

请注意赛普拉斯已正式并入英飞凌科技公司。

此封面页之后的文件标注有“赛普拉斯”的文件即该产品为此公司最初开发的。请注意作为英飞凌产品组合的部分,英飞凌将继续为新的及现有客户提供该产品。

### 文件内容的连续性

事实是英飞凌提供如下产品作为英飞凌产品组合的部分不会带来对于此文件的任何变更。未来的变更将在恰当的时候发生,且任何变更将在历史页面记录。

### 订购零件编号的连续性

英飞凌继续支持现有零件编号的使用。下单时请继续使用数据表中的订购零件编号。

**EZ-USB® FX2LP™ USB 微控制器高速  
USB 外设控制器****特性**

- 已验证的 USB 2.0 USB IF 高速模式 (TID # 40460272)
- 单一芯片集成式 USB 2.0 收发器、智能 SIE 和增强型 8051 微处理器
- 适用性、外观和功能与 FX2 兼容
  - 可兼容引脚
  - 可兼容对象代码
  - 可兼容功能 (FX2LP 是超集)
- 超低功耗:  $I_{CC}$  在任何模式下均不超过 85 mA
  - 适用于总线和电池供电应用
- 软件: 8051 代码从以下部件中运行:
  - 内部 RAM, 8051 代码通过 USB 下载
  - 内部 RAM, 从 EEPROM 加载 8051 代码
  - 外部存储器器件 (128 引脚封装)
- 16 KB 的片上代码 / 数据 RAM
- 4 个可编程 BULK、INTERRUPT 和 ISOCHRONOUS 端点
  - 缓冲选项: 双重、三重和四立体
- 其他可编程 (BULK/INTERRUPT) 64 字节端点
- 8 位或 16 位外部数据接口
- 智能媒体标准 ECC 生成
- GPIF™ (通用可编程接口)
  - 直接连接到大多数并行接口
  - 可编程波形描述符和配置寄存器以用来定义波形
  - 支持多种就绪 (RDY) 输入和控制 (CTL) 输出
- 集成式工业级标准的增强型 8051
  - 48 MHz、24 MHz 或 12 MHz CPU 工作频率
  - 每个指令周期有 4 个时钟
  - 2 个 USART
  - 3 个计数器 / 定时器
  - 扩展的中断系统
  - 2 个数据指针

- 3.3 V 工作电压, 5 V 容限输入
- 矢量式 USB 中断和 GPIF/FIFO 中断
- CONTROL 传输设置和数据部分的单独数据缓冲区
- 集成式 I<sup>2</sup>C 控制器, 运行频率达 100 或 400 kHz
- 4 个集成式 FIFO
  - 集成式胶连逻辑和 FIFO 较低系统成本
  - 16 位总线自动切换
  - 主控或从器件操作
  - 使用外部时钟或异步探针
  - 易于连接到 ASIC 和 DSP IC 接口
- 可在商业级和工具级温度下使用 (除 VFBGA 以外的所有封装)

**特性 (仅限 CY7C68013A/14A)**

- CY7C68014A: 适用于电池供电应用
  - 暂停电流: 100  $\mu$ A (典型)
- CY7C68013A: 适用于非电池供电应用
  - 暂停电流: 300  $\mu$ A (典型)
- 适用于 5 个高达 40 GPIO 的无铅封装
  - 128 引脚 TQFP (40 GPIO), 100 引脚 TQFP (40 GPIO), 56 引脚 QFN (24 GPIO), 56 引脚 SSOP (24 GPIO) 和 56 引脚 VFBGA (24 GPIO)

**特性 (仅限 CY7C68015A/16A)**

- CY7C68016A: 适用于电池供电应用
  - 暂停电流: 100  $\mu$ A (典型)
- CY7C68015A: 适用于非电池供电应用
  - 暂停电流: 300  $\mu$ A (典型)
- 适用于无铅 56 引脚 QFN 封装 (26 GPIO)
- 比 CY7C68013A/14A 多 2 个 GPIO

相关资源的完整列表, 请点击 [here](#)。

**勘误表:** 芯片勘误表信息, 请参阅第64页勘误表, 包括触发条件, 受影响器件, 以及建议解决方法。

## 更多信息

赛普拉斯在 [www.cypress.com](http://www.cypress.com) 为您提供了丰富的数据, 以帮助您在设计中选择正确的器件, 从而快速有效的把器件集成到您的设计之中。相关资源的全面列表, 请参阅应用笔记 [AN65209-Getting Started with FX2LP](#)。

### ■ 概览: [USB 集合](#), [USB Roadmap](#)

### ■ USB 2.0 产品选择: [FX2LP](#), [AT2LP](#), [NX2LP-Flex](#), [SX2](#)

### ■ 应用手册: 赛普拉斯提供了大量的USB应用手册, 主题覆盖了从初级到高级的广大范围。推荐 FX2LP 用户从下列手册起步:

- [AN65209-Getting Started with FX2LP](#)
- [AN15456-Guide to successful EZ-USB® FX2LP™ and EZ-USB FX1™ Hardware Design and Debug](#)
- [AN50963-EZ-USB™ FX1™/FX2LP™ Boot Options](#)
- [AN66806-EZ-USB™ FX2LP™ GPIF Design Guide](#)
- [AN61345-Implementing an FX2LP™-FPGA Interface](#)
- [AN57322-Interfacing SRAM with FX2LP over GPIF](#)
- [AN40353-Streaming Data through Isochronous/Bulk End-points on EZ-USB® FX2 and EXUSB FX2LP](#)
- [AN63787-EZ-USB® FX2LP™ GPIF and Slave FIFO Configuration Examples using 8-bit Asynchronous Interface](#)

完整的应用手册列表, 请点击 [here](#)。

### ■ 代码范例:

- [USB Hi-Speed](#)

### ■ 技术参考手册:

- [EX-USB FX2LP Technical Reference Manual](#)

### ■ 参考设计:

- [CY4661-External USB Hard Disk Drives \(HDD\) with Finger-print Authentication Security](#)
- [FX2LP DM-T/H TV Dongle reference design](#)

### ■ Models: [IBIS](#)

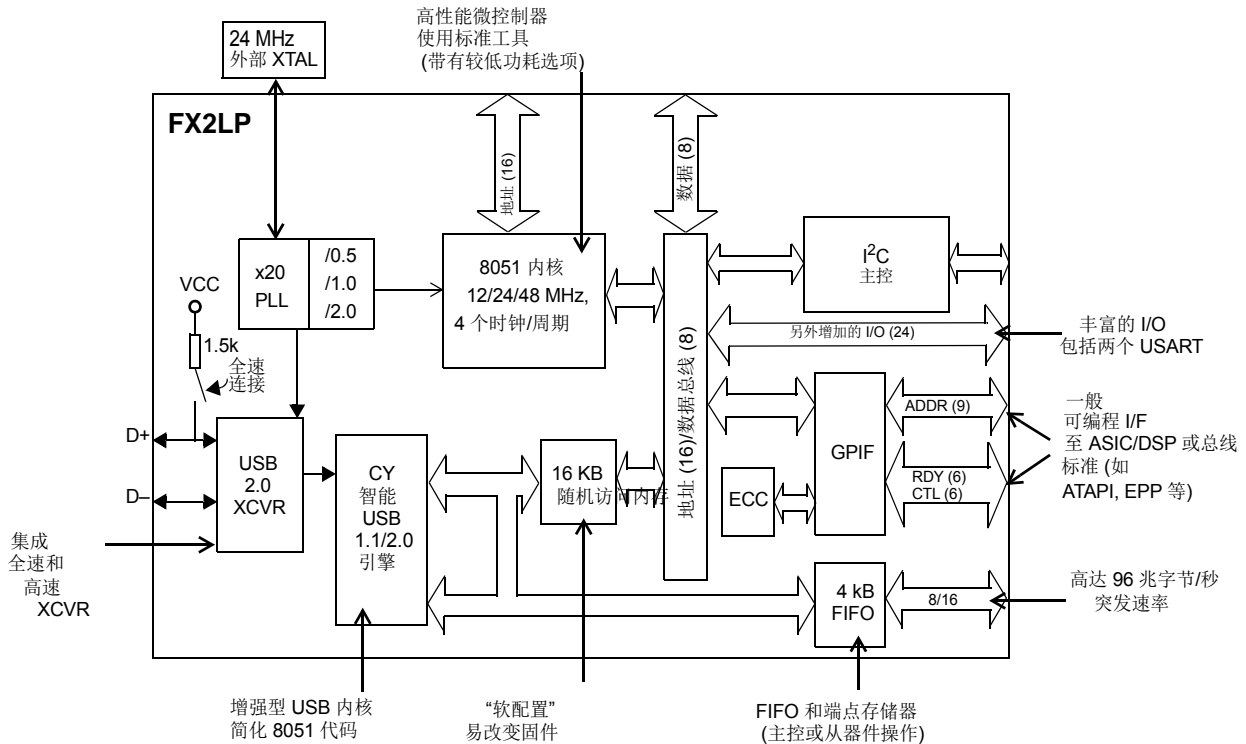
### EX-USB FX2LP 开发套件

[CY3684 EZ-USB FX2LP 开发套件](#) 包含 FX2LP 开发的完整资源。它提供了使用 FX2LP 的客户项目的开发测试平台。开发套件包含了使用 FX2LP 的固件, 硬件, 以及设计中软件相关资源。

### GPIF™ Designer

FX2LP™ 通用可编程接口提供了独立的硬件单元, 可以产生外部接口需要的数据和控制信号。FX2LP GPIF Designer 允许用户为 EZ-USB FX2/FX2LP 家族芯片, 使用图形用户界面, 创建和修改 GPIF 波形描述符。通用 GPIF 讨论的延伸和使用 GPIF Designer 编程包含在随 GPIF Designer 发行的 [FX2LP Technical Reference Manual](#) 和 GPIF Designer User Guide。[AN66806-Getting Started with EZ-USB® FX2LP™](#) 是很好的起点。

## 逻辑框图



赛普拉斯 EZ-USB<sup>®</sup> FX2LP<sup>™</sup> (CY7C68013A/14A) 是 EZ-USB FX2<sup>™</sup> (CY7C68013) 低功耗版本, 它是高度集成、低功耗 USB 2.0 的微控制器。通过在单个芯片上集成 USB 2.0 收发器、串行接口引擎 (SIE)、增强型 8051 微控制器和可编程外设接口,

赛普拉斯已创建成本效益解决方案, 具有迅速投放市场这一优势, 低功耗可以实现总线供电应用。

技术精湛的 FX2LP 架构提供每秒 53 兆字节以上的数据传输速率, 支持最大为 USB 2.0 的带宽, 而仍然使用低成本的 8051 型微控制器, 其封装大小为 56 VFBGA (5 mm x 5 mm)。由于它采用了 USB 2.0 收发器, 因此 FX2LP 更经济, 体积小于 USB 2.0 SIE 或外部操作的收发器。通过 EZ-USB FX2LP, 赛普拉斯智能 SIE (CSS) 可以处理大多数 USB 1.1 和 2.0 硬件协议, 解除应用

特定功能的嵌入式微控制器的限制, 缩短开发时间以确保 USB 的兼容性。

通用可编程接口 (GPIF) 和主控 / 从器件端点 FIFO (8 位或 16 位数据总线) 提供简易的无缝接口, 可以与其他通用接口连接, 例如, ATA, UTOPIA, EPP, PCMCIA 和许多 DSP/ 处理器。

FX2LP 消耗的电流低于 FX2 (CY7C68013), 具有双倍片上代码 / 数据 RAM, 在适用性、外观和功能上与 56、100 和 128 引脚 FX2 一致。

该系列定义了 5 个封装: 56 VFBGA, 56 SSOP, 56 QFN, 100 TQFP 和 128 TQFP。

## 目录

应用 .....	5	数据存储器读取 .....	41
功能概述 .....	5	数据存储器写入 .....	42
USB 信号速度 .....	5	PORTC 探针特性时序 .....	43
8051 微处理器 .....	5	GPIF 同步信号 .....	44
I <sup>2</sup> C 总线 .....	5	从器件 FIFO 同步读取 .....	45
总线 .....	5	从器件 FIFO 异步读取 .....	46
USB 引导方法 .....	6	从器件 FIFO 同步写入 .....	47
ReNumeration .....	6	从器件 FIFO 异步写入 .....	47
总线供电应用 .....	6	从器件 FIFO 同步数据包结束探针 .....	48
中断系统 .....	6	从器件 FIFO 异步数据包结束探针 .....	49
复位和唤醒 .....	9	从器件 FIFO 输出使能 .....	49
程序 / 数据 RAM .....	10	从器件 FIFO 标志 / 数据寻址 .....	50
寄存器地址 .....	11	从器件 FIFO 同步地址 .....	50
端点 RAM .....	12	从器件 FIFO 异步地址 .....	50
外部 FIFO 接口 .....	13	序列图 .....	51
GPIF .....	14	订购信息 .....	55
ECC 生成 [8] .....	14	订购代码定义 .....	55
USB 上载和下载 .....	14	封装图 .....	56
自动指针访问 .....	14	PCB 布局建议 .....	61
I <sup>2</sup> C 控制器 .....	14	四方扁平封装无引脚 (QFN) 封装设计说明 .....	62
与上一代 EZ-USB FX2 兼容 .....	15	缩略语 .....	63
CY7C68013A/14A 和 CY7C68015A/16A 的区别 .....	15	文档规范 .....	63
引脚分配 .....	16	<b>勘误表 .....</b>	<b>64</b>
CY7C68013A/15A 引脚描述 .....	23	文档修订记录页 .....	65
寄存器摘要 .....	31	销售、解决方案和法律信息 .....	66
绝对最大额定值 .....	38	全球销售和 design 支持 .....	66
运行条件 .....	38	产品 .....	66
热特性 .....	38	PSoC® 解决方案 .....	66
直流电特性 .....	39	赛普拉斯开发者社区 .....	66
USB 收发器 .....	39	技术支持 .....	66
交流电气特性 .....	40		
USB 收发器 .....	40		
程序存储器读取 .....	40		

## 应用

- 便携式视频记录仪
- MPEG/TV 转换
- DSL 调制解调器
- ATA 接口
- 存储器读卡器
- 传统的转换器件
- 摄像机
- 扫描仪
- 无线局域网
- MP3 播放器
- 网络

赛普拉斯网站“参考设计”这一节提供典型 USB 2.0 应用的其他工具。每种参考设计附带固件源和目标代码、原理图和文档。有关更多信息，请访问网站 [www.cypress.com](http://www.cypress.com)。

## 功能概述

### USB 信号速度

FX2LP 以两种速率 (共 3 种) 运行，在 USB 规范 2.0 版本 (2000 年 4 月 27 日) 中定义了这两种速率：

- 全速，信号位速率为 12 Mbps
- 高速，信号位速率为 480 Mbps

FX2LP 不支持低速信号模式 1.5 Mbps。

### 8051 微处理器

FX2LP 系列内嵌式 8051 微处理器包含 256 字节的寄存器 RAM、扩展的中断系统、3 个定时器 / 计数器和 2 个 USART。

#### 8051 时钟频率

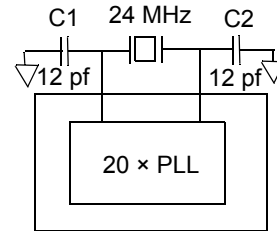
FX2LP 具有片上振荡器电路，该电路使用 24 MHz ( $\pm 100$  ppm) 外部晶振，其特征如下：

- 并联谐振
- 基本模式
- 500  $\mu$ W 驱动电平
- 12 pF (5% 容差) 负载的电容

按照收发器 /PHY 的要求，片上 PLL 成倍数增加 24 MHz 振荡器，最高达到 480 MHz，并且，内部计数器分频使用 8051 时钟。

8051 时钟频率默认值为 12 MHz。8051 时钟频率可能由 8051 通过 CPUCS 寄存器完成动态更改。

图 1. 晶振配置



12 pF 电容假设 4 层 FR4 PCA 每边 3 pF 走线电容

CLKOUT 引脚，三态，使用内部控制位反转，输入 50% 占空比 8051 时钟 (选定的 8051 时钟频率)：48 MHz, 24 MHz 或 12 MHz。USART

FX2LP 包含两个标准的 8051 USART，通过特殊功能寄存器 (SFR) 位来寻址 USART 接口引脚可以用于单独的 I/O 引脚，但不是带有端口引脚的复用式接口。

UART0 和 UART1 使用内部时钟以 230 KBaud 的频率运行，波特率误差不超过 1%。由内部衍生的时钟源实现 230 KBaud 的运行频率，这一时钟源适时生成溢出脉冲。内部时钟根据 8051 时钟频率来调整频率 (48 MHz, 24 MHz 和 12 MHz)，这样，它便始终以正确频率 230 KBaud 来运行。[1]

#### 特殊功能寄存器

包含某些 8051 SFR 地址，以便快速访问关键的 FX2LP 功能。这些新增的 SFR 地址如第 6 页的表 1 所示。粗体字表示非标准的增强型 8051 寄存器。以“0”和“8”结尾的两行 SFR 包含位寻址寄存器。4 个 I/O 端口 A-D 使用 SFR 地址，这是在端口 0-3 的标准 8051 中使用的地址，而尚未在 FX2LP 中实现。由于 SFR 寻址模式不但较快，而且更加有效，因此 FX2LP I/O 端口不在外部 RAM 空间 (使用 MOVX 指令) 中寻址。

### I<sup>2</sup>C 总线

FX2LP 仅在 100/400 kHz 时才支持作为主控的 I<sup>2</sup>C 总线。SCL 和 SDA 引脚具有开漏输出和迟滞输入。即使未连接 I<sup>2</sup>C 器件，这些信号也应必须上拉到 3.3V。

### 总线

所有封装、8 位或 16 位双向数据总线、复用式 I/O 端口 B 和 D。128 引脚封装：添加 16 位仅输出 8051 地址总线、8 位双向数据总线。

#### 注

1. 通过编程将 8051 SMOD0 或 SMOD1 位设为供 UART0, UART1 同时使用或分别使用的“1”位，则可能实现 115 KB 波特的操作。

**表 1. 特殊功能寄存器**

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	INT4CLR	OEA	—	—	—	—
3	DPH0	—	—	OEB	—	—	—	—
4	DPL1	—	—	OEC	—	—	—	—
5	DPH1	—	—	OED	—	—	—	—
6	DPS	—	—	OEE	—	—	—	—
7	PCON	—	—	—	—	—	—	—
8	TCON	SCON0	IE	IP	T2CON	EICON	EIE	EIP
9	TMOD	SBUF0	—	—	—	—	—	—
A	TL0	AUTOPTRH1	EP2468STAT	EP01STAT	RCAP2L	—	—	—
B	TL1	AUTOPTRL1	EP24FIFOFLGS	GPIFTRIG	RCAP2H	—	—	—
C	TH0	保留	EP68FIFOFLGS	—	TL2	—	—	—
D	TH1	AUTOPTRH2	—	GPIFSGLDATH	TH2	—	—	—
E	CKCON	AUTOPTRL2	—	GPIFSGLDATLX	—	—	—	—
F	—	保留	AUTOPTRSET-UP	GPIFSGLDATLNOX	—	—	—	—

## USB 引导方法

在加电顺序中，内部逻辑检查 EEPROM 连接的 I<sup>2</sup>C 端口，其首个字节为 0xC0 或 0xC2。若找到，则使用 EEPROM 中的 VID/PID/DID 值来替换内部存储值 (0xC0)，或使用引导方法加载 EEPROM 内容到 RAM (0xC2) 中。如果未检测到 EEPROM，则 FX2LP 使用内部存储的描述符进行枚举。FX2LP ID 默认值为 VID/PID/DID (0x04B4, 0x8613, 0xAxxx，其中，xxx = 芯片修订版)。[2]

**表 2. FX2LP ID 默认值**

默认 VID/PID/DID		
供货商 ID	0x04B4	赛普拉斯半导体
产品 ID	0x8613	EZ-USB FX2LP
器件释放	0xAxxx	根据芯片修订版 (nnn = 芯片修订版，其中，首个 芯片 = 001)

## ReNumeration

由于 FX2LP 为软配置，一个芯片具有多个不同 USB 器件的特征。

第一次插入 USB 时，FX2LP 自动枚举并通过 USB 电缆下载固件和 USB 描述符表。第二次，FX2LP 再次枚举，这次作为下载信息定义的器件。这个专有的两步流程称作 ReNumeration™，在器件插入时即时发生，不提示初始下载步骤是否发生。

USBCS 中的两个控制位 (USB 控制和状态) 寄存器控制 ReNumeration 流程 DISCON 和 RENUM。要模拟 USB 断开，该固件将 DISCON 设置为 1。要重新连接，该固件将 DISCON 清除为 0。

重新连接前，该固件设置或清除 RENUM 位以表示该固件或默认 USB 器件是否处理端点 0 上的器件请求：若 RENUM = 0，默认 USB 器件处理器件请求；若 RENUM = 1，固件处理该请求。

## 总线供电应用

FX2LP 可以按照 USB 2.0 规范的要求使用低于 100 mA 的电流进行枚举，从而全面支持总线供电设计。

## 中断系统

### INT2 中断请求和使能寄存器

FX2LP 实现了 INT2 和 INT4 自动矢量特性。共有 27 个 INT2 (USB) 矢量和 14 个 INT4 (FIFO/GPIF) 矢量。更多信息，请参见《EZ-USB 技术参考手册》(TRM)。

### USB 中断自动矢量

27 个中断源共同占用 USB 主中断。要保存代码和用来识别单个 USB 中断源所需的处理时间，FX2LP 提供第二级中断矢量，称为自动矢量化。激活 USB 中断时，FX2LP 将程序计数器推至堆栈，然后跳转到地址 0x0043 上，在此期望找到“跳转”至 USB 中断服务例程的指令。

## 注

- I<sup>2</sup>C 总线 SCL 和 SDA 引脚必须上拉，即使 EEPROM 未连接也如此。否则，这种检测方法不会正常工作。



FX2LP 跳转指令编码如下所示:

**表 3. INT2 USB 中断**

INT2 的 USB 中断表			
优先级	INT2VEC 值	Source (源)	注
1	00	SUDAV	设置数据可用
2	04	SOF	启动帧 (或微帧)
3	08	SUTOK	已接收设置令牌
4	0C	SUSPEND	USB 暂停请求
5	10	USB 复位	总线复位
6	14	HISPEED	输入高速操作
7	18	EP0ACK	FX2LP ACK'd 同步交换控制
8	1C		已保留
9	20	EP0-IN	EP0-IN 准备加载数据
10	24	EP0-OUT	EP0-OUT 包含 USB 数据
11	28	EP1-IN	EP1-IN 准备加载数据
12	2C	EP1-OUT	EP1-OUT 包含 USB 数据
13	30	EP2	IN: 缓冲区可用 OUT: 缓冲区包含数据
14	34	EP4	IN: 缓冲区可用 OUT: 缓冲区包含数据
15	38	EP6	IN: 缓冲区可用 OUT: 缓冲区包含数据
16	3C	EP8	IN: 缓冲区可用 OUT: 缓冲区包含数据
17	40	IBN	IN-Bulk-NAK (任何 IN 端点)
18	44		已保留
19	48	EP0PING	EP0 OUT 已发送但已被拒绝
20	4C	EP1PING	EP1 OUT 已发送但已被拒绝
21	50	EP2PING	EP2 OUT 已发送但已被拒绝
22	54	EP4PING	EP4 OUT 已发送但已被拒绝
23	58	EP6PING	EP6 OUT 已发送但已被拒绝
24	5C	EP8PING	EP8 OUT 已发送但已被拒绝
25	60	ERRLIMIT	总线错误率超出程序设定的限制
26	64	—	—
27	68	—	已保留
28	6C	—	已保留
29	70	EP2ISOERR	ISO EP2 OUT PID 序列错误
30	74	EP4ISOERR	ISO EP4 OUT PID 序列错误
31	78	EP6ISOERR	ISO EP6 OUT PID 序列错误
32	7C	EP8ISOERR	ISO EP8 OUT PID 序列错误

如果启用自动矢量化 (在 INTSET-UP 寄存器中 AV2EN = 1), FX2LP 替换 INT2VEC 字节。因此, 如果在位置 0x0044 预加载跳转表地址的高字节 (“页”), 则在 0x0045 自动插入的 INT2VEC 字节将跳转表发送到本页中的正确地址 (共有 27 个地址)。

#### FIFO/GPIF 中断 (INT4)

正如 27 个单独 USB 中断源共享一个 USB 中断一样, FIFO/GPIF 中断由 14 个单独的 FIFO/GPIF 中断源所共享。FIFO/GPIF 中断类似于 USB 中断, 可以应用自动矢量化。第 8 页的表 4 显示 14 个 FIFO/GPIF 中断源的优先级和 INT4VEC 值。



**表 4. 单个 FIFO/GPIF 中断源**

优先级	INT4VEC 值	Source (源)	注
1	80	EP2PF	端点 2 可编程标志
2	84	EP4PF	端点 4 可编程标志
3	88	EP6PF	端点 6 可编程标志
4	8C	EP8PF	端点 8 可编程标志
5	90	EP2EF	端点 2 空标志 <sup>[3]</sup>
6	94	EP4EF	端点 4 空标志
7	98	EP6EF	端点 6 空标志
8	9C	EP8EF	端点 8 空标志
9	A0	EP2FF	端点 2 满标志
10	A4	EP4FF	端点 4 满标志
11	A8	EP6FF	端点 6 满标志
12	AC	EP8FF	端点 8 满标志
13	B0	GPIFDONE	GPIF 操作已完成
14	B4	GPIFWF	GPIF 波形

如果启用自动矢量化 (在 INTSET-UP 寄存器中 AV4EN = 1), FX 2LP 替换 INT4VEC 字节。因此,如果在位置 0x0054 预加载跳转表地址的高字节 (“页”),则在 0x0055 自动插入的 INT4VEC 字节将跳转表发送到本页中的正确地址 (共 14 个地址)。当触发 ISR 时, FX2LP 将程序计数器推至堆栈,然后跳转到地址 0x0053 上,在此希望找到“跳转”至 ISR 中断服务例程的指令。

**注**

3. **勘误表**: In Slave FIFO Asynchronous Word Wide mode, if a single word data is transferred from the USB host to EP2, configured as OUT Endpoint (EP) in the first transaction, then the Empty flag behaves incorrectly. This does not happen if the data size is more than one word in the first transaction. For more information, see the 第 64 页勘误表。

## 复位和唤醒

### 复位引脚

输入引脚 **RESET#** 在激活时复位 **FX2LP**。该引脚具有迟滞现象，低电平有效。当通过 **CY7C680xxA** 使用晶振时，复位周期必须使晶振和 **PLL** 保持稳定。当 **VCC** 达到 **3.0V** 后，此复位周期必须约为 **5 ms**。如果晶振输入引脚由时钟信号予以驱动，则在 **VCC** 达到 **3.0V** 后，内部 **PLL** 将在 **200 μs** 上保持稳定。<sup>[4]</sup>

第 9 页的图 2 显示复位条件上的功耗和工作期间所应用的复位。复位功耗定义如下：当电路正在消耗电量时所激活的时间复位。供电复位是指 **FX2LP** 加电，操作和激活 **RESET#** 引脚。

赛普拉斯提供应用程序笔记，其中包含描述和建议加电复位实现方法。有关 **FX2** 产品系列复位实现方法的更多信息，请访问网站 <http://www.cypress.com>。

图 2. 复位时序图

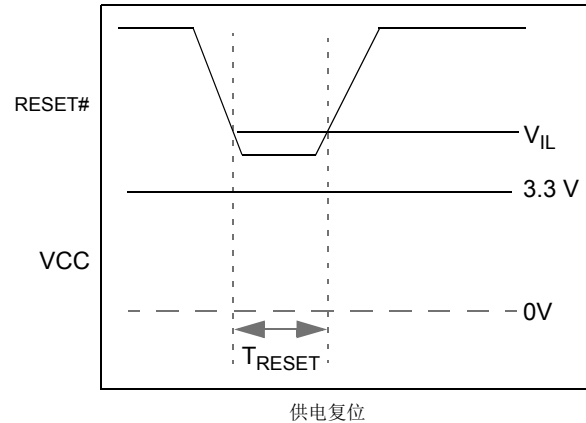
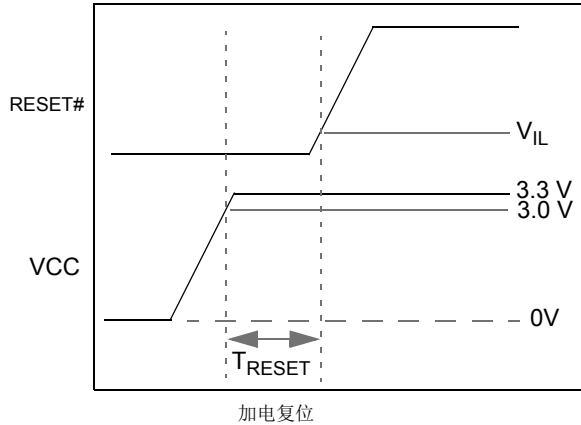


表 5. 复位时序值

条件	T <sub>RESET</sub>
加电复位晶振	5 ms
加电复位外部时钟	200 μs + 时钟稳定性时间
供电复位	200 μs

### 唤醒引脚

8051 通过设置 **PCON.0 = 1** 将自身和其他芯片置于断电模式。这关闭了振荡器和 **PLL**。当通过外部逻辑激活 **WAKEUP** (唤醒) 时，**PLL** 保持移动后，振荡器重新启动，然后 8051 收到唤醒中断。无论 **FX2LP** 是否与 **USB** 连接，这种情况均适用。

**FX2LP** 使用下列一种方法退出断电 (**USB 暂停**) 状态：

- **USB 总线活动** (若 **D+/D<tA;** 线保持悬空，这些线上的噪声表示 **FX2LP** 活动已开始唤醒)
- **外部逻辑激活 WAKEUP** (唤醒) 引脚
- **外部逻辑激活 PA3/WU2** 引脚

此外，第二个唤醒引脚 **WU2** 还配置为通用 **I/O** 引脚。这样便可以使用简单的外部 **R-C** 作为定期唤醒源。默认情况下，**WAKEUP** (唤醒) 为有效的低电平。

