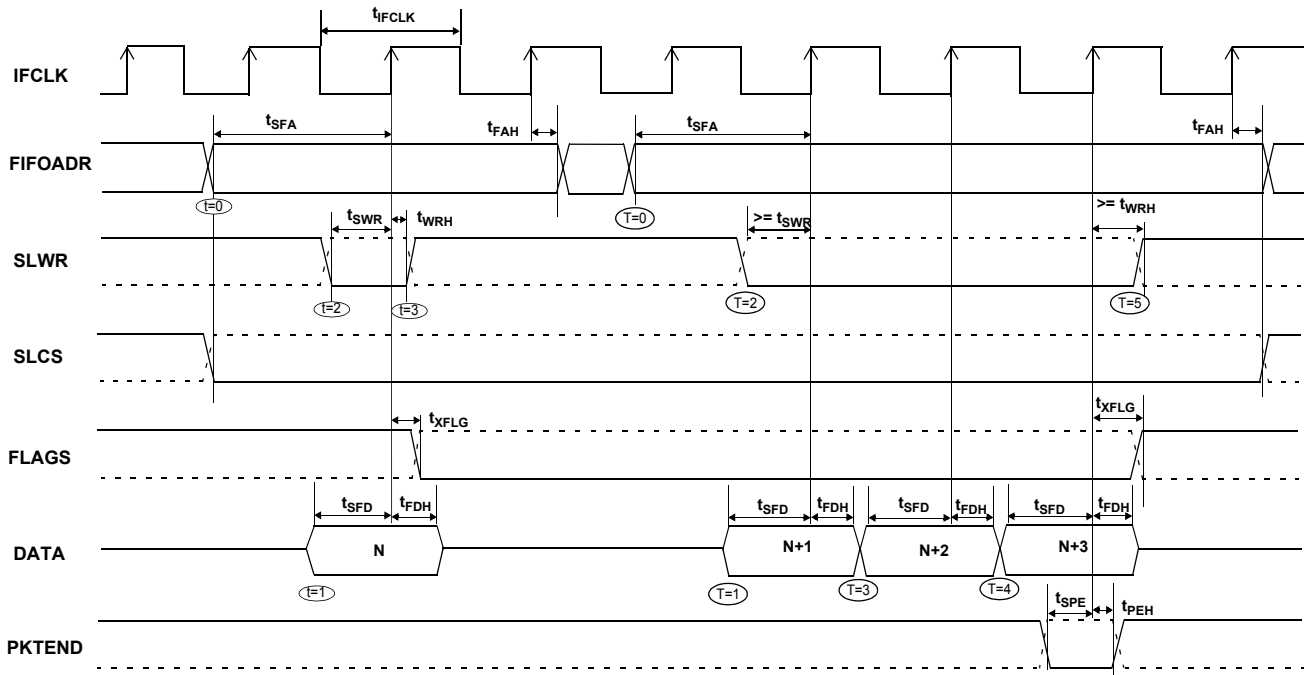


Figure 22 shows the timing relationship of the SLAVE FIFO signals during a synchronous write using IFCLK as the synchronizing clock. The diagram illustrates a single write followed by burst write of 3 bytes and committing all 4 bytes as a short packet using the PKTEND pin.

- 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.

**Note**  
37. Dashed lines denote signals with programmable polarity.

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 34 for further details about this timing.

Sequence Diagram of a Single and Burst Asynchronous Read

Figure 23. Slave FIFO Asynchronous Read Sequence and Timing Diagram<sup>[38]</sup>