### 注

4. 如果随 **CY7C680xxA** 一起同时为外部时钟供电，并且外部时钟具有稳定等待周期，则复位周期必须增加到 **200 μs**。

## 程序 / 数据 RAM

### 大小

FX2LP 包含 16 KB 的内部程序 / 数据 RAM, 其中, PSEN#/RD# 信号是内部 ORed, 该 ORed 使能 8051 将其作为程序和数据存储器进行访问。在此空间不显示 USB 控制寄存器。

下图显示两个存储器映射:

第 10 页的图 3 显示内部代码存储器, EA = 0

第 11 页的图 4 显示外部代码存储器, EA = 1。

### 内部代码存储器, EA = 0

此模式实现了内部 16 KB 的 RAM 模块 (以 0 开头), 作为组合代码和数据存储器。当增加外部 RAM 或 ROM 时, 芯片内部存在的

存储器空间将禁用外部读取和写入探针。这使用户能够连接 64 KB 的存储器, 无需地址解码来清除内部存储器空间。

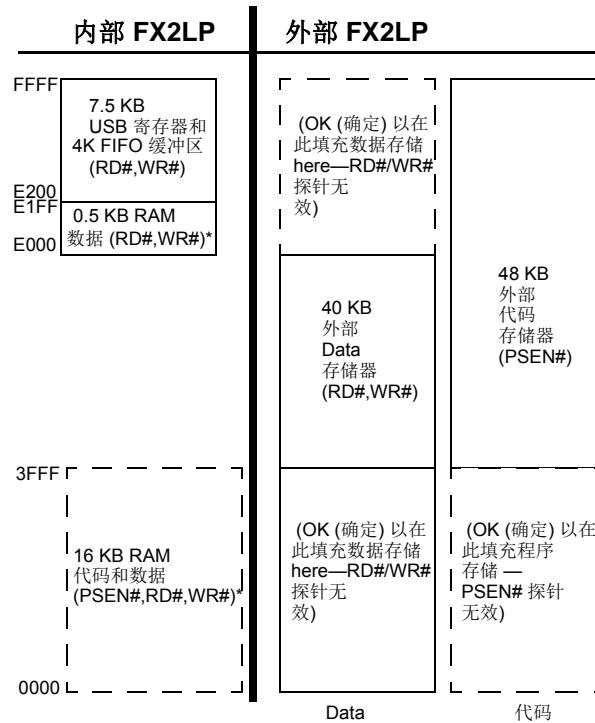
只有内部 16 KB 和暂存器 0.5 KB 的 RAM 空间具有以下访问权限:

- USB 下载
- USB 上载
- 设置数据指针
- I<sup>2</sup>C 接口引导加载。

### 外部代码存储器, EA = 1

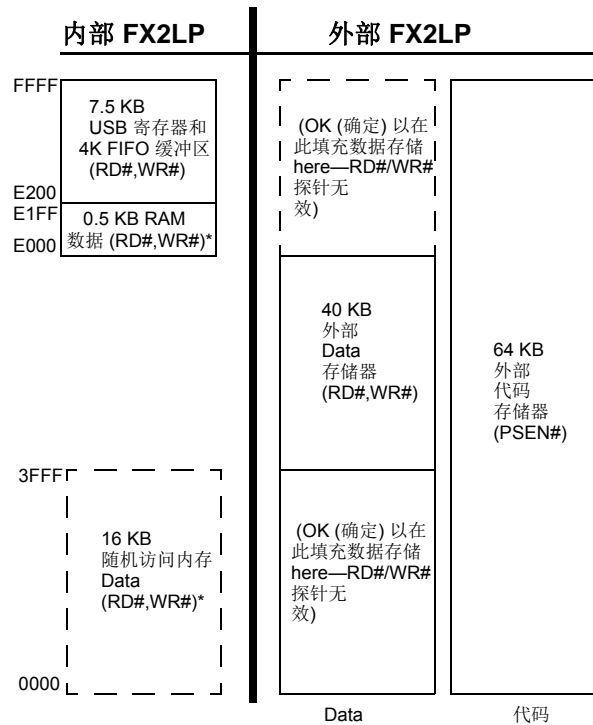
底部 16 KB 程序存储器为外部存储器, 因此底部 16 KB 的内部 RAM 仅作为数据存储器予以访问。

图 3. 内部代码存储器, EA = 0



\*SUDPTR, USB 上载/下载, I<sup>2</sup>C 接口引导访问

图 4. 外部代码存储器, EA = 1


 \*SUDPTR, USB 上载/下载、I<sup>2</sup>C 接口引导访问

### 寄存器地址

FFFF	4 字节 EP2-EP8 缓冲器 (8 x 512)
F000 EFFF	2 KB 已保留
E800 E7FF E7C0	64 字节 EP1IN
E7BF E780	64 字节 EP1OUT
E77F E740	64 字节 EP0 IN/OUT
E73F E700 E6FF	64 字节 已保留
E500	8051 可寻址寄存器 (512)
E4FF E480	已保留 (128)
E47F	128 字节 GPIF 波形
E400 E3FF E200	已保留 (512)
E1FF E000	512 字节 8051 xdata RAM

## 端点 RAM

### 大小

- 3 × 64 字节 (端点 0 和 1)
- 8 × 512 字节 (端点 2, 4, 6, 8)

### 组织

- EP0
- 双向端点 0、64 字节缓冲区
- EP1IN, EP1OUT
- 64 字节缓冲区、批量传输或中断传输
- EP2, 4, 6, 8
- 8 个 512 字节的缓冲区、批量传输、中断传输或同步传输。  
 EP4 和 EP8 是双缓冲；EP2 和 6 是双缓冲、三重或四立体缓冲。有关高速端点配置选项，请参见图 5。

### 设置数据缓冲区

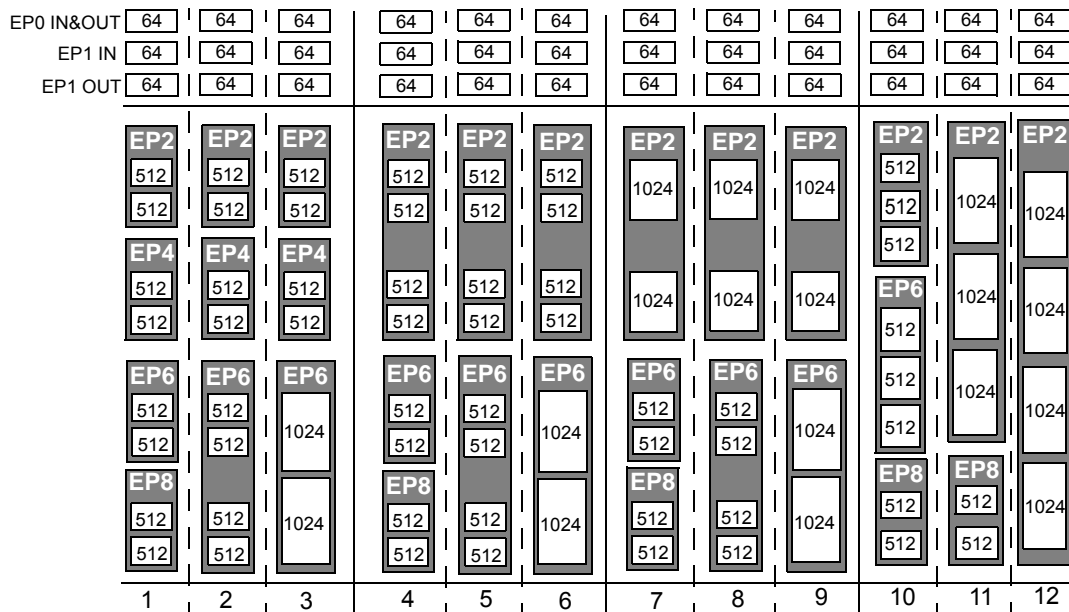
在 0xE6B8-0xE6BF 上的单独的 8 字节缓冲区保持 ONTROL 传输的设置数据。

### 端点配置 (高速模式)

每个配置的端点 0 和 1 相同。端点 0 是唯一的 CONTROL (控制) 端点，端点 1 既可以是 BULK (批量传输) 端点，也可以是 INTERRUPT (中断传输) 端点。

端点缓冲区可以配置为竖列中显示的 12 个配置中的任何一个配置。当在 BULK (批量传输) 全速模式下操作时，仅使用每个缓冲区首个 64 位字节。例如，在高速模式下，最大数据包大小为 512 字节，但在全速模式下，最大数据包为 64 字节。即使缓冲区配置为 512 字节的缓冲区，在全速模式下仍然仅使用第一个 64 位字节。未使用的端点缓冲区空间不可以用于其他操作。示例端点配置为 EP2-1024 双缓冲；EP6-512 是四立体缓冲 (第 8 列)。

图 5. 端点配置



#### 默认全速备用设置

表 6. 默认全速备用设置 [5, 6]

备用设置	0	1	2	3
ep0	64	64	64	64
ep1out	0	64 批量传输	64 int	64 int
ep1in	0	64 批量传输	64 int	64 int
ep2	0	64 批量输出 (2×)	64 int out (2×)	64 iso out (2×)
ep4	0	64 批量输出 (2×)	64 批量输出 (2×)	64 批量输出 (2×)
ep6	0	64 批量输入 (2×)	64 int in (2×)	64 iso in (2×)
ep8	0	64 批量输入 (2×)	64 批量输入 (2×)	64 批量输入 (2×)

#### 默认全速备用设置

表 7. 默认全速备用设置 [5, 6]

备用设置	0	1	2	3
ep0	64	64	64	64
ep1out	0	512 批量传输 [7]	64 int	64 int
ep1in	0	512 批量传输 [7]	64 int	64 int
ep2	0	512 批量输出 (2×)	512 int out (2×)	512 iso out (2×)
ep4	0	512 批量输出 (2×)	512 批量输出 (2×)	512 批量输出 (2×)
ep6	0	512 批量输入 (2×)	512 int in (2×)	512 iso in (2×)
ep8	0	512 批量输入 (2×)	512 批量输入 (2×)	512 批量输入 (2×)

## 外部 FIFO 接口

### 架构

FX2LP 从器件 FIFO 架构在端点 RAM 中有 8 个 512 字节的模块，可直接用作 FIFO 存储器，这些模块受控于 FIFO 控制信号（例如，IFCLK, SLCS#, SLRD, SLWR, SLOE, PKTEND 和标志）。

在操作中，这 8 个 RAM 模块中的某些模块由 SIE 填满或保留为空，而其他模块连接到 I/O 传输逻辑。传输逻辑采用两种形式，分别为 GPIF（用于内部生成控制信号）和从器件 FIFO 接口（用于控制传输）。

### 主控 / 从器件控制信号

FX2LP 端点 FIFOS 可以实现为 8 个物理上截然不同的 256x16 RAM 模块。8051/SIE 可以在 USB (SIE) 域和 8051-I/O 单元域之间切换到任意 RAM 模块。此切换实际上在瞬间即可完成，“USB FIFOS”与“从器件 FIFOS”之间的传输时间基本上为零，因为他们实际上是同一个存储器，在两个缓冲区之间并未发生任何字节传输。

无论何时，某些 RAM 模块均可以使用 SIE 控制下的 USB 数据进行填充 / 清空，而其他 RAM 模块可用于 8051、I/O 控制单元或同时用于二者。RAM 模块在 USB 域中作为单个端口运行，而在 8051-I/O 域中作为双端口运行。这些模块可以配置为单缓冲、双缓冲、三重缓冲或四立体缓冲，如上所述。

I/O 控制单元实现了内部掩模（用于主控的 M）或外部掩模（用于从器件的 S）接口。

在主机 (M) 模式下，GPIF 在内部控制 FIFOADR[1..0]，用以选择 FIFO。RDY 引脚（56 引脚封装中有 2 个、100 引脚和 128 引脚封装中有 6 个）可以用作自外部 FIFO 或其他逻辑（若需要）的标志输入。GPIF 可以从内部派生的时钟或外部供应的时钟（IFCLK）中运行，传输数据的速率高达 96 MB/s（16 位接口 48 MHz IFCLK）。

在从器件 (S) 模式下，FX2LP 从外部逻辑中接受内部派生的时钟或外部供应的时钟（IFCLK，最大频率为 48 MHz）及 SLCS#, SLRD, SLWR, SLOE, PKTEND 信号。使用外部 IFCLK 时，要在切换到 IFCLKSRC 位的外部时钟之前提供外部时钟。通过内部配置位单独选择字节或文字操作的各个端点。从器件 FIFO 输出使能信号 SLOE 使数据达到所选宽度。外部逻辑必须确保输出使能信号在将数据写入从器件 FIFO 时未激活。此外，从器件接口也可以异步运行，其中，SLRD 和 SLWR 信号直接充当探针，而非同步模式下的时钟限定符。信号 SLRD, SLWR, SLOE 和 PKTEND 由信号 SLCS# 来关断。

### GPIF 和 FIFO 时钟频率

8051 寄存器位从内部供应的接口时钟中选择其中一个频率（共 2 个）：30 MHz 和 48 MHz。当 GPIF 和 FIFO 由内部时钟驱动时，IFCLK 可以配置为输出时钟的功能。IFCONFIG 寄存器中的输出使能位关闭此时钟输出（若需要）。IFCONFIG 寄存器中的另一个位反转 IFCLK 信号，无论是内部源还是外部源，无一例外。

### 注

5. “0”表示“未实施”。

6. “2×”表示“双缓冲”。

7. 即使这些缓冲区为 64 字节，也报告为符合 512 字节的 USB 2.0 规范。用户禁止向 EP1 传输大于 64 字节的数据包。

## GPIO

GPIO 是灵活的 8 位或 16 位并行接口，由用户可编辑有限状态机来驱动。该接口使能 CY7C68013A/15A 执行本地总线主控，并可以实施多种协议，例如，ATA 接口、打印机并行端口和 Utopia。

GPIO 有 6 个可编程控制输出 (CTL)、9 个地址输出 (GPIOFADR<sub>x</sub>) 和 6 个通用就绪输入 (RDY) 接口。数据总线宽度可以是 8 位或 16 位。各个 GPIO 矢量定义了控制输出状态，并确定就绪输入 (多重输入) 在继续执行之前的状态。GPIO 矢量需要程序化，从而可以使 FIFO 进入下一个数据值，进入一个地址等。GPIO 矢量序列组成单独的波形，运行该波形时，可以在 FX2LP 与外部器件之间随意移动数据。

### 6 个控制输出信号

100 引脚和 128 引脚封装生成所有这 6 种控制输出引脚 (CTL0-CTL5)。8051 程序化 GPIO 单元，以用来定义 CTL 波形。56 引脚封装生成这些信号中所包含的其中 3 种信号 CTL0-CTL2。CTL<sub>x</sub> 波形边沿可以被程序化，从而一次性快速切换到每个时钟 (使用 48 MHz 时钟为 20.8 ns)。

### 6 个就绪 IN 信号

100 引脚和 128 引脚封装生成所有这 6 个就绪输入信号 (RDY0-RDY5)。8051 程序化 GPIO 单元以用来测试 GPIO 分支的 RDY 引脚。56 引脚封装生成这些信号中所包含的其中 2 种信号。

### 9 个 GPIO 地址 OUT 信号

9 个 GPIO 地址行可用于 100 引脚和 128 引脚的封装 GPIOFADR[8..0]。GPIO 地址行能够完成高达 512 字节 RAM 模块的索引编制。如果需要更多地址行，则使用 I/O 端口引脚。

### 远程传输模式

在主机模式下，8051 为多达 2<sup>32</sup> 种任务的无伴随传输设置了相应的 GPIO 任务计数寄存器 (GPIOFCB3, GPIOFCB2, GPIOFCB1 或 GPIOFCB0)。GPIO 自动抑制数据流，以便防止数据流下溢或溢出，直到全部请求的任务完成时为止。GPIO 递减这些寄存器中的值，从而表示该任务的当前状态。

## ECC 生成<sup>[8]</sup>

EZ-USB 可以对跨越 GPIO 或从器件 FIFO 接口传递的数据计算 ECC (错误修正代码)。共有两种 ECC 配置：这两种 ECC 各自的计算值超出 256 字节 (SmartMedia 标准)；其中一种 ECC 的计算值超出 512 字节。

ECC 可以修正任何一位误差或检测任何两位误差。

### ECC 实现

通过 ECCM 位来选择这两个 ECC 配置：

ECCM = 0

两个 3 字节 ECC，各自的计算值超出 256 字节的数据模块。这种配置符合 SmartMedia 的标准。

将任意值写入 ECCRESET，然后将数据传递到 GPIO 或从器件 FIFO 接口。计算第一个 256 字节数据的 ECC，然后将其存储在 ECC1 中。计算下一个 256 字节数据的 ECC，然后存储在 ECC2 中。第二个 ECC 计算完成后，ECC<sub>x</sub> 寄存器中的值保持不变，直到再次写入 ECCRESET 时为上，即使有更多数据序列传递到该接口也一样。

### 注

8. 要使用 ECC 逻辑，GPIO 或从器件 FIFO 接口必须针对字节宽度操作进行配置。
9. 从主机下载数据完成后，“加载器”可能从内部 RAM 运行以将下载数据传输到外部存储器。

ECCM = 1

其中一个 3 字节 ECC 的计算值超出 512 字节的数据模块。

将任意值写入 ECCRESET，然后将数据传递到 GPIO 或从器件 FIFO 接口。计算第一个 512 字节数据的 ECC，然后将其存储在 ECC1；ECC2 尚未使用。ECC 计算完成后，ECC1 中的值保持不变，即使有更多数据序列传递到该接口也如此，直到再次写入 ECCRESET 时为上。

## USB 上载和下载

该内核可以通过供应商特定的命令直接编辑内部 16 KB RAM 和内部 512 字节暂存器 RAM 的数据内容。通常，在下载用户代码时使用该功能，但仅限于从内部 RAM 中读取和写入，仅在 8051 保持复位时方可。RAM 可用空间为 16 KB 0x0000-0x3FFF (代码 / 数据) 和 512 字节 0xE000-0xE1FF (暂存器数据 RAM)。<sup>[9]</sup>

### 自动指针访问

FX2LP 提供两个相同的自动指针。这两个指针类似于内部 8051 数据指针，但具有额外特性：每次访问存储器后，指针递增 (可选)。此功能可用于从内部和外部 RAM 读取和写入。这两个自动指针可用于受控于模式位的 FX2LP 寄存器 (AUTOPTRESET-UP.0)。使用外部 FX2LP 自动指针访问 (在 0xE67B-0xE67C 上) 可以实现自动指针对该器件所有的内部和外部 RAM 的访问。

此外，自动指针还可以指向任何 FX2LP 寄存器或端点缓冲区空间。当自动指针访问外部存储器时，无法使用 XDATA 中的位存 0xE67B 和 0xE67C 及代码空间。

## I<sup>2</sup>C 控制器

FX2LP 具有一个 I<sup>2</sup>C 端口，它由两个内部控制器来驱动，其中一个控制器可以在启动时自动加载 VID/PID/DID 和配置信息，另一个由 8051 在运行时用来控制外部 I<sup>2</sup>C 器件。I<sup>2</sup>C 端口仅运行在主机模式下。

### I<sup>2</sup>C 端口引脚

即使 EEPROM 未与 FX2LP 连接，I<sup>2</sup>C 引脚 SCL 和 SDA 也必须具有 2.2 kΩ 外部上拉电阻。外部 EEPROM 器件地址引脚配置必须正确。有关器件地址引脚的更多信息，请参见表 8。

表 8. 引导这些值的 EEPROM 地址行

字节	示例 EEPROM	A2	A1	A0
16	24LC00 <sup>[10]</sup>	不可用	不可用	不可用
128	24LC01	0	0	0
256	24LC02	0	0	0
4K	24LC32	0	0	1
8 K	24LC64	0	0	1
16 K	24LC128	0	0	1

### I<sup>2</sup>C 接口引导加载访问

在加电复位时，I<sup>2</sup>C 接口引导加载程序加载 VID/PID/DID 配置字节，高达 16 KB 的程序 / 数据。RAM 可用空间是 16 KB 0x0000-0x3FFF 和 512 字节 0xE000-0xE1FF。8051 处于复位状态。I<sup>2</sup>C 接口引导加载仅在加电复位后发生。



## I<sup>2</sup>C 接口通用访问

8051 使用 I<sup>2</sup>CTL 和 I2DAT 寄存器控制连接至 I<sup>2</sup>C 总线的外设。FX2LP 仅提供 I<sup>2</sup>C 主控，但从不提供 I<sup>2</sup>C 从器件。

## 与上一代 EZ-USB FX2 兼容

EZ-USB FX2LP 的外观、适用性及其功能（异常情况极少）均与上一代 EZ-USB FX2 完全兼容。这样，设计者便可以轻松转换以将其系统从 FX2 升级至 FX2LP。在 FX2LP 中，引脚分布和封装选项是过去针对 FX2 功能开发的类似的大多数固件。

对于从 FX2 迁移至 FX2LP 的设计者而言，此迁移要求变更材料单和评估存储器的分配（因为内部存储器日益增加）。有关从 EZ-USB FX2 迁移至 EZ-USB FX2LP 的更多信息，请参考标题为“*Migrating from EZ-USB FX2 to EZ-USB FX2LP*”的应用笔记，这可以在赛普拉斯网站上获得。

**表 9. 部件编号转换表**

EZ-USB FX2 部件编号	EZ-USB FX2LP 部件编号	封装描述
CY7C68013-56PVC	CY7C68013A-56PVXC 或 CY7C68014A-56PVXC	56 引脚 SSOP
CY7C68013-56PVCT	CY7C68013A-56PVXCT 或 CY7C68014A-56PVXCT	56 引脚 SSOP – 盘 带封装
CY7C68013-56LFC	CY7C68013A-56LFXC 或 CY7C68014A-56LFXC	56 引脚 QFN
CY7C68013-100AC	CY7C68013A-100AXC 或 CY7C68014A-100AXC	100 引脚 TQFP
CY7C68013-128AC	CY7C68013A-128AXC 或 CY7C68014A-128AXC	128 引脚 TQFP

## CY7C68013A/14A 和 CY7C68015A/16A 的区别

CY7C68013A 在外观、适用性和功能上与 CY7C68014A 类似。CY7C68015A 在外观、适用性和功能上与 CY7C68016A 类似。CY7C68014A 和 CY7C68016A 的暂停电流分别低于 CY7C68013A 和 CY7C68015A，对电量敏感的蓄电池应用而言是非常理想的。

CY7C68015A 和 CY7C68016A 仅可以用于 56 引脚 QFN 封装。在 CY7C68015A 和 CY7C68016A 上可以获得两个额外信号，当 56 引脚的封装中既不需要 IFCLK，也不需要 CLKOUT 时，这两种信号可以提供超高的灵活性。

USB 开发人员期望将 FX2 56 引脚应用转换成总线供电系统，由此直接从这些额外信号中获益。这两个 GPIO 为开发人员提供总线供电式应用耗电控制电路所需的信号，而无需升级到较高引脚数目的 FX2LP 版本。

CY7C68015A 仅用于 56 引脚的 QFN 封装

**表 10. CY7C68013A/14A 和 CY7C68015A/16A 引脚区别**

CY7C68013A/CY7C68014A	CY7C68015A/CY7C68016A
IFCLK	PE0
CLKOUT	PE1

## 注

10. 此 EEPROM 不包含地址引脚。

## 引脚分配

第 17 页的图 6 识别 5 种封装类型的所有信号。后续页面给出单个引脚图和组合图，用来阐明所有信号中的哪一个信号可以用于 128 引脚、100 引脚和 56 引脚的封装。

在第 17 页的图 6 中，56 引脚封装左边沿上的信号对 FX2LP 系列中的所有版本通用，包含 CY7C68013A/14A 和 CY7C68015A/16A 之间区别的说明。

所有封装版本均可以使用的模式有三种：端口、GPIF 主控和从器件 FIFO。这些模式定义了该图右侧的信号。8051 使用 IFCONFIG[1:0] 寄存器位选择接口模式。端口模式是加电默认配置。

100 引脚的封装通过添加这些引脚将功能添加到 56 引脚的封装中：

- PORTC 或 GPIFADR[7:0] 备用地址信号
- PORTE 或 GPIFADR[8] 备用地址信号和 7 个额外的 8051 信号
- 3 个 GPIF 控制信号
- 4 个 GPIF 就绪信号
- 9 个 8051 信号 (2 个 USART, 3 个定时器输入, INT4 和 INT5#)
- BKPT, RD#, WR#。

128 引脚的封装负责添加 8051 地址和数据总线加控制信号。

注：在 100 引脚的版本中提供 2 个所需信号、RD# 和 WR#。

在 100 引脚和 128 引脚的版本中，当 8051 读取 / 写入 PORTC 时，可以设置 8051 控制位来驱动 RD# 和 WR# 引脚。此功能通过在 CPUCS 寄存器中设置 PORTCSTB 来得以实现。