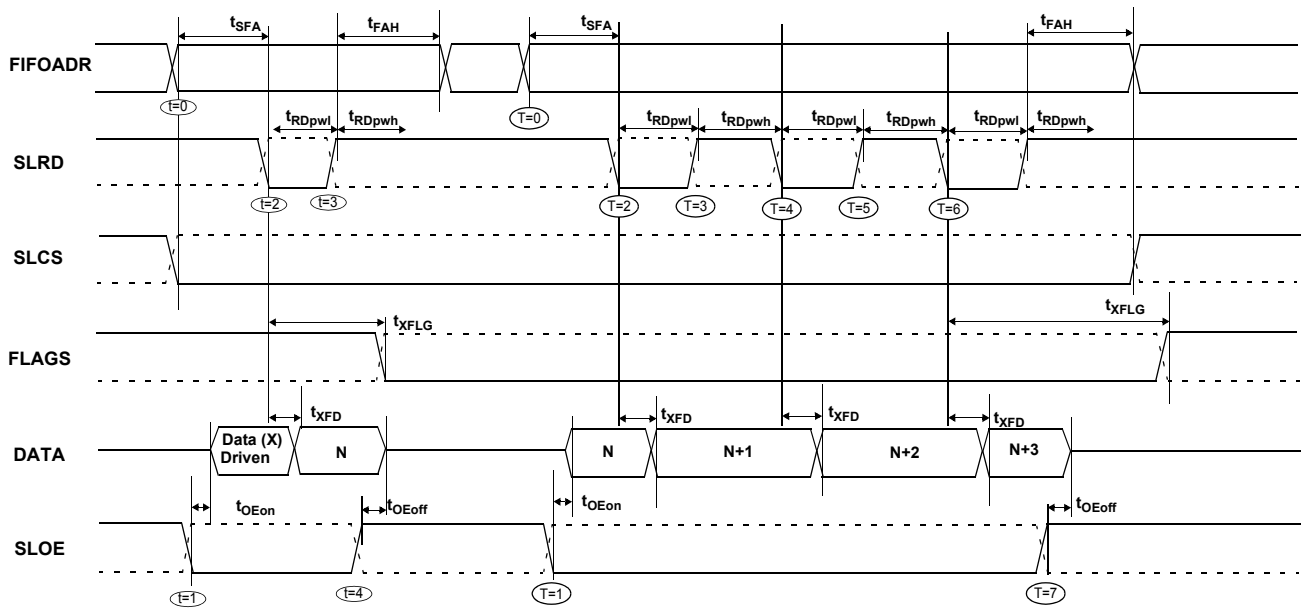


Figure 24. Slave FIFO Asynchronous Read Sequence of Events Diagram

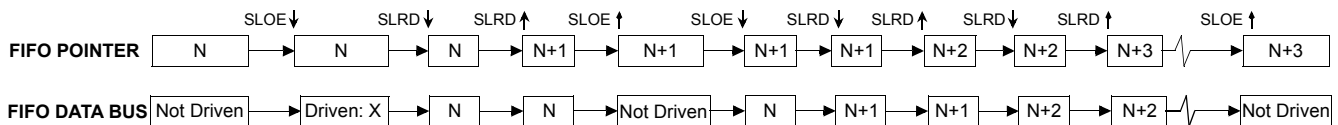


Figure 23 illustrates the timing relationship of the SLAVE FIFO signals during an asynchronous FIFO read. It shows a single read followed by a burst read.

- 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

38. 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<sup>[39]</sup>

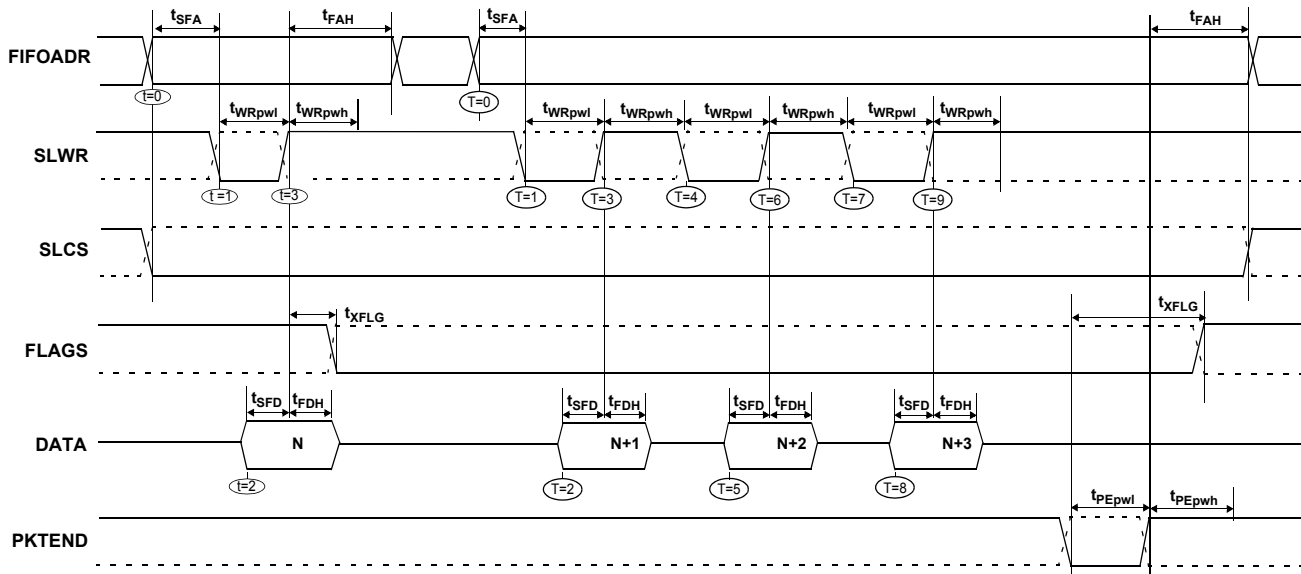


Figure 25 illustrates the timing relationship of the SLAVE FIFO write in an asynchronous mode. The diagram shows a single write followed by a burst write of 3 bytes and committing the 4-byte-short packet using PKTEND.

- 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**  
39. Dashed lines denote signals with programmable polarity.

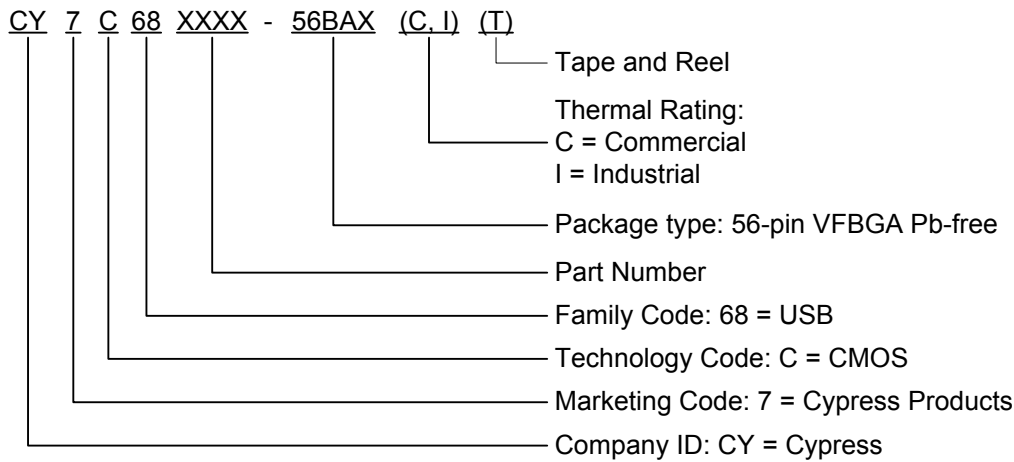
## Ordering Information

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](http://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                  | –                       |
| <b>Development Tool Kit</b> |                                  |          |                     |                         |
| CY3687                      | MoBL-USB FX2LP18 Development Kit |          |                     |                         |