节 显示访问 PORTC 时读取和写入探针函数的时序图。

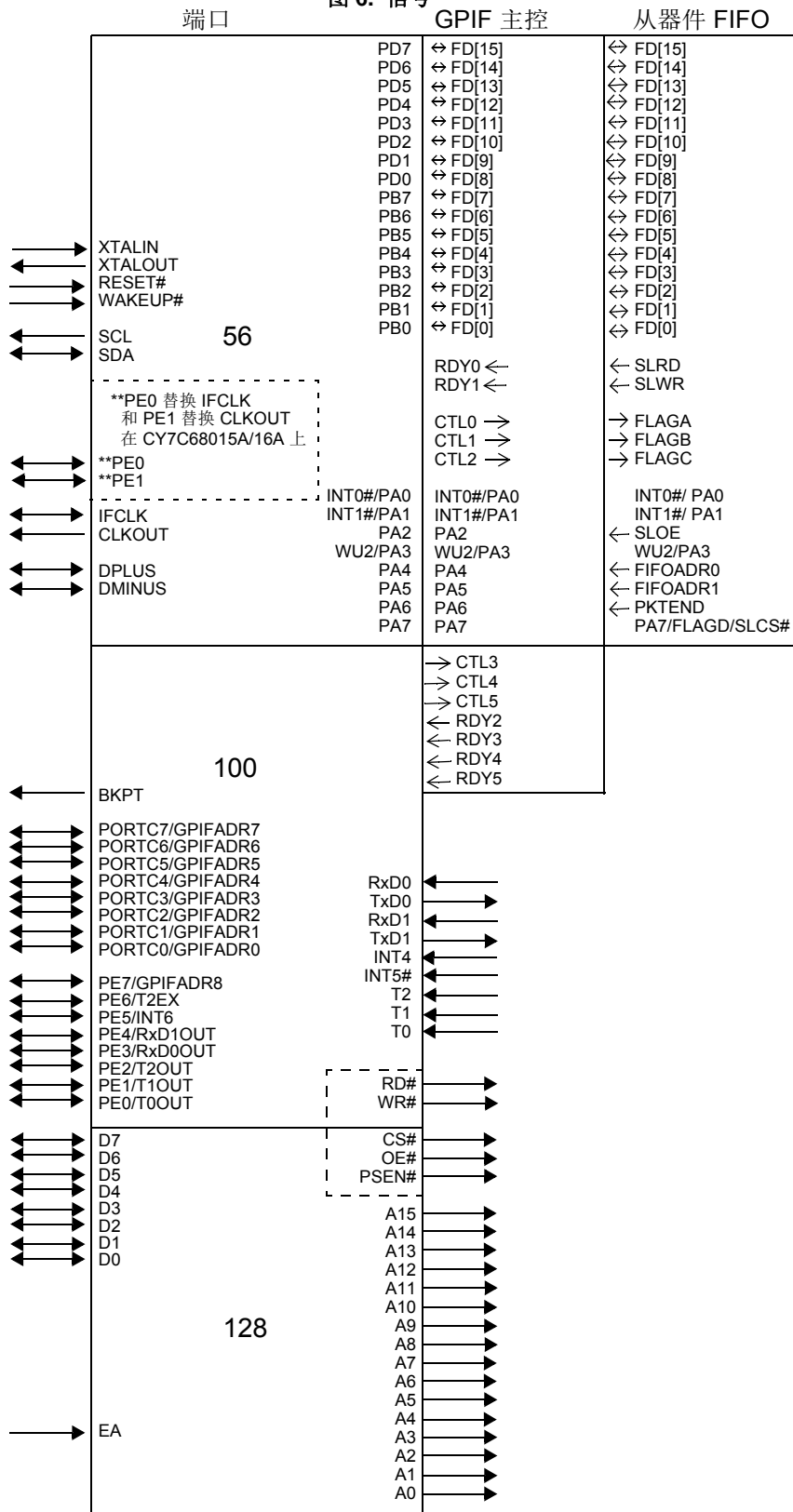
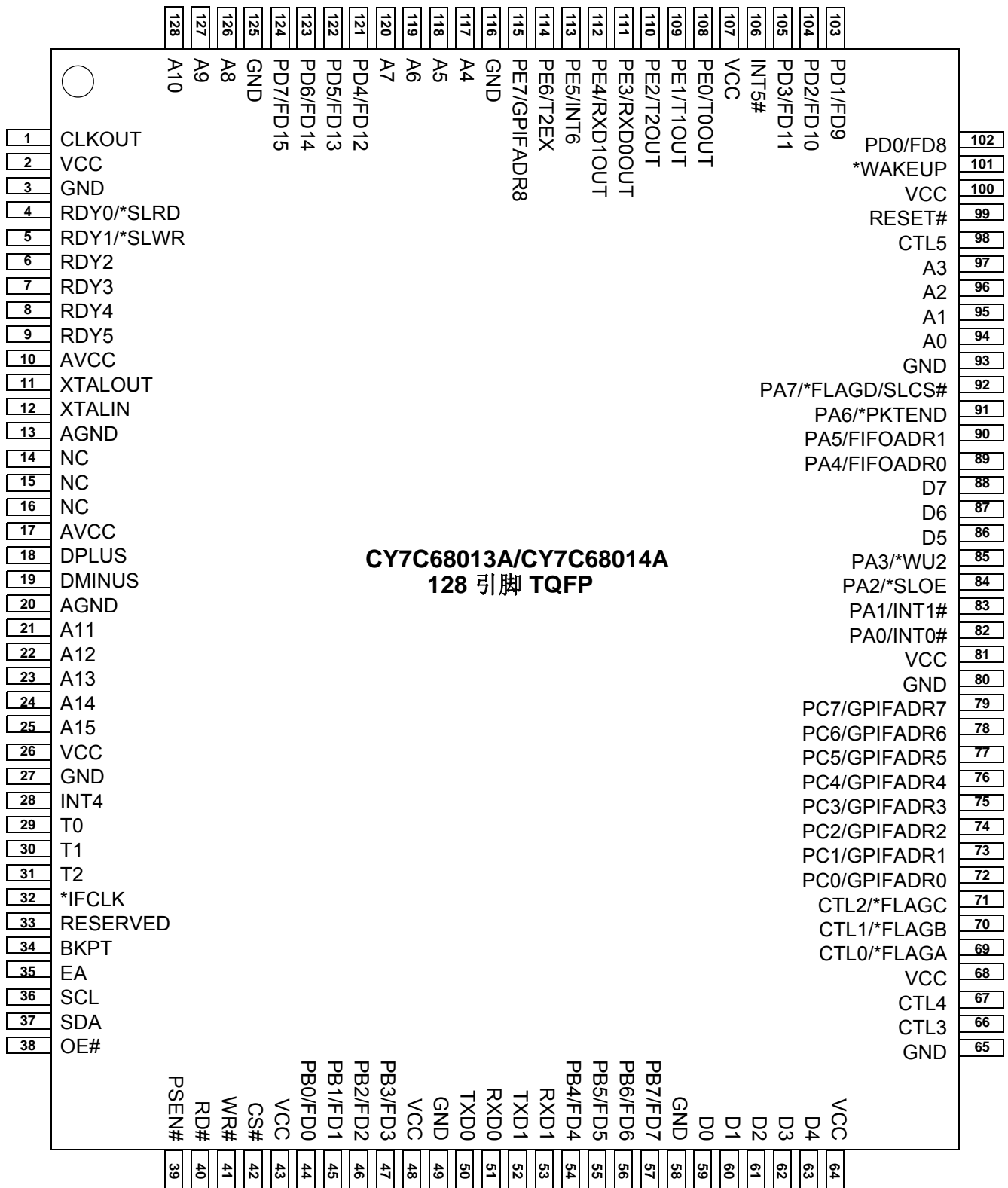
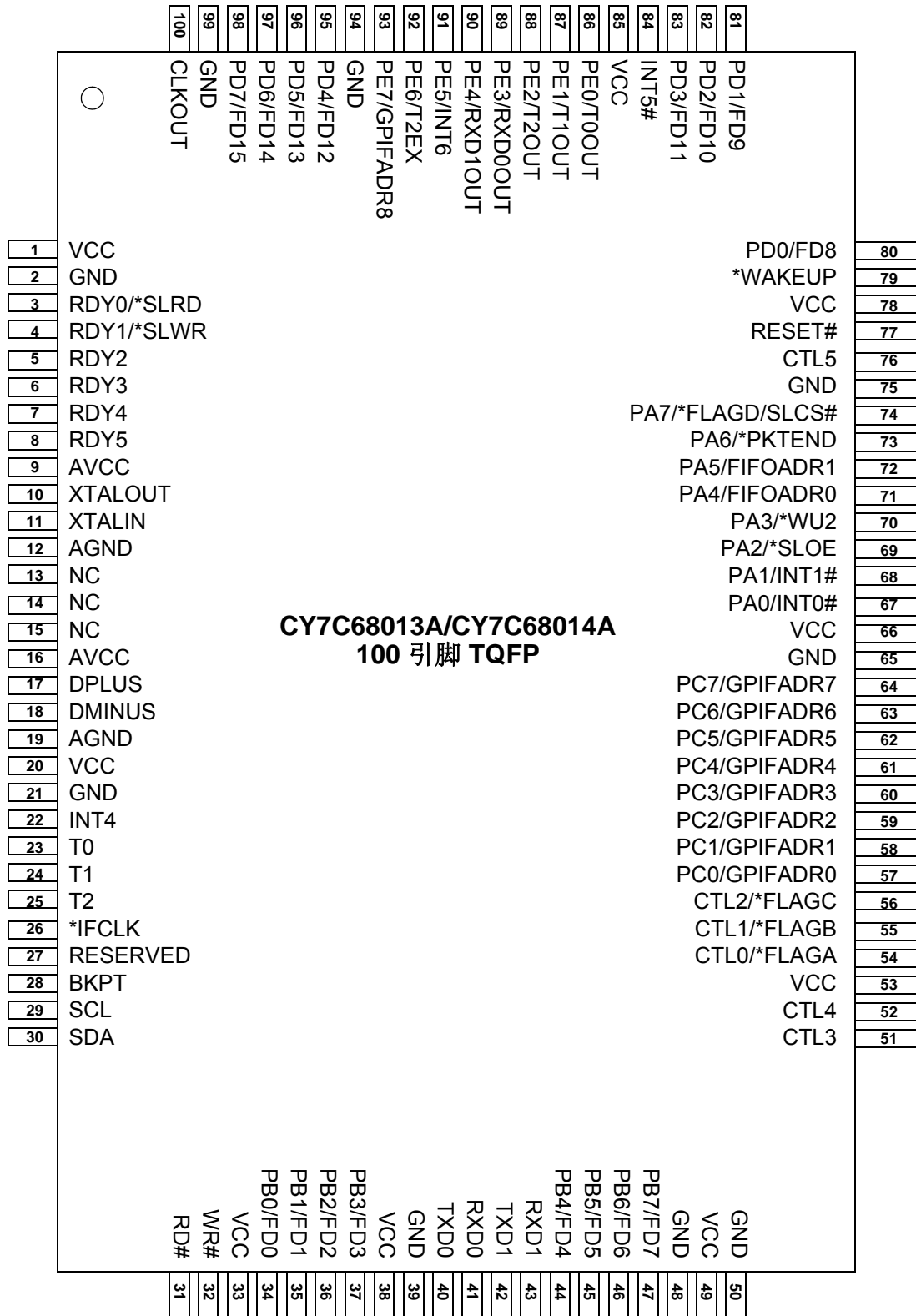
**图 6. 信号**


图 7. CY7C68013A/CY7C68014A 128 引脚 TQFP 的引脚分配



\* 表示可编程极性

图 8. CY7C68013A/CY7C68014A 100 引脚 TQFP 的引脚分配



\* 表示可编程极性

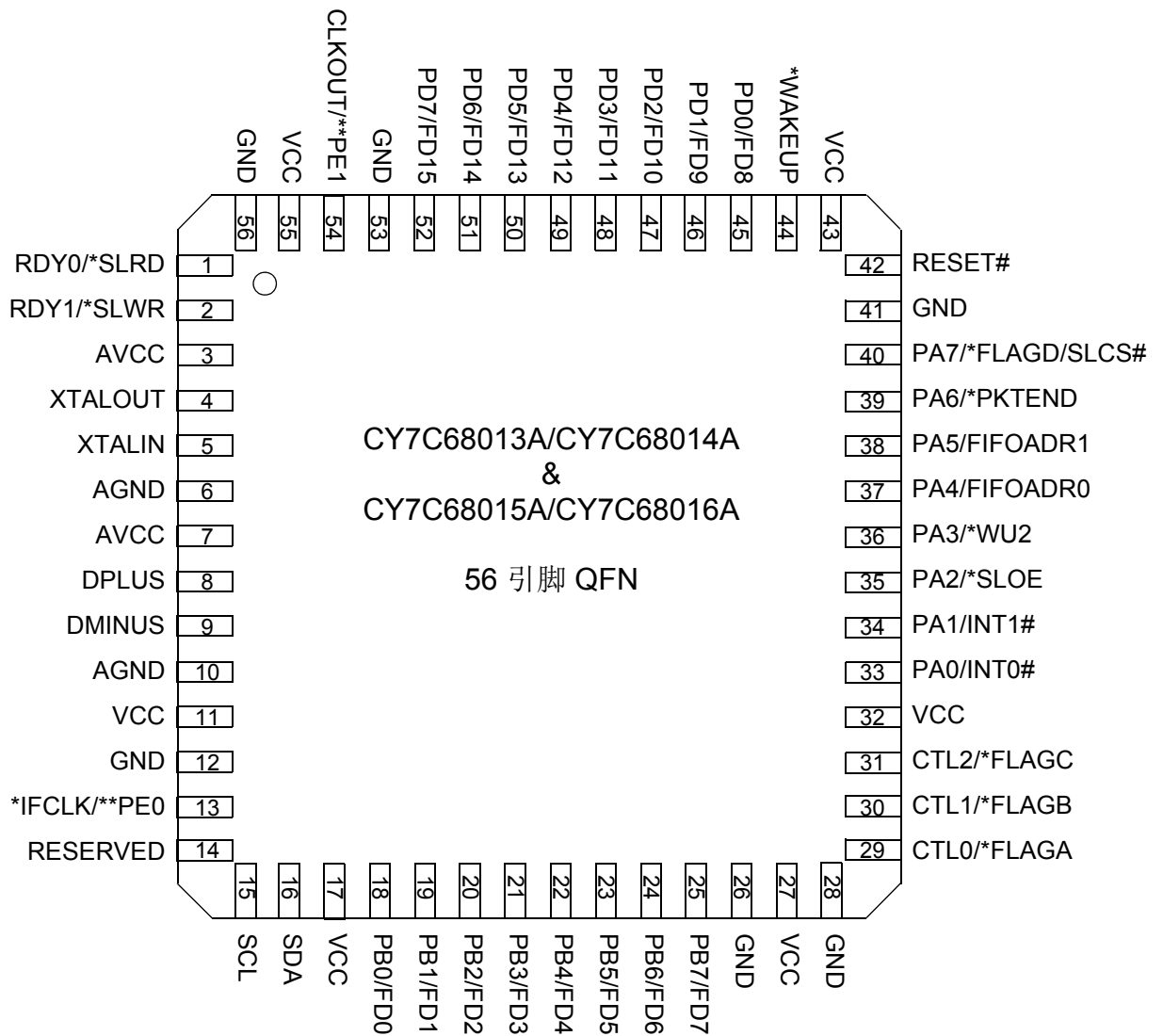
图 9. CY7C68013A/CY7C68014A 56 引脚 SSOP 的引脚分配

**CY7C68013A/CY7C68014A**  
**56 引脚 SSOP**

1	PD5/FD13	PD4/FD12	56
2	PD6/FD14	PD3/FD11	55
3	PD7/FD15	PD2/FD10	54
4	GND	PD1/FD9	53
5	CLKOUT	PD0/FD8	52
6	VCC	*WAKEUP	51
7	GND	VCC	50
8	RDY0/*SLRD	RESET#	49
9	RDY1/*SLWR	GND	48
10	AVCC	PA7/*FLAGD/SLCS#	47
11	XTALOUT	PA6/PKTEND	46
12	XTALIN	PA5/FIFOADR1	45
13	AGND	PA4/FIFOADR0	44
14	AVCC	PA3/*WU2	43
15	DPLUS	PA2/*SLOE	42
16	DMINUS	PA1/INT1#	41
17	AGND	PA0/INT0#	40
18	VCC	VCC	39
19	GND	CTL2/*FLAGC	38
20	*IFCLK	CTL1/*FLAGB	37
21	RESERVED	CTL0/*FLAGA	36
22	SCL	GND	35
23	SDA	VCC	34
24	VCC	GND	33
25	PB0/FD0	PB7/FD7	32
26	PB1/FD1	PB6/FD6	31
27	PB2/FD2	PB5/FD5	30
28	PB3/FD3	PB4/FD4	29

\* 表示可编程极性

图 10. CY7C68013A/14A/15A/16A 56 引脚 QFN 的引脚分配

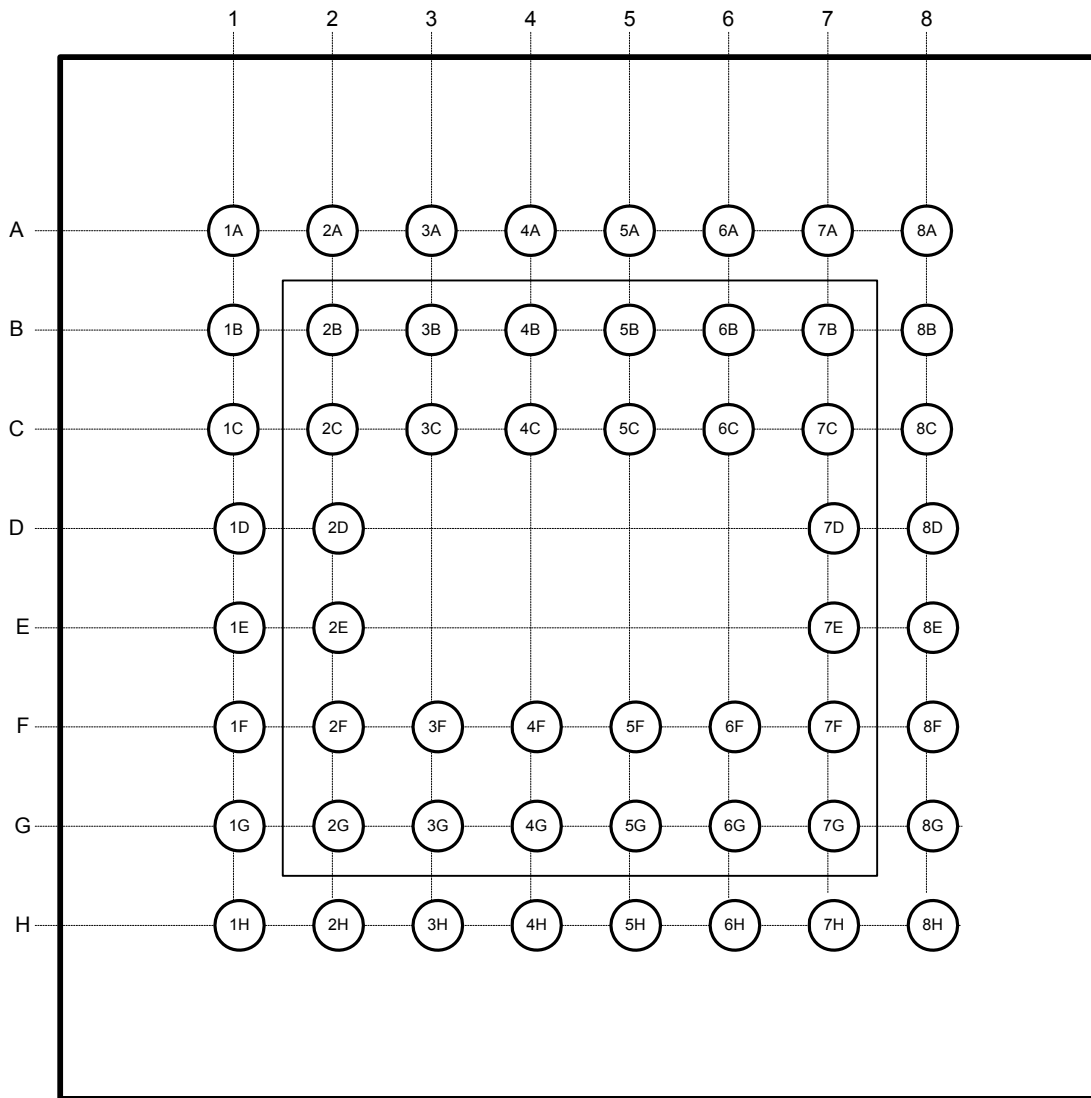


\* 表示可编程极性

\*\* 表示 CY7C68015A/CY7C68016A 引脚分布



图 11. CY7C68013A 56 引脚 VFBGA 的引脚分配 – 上视图



**CY7C68013A/15A 引脚描述**

FX2LP 引脚说明如下。 [11]

**表 11. 引脚描述**

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	名称	类型	默认值	复位 [12]	说明
10	9	10	3	2D	AVCC	功耗	不可用	不可用	<b>模拟 VCC</b> 。连接此引脚至 3.3V 的电源。此信号提供芯片模拟部分的功耗。
17	16	14	7	1D	AVCC	功耗	不可用	不可用	<b>模拟 VCC</b> 。连接此引脚至 3.3V 的电源。此信号提供芯片模拟部分的功耗。
13	12	13	6	2F	AGND	接地	不可用	不可用	<b>模拟接地</b> 。通过最短路径连接至接地。
20	19	17	10	1F	AGND	接地	不可用	不可用	<b>模拟接地</b> 。通过最短路径连接至接地。
19	18	16	9	1E	DMINUS	I/O/Z	Z	不可用	<b>USB D- 信号</b> 。连接至 USB D- 信号。
18	17	15	8	2E	DPLUS	I/O/Z	Z	不可用	<b>USB D+ 信号</b> 。连接至 USB D+ 信号。
94	—	—	—	—	A0	输出	L	L	<b>8051 地址总线</b> 。此总线可以在全天候下驱动。当 8051 寻址内部 RAM 时，它可以返回内部地址。
95	—	—	—	—	A1	输出	L	L	
96	—	—	—	—	A2	输出	L	L	
97	—	—	—	—	A3	输出	L	L	
117	—	—	—	—	A4	输出	L	L	
118	—	—	—	—	A5	输出	L	L	
119	—	—	—	—	A6	输出	L	L	
120	—	—	—	—	A7	输出	L	L	
126	—	—	—	—	A8	输出	L	L	
127	—	—	—	—	A9	输出	L	L	
128	—	—	—	—	A10	输出	L	L	
21	—	—	—	—	A11	输出	L	L	
22	—	—	—	—	A12	输出	L	L	
23	—	—	—	—	A13	输出	L	L	
24	—	—	—	—	A14	输出	L	L	
25	—	—	—	—	A15	输出	L	L	
59	—	—	—	—	D0	I/O/Z	Z	Z	<b>8051 数据总线</b> 。此双向总线在静态、总线读取输入和总线写入输出时处于高阻抗状态。该数据总线用于外部 8051 程序和数据存储器。该数据总线仅对外部总线访问有效，并在暂停时以低电平驱动。
60	—	—	—	—	D1	I/O/Z	Z	Z	
61	—	—	—	—	D2	I/O/Z	Z	Z	
62	—	—	—	—	D3	I/O/Z	Z	Z	
63	—	—	—	—	D4	I/O/Z	Z	Z	
86	—	—	—	—	D5	I/O/Z	Z	Z	
87	—	—	—	—	D6	I/O/Z	Z	Z	
88	—	—	—	—	D7	I/O/Z	Z	Z	
39	—	—	—	—	PSEN#	输出	H	H	<b>程序存储使能</b> 。此低电平有效信号表示自外部存储器的 8051 代码提取。它仅对以下程序存储器提取源有效：0x4000–0xFFFF (EA 引脚为低电平) 或 0x0000–0xFFFF (EA 引脚为高电平)。

注

11. 未使用的输入不必保留为悬空。按需绑定高电平与低电平。输出应仅为上拉式或下拉式，从而确保信号处于加电和待机模式。注：器件电源关闭时，不应驱动任何引脚。

12. 复位列指示在复位期间 (RESET# 激活) 或上电复位 (POR) 时信号状态。

表 11. 引脚描述 (续)

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	名称	类型	默认值	复位 <sup>[12]</sup>	说明
34	28	—	—		BKPT	输出	L	L	<b>断点。</b> 此引脚在 8051 地址总线与 BPADDRH/L 寄存器匹配并在 BREAKPT 寄存器 (BPEN = 1) 中启用断点时有效 (高电平)。如果 BREAKPT 寄存器中的 BPPULSE 位为 HIGH (高), 则此信号以高脉冲驱动 8 个频率为 12-/24-/48 MHz 的时钟。如果 BPPULSE 位为 LOW (低), 则该信号仍然保持 HIGH (高脉冲), 直到 8051 清除 BREAKPT 寄存器中的 BREAK 位时为止。
99	77	49	42	8B	RESET#	输入	不可用	不可用	<b>有效的 LOW (低电平) 复位。</b> 复位整个芯片。更多信息, 请参见第 “复位和唤醒” on page 9 节。
35	—	—	—	—	EA	输入	不可用	不可用	<b>外部访问。</b> 此引脚确定 8051 在 0x0000 与 0x3FFF 地址之间提取代码的位置。如果 EA = 0, 则 8051 从内部 RAM 中提取此代码。如果 EA = 1, 则 8051 从外部存储器中提取此代码。
12	11	12	5	1C	XTALIN	输入	不可用	不可用	<b>晶振输入。</b> 连接此信号至 24 MHz 的并联谐振电路, 连接主模晶振和负载电容至 GND。此外, 它还可以正确驱动从另一个时钟源衍生的具有 24-MHz 方波的 XTALIN。当从外部源驱动时, 驱动信号应为 3.3V 方波。
11	10	11	4	2C	XTALOUT	输出	不可用	不可用	<b>晶振输出。</b> 连接此信号至 24 MHz 的并联谐振电路, 连接主模晶振和负载电容至 GND。如果外部时钟用于驱动 XTALIN, 则此引脚保持为开路。
1	100	5	54	2B	在 CY7C68013A 上的 CLKOUT 和 CY7C68014A ----- 在 CY7C68015A 和 CY7C68016A 上的 PE1	O/Z  I/O/Z	12 MHz  I	时钟驱动  Z	<b>CLKOUT:</b> 12, 24 或 48 MHz 时钟, 相位锁定到 24 MHz 的输入时钟 8051 默认为 12 MHz 操作。8051 通过设置 CPUCS.1 = 1 来使此输出具有三种状态。  <b>PE1</b> 是双向 I/O 端口引脚。
<b>端口 A</b>									
82	67	40	33	8G	PA0 或 INT0#	I/O/Z	I (PA0)	I (PA0)	通过 PORTACFG.0 来选择函数的复用式引脚 <b>PA0</b> 是双向 I/O 端口引脚。 <b>INT0#</b> 是有效低电平 8051 INT0 中断输入信号, 其触发模式即可以是边沿触发 (IT0 = 1), 也可以是电平触发 (IT0 = 0)。
83	68	41	34	6G	PA1 或 INT1#	I/O/Z	I (PA1)	I (PA1)	通过以下方式选择函数的复用式引脚: PORTACFG.1 <b>PA1</b> 是双向 I/O 端口引脚。 <b>INT1#</b> 是有效低电平 8051 INT1 中断输入信号, 其触发模式即可以是边沿触发 (IT1 = 1), 也可以是电平触发 (IT1 = 0)。
84	69	42	35	8F	PA2 或 SLOE 或	I/O/Z	I (PA2)	Z (PA2)	通过两个位来选择函数的复用式引脚: IFCONFIG[1:0]。 <b>PA2</b> 是双向 I/O 端口引脚。 <b>SLOE</b> 是仅输入 / 输出使能, 带有与 FD[7..0] or FD[15..0] 连接的从器件 FIFO 可编程极性 (FIFOPINPOLAR.4)。

表 11. 引脚描述 (续)

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	名称	类型	默认值	复位 <sup>[12]</sup>	说明
85	70	43	36	7F	PA3 或 WU2	I/O/Z	I (PA3)	Z (PA3)	通过以下方式选择函数的复用式引脚: WAKEUP.7 和 OEA.3 <b>PA3</b> 是双向 I/O 端口引脚。 <b>WU2</b> 是备用的 <b>USB 唤醒源</b> , 通过 WU2POL (WAKEUP.4) 设置的 WU2EN 位 (WAKEUP.1) 和极性得以实现。如果 8051 处于暂停模式, 并且 WU2EN = 1, 则在此引脚上切换将启动 振荡器, 并中断 8051 以使其退出暂停模式。 如果 WU2EN = 1, 激活此引脚将禁止芯片暂 停。
89	71	44	37	6F	PA4 或 FIFOADR0	I/O/Z	I (PA4)	Z (PA4)	通过以下方式选择函数的复用式引脚: IFCONFIG[1..0]。 <b>PA4</b> 是双向 I/O 端口引脚。 <b>FIFOADR0</b> 是针对连接至 FD[7..0] 或 FD[15..0] 的从器件 FIFO 选择的仅输入地址。
90	72	45	38	8C	PA5 或 FIFOADR1	I/O/Z	I (PA5)	Z (PA5)	通过以下方式选择函数的复用式引脚: IFCONFIG[1..0]。 <b>PA5</b> 是双向 I/O 端口引脚。 <b>FIFOADR1</b> 是针对连接至 FD[7..0] 或 FD[15..0] 的从器件 FIFO 选择的仅输入地址。
91	73	46	39	7C	PA6 或 PKTEND	I/O/Z	I (PA6)	Z (PA6)	通过 IFCONFIG[1:0] 位来选择函数的复用式 引脚。 <b>PA6</b> 是双向 I/O 端口引脚。 <b>PKTEND</b> 是用于提交 FIFO 数据包到端点的 输入接口, 其极性可以通过 FIFOPINPOLAR.5 来编程。
92	74	47	40	6C	PA7 或 FLAGD 或 SLCS#	I/O/Z	I (PA7)	Z (PA7)	通过 IFCONFIG[1:0] 和 PORTACFG.7 位来选 择函数的复用式引脚。 <b>PA7</b> 是双向 I/O 端口引脚。 <b>FLAGD</b> 是可编程的从器件 FIFO 输出状态标 志信号。 <b>SLCS#</b> 传送其他所有从器件 FIFO 使能 / 探 针。
<b>端口 B</b>									
44	34	25	18	3H	PB0 或 FD[0]	I/O/Z	I (PB0)	Z (PB0)	通过以下位来选择函数的复用式引脚: IFCONFIG[1..0]。 <b>PB0</b> 是双向 I/O 端口引脚。 FD[0] 是双向 FIFO/GPIF 数据总线。
45	35	26	19	4F	PB1 或 FD[1]	I/O/Z	I (PB1)	Z (PB1)	通过以下位来选择函数的复用式引脚: IFCONFIG[1..0]。 <b>PB1</b> 是双向 I/O 端口引脚。 FD[1] 是双向 FIFO/GPIF 数据总线。
46	36	27	20	4H	PB2 或 FD[2]	I/O/Z	I (PB2)	Z (PB2)	通过以下位来选择函数的复用式引脚: IFCONFIG[1..0]。 <b>PB2</b> 是双向 I/O 端口引脚。 FD[2] 是双向 FIFO/GPIF 数据总线。
47	37	28	21	4G	PB3 或 FD[3]	I/O/Z	I (PB3)	Z (PB3)	通过以下位来选择函数的复用式引脚: IFCONFIG[1..0]。 <b>PB3</b> 是双向 I/O 端口引脚。 FD[3] 是双向 FIFO/GPIF 数据总线。
54	44	29	22	5H	PB4 或 FD[4]	I/O/Z	I (PB4)	Z (PB4)	通过以下位来选择函数的复用式引脚: IFCONFIG[1..0]。 <b>PB4</b> 是双向 I/O 端口引脚。 FD[4] 是双向 FIFO/GPIF 数据总线。

表 11. 引脚描述 (续)

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	名称	类型	默认值	复位 <sup>[12]</sup>	说明
55	45	30	23	5G	PB5 或 FD[5]	I/O/Z	I (PB5)	Z (PB5)	通过以下位来选择函数的复用式引脚: IFCONFIG[1..0]。 <b>PB5</b> 是双向 I/O 端口引脚。 <b>FD[5]</b> 是双向 FIFO/GPIF 数据总线。
56	46	31	24	5F	PB6 或 FD[6]	I/O/Z	I (PB6)	Z (PB6)	通过以下位来选择函数的复用式引脚: IFCONFIG[1..0]。 <b>PB6</b> 是双向 I/O 端口引脚。 <b>FD[6]</b> 是双向 FIFO/GPIF 数据总线。
57	47	32	25	6H	PB7 或 FD[7]	I/O/Z	I (PB7)	Z (PB7)	通过以下位来选择函数的复用式引脚: IFCONFIG[1..0]。 <b>PB7</b> 是双向 I/O 端口引脚。 <b>FD[7]</b> 是双向 FIFO/GPIF 数据总线。
<b>端口 C</b>									
72	57	—	—	—	PC0 或 GPIFADR0	I/O/Z	I (PC0)	Z (PC0)	通过 PORTCCFG.0 来选择函数的复用式引脚 <b>PC0</b> 是双向 I/O 端口引脚。 <b>GPIFADR0</b> 是 GPIF 地址输出引脚。
73	58	—	—	—	PC1 或 GPIFADR1	I/O/Z	I (PC1)	Z (PC1)	通过 PORTCCFG.1 来选择函数的复用式引脚 <b>PC1</b> 是双向 I/O 端口引脚。 <b>GPIFADR1</b> 是 GPIF 地址输出引脚。
74	59	—	—	—	PC2 或 GPIFADR2	I/O/Z	I (PC2)	Z (PC2)	通过 PORTCCFG.2 选择函数的复用式引脚 <b>PC2</b> 是双向 I/O 端口引脚。 <b>GPIFADR2</b> 是 GPIF 地址输出引脚。
75	60	—	—	—	PC3 或 GPIFADR3	I/O/Z	I (PC3)	Z (PC3)	通过 PORTCCFG.3 选择函数的复用式引脚 <b>PC3</b> 是双向 I/O 端口引脚。 <b>GPIFADR3</b> 是 GPIF 地址输出引脚。
76	61	—	—	—	PC4 或 GPIFADR4	I/O/Z	I (PC4)	Z (PC4)	通过 PORTCCFG.4 选择函数的复用式引脚 <b>PC4</b> 是双向 I/O 端口引脚。 <b>GPIFADR4</b> 是 GPIF 地址输出引脚。
77	62	—	—	—	PC5 或 GPIFADR5	I/O/Z	I (PC5)	Z (PC5)	通过 PORTCCFG.5 选择函数的复用式引脚 <b>PC5</b> 是双向 I/O 端口引脚。 <b>GPIFADR5</b> 是 GPIF 地址输出引脚。
78	63	—	—	—	PC6 或 GPIFADR6	I/O/Z	I (PC6)	Z (PC6)	通过 PORTCCFG.6 选择函数的复用式引脚 <b>PC6</b> 是双向 I/O 端口引脚。 <b>GPIFADR6</b> 是 GPIF 地址输出引脚。
79	64	—	—	—	PC7 或 GPIFADR7	I/O/Z	I (PC7)	Z (PC7)	通过 PORTCCFG.7 选择函数的复用式引脚 <b>PC7</b> 是双向 I/O 端口引脚。 <b>GPIFADR7</b> 是 GPIF 地址输出引脚。
<b>端口 D</b>									
102	80	52	45	8A	PD0 或 FD[8]	I/O/Z	I (PD0)	Z (PD0)	通过 IFCONFIG[1..0] 和 EPxFIFOCFG.0 (全局) 位来选择函数的复用式引脚。 <b>FD[8]</b> 是双向 FIFO/GPIF 数据总线。
103	81	53	46	7A	PD1 或 FD[9]	I/O/Z	I (PD1)	Z (PD1)	通过 IFCONFIG[1..0] 和 EPxFIFOCFG.0 (全局) 位来选择函数的复用式引脚。 <b>FD[9]</b> 是双向 FIFO/GPIF 数据总线。
104	82	54	47	6B	PD2 或 FD[10]	I/O/Z	I (PD2)	Z (PD2)	通过 IFCONFIG[1..0] 和 EPxFIFOCFG.0 (全局) 位来选择函数的复用式引脚。 <b>FD[10]</b> 是双向 FIFO/GPIF 数据总线。
105	83	55	48	6A	PD3 或 FD[11]	I/O/Z	I (PD3)	Z (PD3)	通过 IFCONFIG[1..0] 和 EPxFIFOCFG.0 (全局) 位来选择函数的复用式引脚。 <b>FD[11]</b> 是双向 FIFO/GPIF 数据总线。

表 11. 引脚描述 (续)

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	名称	类型	默认值	复位 <sup>[12]</sup>	说明
121	95	56	49	3B	PD4 或 FD[12]	I/O/Z	I (PD4)	Z (PD4)	通过 IFCONFIG[1..0] 和 EPxFIFOCFG.0 (全局) 位来选择函数的复用式引脚。 <b>FD[12]</b> 是双向 FIFO/GPIF 数据总线。
122	96	1	50	3A	PD5 或 FD[13]	I/O/Z	I (PD5)	Z (PD5)	通过 IFCONFIG[1..0] 和 EPxFIFOCFG.0 (全局) 位来选择函数的复用式引脚。 <b>FD[13]</b> 是双向 FIFO/GPIF 数据总线。
123	97	2	51	3C	PD6 或 FD[14]	I/O/Z	I (PD6)	Z (PD6)	通过 IFCONFIG[1..0] 和 EPxFIFOCFG.0 (全局) 位来选择函数的复用式引脚。 <b>FD[14]</b> 是双向 FIFO/GPIF 数据总线。
124	98	3	52	2A	PD7 或 FD[15]	I/O/Z	I (PD7)	Z (PD7)	通过 IFCONFIG[1..0] 和 EPxFIFOCFG.0 (全局) 位来选择函数的复用式引脚。 <b>FD[15]</b> 是双向 FIFO/GPIF 数据总线。
<b>端口 E</b>									
108	86	—	—	—	PE0 或 T0OUT	I/O/Z	I (PE0)	Z (PE0)	通过 PORTECFG.0 位来选择函数的复用式引脚。 <b>PE0</b> 是双向 I/O 端口引脚。 <b>T0OUT</b> 是源自 8051 定时器 - 计数器 0 的高电平有效信号。T0OUT 在定时器 0 溢出时的一个 CLKOUT 时钟周期输出高电平。如果在模式 3 (两个单独的定时器 / 计数器) 下运行定时器 0, 则 T0OUT 在低位字节定时器 / 计数器溢出时有效。
109	87	—	—	—	PE1 或 T1OUT	I/O/Z	I (PE1)	Z (PE1)	通过 PORTECFG.1 位来选择函数的复用式引脚。 <b>PE1</b> 是双向 I/O 端口引脚。 <b>T1OUT</b> 是源自 8051 定时器 - 计数器 1 的高电平有效信号。T1OUT 在 Timer1 溢出时的一个 CLKOUT 时钟周期输出高电平。如果在模式 3 (两个单独的定时器 / 计数器) 下运行定时器 1, 则 T1OUT 在低位字节定时器 / 计数器溢出时有效。
110	88	—	—	—	PE2 或 T2OUT	I/O/Z	I (PE2)	Z (PE2)	通过 PORTECFG.2 位来选择函数的复用式引脚。 <b>PE2</b> 是双向 I/O 端口引脚。 <b>T2OUT</b> 是源自 8051 定时器 2 的高电平有效输出信号。T2OUT 对定时器 / 计数器 2 溢出时的一个时钟周期有效 (高电平)。
111	89	—	—	—	PE3 或 RXD0OUT	I/O/Z	I (PE3)	Z (PE3)	通过 PORTECFG.3 位来选择函数的复用式引脚。 <b>PE3</b> 是双向 I/O 端口引脚。 <b>RXD0OUT</b> 是源自 8051 UART0 的高电平有效信号。如果选择 RXD0OUT, 并且 UART0 处于模式 0 下, 则该引脚仅在处于同步模式时才为 UART0 提供输出数据。否则它处于 1 模式。
112	90	—	—	—	PE4 或 RXD1OUT	I/O/Z	I (PE4)	Z (PE4)	通过 PORTECFG.4 位来选择函数的复用式引脚。 <b>PE4</b> 是双向 I/O 端口引脚。 <b>RXD1OUT</b> 是源自 8051 UART1 的高电平有效输出。如果选择 RXD1OUT, 并且 UART1 处于模式 0 下, 则该引脚仅在处于同步模式时才为 UART1 提供输出数据。在模式 1, 2 和 3 下, 该下引脚为高电平。



表 11. 引脚描述 (续)

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	名称	类型	默认值	复位 <sup>[12]</sup>	说明
113	91	—	—	—	PE5 或 INT6	I/O/Z	I (PE5)	Z (PE5)	通过 PORTECFG.5 位来选择函数的复用式引脚。 <b>PE5</b> 是双向 I/O 端口引脚。 <b>INT6</b> 是 8051 INT6 中断请求输入信号。INT6 引脚是边沿敏感型、有效的高电平。
114	92	—	—	—	PE6 或 T2EX	I/O/Z	I (PE6)	Z (PE6)	通过 PORTECFG.6 位来选择函数的复用式引脚。 <b>PE6</b> 是双向 I/O 端口引脚。 <b>T2EX</b> 是输入到 8051 定时器 2 的高电平有效输入信号。T2EX 在其下降沿上重载定时器 2。只有 EXEN2 位设置在 T2CON 中时，T2EX 才有效。
115	93	—	—	—	PE7 或 GPIFADR8	I/O/Z	I (PE7)	Z (PE7)	通过 PORTECFG.7 位来选择函数的复用式引脚。 <b>PE7</b> 是双向 I/O 端口引脚。 <b>GPIFADR8</b> 是 GPIF 地址输出引脚。
4	3	8	1	1A	RDY0 或 SLRD	输入	不可用	不可用	通过以下位来选择函数的复用式引脚： IFCONFIG[1..0]。 <b>RDY0</b> 是 GPIF 输入信号。 <b>SLRD</b> 是仅输入读取探针，带有与 FD[7..0] 或 FD[15..0] 连接的从器件 FIFO 可编程极性 (FIFOPINPOLAR.3)。
5	4	9	2	1B	RDY1 或 SLWR	输入	不可用	不可用	通过以下位来选择函数的复用式引脚： IFCONFIG[1..0]。 <b>RDY1</b> 是 GPIF 输入信号。 <b>SLWR</b> 是仅输入写入探针，带有与 FD[7..0] 或 FD[15..0] 连接的从器件 FIFO 可编程极性 (FIFOPINPOLAR.2)。
6	5	—	—	—	RDY2	输入	不可用	不可用	<b>RDY2</b> 是 GPIF 输入信号。
7	6	—	—	—	RDY3	输入	不可用	不可用	<b>RDY3</b> 是 GPIF 输入信号。
8	7	—	—	—	RDY4	输入	不可用	不可用	<b>RDY4</b> 是 GPIF 输入信号。
9	8	—	—	—	RDY5	输入	不可用	不可用	<b>RDY5</b> 是 GPIF 输入信号。
69	54	36	29	7H	CTL0 或 FLAGA	O/Z	H	L	通过以下位来选择函数的复用式引脚： IFCONFIG[1..0]。 <b>CTL0</b> 是 GPIF 控制输出。 <b>FLAGA</b> 是可编程的从器件 FIFO 输出状态标志信号。 通过 FIFOADR[1:0] 引脚选择的 FIFO 默认为可编程。
70	55	37	30	7G	CTL1 或 FLAGB	O/Z	H	L	通过以下位来选择函数的复用式引脚： IFCONFIG[1..0]。 <b>CTL1</b> 是 GPIF 控制输出。 <b>FLAGB</b> 是可编程的从器件 FIFO 输出状态标志信号。 通过 FIFOADR[1:0] 引脚选择的 FIFO 默认为 FULL (全速)。
71	56	38	31	8H	CTL2 或 FLAGC	O/Z	H	L	通过以下位来选择函数的复用式引脚： IFCONFIG[1..0]。 <b>CTL2</b> 是 GPIF 控制输出。 <b>FLAGC</b> 是可编程的从器件 FIFO 输出状态标志信号。 通过 FIFOADR[1:0] 引脚选择的 FIFO 默认为 EMPTY (空)。



表 11. 引脚描述 (续)

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	名称	类型	默认值	复位 <sup>[12]</sup>	说明
66	51	—	—	—	CTL3	O/Z	H	L	<b>CTL3</b> 是 GPIF 控制输出。
67	52	—	—	—	CTL4	输出	H	L	<b>CTL4</b> 是 GPIF 控制输出。
98	76	—	—	—	CTL5	输出	H	L	<b>CTL5</b> 是 GPIF 控制输出。
32	26	20	13	2G	在 CY7C68013A 上的 IFCLK 和 CY7C68014A	I/O/Z	Z	Z	接口时钟, 用于脉冲进入或输出从器件 FIFO 的同步时钟数据。IFCLK 还用作所有从器件 FIFO 控制信号和 GPIF 的时序参考。使用内部时钟 (IFCONFIG.7 = 1) 时, IFCLK 引脚可以通过 IFCONFIG.5 和 IFCONFIG.6 位配置到输出 (30/48 MHz)。IFCLK 可以通过设置 IFCONFIG.4 = 1 位进行反转, 无论是内部源还是外部源均如此。
					PE0 在 CY7C68015A 和 CY7C68016A	I/O/Z	I	Z	<b>PE0</b> 是双向 I/O 端口引脚。
28	22	—	—	—	INT4	输入	不可用	不可用	<b>INT4</b> 是 8051 INT4 中断请求输入信号。INT4 引脚是边沿敏感型、有效高电平。
106	84	—	—	—	INT5#	输入	不可用	不可用	<b>INT5#</b> 是 8051 INT5 中断请求输入信号。INT5 引脚是边沿敏感型、有效的低电平。
31	25	—	—	—	T2	输入	不可用	不可用	<b>T2</b> 是输入到 8051 定时器 2 的有效高电平 T2, 当 C/T2 = 1 时, 它对定时器 2 提供输入。当 C/T2 = 0 时, 定时器 2 不予使用此引脚。
30	24	—	—	—	T1	输入	不可用	不可用	<b>T1</b> 是 8051 定时器 1 的有效高电平 T1, 当 C/T1 = 1 时, 它对定时器 1 提供输入。当 C/T1 = 0 时, 定时器 1 不予使用该位。
29	23	—	—	—	T0	输入	不可用	不可用	<b>T0</b> 是 8051 定时器 0 的有效高电平 T0, 当 C/T0 = 1 时, 它对定时器 0 提供输入。当 C/T0 = 0 时, 定时器 0 不予使用该位。
53	43	—	—	—	RXD1	输入	不可用	不可用	<b>RXD1</b> 是 8051 UART1 的有效高电平输入信号, 在所有模式下, 它均可以向 UART 提供数据。
52	42	—	—	—	TXD1	输出	H	L	<b>TXD1</b> 是源自 8051 UART1 的有效高电平输出引脚, 它可以在同步模式下提供输出时钟, 在异步模式下提供输出数据。
51	41	—	—	—	RXD0	输入	不可用	不可用	<b>RXD0</b> 是输入到 8051 UART0 的有效高电平 RXD0 输入, 在所有模式下, 它均可以向 UART 提供数据。
50	40	—	—	—	TXD0	输出	H	L	<b>TXD0</b> 是源自 8051 UART0 的有效高电平 TXD0 输出, 它可以在同步模式下提供输出时钟, 在异步模式下提供输出数据。
42		—	—	—	CS#	输出	H	H	<b>CS#</b> 是针对外部存储器选择的有效低电平芯片。
41	32	—	—	—	WR#	输出	H	H	<b>WR#</b> 是针对外部存储器的有效低电平写入探针输出。
40	31	—	—	—	RD#	输出	H	H	<b>RD#</b> 是针对外部存储器的有效低电平读取探针输出。
38		—	—	—	OE#	输出	H	H	<b>OE#</b> 是针对外部存储器的有效低电平输出使能。
33	27	21	14	2H	已保留	输入	不可用	不可用	<b>已保留。</b> 连接至接地。

表 11. 引脚描述 (续)

128 TQFP	100 TQFP	56 SSOP	56 QFN	56 VFBGA	名称	类型	默认值	复位 <sup>[12]</sup>	说明
101	79	51	44	7B	WAKEUP (唤醒)	输入	不可用	不可用	<b>USB 唤醒。</b> 如果 8051 处于暂停模式, 则激活此引脚可以启动振荡器, 并中断 8051 以使其退出暂停模式。保持 WAKEUP (唤醒) 为激活状态, 这样将阻止 EZ-USB 芯片暂停。此引脚具有可编程极性 (WAKEUP.4)。
36	29	22	15	3F	SCL	OD	Z	Z (如果引导结束)	<b>I<sup>2</sup>C 接口的时钟。</b> 连接至 VCC, 电阻为 2.2K, 不考虑是否安装 I <sup>2</sup> C 外设。
37	30	23	16	3G	SDA	OD	Z	Z (如果引导结束)	<b>I<sup>2</sup>C 兼容接口的数据。</b> 连接至 VCC, 电阻为 2.2K, 不考虑是否安装 I <sup>2</sup> C 兼容外设。
2	1	6	55	5A	VCC	功耗	不可用	不可用	<b>VCC。</b> 连接至 3.3 V 电源。
26	20	18	11	1G	VCC	功耗	不可用	不可用	<b>VCC。</b> 连接至 3.3 V 电源。
43	33	24	17	7E	VCC	功耗	不可用	不可用	<b>VCC。</b> 连接至 3.3 V 电源。
48	38	—	—	—	VCC	功耗	不可用	不可用	<b>VCC。</b> 连接至 3.3 V 电源。
64	49	34	27	8E	VCC	功耗	不可用	不可用	<b>VCC。</b> 连接至 3.3 V 电源。
68	53	—	—	—	VCC	功耗	不可用	不可用	<b>VCC。</b> 连接至 3.3 V 电源。
81	66	39	32	5C	VCC	功耗	不可用	不可用	<b>VCC。</b> 连接至 3.3 V 电源。
100	78	50	43	5B	VCC	功耗	不可用	不可用	<b>VCC。</b> 连接至 3.3 V 电源。
107	85	—	—	—	VCC	功耗	不可用	不可用	<b>VCC。</b> 连接至 3.3 V 电源。
3	2	7	56	4B	GND	接地	不可用	不可用	<b>接地。</b>
27	21	19	12	1H	GND	接地	不可用	不可用	<b>接地。</b>
49	39	—	—	—	GND	接地	不可用	不可用	<b>接地。</b>
58	48	33	26	7D	GND	接地	不可用	不可用	<b>接地。</b>
65	50	35	28	8D	GND	接地	不可用	不可用	<b>接地。</b>
80	65	—	—	—	GND	接地	不可用	不可用	<b>接地。</b>
93	75	48	41	4C	GND	接地	不可用	不可用	<b>接地。</b>
116	94	—	—	—	GND	接地	不可用	不可用	<b>接地。</b>
125	99	4	53	4A	GND	接地	不可用	不可用	<b>接地。</b>
14	13	—	—	—	NC	不可用	不可用	不可用	<b>未连接。</b> 此引脚必须保持为开路。
15	14	—	—	—	NC	不可用	不可用	不可用	<b>未连接。</b> 此引脚必须保持为开路。
16	15	—	—	—	NC	不可用	不可用	不可用	<b>未连接。</b> 此引脚必须保持为开路。

## 寄存器摘要

在 FX2LP TRM 中非常详细地描述了 FX2LP 寄存器位的定义。

**表 12. FX2LP 寄存器摘要**

Hex	大小	名称	说明	b7	b6	b5	b4	b3	b2	b1	b0	默认值	访问
<b>GPIO 波形存储器</b>													
E400	128	WAVEDATA	GPIO 波形描述符 0, 1, 2, 3 数据	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
E480	128	已保留											
<b>GENERAL CONFIGURATION (通用配置)</b>													
E50D		GPCR2	通用配置寄存器 2	已保留	已保留	已保留	FULL_SPEED_ONLY	已保留	已保留	已保留	已保留	00000000	R
E600	1	CPUCS	CPU 控制与状态	0	0	PORTCSTB	CLKSPD1	CLKSPD0	CLKINV	CLKOE	8051RES	00000010	rrbbbbb
E601	1	IFCONFIG	接口配置 (端口, GPIO, 从器件 FIFO)	IFCLKSRC	3048MHZ	IFCLKOE	IFCLKPOL	ASYN	GSTATE	IFCFG1	IFCFG0	10000000	RW
E602	1	PINFLAGABSAB <sup>[13]</sup>	从器件 FIFO FLAGA 和 FLAGB 引脚配置	FLAGB3	FLAGB2	FLAGB1	FLAGB0	FLAGA3	FLAGA2	FLAGA1	FLAGA0	00000000	RW
E603	1	PINFLAGSCD <sup>[13]</sup>	从器件 FIFO FLAGC 和 FLAGD 引脚配置	FLAGD3	FLAGD2	FLAGD1	FLAGD0	FLAGC3	FLAGC2	FLAGC1	FLAGC0	00000000	RW
E604	1	FIFORESET <sup>[13]</sup>	恢复 FIFOs 到默认状态	NAKALL	0	0	0	EP3	EP2	EP1	EP0	xxxxxxx	W
E605	1	BREAKPT	断点控制	0	0	0	0	BREAK (断点)	BPPULSE	BPEN	0	00000000	rrrrbbb
E606	1	BPADDRH	断点地址 H	A15	A14	A13	A12	A11	A10	A9	A8	xxxxxxx	RW
E607	1	BPADDRL	断点地址 L	A7	A6	A5	A4	A3	A2	A1	A0	xxxxxxx	RW
E608	1	UART230	内部生成的参考时钟 230 Kbaud	0	0	0	0	0	0	230UART1	230UART0	00000000	rrrrbbb
E609	1	FIFOPINPOLAR <sup>[13]</sup>	从器件 FIFO 接口引脚极性	0	0	PKTEND	SLOE	SLRD	SLWR	EF	FF	00000000	rrbbbbb
E60A	1	REVID	芯片修订版	rv7	rv6	rv5	rv4	rv3	rv2	rv1	rv0	RevA 00000001	R
E60B	1	REVCTL <sup>[13]</sup>	芯片修订版控制	0	0	0	0	0	0	dyn_out	enh_pkt	00000000	rrrrbbb
<b>UDMA</b>													
E60C	1	GPIOHOLDAMOUNT	MSTB 保持时间 (针对 UDMA)	0	0	0	0	0	0	HOLDTIME1	HOLDTIME0	00000000	rrrrbbb
	3	已保留											
<b>ENDPOINT CONFIGURATION (断点配置)</b>													
E610	1	EP1OUTCFG	端点 1-OUT 配置	VALID (有效期)	0	TYPE1 (型号 1)	TYPE0 (型号 0)	0	0	0	0	10100000	brbbrrr
E611	1	EP1INCFG	端点 1-IN 配置	VALID (有效期)	0	TYPE1 (型号 1)	TYPE0 (型号 0)	0	0	0	0	10100000	brbbrrr
E612	1	EP2CFG	端点 2 配置	VALID (有效期)	DIR	TYPE1 (型号 1)	TYPE0 (型号 0)	SIZE	0	BUF1	BUF0	10100010	bbbbbbrb
E613	1	EP4CFG	端点 4 配置	VALID (有效期)	DIR	TYPE1 (型号 1)	TYPE0 (型号 0)	0	0	0	0	10100000	bbbbbrrr
E614	1	EP6CFG	端点 6 配置	VALID (有效期)	DIR	TYPE1 (型号 1)	TYPE0 (型号 0)	SIZE	0	BUF1	BUF0	11100010	bbbbbbrb
E615	1	EP8CFG	端点 8 配置	VALID (有效期)	DIR	TYPE1 (型号 1)	TYPE0 (型号 0)	0	0	0	0	11100000	bbbbbrrr
	2	已保留											
E618	1	EP2FIFOCFG <sup>[13]</sup>	端点 2/ 从器件 FIFO 配置	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E619	1	EP4FIFOCFG <sup>[13]</sup>	端点 4/ 从器件 FIFO 配置	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61A	1	EP6FIFOCFG <sup>[13]</sup>	端点 6/ 从器件 FIFO 配置	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61B	1	EP8FIFOCFG <sup>[13]</sup>	端点 8/ 从器件 FIFO 配置	0	INFM1	OEP1	AUTOOUT	AUTOIN	ZEROLENIN	0	WORDWIDE	00000101	rbbbbbrb
E61C	4	已保留											
E620	1	EP2AUTOINLENH <sup>[13]</sup>	端点 2 AUTOIN 数据包长度 H	0	0	0	0	0	PL10	PL9	PL8	00000010	rrrrbbb
E621	1	EP2AUTOINLENL <sup>[13]</sup>	端点 2 AUTOIN 数据包长度 L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E622	1	EP4AUTOINLENH <sup>[13]</sup>	端点 4 AUTOIN 数据包长度 H	0	0	0	0	0	0	PL9	PL8	00000010	rrrrbbb
E623	1	EP4AUTOINLENL <sup>[13]</sup>	端点 4 AUTOIN 数据包长度 L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E624	1	EP6AUTOINLENH <sup>[13]</sup>	端点 6 AUTOIN 数据包长度 H	0	0	0	0	0	PL10	PL9	PL8	00000010	rrrrbbb
E625	1	EP6AUTOINLENL <sup>[13]</sup>	端点 6 AUTOIN 数据包长度 L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E626	1	EP8AUTOINLENH <sup>[13]</sup>	端点 8 AUTOIN 数据包长度 H	0	0	0	0	0	0	PL9	PL8	00000010	rrrrbbb
E627	1	EP8AUTOINLENL <sup>[13]</sup>	端点 8 AUTOIN 数据包长度 L	PL7	PL6	PL5	PL4	PL3	PL2	PL1	PL0	00000000	RW
E628	1	ECCCFG	ECC 配置	0	0	0	0	0	0	0	ECCM	00000000	rrrrrb
E629	1	ECCRESET	ECC 复位	x	x	x	x	x	x	x	x	00000000	W
E62A	1	ECC1B0	ECC1 字节 0 地址	LINE15 (行 15)	LINE14 (行 14)	LINE13 (行 13)	LINE12 (行 12)	LINE11 (行 11)	LINE10 (行 10)	LINE9 (行 9)	LINE8 (行 8)	00000000	R

注

13. 读取和写入这些寄存器时, 可能要求同步延迟。有关“同步延迟”的详细信息, 请参见《技术参考手册》。

**表 12. FX2LP 寄存器摘要 (续)**

Hex	大小	名称	说明	b7	b6	b5	b4	b3	b2	b1	b0	默认值	访问
E62B	1	ECC1B1	ECC1 字节 1 地址	LINE7 (行 7)	LINE6 (行 6)	LINE5 (行 5)	LINE4 (行 4)	LINE3 (行 3)	LINE2 (行 2)	LINE1 (行 1)	LINE0 (行 0)	00000000	R
E62C	1	ECC1B2	ECC1 字节 2 地址	COL5	COL4	COL3	COL2	COL1	COL0	LINE17 (行 17)	LINE16 (行 16)	00000000	R
E62D	1	ECC2B0	ECC2 字节 0 地址	LINE15 (行 15)	LINE14 (行 14)	LINE13 (行 13)	LINE12 (行 12)	LINE11 (行 11)	LINE10 (行 10)	LINE9 (行 9)	LINE8 (行 8)	00000000	R
E62E	1	ECC2B1	ECC2 字节 1 地址	LINE7 (行 7)	LINE6 (行 6)	LINE5 (行 5)	LINE4 (行 4)	LINE3 (行 3)	LINE2 (行 2)	LINE1 (行 1)	LINE0 (行 0)	00000000	R
E62F	1	ECC2B2	ECC2 字节 2 地址	COL5	COL4	COL3	COL2	COL1	COL0	0	0	00000000	R
E630	1	EP2FIFOPFH <sup>[13]</sup>	端点 2/ 从器件 FIFO 可编程标志 H	DECIS	PKTSTAT	IN:PKTS[2] OUT:PFC12	IN:PKTS[1] OUT:PFC11	IN:PKTS[0] OUT:PFC10	0	PFC9	PFC8	10001000	bbbbbrbb
E630	1	EP2FIFOPFH <sup>[13]</sup>	端点 2/ 从器件 FIFO 可编程标志 L	DECIS	PKTSTAT	OUT:PFC12	OUT:PFC11	OUT:PFC10	0	PFC9	IN:PKTS[2] OUT:PFC8	10001000	bbbbbrbb
E631	1	EP2FIFOPFL <sup>[13]</sup>	端点 2/ 从器件 FIFO 可编程标志 L	PFC7	PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E631	1	EP2FIFOPFL <sup>[13]</sup>	端点 2/ 从器件 FIFO 可编程标志 L	IN:PKTS[1] OUT:PFC7	IN:PKTS[0] OUT:PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E632	1	EP4FIFOPFH <sup>[13]</sup>	端点 4/ 从器件 FIFO 可编程标志 H	DECIS	PKTSTAT	0	IN: PKTS[1] OUT:PFC10	IN: PKTS[0] OUT:PFC9	0	0	PFC8	10001000	bbrbrrrb
E632	1	EP4FIFOPFH <sup>[13]</sup>	端点 4/ 从器件 FIFO 可编程标志 L	DECIS	PKTSTAT	0	OUT:PFC10	OUT:PFC9	0	0	PFC8	10001000	bbrbrrrb
E633	1	EP4FIFOPFL <sup>[13]</sup>	端点 4/ 从器件 FIFO 可编程标志 L	PFC7	PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E633	1	EP4FIFOPFL <sup>[13]</sup>	端点 4/ 从器件 FIFO 可编程标志 L	IN: PKTS[1] OUT:PFC7	IN: PKTS[0] OUT:PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E634	1	EP6FIFOPFH <sup>[13]</sup>	端点 6/ 从器件 FIFO 可编程标志 H	DECIS	PKTSTAT	IN:PKTS[2] OUT:PFC12	IN:PKTS[1] OUT:PFC11	IN:PKTS[0] OUT:PFC10	0	PFC9	PFC8	00001000	bbbbbrbb
E634	1	EP6FIFOPFH <sup>[13]</sup>	端点 6/ 从器件 FIFO 可编程标志 L	DECIS	PKTSTAT	OUT:PFC12	OUT:PFC11	OUT:PFC10	0	PFC9	IN:PKTS[2] OUT:PFC8	00001000	bbbbbrbb
E635	1	EP6FIFOPFL <sup>[13]</sup>	端点 6/ 从器件 FIFO 可编程标志 L	PFC7	PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E635	1	EP6FIFOPFL <sup>[13]</sup>	端点 6/ 从器件 FIFO 可编程标志 L	IN:PKTS[1] OUT:PFC7	IN:PKTS[0] OUT:PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E636	1	EP8FIFOPFH <sup>[13]</sup>	端点 8/ 从器件 FIFO 可编程标志 H	DECIS	PKTSTAT	0	IN: PKTS[1] OUT:PFC10	IN: PKTS[0] OUT:PFC9	0	0	PFC8	00001000	bbrbrrrb
E636	1	EP8FIFOPFH <sup>[13]</sup>	端点 8/ 从器件 FIFO 可编程标志 L	DECIS	PKTSTAT	0	OUT:PFC10	OUT:PFC9	0	0	PFC8	00001000	bbrbrrrb
E637	1	EP8FIFOPFL <sup>[13]</sup>	端点 8/ 从器件 FIFO 可编程标志 L	PFC7	PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
E637	1	EP8FIFOPFL <sup>[13]</sup>	端点 8/ 从器件 FIFO 可编程标志 L	IN: PKTS[1] OUT:PFC7	IN: PKTS[0] OUT:PFC6	PFC5	PFC4	PFC3	PFC2	PFC1	PFC0	00000000	RW
8		已保留											
E640	1	EP2ISOINPKTS	每个帧 (1-3) EP2 (若 ISO)IN 数据包	AADJ	0	0	0	0	0	INPPF1	INPPF0	00000001	brrrrrrb
E641	1	EP4ISOINPKTS	每个帧 (1-3) EP4 (若 ISO)IN 数据包	AADJ	0	0	0	0	0	INPPF1	INPPF0	00000001	brrrrrrr
E642	1	EP6ISOINPKTS	每个帧 (1-3) EP6 (若 ISO)IN 数据包	AADJ	0	0	0	0	0	INPPF1	INPPF0	00000001	brrrrrrb
E643	1	EP8ISOINPKTS	每个帧 (1-3) EP8 (若 ISO)IN 数据包	AADJ	0	0	0	0	0	INPPF1	INPPF0	00000001	brrrrrrr
E644	4	已保留											
E648	1	INPKTEND <sup>[13]</sup>	强制 IN 数据包结束	跳过	0	0	0	EP3	EP2	EP1	EP0	xxxxxxx	W
E649	7	OUTPKTEND <sup>[13]</sup>	强制 OUT 数据包结束	跳过	0	0	0	EP3	EP2	EP1	EP0	xxxxxxx	W
		INTERRUPTS (中断)											
E650	1	EP2FIFOIE <sup>[13]</sup>	端点 2/ 从器件 FIFO 标志中断使能	0	0	0	0	EDGE PF	PF	EF	FF	00000000	RW
E651	1	EP2FIFOIRQ <sup>[13,14]</sup>	端点 2/ 从器件 FIFO 标志中断请求	0	0	0	0	0	PF	EF	FF	00000000	rrrrrrbb
E652	1	EP4FIFOIE <sup>[13]</sup>	端点 4/ 从器件 FIFO 标志中断使能	0	0	0	0	EDGE PF	PF	EF	FF	00000000	RW
E653	1	EP4FIFOIRQ <sup>[13,14]</sup>	端点 4/ 从器件 FIFO 标志中断请求	0	0	0	0	0	PF	EF	FF	00000000	rrrrrrbb
E654	1	EP6FIFOIE <sup>[13]</sup>	端点 6/ 从器件 FIFO 标志中断使能	0	0	0	0	EDGE PF	PF	EF	FF	00000000	RW
E655	1	EP6FIFOIRQ <sup>[13,14]</sup>	端点 6/ 从器件 FIFO 标志中断请求	0	0	0	0	0	PF	EF	FF	00000000	rrrrrrbb
E656	1	EP8FIFOIE <sup>[13]</sup>	端点 8/ 从器件 FIFO 标志中断使能	0	0	0	0	EDGE PF	PF	EF	FF	00000000	RW
E657	1	EP8FIFOIRQ <sup>[13,14]</sup>	端点 8/ 从器件 FIFO 标志中断请求	0	0	0	0	0	PF	EF	FF	00000000	rrrrrrbb
E658	1	IBNIE	IN-BULK-NAK 中断使能	0	0	EP8	EP6	EP4	EP2	EP1	EP0	00000000	RW
E659	1	IBNIRQ <sup>[14]</sup>	IN-BULK-NAK 中断请求	0	0	EP8	EP6	EP4	EP2	EP1	EP0	00xxxxxx	rrrrrrbb
E65A	1	NAKIE	端点 Ping-NAK/IBN 中断使能	EP8	EP6	EP4	EP2	EP1	EP0	0	IBN	00000000	RW
E65B	1	NAKIRQ <sup>[14]</sup>	端点 Ping-NAK/IBN 中断请求	EP8	EP6	EP4	EP2	EP1	EP0	0	IBN	xxxxxx0x	bbbbrrrb
E65C	1	USBIE	USB 中断使能	0	EP0ACK	HSGRANT	URES	SUSP	SUTOK	SOF	SUDAV	00000000	RW