## Ordering Code Definitions

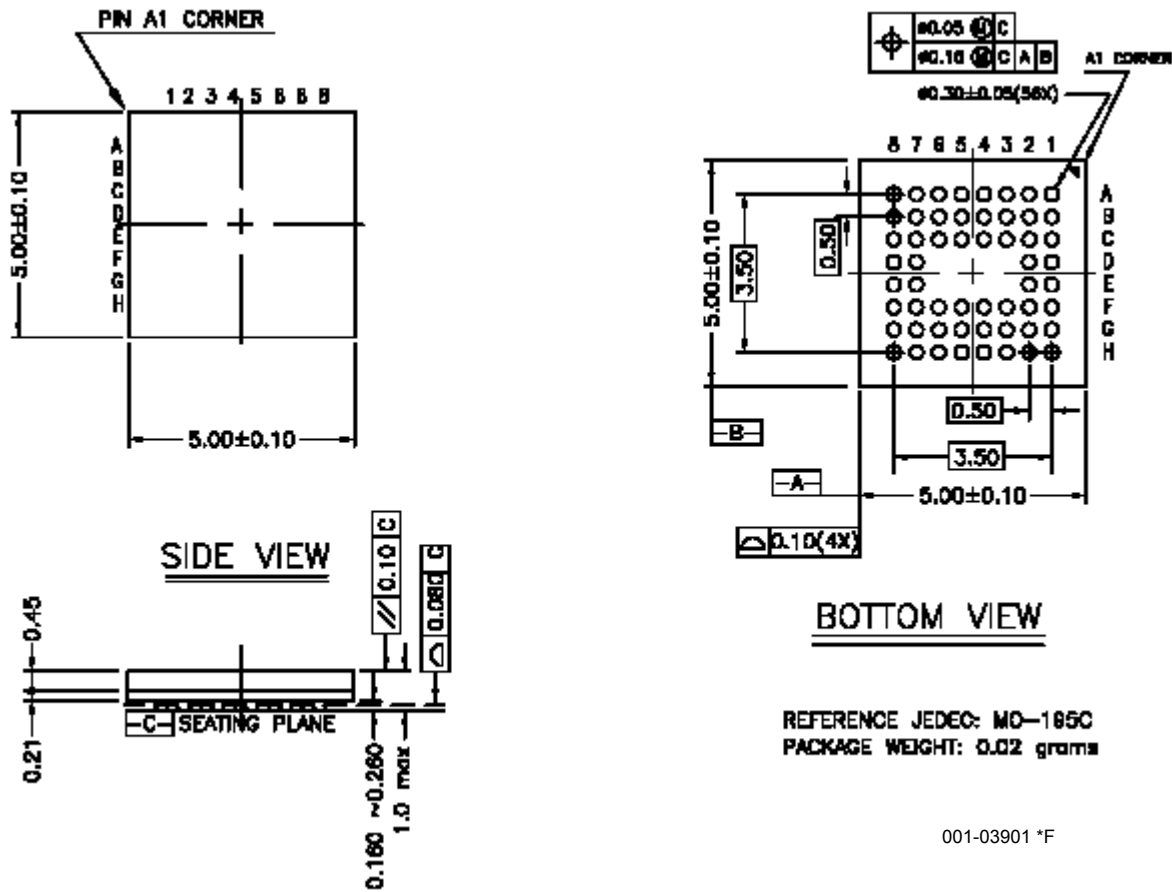




## Package Diagram

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<br>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 <sub>CC</sub> or V <sub>CC_IO</sub><br>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 <sup>2</sup> C interface<br>In pin description table, renamed pin 2H (Reserved) to Ground<br>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"<br>In Section 8, added a footnote to the DC char table stating that AV <sub>cc</sub> can be floated in low power mode<br>In Section 8, changed V <sub>IH</sub> max in DC char table from 3.6 V to V <sub>CC_IO</sub> + 10% |
| *B   | 465471  | OSG             | See ECN         | Changed the recommendation for the pull up resistors on I <sup>2</sup> C<br>Split I <sub>cc</sub> into 4 different values, corresponding to the different voltage supplies<br>Changed I <sub>sus</sub> typical to 20uA and 220uA<br>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 I <sub>cc</sub> 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.<br>Added min spec for t <sub>OEOFF</sub><br>Changed I <sub>cc</sub> and power dissipation numbers   |
| *G   | 1128404 | OSG / ARI       | See ECN         | Removed SLCS from figure in Section 9.6 Slave FIFO Asynchronous Write<br>Changed SLWR Pulse HIGH parameter to 50ns<br>Section 9.13.1 9 V Removed the indication that SLCS and SLRD can be asserted together<br>Section 9.13.3 - Removed the indication that SLCS and SLRD can be asserted together<br>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 AV <sub>cc</sub> parameter in DC Characteristics table   |
| *J   | 3072698 | ODC             | 10/27/10        | Template update and styles update.<br>Included Table of Contents, Ordering Code Definitions, Acronyms, and Units of measure.<br>Updated package diagram revision from *B to *C.  |
| *K   | 4212673 | DBIR            | 12/06/2013      | Updated <a href="#">Package Diagram</a> :<br>spec 001-03901 – Changed revision from *C to *E.<br>Updated in new template.<br>Completing Sunset Review.   |
| *L   | 4791911 | MDDD            | 10/28/2015      | Added <a href="#">More Information</a> .<br>Updated <a href="#">Figure 26</a> (spec 001-03901 *E to *F) in <a href="#">Package Diagram</a> .   |
| *M   | 5718683 | AESATMP7        | 04/28/2017      | Updated Cypress Logo and Copyright.  |

## 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

|  |  |
|--|--|
| <a href="#">Automotive</a>                   | <a href="http://cypress.com/go/automotive">cypress.com/go/automotive</a> |
| <a href="#">Clocks &amp; Buffers</a>         | <a href="http://cypress.com/go/clocks">cypress.com/go/clocks</a>         |
| <a href="#">Interface</a>                    | <a href="http://cypress.com/go/interface">cypress.com/go/interface</a>   |
| <a href="#">Lighting &amp; Power Control</a> | <a href="http://cypress.com/go/powerpsoc">cypress.com/go/powerpsoc</a>   |
| <a href="#">Memory</a>                       | <a href="http://cypress.com/go/memory">cypress.com/go/memory</a>         |
| <a href="#">PSoC</a>                         | <a href="http://cypress.com/go/psoc">cypress.com/go/psoc</a>             |
| <a href="#">Touch Sensing</a>                | <a href="http://cypress.com/go/touch">cypress.com/go/touch</a>           |
| <a href="#">USB Controllers</a>              | <a href="http://cypress.com/go/USB">cypress.com/go/USB</a>               |
| <a href="#">Wireless/RF</a>                  | <a href="http://cypress.com/go/wireless">cypress.com/go/wireless</a>     |

#### PSoC® Solutions

[psoc.cypress.com/solutions](http://psoc.cypress.com/solutions)  
[PSoC 1](#) | [PSoC 3](#) | [PSoC 4](#) | [PSoC 5LP](#) | [PSoC 6](#)

#### Cypress Developer Community

[Community](#) | [Forums](#) | [Blogs](#) | [Video](#) | [Training](#)

#### Technical Support

[cypress.com/go/support](http://cypress.com/go/support)

© Cypress Semiconductor Corporation, 2006-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

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

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