**注**

14. 仅可以复位寄存器，而无法设置。

表 12. FX2LP 寄存器摘要 (续)

Hex	大小	名称	说明	b7	b6	b5	b4	b3	b2	b1	b0	默认值	访问
E65D	1	USBIRQ <sup>[14]</sup>	USB 中断请求	0	EP0ACK	HSGRANT	URES	SUSP	SUTOK	SOF	SUDAV	0xxxxxxx	rbrrrrrr
E65E	1	EPIE	端点中断使能	EP8	EP6	EP4	EP2	EP1OUT	EP1IN	EP0OUT	EP0IN	00000000	RW
E65F	1	EPIRQ <sup>[14]</sup>	端点中断请求	EP8	EP6	EP4	EP2	EP1OUT	EP1IN	EP0OUT	EP0IN	0	RW
E660	1	GPIFIE <sup>[13]</sup>	GPIF 中断使能	0	0	0	0	0	0	GPIFWF	GPIFDONE	00000000	RW
E661	1	GPIFIRQ <sup>[13]</sup>	GPIF 中断请求	0	0	0	0	0	0	GPIFWF	GPIFDONE	000000xx	RW
E662	1	USBERRIE	USB 错误中断使能	ISOEP8	ISOEP6	ISOEP4	ISOEP2	0	0	0	ERRLIMIT	00000000	RW
E663	1	USBERRIRQ <sup>[14]</sup>	USB 错误中断请求	ISOEP8	ISOEP6	ISOEP4	ISOEP2	0	0	0	ERRLIMIT	0000000x	bbrrrrrr
E664	1	ERRCNTLIM	USB 错误计数器和限值	EC3	EC2	EC1	EC0	LIMIT3 ( 限值 3)	LIMIT2 ( 限值 2)	LIMIT1 ( 限值 1)	LIMIT0 ( 限值 0)	xxxx0100	rrrrrrrr
E665	1	CLRERRCNT	清除错误计数器 EC3:0	x	x	x	x	x	x	x	x	xxxxxxx	W
E666	1	INT2IVEC	中断 2 (USB) 自动矢量	0	I2V4	I2V3	I2V2	I2V1	I2V0	0	0	00000000	R
E667	1	INT4IVEC	中断 4 ( 从器件 FIFO 和 GPIF) 自动矢量	1	0	I4V3	I4V2	I4V1	I4V0	0	0	10000000	R
E668	1	INTSET-UP	中断 2 和 4 设置	0	0	0	0	AV2EN	0	INT4SRC	AV4EN	00000000	RW
E669	7	已保留											
		INPUT/OUTPUT ( 输入 / 输出 )											
E670	1	PORTACFG	I/O PORTA 备用配置	FLAGD	SLCS	0	0	0	0	INT1	INT0	00000000	RW
E671	1	PORTCCFG	I/O PORTC 备用配置	GPIFA7	GPIFA6	GPIFA5	GPIFA4	GPIFA3	GPIFA2	GPIFA1	GPIFA0	00000000	RW
E672	1	PORTECFG	I/O PORTE 备用配置	GPIFA8	T2EX	INT6	RXD1OUT	RXD0OUT	T2OUT	T1OUT	T0OUT	00000000	RW
E673	4	已保留											
E677	1	已保留											
E678	1	I <sup>2</sup> CS	I <sup>2</sup> C 总线控制与状态	START	STOP ( 停止 )	LASTRD	ID1	ID0	BERR	ACK	DONE ( 完成 )	000xx000	bbrrrrrr
E679	1	I2DAT	I <sup>2</sup> C 总线 Data	d7	d6	d5	d4	d3	d2	d1	d0	xxxxxxx	RW
E67A	1	I <sup>2</sup> CTL	I <sup>2</sup> C 总线控制	0	0	0	0	0	0	STOPIE	400KHZ	00000000	RW
E67B	1	XAUTODAT1	当 APTREN=1 时, Autoptr1 MOVX 访问	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
E67C	1	XAUTODAT2	当 APTREN=1 时, Autoptr2 MOVX 访问	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	RW
		UDMA CRC											
E67D	1	UDMACRCH <sup>[13]</sup>	UDMA CRC MSB	CRC15	CRC14	CRC13	CRC12	CRC11	CRC10	CRC9	CRC8	01001010	RW
E67E	1	UDMACRCL <sup>[13]</sup>	UDMA CRC LSB	CRC7	CRC6	CRC5	CRC4	CRC3	CRC2	CRC1	CRC0	10111010	RW
E67F	1	UDMACRC-QUALIFIER	UDMA CRC 限定符	QENABLE	0	0	0	QSTATE	QSIGNA2	QSIGNA1	QSIGNA0	00000000	bbrrrrrr
		USB CONTROL (USB 控制)											
E680	1	USBCS	USB 控制与状态	HSM	0	0	0	DISCON	NOSYNSOF	RENUM	SIGRSUME	x0000000	rrrrrrrr
E681	1	SUSPEND	将芯片置于暂停	x	x	x	x	x	x	x	x	xxxxxxx	W
E682	1	WAKEUPCS	唤醒控制与状态	WU2	WU	WU2POL	WUPOL	0	DPEN	WU2EN	WUEN	xx000101	bbrrrrrr
E683	1	TOGCTL	切换控制	Q	S	R	I/O	EP3	EP2	EP1	EP0	x0000000	rrrrrrrr
E684	1	USBFRAMEH	USB 帧计数 H	0	0	0	0	0	FC10	FC9	FC8	00000xxx	R
E685	1	USBFRAMEL	USB 帧计数 L	FC7	FC6	FC5	FC4	FC3	FC2	FC1	FC0	xxxxxxx	R
E686	1	MICROFRAME	微帧计数, 0-7	0	0	0	0	0	MF2	MF1	MF0	00000xxx	R
E687	1	FNADDR	USB 函数地址	0	FA6	FA5	FA4	FA3	FA2	FA1	FA0	0xxxxxxx	R
E688	2	保留											
		ENDPOINTS ( 端点 )											
E68A	1	EP0BCH <sup>[13]</sup>	端点 0 字节计数 H	(BC15)	(BC14)	(BC13)	(BC12)	(BC11)	(BC10)	(BC9)	(BC8)	xxxxxxx	RW
E68B	1	EP0BCL <sup>[13]</sup>	端点 0 字节计数 L	(BC7)	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxx	RW
E68C	1	保留											
E68D	1	EP1OUTBC	端点 1 OUT ( 输出 ) 字节计数	0	BC6	BC5	BC4	BC3	BC2	BC1	BC0	0xxxxxxx	RW
E68E	1	保留											
E68F	1	EP1INBC	端点 1 IN 字节计数	0	BC6	BC5	BC4	BC3	BC2	BC1	BC0	0xxxxxxx	RW
E690	1	EP2BCH <sup>[13]</sup>	端点 2 字节计数 H	0	0	0	0	0	BC10	BC9	BC8	00000xxx	RW
E691	1	EP2BCL <sup>[13]</sup>	端点 2 字节计数 L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxx	RW
E692	2	保留											
E694	1	EP4BCH <sup>[13]</sup>	端点 4 字节计数 H	0	0	0	0	0	0	BC9	BC8	00000xxx	RW
E695	1	EP4BCL <sup>[13]</sup>	端点 4 字节计数 L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxx	RW
E696	2	保留											
E698	1	EP6BCH <sup>[13]</sup>	端点 6 字节计数 H	0	0	0	0	0	BC10	BC9	BC8	00000xxx	RW
E699	1	EP6BCL <sup>[13]</sup>	端点 6 字节计数 L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxx	RW
E69A	2	保留											
E69C	1	EP8BCH <sup>[13]</sup>	端点 8 字节计数 H	0	0	0	0	0	0	BC9	BC8	00000xxx	RW
E69D	1	EP8BCL <sup>[13]</sup>	端点 8 字节计数 L	BC7/SKIP	BC6	BC5	BC4	BC3	BC2	BC1	BC0	xxxxxxx	RW
E69E	2	保留											
E6A0	1	EP0CS	端点 0 控制和状态	HSNAK	0	0	0	0	0	BUSY	STALL	10000000	bbrrrrrr
E6A1	1	EP1OUTCS	端点 1 OUT 控制和状态	0	0	0	0	0	0	BUSY	STALL	00000000	bbrrrrrr
E6A2	1	EP1INCS	端点 1 IN 控制和状态	0	0	0	0	0	0	BUSY	STALL	00000000	bbrrrrrr

表 12. FX2LP 寄存器摘要 (续)

Hex	大小	名称	说明	b7	b6	b5	b4	b3	b2	b1	b0	默认值	访问
E6A3	1	EP2CS	端点 2 控制和状态	0	NPAK2	NPAK1	NPAK0	FULL (满)	EMPTY (空)	0	STALL	00101000	rrrrrrb
E6A4	1	EP4CS	端点 4 控制和状态	0	0	NPAK1	NPAK0	FULL (满)	EMPTY (空)	0	STALL	00101000	rrrrrrb
E6A5	1	EP6CS	端点 6 控制和状态	0	NPAK2	NPAK1	NPAK0	FULL (满)	EMPTY (空)	0	STALL	00000100	rrrrrrb
E6A6	1	EP8CS	端点 8 控制和状态	0	0	NPAK1	NPAK0	FULL (满)	EMPTY (空)	0	STALL	00000100	rrrrrrb
E6A7	1	EP2FIFOFLGS	端点 2/ 从器件 FIFO 标志	0	0	0	0	0	PF	EF	FF	00000010	R
E6A8	1	EP4FIFOFLGS	端点 4/ 从器件 FIFO 标志	0	0	0	0	0	PF	EF	FF	00000010	R
E6A9	1	EP6FIFOFLGS	端点 6/ 从器件 FIFO 标志	0	0	0	0	0	PF	EF	FF	00000110	R
E6AA	1	EP8FIFOFLGS	端点 8/ 从器件 FIFO 标志	0	0	0	0	0	PF	EF	FF	00000110	R
E6AB	1	EP2FIFOBCH	端点 2/ 从器件 FIFO 总字节计数 H	0	0	0	BC12	BC11	BC10	BC9	BC8	00000000	R
E6AC	1	EP2FIFOBCL	端点 2/ 从器件 FIFO 总字节计数 L	BC7	BC6	BC5	BC4	BC3	BC2	BC1	BC0	00000000	R
E6AD	1	EP4FIFOBCH	端点 4/ 从器件 FIFO 总字节计数 H	0	0	0	0	0	BC10	BC9	BC8	00000000	R
E6AE	1	EP4FIFOBCL	端点 4/ 从器件 FIFO 总字节计数 L	BC7	BC6	BC5	BC4	BC3	BC2	BC1	BC0	00000000	R
E6AF	1	EP6FIFOBCH	端点 6/ 从器件 FIFO 总字节计数 H	0	0	0	0	BC11	BC10	BC9	BC8	00000000	R
E6B0	1	EP6FIFOBCL	端点 6/ 从器件 FIFO 总字节计数 L	BC7	BC6	BC5	BC4	BC3	BC2	BC1	BC0	00000000	R
E6B1	1	EP8FIFOBCH	端点 8/ 从器件 FIFO 总字节计数 H	0	0	0	0	0	BC10	BC9	BC8	00000000	R
E6B2	1	EP8FIFOBCL	端点 8/ 从器件 FIFO 总字节计数 L	BC7	BC6	BC5	BC4	BC3	BC2	BC1	BC0	00000000	R
E6B3	1	SUDPTRH	设置数据指针地址高字节	A15	A14	A13	A12	A11	A10	A9	A8	xxxxxxx	RW
E6B4	1	SUDPTL	设置数据指针地址低字节	A7	A6	A5	A4	A3	A2	A1	0	xxxxxxx0	bbbbbbbr
E6B5	1	SUDPTRCTL	设置数据指针自动模式	0	0	0	0	0	0	0	SDPAUTO	00000001	RW
	2	保留											
E6B8	8	SET-UPDAT	8 个字节的设置数据	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxx	R
			SET-UPDAT[0] = bmRequestType										
			SET-UPDAT[1] = bmRequest										
			SET-UPDAT[2:3] = wValue										
			SET-UPDAT[4:5] = wIndex										
			SET-UPDAT[6:7] = wLength										
		GPIF											
E6C0	1	GPWFSELECT	波形选择器	SINGLEWR1	SINGLEWR0	SINGLERD1	SINGLERD0	FIFOWR1	FIFOWR0	FIFORD1	FIFORD0	11100100	RW
E6C1	1	GPIDLECS	GPIF 已完成, GPIF IDLE 驱动模式	DONE (完成)	0	0	0	0	0	0	IDLEDRV	10000000	RW
E6C2	1	GPIDLECTL	静态总线, CTL 状态	0	0	CTL5	CTL4	CTL3	CTL2	CTL1	CTL0	11111111	RW
E6C3	1	GPICLTCFG	CTL 驱动类型	TRICTL	0	CTL5	CTL4	CTL3	CTL2	CTL1	CTL0	00000000	RW
E6C4	1	GPIFADRH <sup>[13]</sup>	GPIF 地址 H	0	0	0	0	0	0	0	GPIFA8	00000000	RW
E6C5	1	GPIFADRL <sup>[13]</sup>	GPIF 地址 L	GPIFA7	GPIFA6	GPIFA5	GPIFA4	GPIFA3	GPIFA2	GPIFA1	GPIFA0	00000000	RW
		FLOWSTATE (流态)											
E6C6	1	FLOWSTATE (流态)	流态使能和选择器	FSE	0	0	0	0	FS2	FS1	FS0	00000000	brrrrbbb
E6C7	1	FLOWLOGIC (流逻辑)	流态逻辑	LFUNC1	LFUNC0	TERMA2	TERMA1	TERMA0	TERMB2	TERMB1	TERMB0	00000000	RW
E6C8	1	FLOWEQ0CTL	在流态中的 CTL 引脚状态 (当逻辑 = 0 时)	CTL0E3	CTL0E2	CTL0E1/CTL5	CTL0E0/CTL4	CTL3	CTL2	CTL1	CTL0	00000000	RW
E6C9	1	FLOWEQ1CTL	在流态中的 CTL 引脚状态 (当逻辑 = 1 时)	CTL0E3	CTL0E2	CTL0E1/CTL5	CTL0E0/CTL4	CTL3	CTL2	CTL1	CTL0	00000000	RW
E6CA	1	FLOWHOLDOFF	触发抑制配置	HOPERIOD3	HOPERIOD2	HOPERIOD1	HOPERIOD0	HOSTATE	HOCTL2	HOCTL1	HOCTL0	00010010	RW
E6CB	1	FLOWSTB	触发抑制探针配置	SLAVE (从器件)	RDYASYNC	CTLTOGL	SUSTAIN (维持)	0	MSTB2	MSTB1	MSTB0	00100000	RW
E6CC	1	FLOWSTBEDGE	流态上升/下降沿配置	0	0	0	0	0	0	FALLING (下降)	RISING (上升)	00000001	rrrrrrbb
E6CD	1	FLOWSTBPERIOD	主控 - 探针半周期	D7	D6	D5	D4	D3	D2	D1	D0	00000010	RW
E6CE	1	GPIFTCB3 <sup>[13]</sup>	GPIF 任务计数字节 3	TC31	TC30	TC29	TC28	TC27	TC26	TC25	TC24	00000000	RW
E6CF	1	GPIFTCB2 <sup>[13]</sup>	GPIF 任务计数字节 2	TC23	TC22	TC21	TC20	TC19	TC18	TC17	TC16	00000000	RW
E6D0	1	GPIFTCB1 <sup>[13]</sup>	GPIF 任务计数字节 1	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8	00000000	RW
E6D1	1	GPIFTCB0 <sup>[13]</sup>	GPIF 任务计数字节 0	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0	00000001	RW
	2	保留										00000000	RW
		保留											
		保留											
E6D2	1	EP2GPIFFLGSEL <sup>[13]</sup>	端点 2 GPIF 标志选择	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6D3	1	EP2GPIFPFSTOP	端点 2 GPIF 在传输标志上停止操作	0	0	0	0	0	0	0	FIFO2FLAG	00000000	RW
E6D4	1	EP2GPIFTRIG <sup>[13]</sup>	端点 2 GPIF 触发器	x	x	x	x	x	x	x	x	xxxxxxx	W
	3	保留											
		保留											
		保留											



**表 12. FX2LP 寄存器摘要 (续)**

Hex	大小	名称	说明	b7	b6	b5	b4	b3	b2	b1	b0	默认值	访问
E6DA	1	EP4GPIFFLGSEL <sup>[13]</sup>	端点 4 GPIF 标志选择	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6DB	1	EP4GPIPFSTOP	端点 4 GPIF 在 GPIF 标志上停止操作	0	0	0	0	0	0	0	FIFO4FLAG	00000000	RW
E6DC	1	EP4GPIFTRIG <sup>[13]</sup>	端点 4 GPIF 触发器	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	保留											
		保留											
		保留											
E6E2	1	EP6GPIFFLGSEL <sup>[13]</sup>	端点 6 GPIF 标志选择	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6E3	1	EP6GPIPFSTOP	端点 6 GPIF 在传输标志上停止操作	0	0	0	0	0	0	0	FIFO6FLAG	00000000	RW
E6E4	1	EP6GPIFTRIG <sup>[13]</sup>	端点 6 GPIF 触发器	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	保留											
		保留											
		保留											
E6EA	1	EP8GPIFFLGSEL <sup>[13]</sup>	端点 8 GPIF 标志选择	0	0	0	0	0	0	FS1	FS0	00000000	RW
E6EB	1	EP8GPIPFSTOP	端点 8 GPIF 在传输标志上停止操作	0	0	0	0	0	0	0	FIFO8FLAG	00000000	RW
E6EC	1	EP8GPIFTRIG <sup>[13]</sup>	端点 8 GPIF 触发器	x	x	x	x	x	x	x	x	xxxxxxxx	W
	3	保留											
E6F0	1	XGPIFSGLDATH	GPIF 数据 H (仅限 16 位模式)	D15	D14	D13	D12	D11	D10	D9	D8	xxxxxxxx	RW
E6F1	1	XGPIFSGLDATLX	读取 / 写入 GPIF 数据 L 和操作触发	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E6F2	1	XGPIFSGLDATLNOX	读取 / 写入 GPIF 数据 L 和触发操作	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	R
E6F3	1	GPIFREADYCFG	内部 RDY, 同步 / 异步, RDY 引脚状态	INTRDY	SAS	TCXRDY5	0	0	0	0	0	00000000	bbbbrrrr
E6F4	1	GPIFREADYSTAT	GPIF 就绪状态	0	0	RDY5	RDY4	RDY3	RDY2	RDY1	RDY0	00xxxxxx	R
E6F5	1	GPIFABORT	中止 GPIF 波形	x	x	x	x	x	x	x	x	xxxxxxxx	W
E6F6	2	保留											
		ENDPOINT BUFFERS (端点缓冲区)											
E740	64	EP0BUF	EP0-IN/-OUT 缓冲区	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E780	64	EP10OUTBUF	EP1-OUT 缓冲区	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E7C0	64	EP1INBUF	EP1-IN 缓冲区	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
E800	2048	保留											RW
F000	1024	EP2FIFOBUF	512/1024 字节 EP 2/ 从器件 FIFO 缓冲区 (输入或输出)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
F400	512	EP4FIFOBUF	512 字节 EP 4/ 从器件 FIFO 缓冲区 (输入或输出)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
F600	512	保留											
F800	1024	EP6FIFOBUF	512/1024 字节 EP 6/ 从器件 FIFO 缓冲区 (输入或输出)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
FC00	512	EP8FIFOBUF	512 字节 EP 8/ 从器件 FIFO 缓冲区 (输入或输出)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
FE00	512	保留											



**表 12. FX2LP 寄存器摘要 (续)**

Hex	大小	名称	说明	b7	b6	b5	b4	b3	b2	b1	b0	默认值	访问
xxxx		PC 配置字节		0	DISCON	0	0	0	0	0	400KHZ	xxxxxxxx [16]	n/a (不适用)
		特殊功能寄存器 (SFRs)											
80	1	IOA <sup>[15]</sup>	端口 A (位寻址)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
81	1	SP	堆栈指针	D7	D6	D5	D4	D3	D2	D1	D0	00000111	RW
82	1	DPL0	数据指针 0 L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
83	1	DPH0	数据指针 0 H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
84	1	DPL1 <sup>[15]</sup>	数据指针 1 L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
85	1	DPH1 <sup>[15]</sup>	数据指针 1 H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
86	1	DPS <sup>[15]</sup>	数据指针 0/1 选择	0	0	0	0	0	0	SEL		00000000	RW
87	1	PCON	电源控制	SMOD0	x	1	1	x	x	x	IDLE	00110000	RW
88	1	TCON	定时器 / 计数器控制 (位寻址)	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00000000	RW
89	1	TMOD	定时器 / 计数器模式控制	GATE	CT	M1	M0	GATE	CT	M1	M0	00000000	RW
8A	1	TL0	定时器 0 重载 L	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
8B	1	TL1	定时器 1 重载 L	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
8C	1	TH0	定时器 0 重载 H	D15	D14	D13	D12	D11	D10	D9	D8	00000000	RW
8D	1	TH1	定时器 1 重载 H	D15	D14	D13	D12	D11	D10	D9	D8	00000000	RW
8E	1	CKCON <sup>[15]</sup>	时钟控制	x	x	T2M	T1M	T0M	MD2	MD1	MD0	00000001	RW
8F	1	保留											
90	1	IOB <sup>[15]</sup>	端口 B (位寻址)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
91	1	EXIF <sup>[15]</sup>	外部中断标志	IE5	IE4	PCINT	USBNT	1	0	0	0	00001000	RW
92	1	MPAGE <sup>[15]</sup>	使用 @R0 / @R1 MOVX 上位地址字节	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
93	5	保留											
98	1	SCON0	串行端口 0 控制 (位寻址)	SM0_0	SM1_0	SM2_0	REN_0	TB8_0	RB8_0	TI_0	RI_0	00000000	RW
99	1	SBUF0	串行端口 0 数据缓冲区	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
9A	1	AUTOPTRH1 <sup>[15]</sup>	自动指针 1 地址 H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
9B	1	AUTOPTRL1 <sup>[15]</sup>	自动指针 1 地址 L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
9C	1	保留											
9D	1	AUTOPTRH2 <sup>[15]</sup>	自动指针 2 地址 H	A15	A14	A13	A12	A11	A10	A9	A8	00000000	RW
9E	1	AUTOPTRL2 <sup>[15]</sup>	自动指针 2 地址 L	A7	A6	A5	A4	A3	A2	A1	A0	00000000	RW
9F	1	保留											
A0	1	IOC <sup>[15]</sup>	端口 C (位寻址)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
A1	1	INT2CLR <sup>[15]</sup>	中断 2 清除	x	x	x	x	x	x	x	x	xxxxxxxx	W
A2	1	INT4CLR <sup>[15]</sup>	中断 4 清除	x	x	x	x	x	x	x	x	xxxxxxxx	W
A3	5	保留											
A8	1	IE	中断使能 (位寻址)	EA	ES1	ET2	ES0	ET1	EX1	ET0	EX0	00000000	RW
A9	1	保留											
AA	1	EP2468STAT <sup>[15]</sup>	端点 2, 4, 6, 8 状态标志	EP8F	EP8E	EP6F	EP6E	EP4F	EP4E	EP2F	EP2E	01011010	R
AB	1	EP24FIFOFLGS <sup>[15]</sup>	端点 2, 4/ 从器件 FIFO 状态标志	0	EP4PF	EP4EF	EP4FF	0	EP2PF	EP2EF	EP2FF	00100010	R
AC	1	EP68FIFOFLGS <sup>[15]</sup>	端点 6, 8/ 从器件 FIFO 状态标志	0	EP8PF	EP8EF	EP8FF	0	EP6PF	EP6EF	EP6FF	01100110	R
AD	2	保留											
AF	1	AUTOPTRSETUP <sup>[15]</sup>	自动指针 1 和 2 设置	0	0	0	0	0	APTR2INC	APTR1INC	APTREN	00000110	RW
B0	1	IOD <sup>[15]</sup>	端口 D (位寻址)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
B1	1	IOE <sup>[15]</sup>	端口 E (无位寻址)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
B2	1	OEA <sup>[15]</sup>	端口 A 输出使能	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B3	1	OEB <sup>[15]</sup>	端口 B 输出使能	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B4	1	OEC <sup>[15]</sup>	端口 C 输出使能	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B5	1	OED <sup>[15]</sup>	端口 D 输出使能	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B6	1	OEE <sup>[15]</sup>	端口 E 输出使能	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
B7	1	保留											
B8	1	IP	中断优先级 (位寻址)	1	PS1	PT2	PS0	PT1	PX1	PT0	PX0	10000000	RW
B9	1	保留											
BA	1	EP01STAT <sup>[15]</sup>	端点 0 和 1 的状态	0	0	0	0	0	EP1INBSY	EP1OUTBSY	EP0BSY	00000000	R
BB	1	GPIFTRIG <sup>[15, 13]</sup>	端点 2, 4, 6, 8 GPIF 从器件 FIFO 触发器	DONE (完成)	0	0	0	0	RW	EP1	EP0	10000xxx brrrrbbb	
BC	1	保留											
BD	1	GPIFSGLDATH <sup>[15]</sup>	GPIF 数据 H (仅限 16 位模式)	D15	D14	D13	D12	D11	D10	D9	D8	xxxxxxxx	RW

**注**

15. SFR 不是标准的 8051 架构部件。

16. 如果未通过 SIE 检测 EEPROM, 那么默认值为 00000000。

表 12. FX2LP 寄存器摘要 (续)

Hex	大小	名称	说明	b7	b6	b5	b4	b3	b2	b1	b0	默认值	访问
BE	1	GPIFSGLDATLX <sup>[15]</sup>	GPIF 数据 L (带有触发器)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	RW
BF	1	GPIFSGLDATLNOX <sup>[15]</sup>	GPIF 数据 L (不带有触发器)	D7	D6	D5	D4	D3	D2	D1	D0	xxxxxxxx	R
C0	1	SCON1 <sup>[15]</sup>	串行端口 1 控制 (位寻址)	SM0_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1	00000000	RW
C1	1	SBUF1 <sup>[15]</sup>	串行端口 1 数据缓冲区	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
C2	6	保留											
C8	1	T2CON	定时器 / 计数器 2 控制 (位寻址)	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CT2	CPRL2	00000000	RW
C9	1	保留											
CA	1	RCAP2L	捕捉定时器 2, 自动重新加载, 计数器增加	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
CB	1	RCAP2H	捕捉定时器 2, 自动重新加载, 计数器增加	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
CC	1	TL2	定时器 2 重载 L	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
CD	1	TH2	定时器 2 重载 H	D15	D14	D13	D12	D11	D10	D9	D8	00000000	RW
CE	2	保留											
D0	1	PSW	程序状态字 (位寻址)	CY	AC	F0	RS1	RS0	OV	F1	P	00000000	RW
D1	7	保留											
D8	1	EICON <sup>[15]</sup>	外部中断控制	SMOD1	1	ERESI	RESI	INT6	0	0	0	01000000	RW
D9	7	保留											
E0	1	ACC	累加器 (位寻址)	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
E1	7	保留											
E8	1	EIE <sup>[15]</sup>	外部中断使能	1	1	1	EX6	EX5	EX4	EI <sup>2</sup> C	EUSB	11100000	RW
E9	7	保留											
F0	1	B	B (位寻址)	D7	D6	D5	D4	D3	D2	D1	D0	00000000	RW
F1	7	保留											
F8	1	EIP <sup>[15]</sup>	外部中断优先级控制	1	1	1	PX6	PX5	PX4	PI <sup>2</sup> C	PUSB	11100000	RW
F9	7	保留											

R = 所有只读位  
 W = 所有只写位  
 r = 只读位  
 w = 只写位  
 b = 读/写位

## 绝对最大额定值

超过最大额定值可能会缩短器件的使用寿命。用户指导未经过测试。

存储温度 .....	-65 °C 到 +150 °C
供电 (商业级) 环境温度 .....	0 °C 到 +70 °C
供电 (工业级) 环境温度 .....	-40 °C 到 +105 °C
接地潜能的供电电压 .....	0.5 V 到 +4.0 V
任何输入引脚的直流输入电压 <sup>[17]</sup> .....	5.25 V
在高 Z 状态下用来输出的直流电压	0.5 V 到 $V_{CC} + 0.5 V$
功率耗散 .....	300 mW
静态放电电压 .....	>2000 V
最大输出电流, 每个 I/O 端口 .....	10 mA
最大输出电流, 所有 5 个 I/O 端口 (128 引脚和 100 引脚的封装) .....	50 mA

## 热特性

下表显示了各种封装的热特性:

**表 13. 热特性**

工具包	环境温度 (°C)	$\theta_{Jc}$ 壳温热电阻 (°C/W)	$\theta_{Ja}$ 结温热电阻 (°C/W)
56 SSOP	70	24.4	47.7
100 TQFP	70	11.9	45.9
128 TQFP	70	15.5	43.2
56 QFN	70	10.6	25.2
56 VFBGA	70	30.9	58.6

使用下列公式, 可以计算结温  $\theta_j$ :  $\theta_j = P \cdot \theta_{Ja} + \theta_a$

其中:

P = 功耗

$\theta_{Ja}$  = 结温 ( $\theta_{Jc} + \theta_{Ca}$ )

$\theta_a$  = 环境温度 (70 °C)

使用下列公式, 可以计算壳温  $\theta_c$ :  $\theta_c = P \cdot \theta_{Ca} + \theta_a$

其中:

P = 功耗

$\theta_{Ca}$  = 壳温

$\theta_a$  = 环境温度 (70 °C)

## 运行条件

$T_A$ (偏压环境温度) (商业级) .....	0 °C 到 +70 °C
$T_A$ (偏压环境温度) (工业级) .....	-40 °C 到 +105 °C
供电电压 .....	+3.00 V 到 +3.60 V
接电电压 .....	0 V
$F_{OSC}$ (振荡器或晶振频率) .....	24 MHz $\pm$ 100 ppm, 并行谐振

### 注

17. 芯片断电时, 不为 I/O 供电。

## 直流电特性

表 14. 直流特性

参数	说明	条件	最小值	典型值	最大值	单位
VCC	供电电压	—	3.00	3.3	3.60	V
VCC 上升	0 - 3.3 V	—	200	—	—	ns
V <sub>IH</sub>	输入高电平电压	—	2	—	5.25	V
V <sub>IL</sub>	输入低电平电压	—	-0.5	—	0.8	V
V <sub>IH_X</sub>	晶振输入高电压	—	2	—	5.25	V
V <sub>IL_X</sub>	晶振输入低电压	—	-0.5	—	0.8	V
I <sub>I</sub>	输入漏电流	0 < V <sub>IN</sub> < VCC	—	—	±10	µA
V <sub>OH</sub>	输出电压高电平	I <sub>OUT</sub> = 4 mA	2.4	—	—	V
V <sub>OL</sub>	输出低电平电压	I <sub>OUT</sub> = -4 mA	—	—	0.4	V
I <sub>OH</sub>	输出电流偏高	—	—	—	4	mA
I <sub>OL</sub>	输出电流偏低	—	—	—	4	mA
C <sub>IN</sub>	输入引脚电容	D+/D- 除外	—	—	10	pF
		D+/D-	—	—	15	pF
I <sub>SUSP</sub>	暂停电流	已连接	—	300	380 <sup>[18]</sup>	µA
	CY7C68014/CY7C68016	已断开	—	100	150 <sup>[18]</sup>	µA
	暂停电流	已连接	—	0.5	1.2 <sup>[18]</sup>	µA
	CY7C68013/CY7C68015	已断开	—	0.3	1.0 <sup>[18]</sup>	µA
I <sub>CC</sub>	供电电流	8051 运行, 连接至 USB HS	—	50	85	mA
		8051 运行, 连接至 USB FS	—	35	65	mA
T <sub>RESET</sub>	有效功耗后的复位时间	V <sub>CC</sub> min = 3.0 V	5.0	—	—	ms
	引脚加电后复位		200	—	—	µs

## USB 收发器

USB 2.0 支持全速或高速模式。

### 注

18. 在 25 °C 及 VCC 电压最大时测得。

## 交流电气特性

### USB 收发器

USB 2.0 支持全速或高速模式。

### 程序存储器读取

图 12. 程序存储器读取时序图

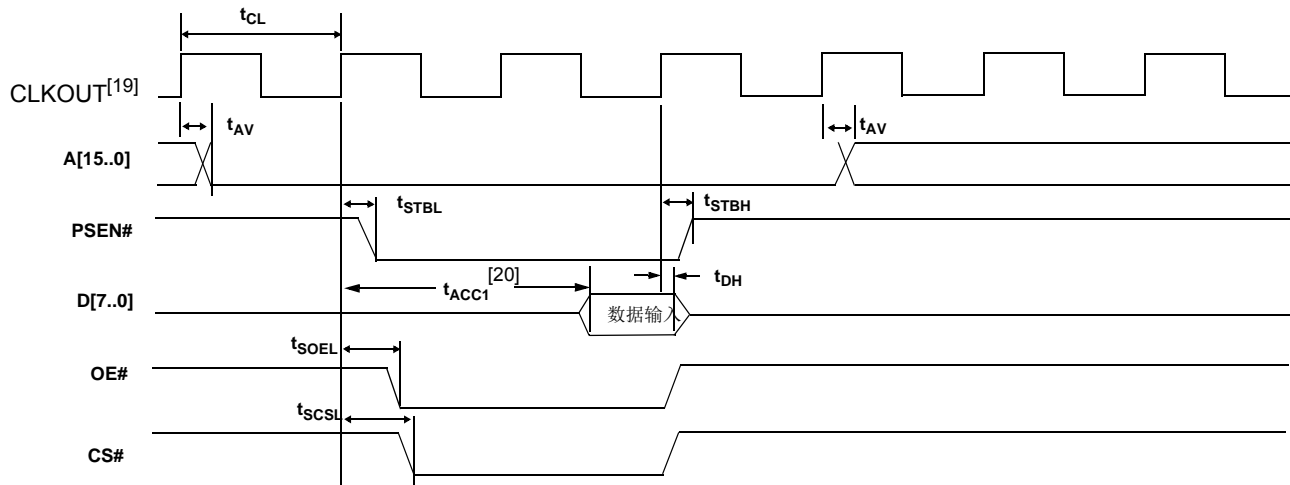


表 15. 程序存储器读取参数

参数	说明	最小值	典型值	最大值	单位	注
$t_{CL}$	1/CLKOUT 频率	—	20.83	—	ns	48 MHz
		—	41.66	—	ns	24 MHz
		—	83.2	—	ns	12 MHz
$t_{AV}$	从时钟到有效地址的延迟	0	—	10.7	ns	—
$t_{STBL}$	从时钟到 PSEN 低电平	0	—	8	ns	—
$t_{STBH}$	从时钟到 PSEN 高电平	0	—	8	ns	—
$t_{SOEL}$	从时钟到 OE 低电平	—	—	11.1	ns	—
$t_{SCSL}$	从时钟到 CS 低电平	—	—	13	ns	—
$t_{DSU}$	从数据设置到时钟	9.6	—	—	ns	—
$t_{DH}$	数据保持时间	0	—	—	ns	—

#### 注

19. CLKOUT 显示为正极性。

20. 根据以下这些参数计算  $t_{ACC1}$ :

$$t_{ACC1}(24 \text{ MHz}) = 3 \cdot t_{CL} - t_{AV} - t_{DSU} = 106 \text{ ns.}$$

$$t_{ACC1}(48 \text{ MHz}) = 3 \cdot t_{CL} - t_{AV} - t_{DSU} = 43 \text{ ns.}$$

## 数据存储器读取 [21]

图 13. 数据存储器读取时序图

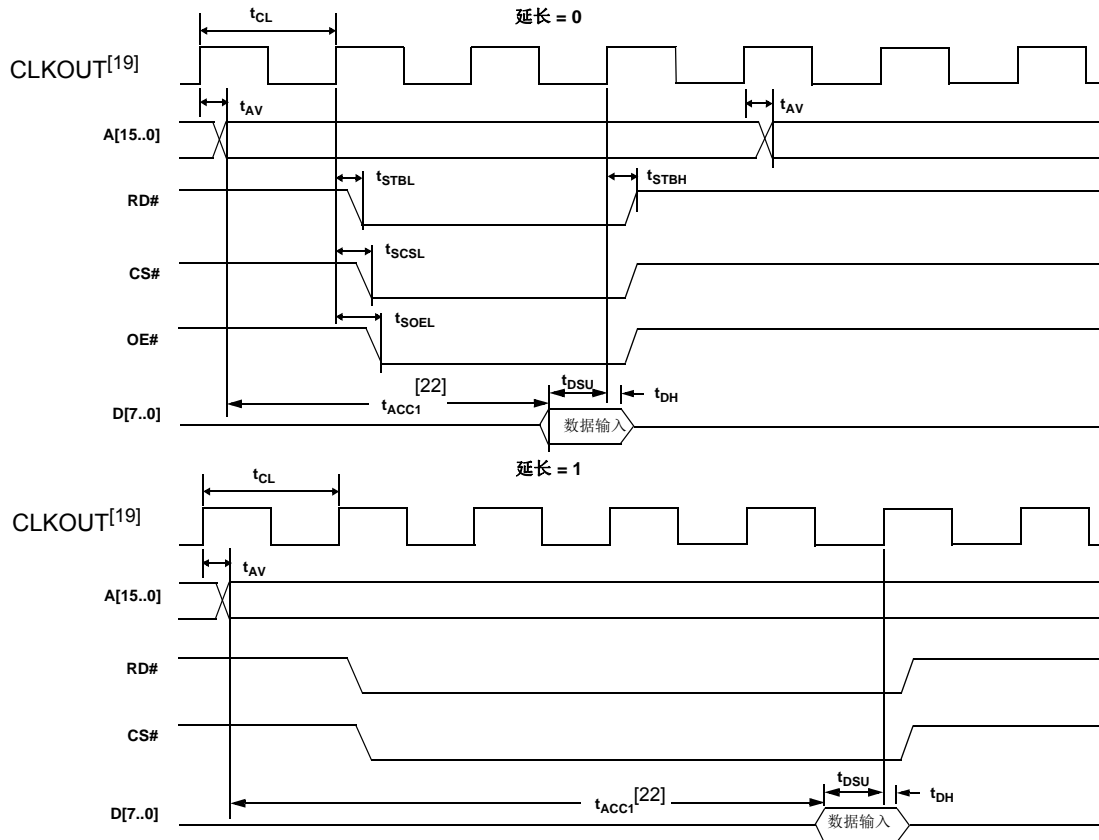


表 16. 数据存储器读取参数

参数	说明	最小值	典型值	最大值	单位	注
$t_{CL}$	1/CLKOUT 频率	—	20.83	—	ns	48 MHz
		—	41.66	—	ns	24 MHz
		—	83.2	—	ns	12 MHz
$t_{AV}$	从时钟到有效地址的延迟	—	—	10.7	ns	—
$t_{STBL}$	从时钟到 RD 低电平	—	—	11	ns	—
$t_{STBH}$	从时钟 RD HIGH	—	—	11	ns	—
$t_{SCSL}$	从时钟到 CS 低电平	—	—	13	ns	—
$t_{SOEL}$	从时钟到 OE 低电平	—	—	11.1	ns	—
$t_{DSU}$	从数据设置到时钟	9.6	—	—	ns	—
$t_{DH}$	数据保留时间	0	—	—	ns	—

使用 AUTOPTTR1 或 AUTOPTTR2 来寻址外部存储器时，AUTOPTTR1 地址仅在 RD# 或 WR# 为活动时才有效。AUTOPTTR2 地址在整个周期内均有效，并满足基于延长值的地址有效时间

### 注

21. 内存周期延长特性允许 EZ-USB 固件调整数据存储器访问速度而非代码存储访问速度。可在技术手册第 12.1.2 节找到包括选通宽度时序典型值具体信息。

22. 根据以下这些参数计算  $t_{ACC2}$  和  $t_{ACC3}$ ：  
 $t_{ACC2}(24 \text{ MHz}) = 3 \cdot t_{CL} - t_{AV} - t_{DSU} = 106 \text{ ns}$ .  
 $t_{ACC2}(48 \text{ MHz}) = 3 \cdot t_{CL} - t_{AV} - t_{DSU} = 43 \text{ ns}$ .  
 $t_{ACC3}(24 \text{ MHz}) = 5 \cdot t_{CL} - t_{AV} - t_{DSU} = 190 \text{ ns}$ .  
 $t_{ACC3}(48 \text{ MHz}) = 5 \cdot t_{CL} - t_{AV} - t_{DSU} = 86 \text{ ns}$ .

## 数据存储器写入 [23]

图 14. 数据存储器写入时序图

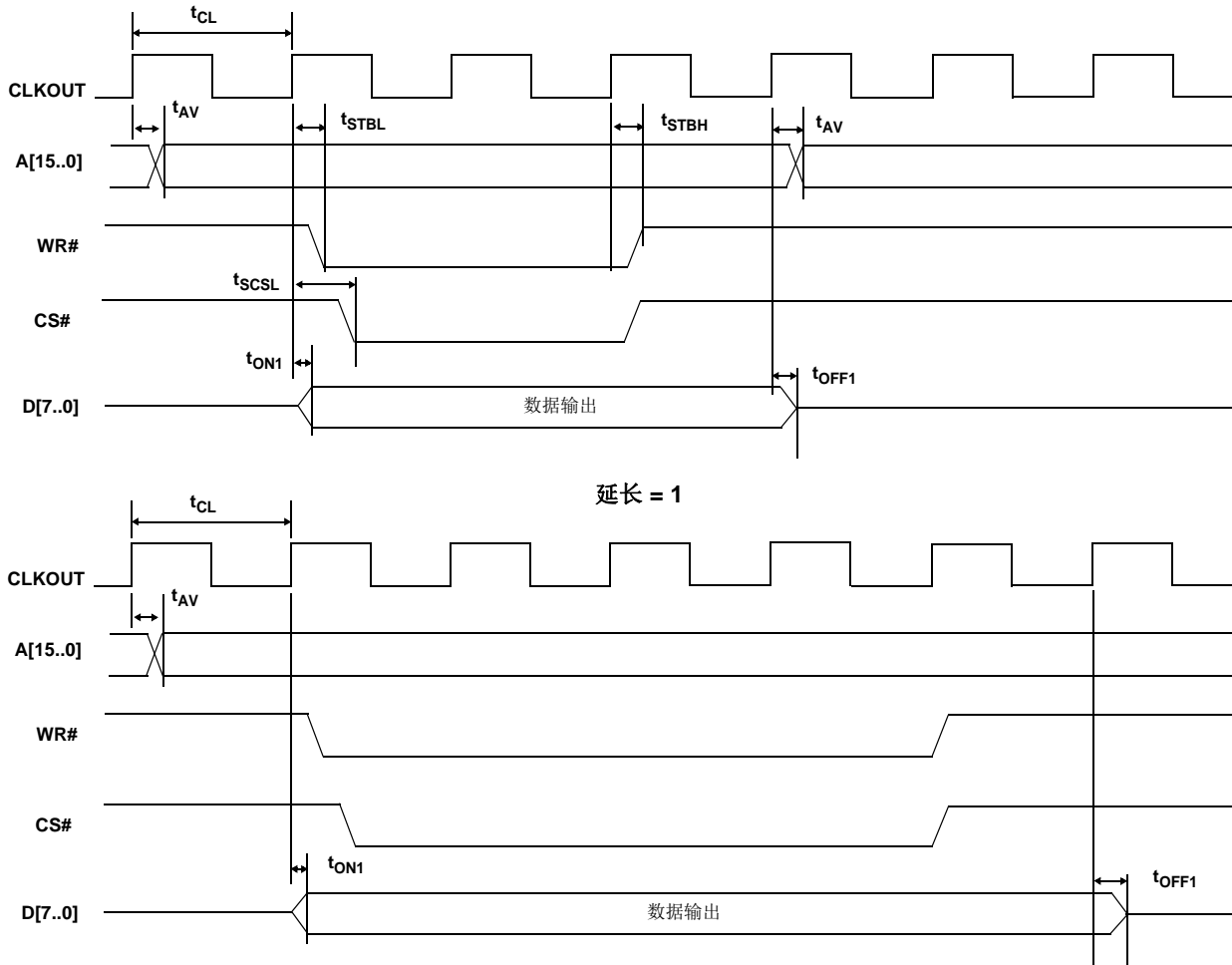


表 17. 数据存储器写入参数

参数	说明	最小值	最大值	单位	注
$t_{AV}$	从时钟到有效地址的延迟	0	10.7	ns	—
$t_{STBL}$	从时钟到 WR 脉冲低电平	0	11.2	ns	—
$t_{STBH}$	从时钟到 WR 脉冲高电平	0	11.2	ns	—
$t_{SCSL}$	从时钟到 CS 脉冲低电平	—	13.0	ns	—
$t_{ON1}$	从时钟到数据启用	0	13.1	ns	—
$t_{OFF1}$	从时钟到数据保持时间	0	13.1	ns	—

使用 AUTOPTTR1 或 AUTOPTTR2 来寻址外部存储器时，AUTOPTTR1 地址仅在 RD# 或 WR# 为活动时才有效。AUTOPTTR2 地址在整个周期内均有效，并满足基于延长值的地址有效时间。

### 注

23. 内存周期延长特性允许 EZ-USB 固件调整数据存储器访问速度而非代码存储器访问速度。可在技术手册第 12.1.2 节找到包括选通宽度时序典型值具体信息。地址周期宽度可以据此解释。



## PORTC 探针特性时序

在 100 引脚版本和 128 引脚封装中提供 RD# 和 WR#。在 100 引脚和 128 引脚版本中，当 8051 读取 / 写入 PORTC 时，可以设置 8051 控制位来触发 RD# 和 WR# 引脚。此功能通过在 CPUCS 寄存器中设置 PORTCSTB 来得以实现。

RD# 和 WR# 探针在访问 PORTC 时的两个 CLKOUT 周期内激活。

在 PORTC 更新之后的两个时钟周期内激活 WR# 探针，因此，该探针在这两个时钟周期内保持有效，如图 16 所示。

就读取来说，在激活 RD# 之前的 3 个时钟周期内，PORTC 的值是 8051 读取输入的值。当 8051 已完成读取 PORTC 函数时，从

这一点开始算起，在 3 个时钟周期后的 2 个时钟周期内触发 RD#。

RD# 信号负责提示外部逻辑准备下一个数据字节。激活 RD# 信号本身时，内部并无任何采样，它仅仅是一个用来获取下一个字节准备数据的预提取式信号。因此，使用该信号时要考虑是否满足延续到下一次读取的设置时间。

此 RD# 脉冲的目的是通知外设 8051 在激活 RD# 前的 3 个 CLKOUT 周期内是否完成读取 PORTC 及数据是否被锁存到 PORTC。脉动 RD# 后，外部逻辑可以更新 PORTC 上的数据。

下面是访问 PORTC 时读取和写入 探针函数的时序图。有关 RD# 和 WR# 信号传输延迟的详细信息，请参考 节 和 节。

图 16. 通过 8051 访问 PORTC 时的 WR# 探针函数

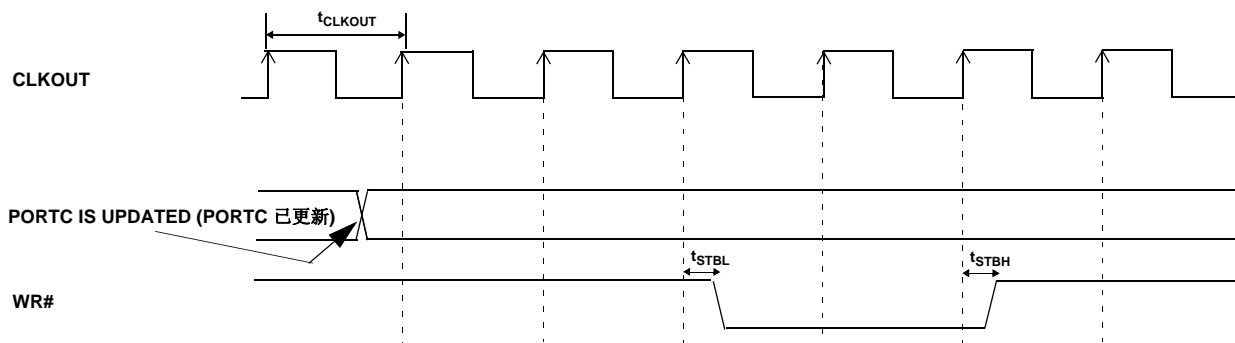
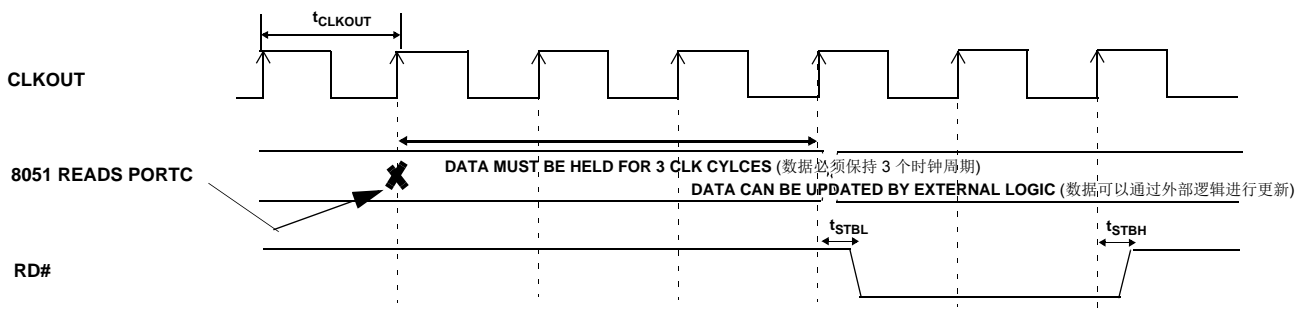
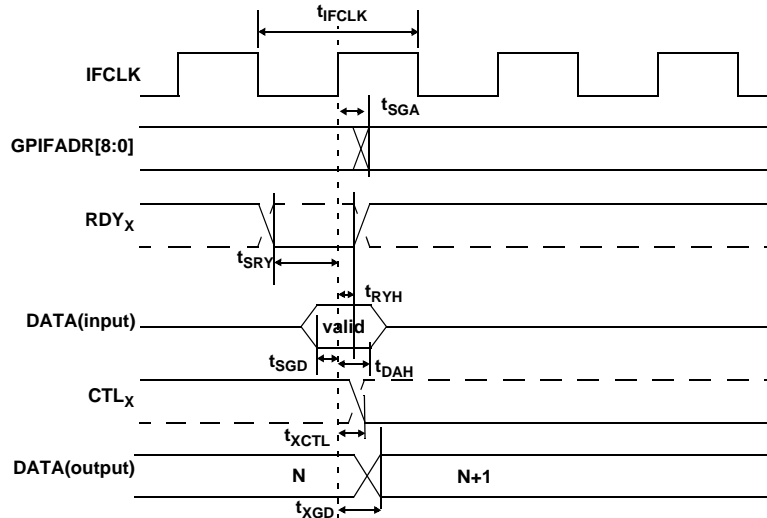


图 17. 通过 8051 访问 PORTC 时的 RD# 探针函数



**GPIF 同步信号**
**图 18. GPIF 同步信号时序图** <sup>[24]</sup>

**表 18. GPIF 同步信号参数 (包含内部源 IFCLK)** <sup>[24, 25]</sup>

参数	说明	最小值	最大值	典型值		单位
				最小值	最大值	
$t_{IFCLK}$	IFCLK 周期	20.83	—	—	—	ns
$t_{SRY}$	从 $RDY_x$ 到时钟设置时间	8.9	—	—	—	ns
$t_{RYH}$	从时钟到 $RDY_x$	0	—	—	—	ns
$t_{SGD}$	从 GPIF 数据到时钟设置时间	9.2	—	—	—	ns
$t_{DAH}$	GPIF 数据保持时间	0	—	—	—	ns
$t_{SGA}$	从时钟到 GPIF 地址的传输延迟	—	7.5	—	—	ns
$t_{XGD}$	从时钟到 GPIF 数据输出传输延迟	—	10	—	—	ns
$t_{XCTL}$	从时钟到 $CTL_x$ 输出传输延迟	—	6.7	—	—	ns
$t_{IFCLKR}$	IFCLK 上升时间	—	—	—	900	ps
$t_{IFCLKF}$	IFCLK 下降时间	—	—	—	900	ps
$t_{IFCLKOD}$	IFCLK 输出占空比	—	—	49	51	%
$t_{IFCLKJ}$	IFCLK 抖动 (峰值到峰值)	—	—	—	300	ps

**表 19. GPIF 同步信号参数 (包含内部源 IFCLK)** <sup>[25]</sup>

参数	说明	最小值	最大值	单位
$t_{IFCLK}$	IFCLK 周期 <sup>[26]</sup>	20.83	200	ns
$t_{SRY}$	从 $RDY_x$ 到时钟设置时间	2.9	—	ns
$t_{RYH}$	从时钟到 $RDY_x$	3.7	—	ns
$t_{SGD}$	从 GPIF 数据到时钟设置时间	3.2	—	ns
$t_{DAH}$	GPIF 数据保持时间	4.5	—	ns
$t_{SGA}$	从时钟到 GPIF 地址的传输延迟	—	11.5	ns
$t_{XGD}$	从时钟到 GPIF 数据输出传输延迟	—	15	ns
$t_{XCTL}$	从时钟到 $CTL_x$ 输出传输延迟	—	10.7	ns

**注**

24. 短划线表示带有可编程极性的信号。

25. 使用内部 48 MHz IFCLK 时, GPIF 异步  $RDY_x$  最低设置时间为 50 ns。

26. IFCLK 必须不超过 48 MHz。

## 从器件 FIFO 同步读取

图 19. 从器件 FIFO 同步读取时序图 [24]

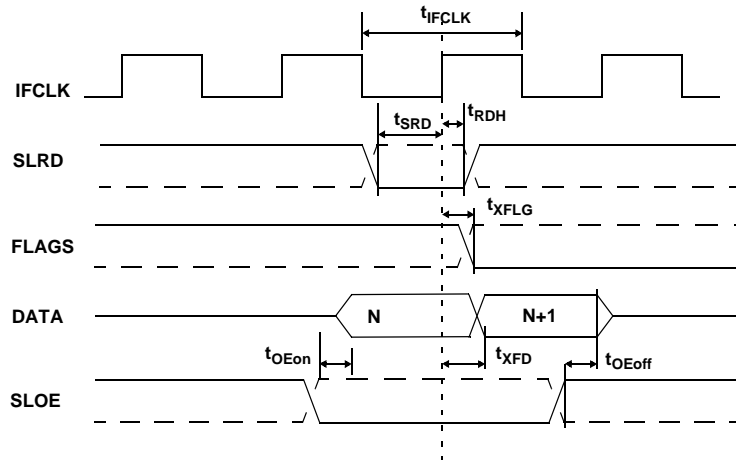
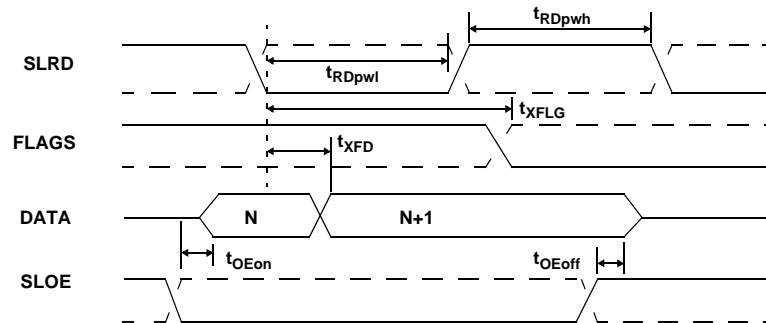


表 20. 从器件 FIFO 同步读取参数 (包含内部源 IFCLK) [25]

参数	说明	最小值	最大值	典型值		单位
				最小值	最大值	
$t_{IFCLK}$	IFCLK 周期	20.83	—	—	—	ns
$t_{SRD}$	从 SLRD 到时钟设置时间	18.7	—	—	—	ns
$t_{RDH}$	从时钟到 SLRD 保持时间	0	—	—	—	ns
$t_{OEon}$	从 SLOE 启用到 FIFO 数据生效	—	10.5	—	—	ns
$t_{OEoff}$	从 SLOE 关闭到 FIFO 数据保持	—	10.5	—	—	ns
$t_{XFLG}$	从时钟到 FLAGS 输出传输延迟	—	9.5	—	—	ns
$t_{XFD}$	从时钟到 FIFO 数据输出传输延迟	—	11	—	—	ns
$t_{IFCLKR}$	IFCLK 上升时间	—	—	—	900	ps
$t_{IFCLKF}$	IFCLK 下降时间	—	—	—	900	ps
$t_{IFCLKOD}$	IFCLK 输出占空比	—	—	49	51	%
$t_{IFCLKJ}$	IFCLK 抖动 (峰值到峰值)	—	—	—	300	ps

**表 21. 从器件 FIFO 同步读取参数 (包含内部源 IFCLK) [25]**

参数	说明	最小值	最大值	单位
$t_{IFCLK}$	IFCLK 周期	20.83	200	ns
$t_{SRD}$	从 SLRD 到时钟设置时间	12.7	—	ns
$t_{RDH}$	从时钟到 SLRD 保持时间	3.7	—	ns
$t_{OEon}$	从 SLOE 启用到 FIFO 数据生效	—	10.5	ns
$t_{OEoff}$	从 SLOE 关闭到 FIFO 数据保持	—	10.5	ns
$t_{XFLG}$	从时钟到 FLAGS 输出传输延迟	—	13.5	ns
$t_{XFD}$	从时钟到 FIFO 数据输出传输延迟	—	15	ns

**从器件 FIFO 异步读取**
**图 20. 从器件 FIFO 异步读取时序图 [24]**

**表 22. 从器件 FIFO 异步读取参数 [27]**

参数	说明	最小值	最大值	单位
$t_{RDpwl}$	SLRD 脉冲宽度偏低	50	—	ns
$t_{RDpwh}$	SLRD 脉冲宽度偏高	50	—	ns
$t_{XFLG}$	从 SLRD 到 FLAGS 输出传输延迟	—	70	ns
$t_{XFD}$	从 SLRD 到 FIFO 数据输出传输延迟	—	15	ns
$t_{OEon}$	从 SLOE 启用到 FIFO 数据生效	—	10.5	ns
$t_{OEoff}$	从 SLOE 关闭到 FIFO 数据保持	—	10.5	ns

**注**

27. 从器件 FIFO 异步参数值在 48 MHz 时使用内部 IFCLK 设置。

## 从器件 FIFO 同步写入

图 21. 从器件 FIFO 同步写入时序图<sup>[24]</sup>

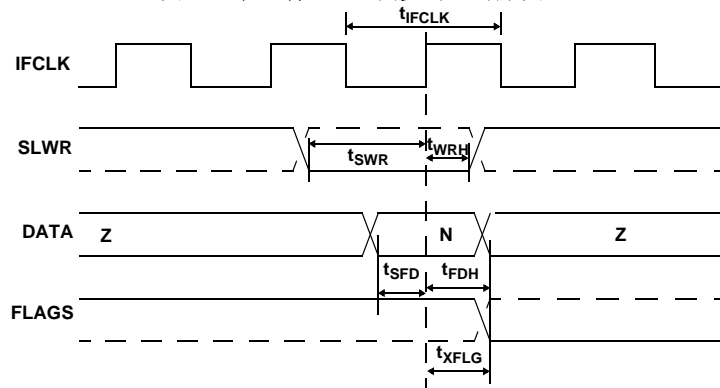


表 23. 从器件 FIFO 同步写入参数 (包含内部源 IFCLK)<sup>[25]</sup>

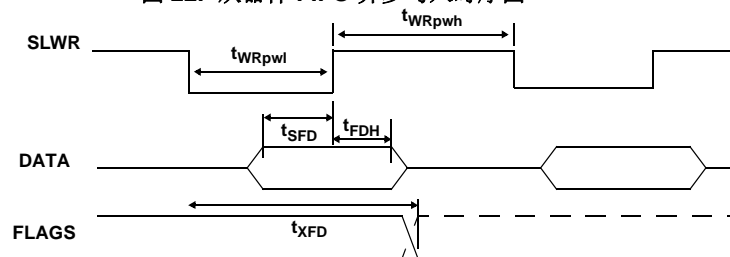
参数	说明	最小值	最大值	单位
$t_{IFCLK}$	IFCLK 周期	20.83	—	ns
$t_{SWR}$	从 SLWR 到时钟设置时间	10.4	—	ns
$t_{WRH}$	从时钟到 SLWR 保持时间	0	—	ns
$t_{SFD}$	从 FIFO 数据到时钟设置时间	9.2	—	ns
$t_{FDH}$	从时钟到 FIFO 数据保持时间	0	—	ns
$t_{XFLG}$	从时钟到 FLAGS 输出传输时间	—	9.5	ns

表 24. 从器件 FIFO 同步写入参数 (包含内部源 IFCLK)<sup>[25]</sup>

参数	说明	最小值	最大值	单位
$t_{IFCLK}$	IFCLK 周期	20.83	200	ns
$t_{SWR}$	从 SLWR 到时钟设置时间	12.1	—	ns
$t_{WRH}$	从时钟到 SLWR 保持时间	3.6	—	ns
$t_{SFD}$	从 FIFO 数据到时钟设置时间	3.2	—	ns
$t_{FDH}$	从时钟到 FIFO 数据保持时间	4.5	—	ns
$t_{XFLG}$	从时钟到 FLAGS 输出传输时间	—	13.5	ns

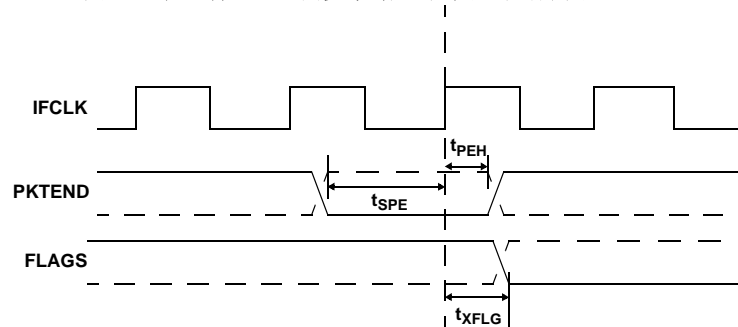
## 从器件 FIFO 异步写入

图 22. 从器件 FIFO 异步写入时序图<sup>[24]</sup>



**表 25. 从器件 FIFO 异步写入参数 ( 包含内部源 IFCLK) [27]**

参数	说明	最小值	最大值	单位
$t_{WRpwl}$	SLWR 脉冲偏低	50	–	ns
$t_{WRpwh}$	SLWR 脉冲偏高	70	–	ns
$t_{SFD}$	从 SLWR 到 FIFO DATA 数据设置时间	10	–	ns
$t_{FDH}$	从 FIFO 数据到 SLWR 的保持时间	10	–	ns
$t_{XFD}$	从 SLWR 到 FLAGS 输出的传输时间延迟	–	70	ns

**从器件 FIFO 同步数据包结束探针**
**图 23. 从器件 FIFO 同步数据包结束探针时序图 [24]**

**表 26. 从器件 FIFO 同步数据包结束探针参数与 IFCLK 内部源 [25]**

参数	说明	最小值	最大值	单位
$t_{IFCLK}$	IFCLK 周期	20.83	–	ns
$t_{SPE}$	PKTEND 至时钟设置时间	14.6	–	ns
$t_{PEH}$	从时钟到 PKTEND 的保持时间	0	–	ns
$t_{XFLG}$	从时钟到 FLAGS 输出传输延迟	–	9.5	ns

**表 27. 从器件 FIFO 同步数据包结束探针参数与 IFCLK 外部源 [25]**

参数	说明	最小值	最大值	单位
$t_{IFCLK}$	IFCLK 周期	20.83	200	ns
$t_{SPE}$	PKTEND 至时钟设置时间	8.6	–	ns
$t_{PEH}$	从时钟到 PKTEND 的保持时间	2.5	–	ns
$t_{XFLG}$	从时钟到 FLAGS 输出传输延迟	–	13.5	ns

从激活 PKTEND 引脚到激活 SLWR 的时间不满足任何特定时序要求。可以激活 PKTEND, 包含时钟脉冲打入 FIFO 或 FIFO 之后的最终数据值。设置时间  $t_{SPE}$  和保持时间  $t_{PEH}$  必须满足条件。

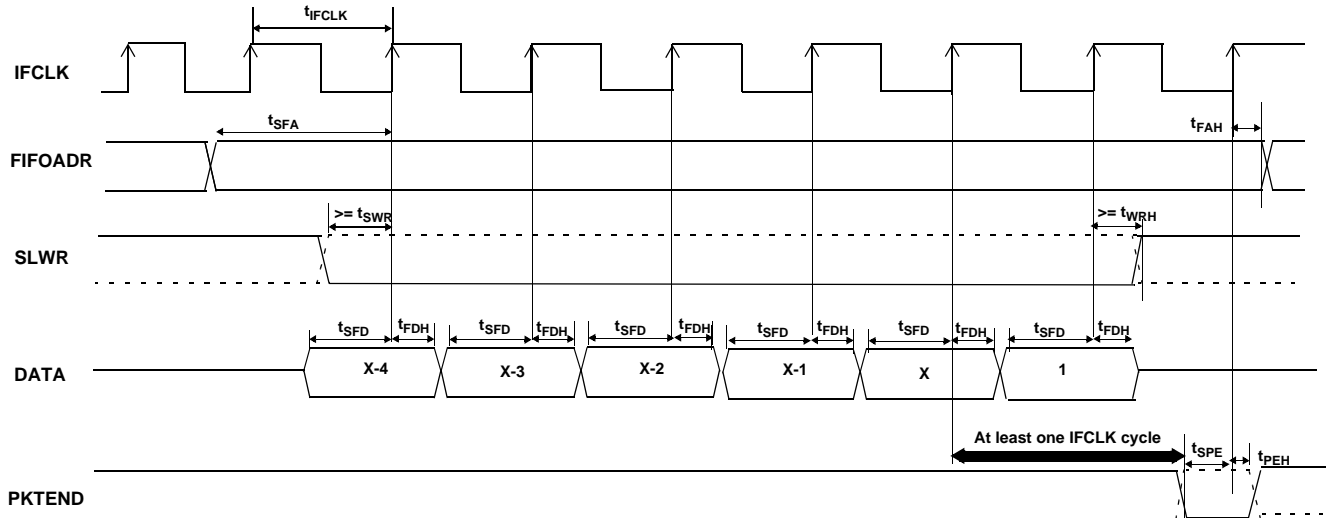
虽然激活 PKTEND 没有特定的时序要求, 但是, 在使用 PKTEND 来提交一个字节或文字数据包时, 需要注意特定边界的条件。当配置 FIFO 以实现在自动模式下运行时, 需要满足额外的时序要求, 即背对背发送两个数据包: 完整数据包 (FIFO 满足 AUTOINLEN 寄存器中设置的级别时全部定义为多字节数据包) 自动提交, 然后再使用 PKTEND 引脚手动提交一个字节或文字的简短数据包。在这种情况下, 用户必须确保在最小时钟周期内激活一个 PKTEND, 并且, 该周期须接在触发以下事件的上升沿

之后发生, 即最后字节或文字的时钟脉冲打入上一个自动提交的数据包中。图 24 X 是 AUTOINLEN 寄存器在 IN 端点配置为自动模式时所设置的值。

图 24 显示两个数据包的提交情况。当 FIFO 中的大量字节达到 X (在 AUTOINLEN 寄存器中设置的值) 时, 自动提交首个数据包, 使用 PKTEND 手动提交第二个包含一个字节 / 文字的简短数据包。

注: 在激活 PKTEND 与时钟打入上一个数据包的最后字节 (导致自动提交该数据包) 之间至少有一个 IFCLK 时钟时序。未遵守该时序会导致 FX2 发送一个字节或一个文字的简短数据包失败。

图 24. 从器件 FIFO 同步写入序列和时序图 [24]



从器件 FIFO 异步数据包结束探针

图 25. 从器件 FIFO 异步数据包结束探针时序图 [24]

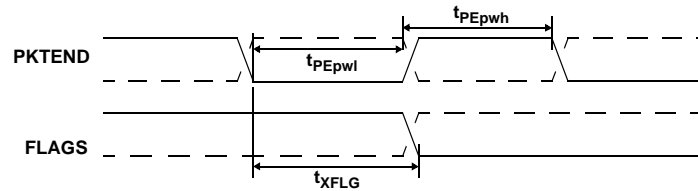


表 28. 从器件 FIFO 异步数据包结束探针参数 [27]

参数	说明	最小值	最大值	单位
$t_{PEpwl}$	PKTEND 脉冲宽度偏低	50	—	ns
$t_{PEpwh}$	PKTEND 脉冲宽度偏高	50	—	ns
$t_{XFLG}$	从 PKTEND 到 FLAGS 输出传输时间延迟	—	115	ns

从器件 FIFO 输出使能

图 26. 从器件 FIFO 输出使能时序图 [24]

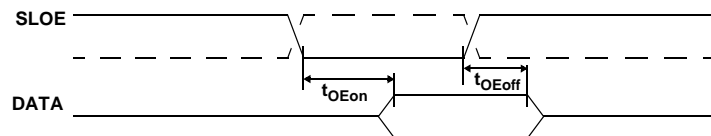


表 29. 从器件 FIFO 输出使能参数

参数	说明	最小值	最大值	单位
$t_{OEon}$	激活 SLOE 至 FIFO DATA 数据输出		10.5	ns
$t_{OEoff}$	取消激活 SLOE 至 FIFO DATA 数据保持		10.5	ns



### 从器件 FIFO 标志 / 数据寻址

图 27. 从器件 FIFO 标志 / 数据时序图寻址 [24]

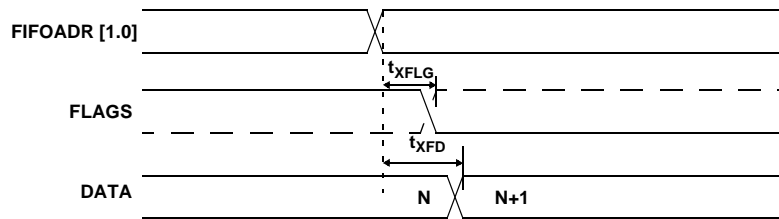


表 30. 从器件 FIFO 标志 / 数据参数寻址

参数	说明	最小值	最大值	单位
$t_{XFLG}$	从 FIFOADR[1:0] 至 FLAGS 输出传输时间延迟	–	10.7	ns
$t_{XFD}$	从 FIFOADR[1:0] 至 FIFODATA 输出的传输时间延迟	–	14.3	ns

### 从器件 FIFO 同步地址

图 28. 从器件 FIFO 同步地址时序图 [24]

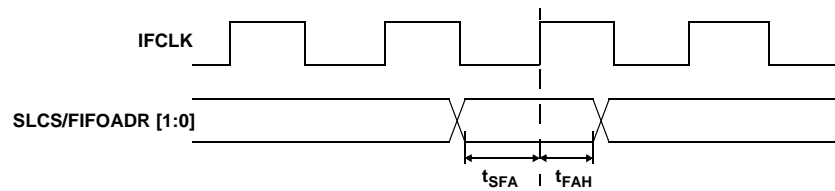


表 31. 从器件 FIFO 同步地址参数 [25]

参数	说明	最小值	最大值	单位
$t_{IFCLK}$	接口时钟周期	20.83	200	ns
$t_{SFA}$	从 FIFOADR[1:0] 到时钟设置时间	25	–	ns
$t_{FAH}$	从时钟到 FIFOADR[1:0] 的保持时间	10	–	ns

### 从器件 FIFO 异步地址

图 29. 从器件 FIFO 异步地址时序图 [24]

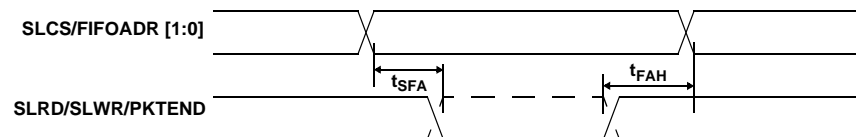


表 32. 从器件 FIFO 异步地址参数 [27]

参数	说明	最小值	最大值	单位
$t_{SFA}$	从 FIFOADR[1:0] 到 SLRD/SLWR/PKTEND 设置时间	10	–	ns
$t_{FAH}$	从 RD/WR/PKTEND 到 FIFOADR[1:0] 的保持时间	10	–	ns

## 序列图

单独的和突发的同步读取示例

图 30. 从器件 FIFO 同步读取序列和时序图 [24]

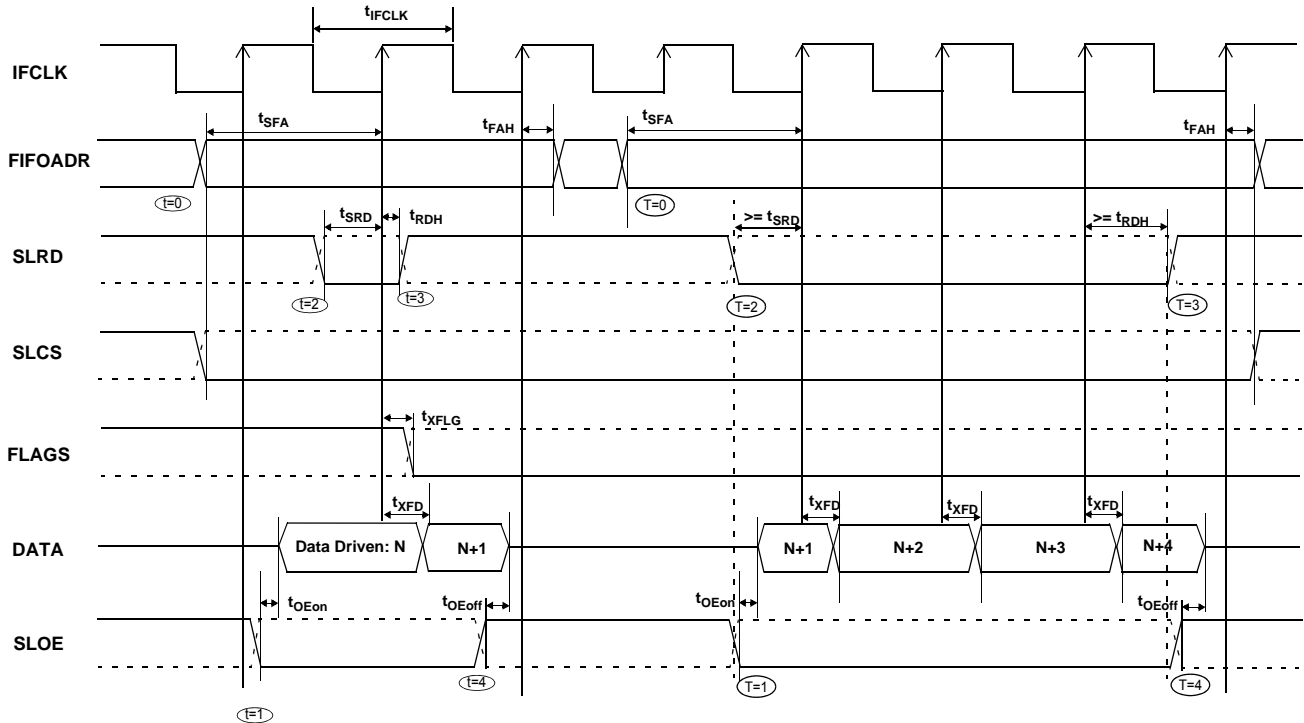
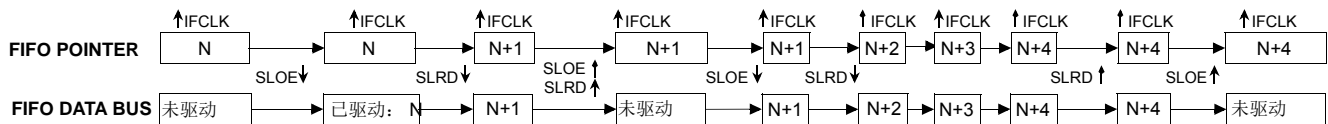


图 31. 从器件 FIFO 同步事件时序图



第 51 页的图 30 显示使用 IFCLK 作为同步化时钟来执行 FIFO 同步读取过程中从器件 FIFO 信号之间的时序关系。该框图阐明了突发读取之前的单个读取。

- At  $t = 0$  时, FIFO 地址是稳定的, 信号 SLCS 已激活 (SLCS 可以连接到某些低电平应用中)。注:  $t_{SFA}$  最低为 25 ns。这意味着 IFCLK 以 48 MHz 运行, FIFO 地址设置时间超出一个 IFCLK 周期。
- At  $t = 1$  时, SLOE 激活。SLOE 仅是输入使能, 其唯一功能是驱动数据总线。总线上驱动的数据是内部 FIFO 指针当前指向的数据。在此示例中, 该数据是 FIFO 中的首个数据值。注: 激活 SLOE 时, 在总线上驱动和预提取数据。
- At  $t = 2$  时, SLRD 激活。SLRD 必须满足  $t_{SRD}$  设置时间 (从激活 SLRD 信号到 IFCLK 上升沿之间的时间) 的要求, 并维持  $t_{RDH}$  最低保持时间 (从 IFCLK 沿到取消激活 SLRD 信号之间的时间)。如果使用 SLCS 信号, 则必须在激活 SLRD 前予以激活该信号 (要同时激活 SLCS 和 SLRD 信号以启动有效读取条件)。

- FIFO 指针在 IFCLK 上升沿时得以更新, 同时 SLRD 被激活。这会启动从新寻址位置到数据总线之间的数据传输。 $t_{XFD}$  传输延迟 (从 IFCLK 的上升沿测得) 后即可显示新数据值。N 是自 FIFO 读取的首个数据值。要在 FIFO 数据总线上保留数据, 还必须同时激活 SLOE。

显示突发读取事件的相同序列, 用  $T = 0 - 5$  来标记该序列。

**注：**对于突发模式而言, SLRD 和 SLOE 会留待读取的完整期间内激活。在突发模式下, 激活 SLOE 时, 由 FIFO 指针索引的数据位于数据总线上。在第一个读循环中, 更新时钟上升沿上的 FIFO 指针, 并增加 FIFO 指针, 使其指向地址  $N+1$ 。对于每个 IFCLK 序列上升沿而言, 当 SLRD 被激活时, FIFO 指针会递增, 而下一个数据值会置于数据总线上。

## 单个和突发的同步写入

图 32. 从器件 FIFO 同步写入序列和时序图 [24]

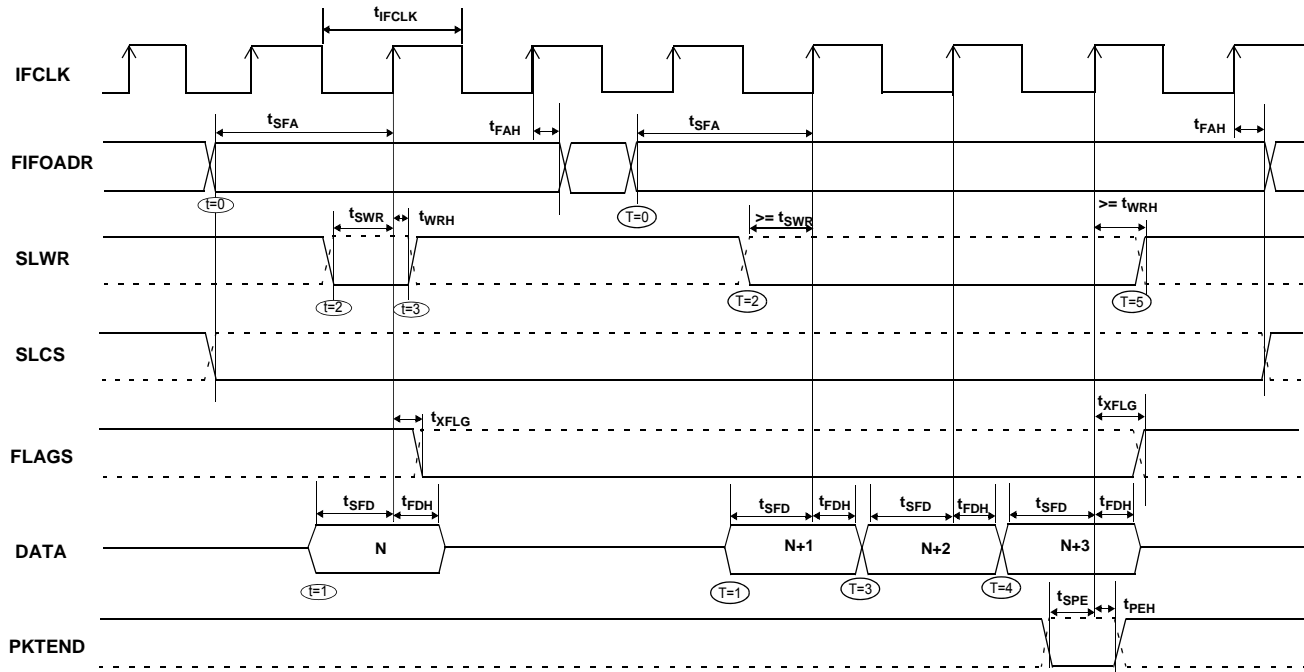


图 32 显示使用 IFCLK 作为同步化时钟来执行同步写入过程中从器件 FIFO 信号之间的时序关系。该图阐明了后接 3 个字节的突发写入的单个写入, 并使用 PKTEND 引脚将这 4 个字节作为一个简短数据包来提交。

- Att=0 时, FIFO 地址是稳定的, 并且, SLCS 已激活。(SLCS 可能连接至某些低电平的应用中) 注:  $t_{SFA}$  最低为 25 ns。这意味着 IFCLK 以 48 MHz 运行, FIFO 地址设置时间超出一个 IFCLK 周期。
- Att=1, 外部主控 / 外设必须在 IFCLK 升高沿之前, 将数据值输出到数据总线上, 最低设置时间为  $t_{SFD}$ 。
- Att=2 时, SLWR 激活。SLWR 必须满足  $t_{SWR}$  设置时间 (从激活 SLWR 信号到 IFCLK 上升沿之间的时间) 的要求, 并维持  $t_{WRH}$  最低保持时间 (从 IFCLK 边沿到取消激活 SLWR 信号之间的时间)。如果使用 SLCS 信号, 则必须与 SLWR 同时激活或在激活 SLWR 前予以激活该信号 (要同时激活 SLCS 和 SLWR 信号以启动有效写入条件)。
- 当 SLWR 被激活时, 数据写至 FIFO 和 IFCLK 的上升沿上, FIFO 指针递增。此外, FIFO 标志也在  $t_{XFLG}$  从时钟上升沿开始延迟之后更新。

显示突发写入事件的相同序列, 用时间指示符 T = 0 - 5 来标记该序列。

**注:** 对于突发模式, SLWR 和 SLCS 会留待写入所有所需数据值的完整期间内激活。在突发写入模式中, SLWR 被激活后, FIFO 数据总线上的数据会写入 IFCLK 的每个上升沿的 FIFO 中。FIFO 指针在 IFCLK 的每个上升沿上更新。在图 32 中, 将这 4 个字节写入 FIFO 后, 取消激活 SLWR。通过激活 PKTEND 信号, 将这个包含 4 个字节的简短数据包提交至主机。

由于激活 PKTEND 信号与激活 SLWR 信号相关联, 因此这不满足任何特定时序要求。可以激活 PKTEND, 包含最终数据值或后续值。设置时间  $t_{SPE}$  和保持时间  $t_{PEH}$  必须满足唯一条件。在图 32 的情况下, 提交的数据值数量中包含写入 FIFO 的最终值。在此示例中, 在 IFCLK 上升沿是打入数据值与 PKTEND 信号的时钟脉冲。此外, 在后续时钟周期中, 可以激活 PKTEND。FIFOADDR 线必须在 PKTEND 激活期间保持不变。

虽然激活 PKTEND 没有特定的时序要求, 但是, 在使用 PKTEND 来提交一个字节或文字数据包时, 需要注意特定边界的条件。当配置 FIFO 以实现在自动模式下运行时, 需要满足额外的时序要求, 即要发送两个数据包: 完整数据包 (FIFO 满足 AUTOINLEN 寄存器中设置的级别时全部定义为多字节) 自动提交, 然后再使用 PKTEND 引脚手动提交一个字节或文字的简短数据包。

在此情况下, 外部主控必须确保在最小时钟周期内激活 PKTEND, 该周期要在触发以下条件的上升沿之后, 即触发最终字节或文字的时钟脉冲打入上一个自动提交的数据包 (该数据包的字节数等于 AUTOINLEN 寄存器中设置的字节数)。有关此时序的更多信息, 请参考第 49 页的图 24。

单个或突发异步读取的序列包

图 33. 从器件 FIFO 异步读取序列和时序图 [24]

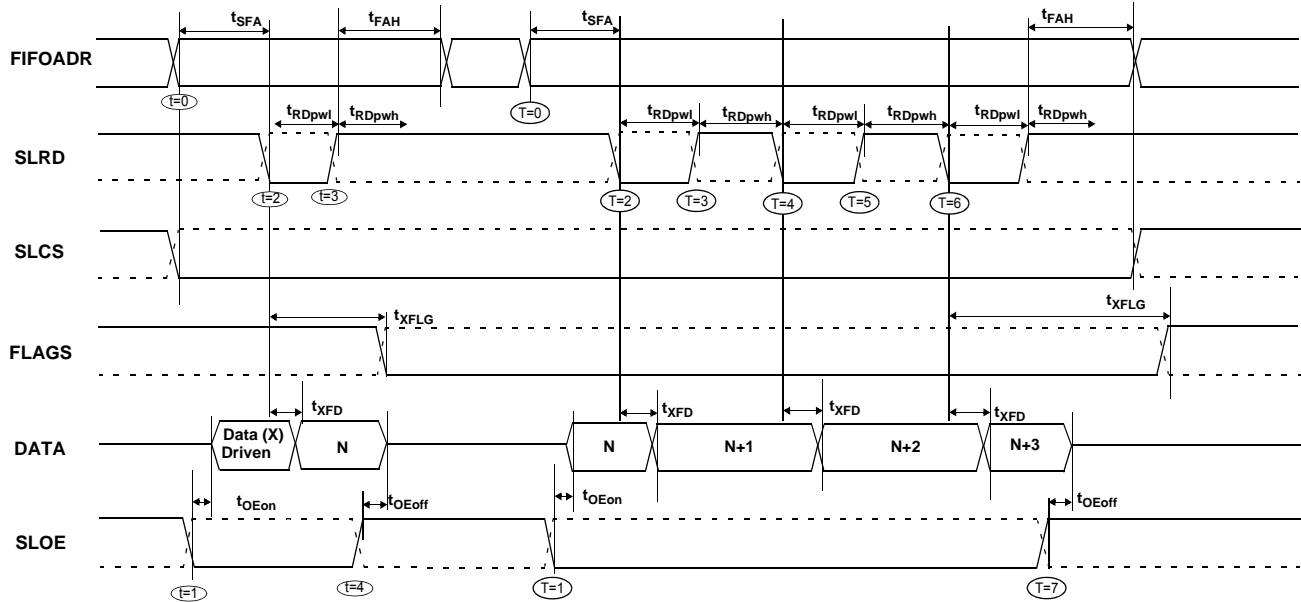


图 34. 从器件 FIFO 异步读取事件序列图

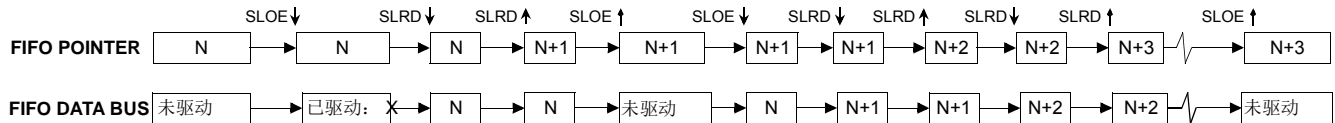


图 33 显示 FIFO 异步读取过程中从器件 FIFO 信号之间的时序关系。该图显示突发读取之前的单个读取。

- At  $t = 0$  时, FIFO 地址是稳定的, 并且, SLCS 信号已激活。
- At  $t = 1$  时, SLOE 激活。这使得数据总线被驱动。在总线上驱动的数据是上一个数据, 该数据位于上一个读循环的 FIFO 中。
- At  $t = 2$  时, SLRD 激活。SLRD 必须满足  $t_{RDpwl}$  的最低活动脉冲和  $t_{RDpwh}$  的最低去活脉冲宽度的要求。如果使用 SLCS 信号, 那么, 在激活 SLRD 前必须激活 SLCS (要同时激活 SLCS 和 SLRD 信号以启动有效读取条件)。

- 激活 SLRD 后被驱动的数据是源自 FIFO 的更新数据。该数据将在 SLRD 激活沿的  $t_{XFD}$  产生传输延迟后生效。在图 33 中, 数据 N 是自 FIFO 读取的首个有效数据。对于读循环 (SLRD 被激活) 数据总线上出现的数据而言, SLOE 必须处于激活状态。SLRD 和 SLOE 也可绑定在一起。

显示突发读取事件的相同序列, 用  $T = 0 - 5$  来标记。

注: 在突发读取模式中, 激活 SLOE 期间, 数据总线处于驱动状态, 并输出上一个数据。SLRD 激活后, 在数据总线上驱动 FIFO 的数据 (SLOE 也必须被激活), 然后, FIFO 指针递增。

单个或突发异步写入的序列图

图 35. 从器件 FIFO 异步写入序列和时序图 [24]

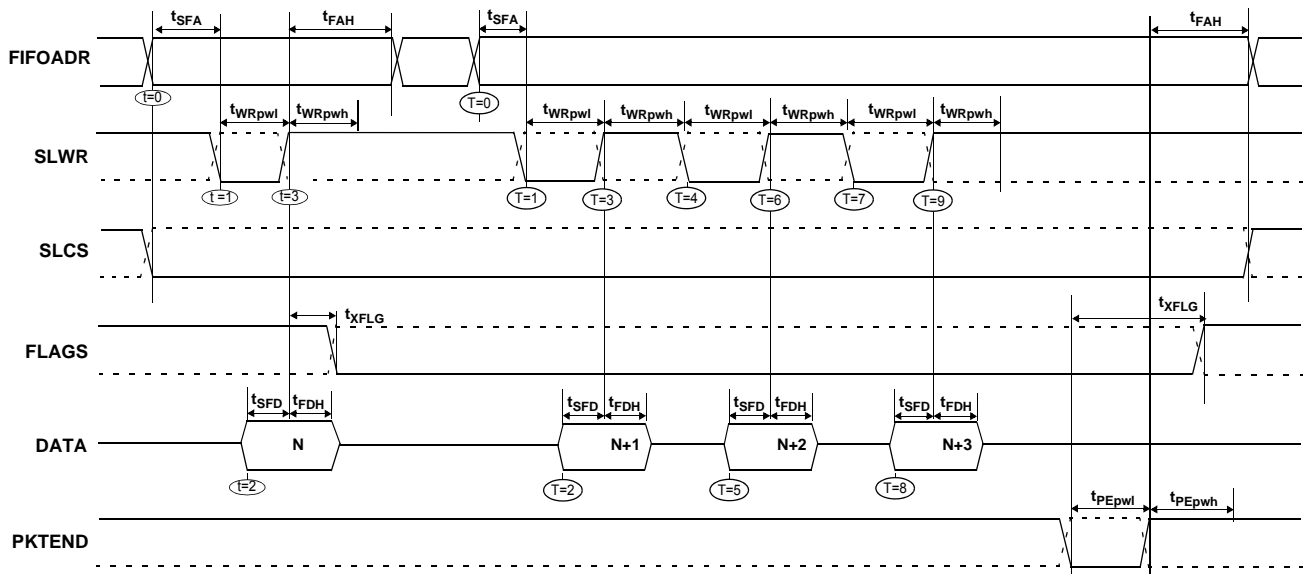


图 35 显示异步模式下从器件 FIFO 写入的时序关系。该图阐明了后接 3 个字节突发写入的单个写入，并使用 PKTEND 提交这 4 个字节简短数据包。

- At  $t = 0$  时，FIFO 地址被应用，确保其满足  $t_{SFA}$  设置时间的要求。如果使用 SLCS，也要激活 SLCS (SLCS 可以连接至某些低电平的应用中)。
- At  $t = 1$  时，SLWR 被激活。SLWR 必须满足  $t_{WRpwl}$  的最低活动脉冲和  $t_{WRpwh}$  的最低去活脉冲宽度的要求。如果使用 SLCS，则必须与 SLWR 同时激活，或先于 SLWR 激活。
- At  $t = 2$  时，在取消激活 SLWR 边沿之前，数据必须显示在  $t_{SFD}$  总线上。
- At  $t = 3$  时，解除激活 SLWR 会使数据从数据总线写入 FIFO，然后 FIFO 指针递增。此外，FIFO 标志也在  $t_{XFLG}$  从 SLWR 解除激活沿开始之后更新。

显示突发写入事件的相同序列，用时间标志  $T = 0 - 5$  来表示该序列。

**注：**在突发写入模式中，SLWR 解除激活后，该数据被写入 FIFO，然后 FIFO 指针递增到 FIFO 中的下一个字节。FIFO 指针后递增。

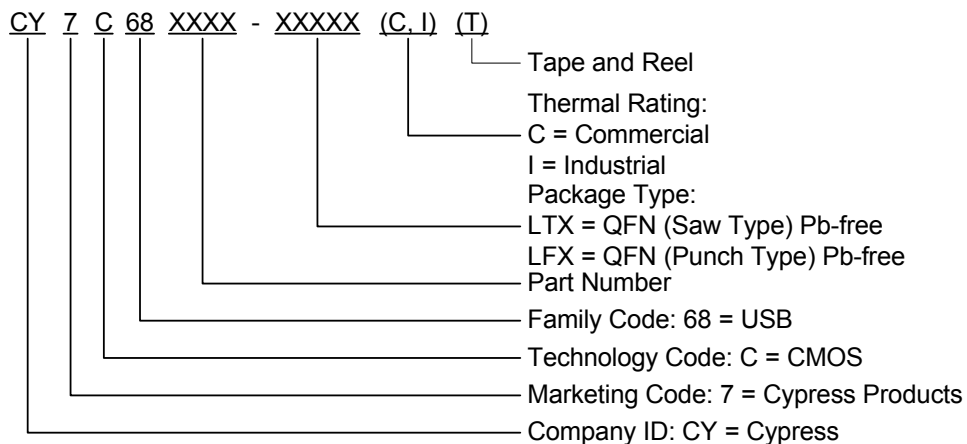
在图 35 中，当这 4 个字节写入 FIFO 并解除激活 SLWR 后，可以使用 PKTEND 将这 4 字节的简短数据包提交至主机。设计外部器件，以便无法同时激活 SLWR 和 PKTEND 信号。该器件的设计原理应是在解除激活 SLWR 之后再行激活 PKTEND，并满足最低解除激活的脉冲宽度。FIFOADDR 线必须在 PKTEND 激活期间保持不变。

## 订购信息

表 33. 订购信息

订购代码	封装类型	RAM 大小	可编程 I/O 编号	8051 地址 / 数据总线	串行调试 [28]
适用于电池供电应用					
CY7C68014A-128AXC	128 TQFP – 无铅	16 K	40	16/8 位	Y
CY7C68014A-100AXC	100 TQFP – 无铅	16 K	40	–	Y
CY7C68014A-56PVXC	56 SSOP – 无铅	16 K	24	–	N
CY7C68014A-56LTXC	56 QFN - 无铅	16 K	24	–	N
CY7C68016A-56LTXC	56 QFN - 无铅	16 K	26	–	N
CY7C68016A-56LTXCT	56 QFN - 无铅	16 K	26	–	N
适用于非电池供电应用					
CY7C68013A-128AXC	128 TQFP – 无铅	16 K	40	16/8 位	Y
CY7C68013A-128AXI	128 TQFP – 无铅 ( 工业级 )	16 K	40	16/8 位	Y
CY7C68013A-100AXC	100 TQFP – 无铅	16 K	40	–	Y
CY7C68013A-100AXI	100 TQFP – 无铅 ( 工业级 )	16 K	40	–	Y
CY7C68013A-56PVXC	56 SSOP – 无铅	16 K	24	–	N
CY7C68013A-56PVXCT	56 SSOP – 无铅	16 K	24	–	N
CY7C68013A-56PVXI	56 SSOP – 无铅 ( 工业级 )	16 K	24	–	N
CY7C68013A-56BAXC	56 VFBGA – 无铅	16 K	24	–	N
CY7C68013A-56BAXCT	56 VFBGA – 无铅	16 K	24	–	N
CY7C68013A-56LTXC	56 QFN – 无铅	16 K	24	–	N
CY7C68013A-56LTXCT	56 QFN – 无铅	16 K	24	–	N
CY7C68013A-56LTXI	56 QFN – 无铅 ( 工业级 )	16 K	24	–	N
CY7C68015A-56LTXC	56 QFN – 无铅	16 K	26	–	N
开发工具包					
CY3684	EZ-USB FX2LP 开发套件				
参考设计套件					
CY4611B	USB 2.0 至 ATA/ATAPI 参考设计 ( 使用 EZ-USB FX2LP)				

## 订购代码定义



### 注

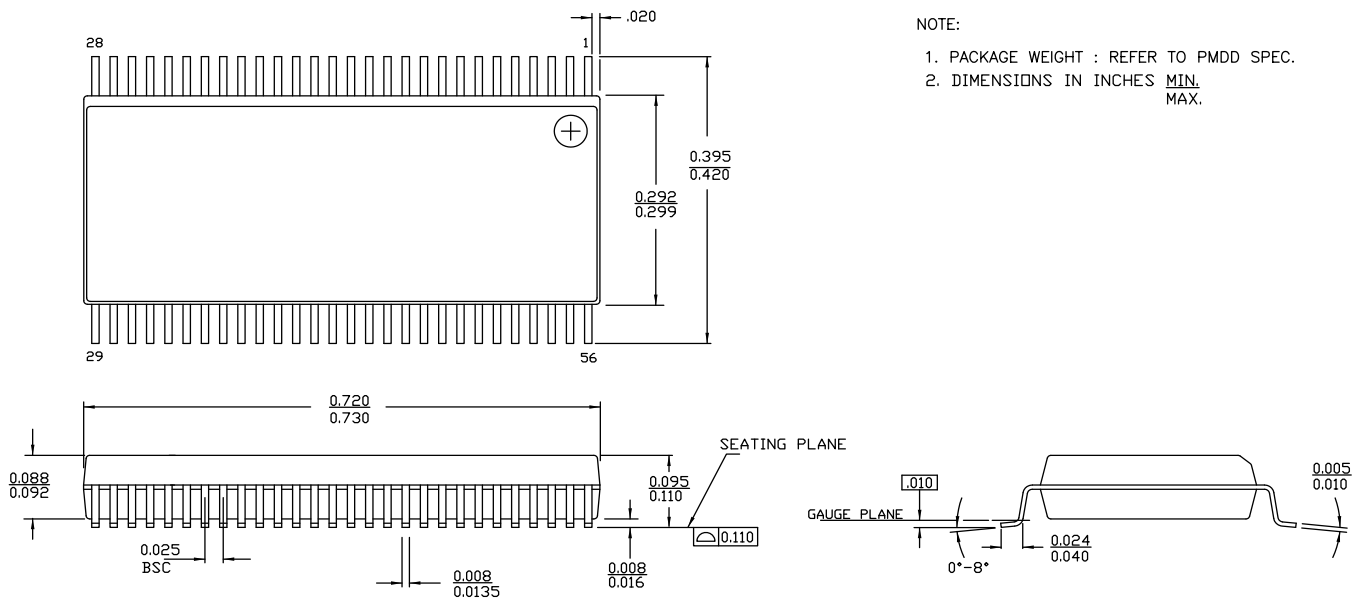
28. 由于 UART 不适用于 CY7C68013A 56 引脚封装, 因此使用 Keil 监测器调试的串行端口无法实现。

## 封装图

FX2LP 可用于 5 个封装:

- 56 引脚 SSOP
- 56 引脚 QFN
- 100 引脚 TQFP
- 128 引脚 TQFP
- 56 脚 VFBGA

图 36. 56 引脚紧缩小外形封装 O56 (51-85062)



51-85062 \*F



图 37. 56 引脚 QFN 8 × 8 mm Sawn 版本 (001-53450)

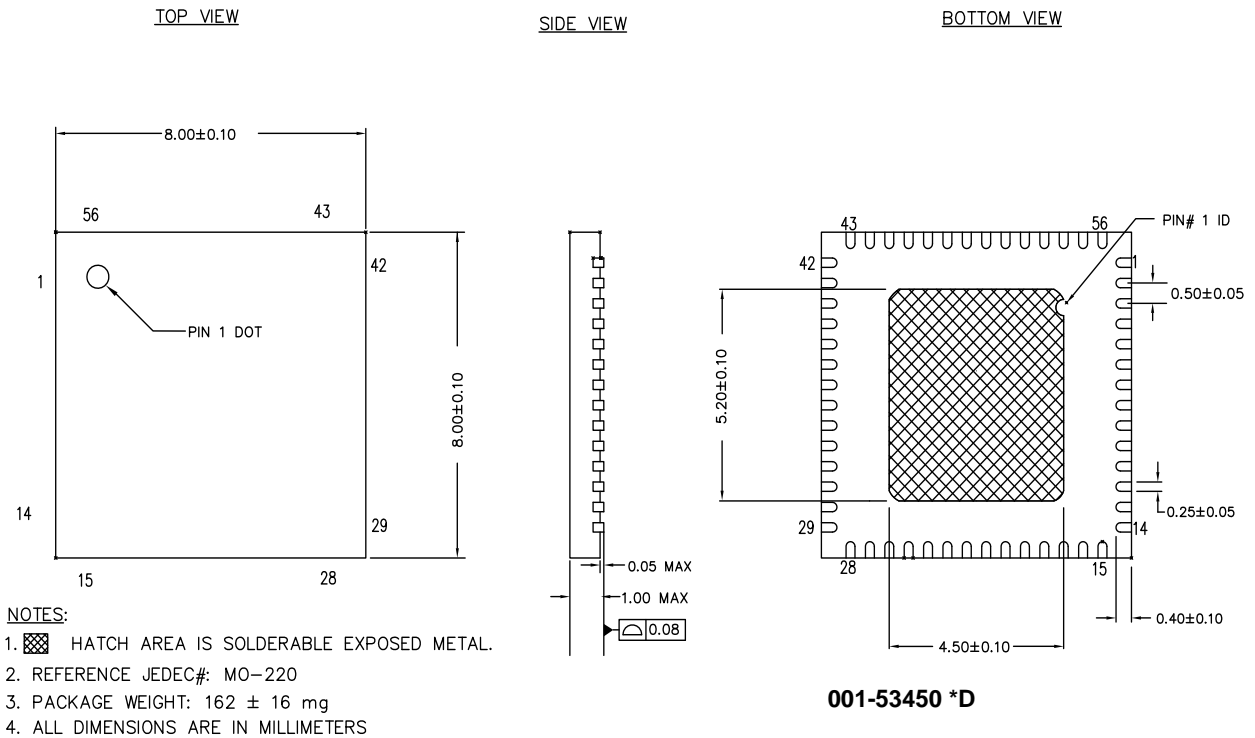
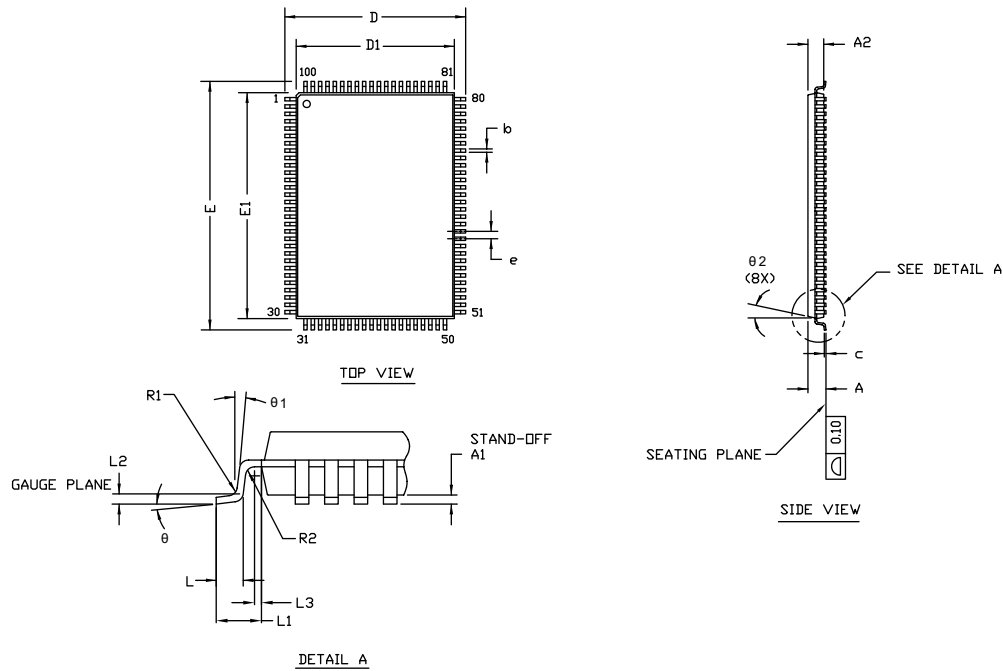


图 38. 100 引脚纤薄四方扁平塑料封装 (14 × 20 × 1.4 mm) A100RA (51-85050)



SYMBOL	DIMENSIONS		
	MIN.	NOM.	MAX.
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
D	15.80	16.00	16.20
D1	13.90	14.00	14.10
E	21.80	22.00	22.20
E1	19.90	20.00	20.10
R1	0.08	—	0.20
R2	0.08	—	0.20
θ	0°	—	7°
θ1	0°	—	—
θ2	11°	12°	13°
c	—	—	0.20
b	0.22	0.30	0.38
L	0.45	0.60	0.75
L1	1.00 REF		
L2	0.25 BSC		
L3	0.20	—	—
e	0.65 TYP		

NOTE:

1. ALL DIMENSIONS ARE IN MILLIMETERS.
2. BODY LENGTH DIMENSION DOES NOT INCLUDE MOLD PROTRUSION/END FLASH. MOLD PROTRUSION/END FLASH SHALL NOT EXCEED 0.0098 in (0.25 mm) PER SIDE. BODY LENGTH DIMENSIONS ARE MAX PLASTIC BODY SIZE INCLUDING MOLD MISMATCH.
3. JEDEC SPECIFICATION NO. REF: MS-026.

51-85050 \*G

Technical drawing of a 128-pin Quad Flat Pack (QFP) package. The drawing includes three views: a top view, a side view, and a detail view (DETAIL A).

**Top View Dimensions:**

- Overall width:  $16.00 \pm 0.20$
- Pin pitch (width between pins):  $14.00 \pm 0.10$
- Overall height:  $22.00 \pm 0.20$
- Pin pitch (height between pins):  $20.00 \pm 0.10$
- Pin 1 indicator: A circle with the number 1.
- Stand-off dimensions:  $0.22 \pm 0.05$  and  $0.50$  TYP.

**Side View Dimensions:**

- Package thickness:  $1.40 \pm 0.05$
- Lead angle:  $12^\circ \pm 1^\circ$  (BX)
- Lead thickness:  $0.20$  MAX.
- Seating plane to lead tip distance:  $1.60$  MAX.
- Seating plane: Indicated by a horizontal line.
- Lead thickness at seating plane:  $0.08$

**DETAIL A (Lead Profile):**

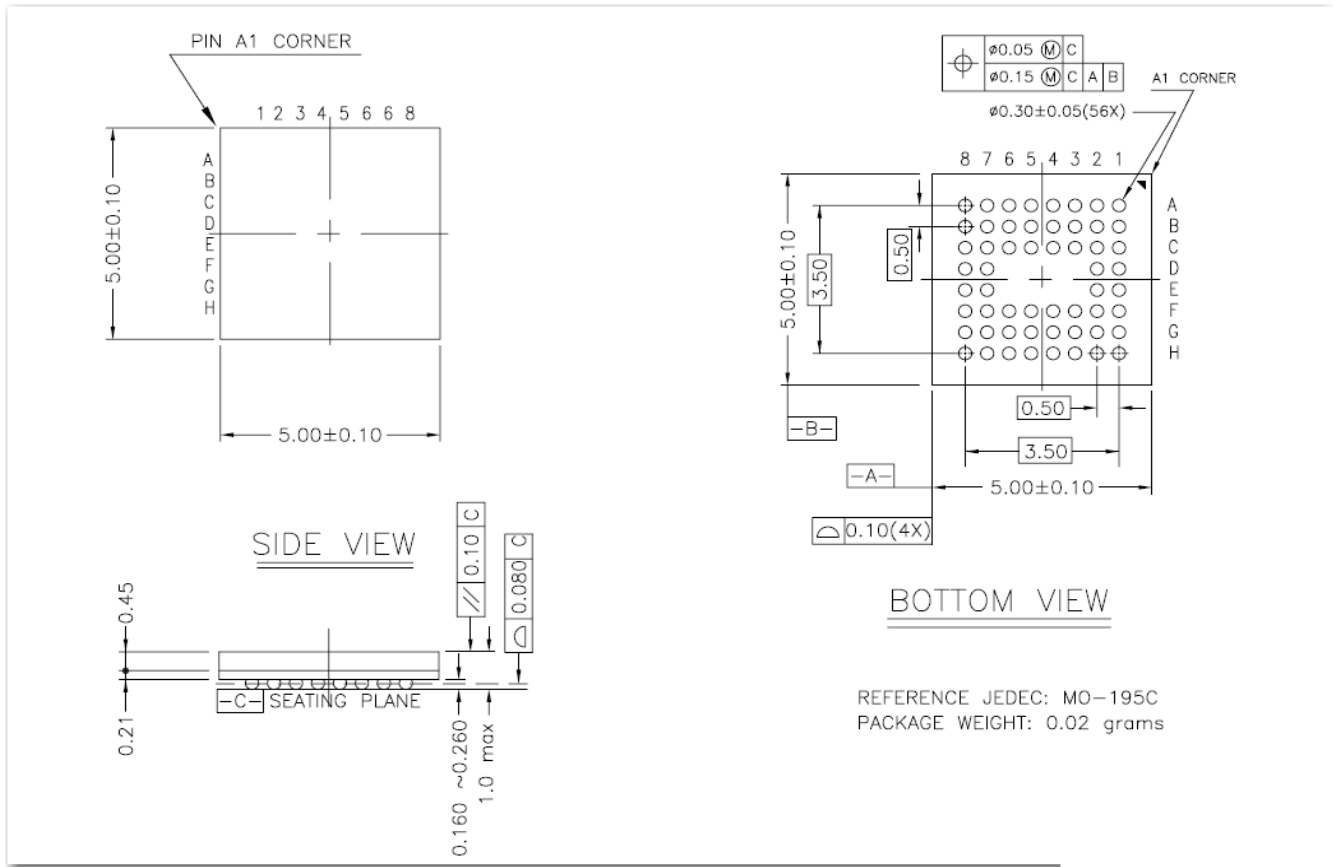
- Lead thickness at gauge plane:  $0.25$
- Gauge plane: Indicated by a vertical line.
- Lead thickness at gauge plane:  $0.60 \pm 0.15$
- Lead thickness at gauge plane:  $0.08$  MIN.  $0.20$  MAX.
- Lead thickness at gauge plane:  $0^\circ$  MIN.
- Lead thickness at gauge plane:  $0^\circ - 7^\circ$
- Stand-off dimensions:  $0.05$  MIN.  $0.15$  MAX.
- Lead thickness at gauge plane:  $R 0.08$  MIN.  $0.20$  MAX.

**NOTE:**

- JEDEC STD REF MS-026
- BODY LENGTH DIMENSION DOES NOT INCLUDE MOLD PROTRUSION/END FLASH  
MOLD PROTRUSION/END FLASH SHALL NOT EXCEED  $0.0098$  in ( $0.25$  mm) PER SIDE  
BODY LENGTH DIMENSIONS ARE MAX PLASTIC BODY SIZE INCLUDING MOLD MISMATCH
- DIMENSIONS IN MILLIMETERS

page 59 of 66

图 40. 56 引脚 VFBGA (5 × 5 × 1.0 mm) 0.50 间距, 0.30 脚 BZ56 (001-03901)



001-03901 \*F

## PCB 布局建议

采纳这些建议以确保可靠的高性能操作：<sup>[29]</sup>

- 4 层阻抗控制电路板需要用来维持信号质量。
- 指定阻抗目标 ( 询问您的电路板供应商能实现什么 )。
- 要控制阻抗, 保持走线宽度和走线空间。
- 最小化端点, 以便最小化反射信号。
- USB 连接器外壳与信号接地之间的连接必须靠近 USB 连接器。
- VBus 上的旁路和反激式电容靠近连接器, 遵从建议设计。
- DPLUS 和 DMINUS 走线长度差应保持在 2 mm 以内, 最适宜的长度应为 20 - 30 mm。
- 保持 DPLUS 和 DMINUS 下面的接地层坚固。不允许开裂这两条走线下面的接地层。
- 请勿在 DPLUS 或 DMINUS 走线路由上放置过孔。
- DPLUS 和 DMINUS 走线要远离其他所有的信号走线, 距离至少为 10 mm。

### 注

29. 建议源: *EZ-USB FX2™ PCB 设计建议*, <http://www.cypress.com> and *高速 USB 平台设计指南*, [http://www.usb.org/developers/docs/hs\\_usb\\_pdg\\_r1\\_0.pdf](http://www.usb.org/developers/docs/hs_usb_pdg_r1_0.pdf)。

## 四方扁平封装无引脚 (QFN) 封装设计说明

通过将封装下表面的引脚焊接到印刷电路板 (PCB) 上, 可以完成 PCB 局部电气接触。因此, 要特别注意封装下面的中心传输区域, 以便为电路板提供良好的热量结合。在封装下面, 将 PCB 中填充的铜 (Cu) 设计为热焊盘。在封装底侧, 中心热量从 FX2LP 传输到器件的金属焊盘上。中心热量从此处传到热焊盘上的 PCB。然后通过  $5 \times 5$  的过孔阵列从热焊盘导入 PCB 内部接地层。过孔镀压穿透 PCB 成孔, 直径为 13 mil。QFN 金属中心焊盘必须焊接到 PCB 的热焊盘上。焊接掩模置于电路板顶侧每个过孔上面, 从而抵制焊接流入过孔。此外, 顶侧掩模还在回流焊过程中最大程度地减少放气率。

有关此封装设计的更多信息, 请参考应用笔记“Application Notes for Surface Mount Assembly of Amkor's MicroLeadFrame (MLF)

Packages”。这可以在 Amkor 网站 <http://www.amkor.com> 上找到。

该应用笔记提供了有关电路板安装指南、流焊和返工流程等的详细信息。

图 41 显示封装底部的横截面积。该横截面仅是一个过孔的横截面积。设计焊膏模板, 以允许至少 50% 焊接面积。焊膏模板的厚度应为 5 mil。使用免清洗焊膏类型 3 安装该部件。在回流过程中建议进行氮气吹洗。

图 42 是焊接掩模模式曲线图, 图 43 显示该装配的 X 射线图 (阴影部分表示焊接区)。

图 41. QFN 封装底部区域的横截面

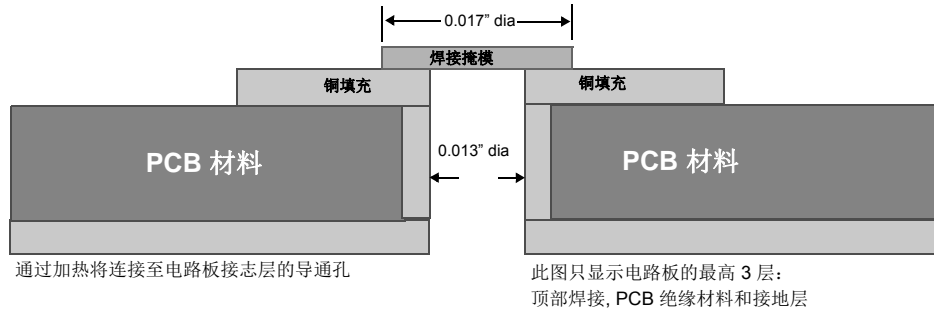


图 42. 焊接掩模曲线图 (白色区域)

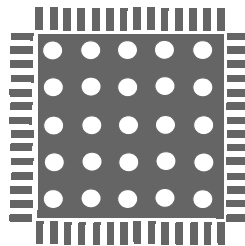
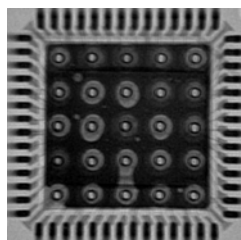


图 43. 封装的 X 射线图像



## 缩略语

### 文档规范

缩略语	说明
ASIC	应用程序特定的集成电路
ATA	先进技术附件
DID	器件标识符
DSL	数据服务线路
DSP	数据信号处理器
ECC	错误更正代码
EEPROM	电可擦可编程只读存储器
EPP	增强型并行端口
FIFO	先进先出
GPIF	通用编程接口
GPIO	通用输入 / 输出
I/O	输入 / 输出
LAN	局域网
MPEG	移动图像专家组
PCMCIA	个人计算机内存卡国际联盟
PID	产品标识符
PLL	锁相环
QFN	四方扁平无引脚器件
随机访问内存	随机存取存储器
SIE	串行接口引擎
SOF	帧起始
SSOP	超小型封装
TQFP	纤薄四方扁平封装
USARTS	通用串行异步接收器 / 发射器
USB	通用串行总线
UTOPIA	通用测试和操作物理层接口
VFBGA	超细间距 BGA
VID	供应商标识符

## 文档规范

### 测量单位

符号	测量单位
kHz	千赫兹
mA	毫安
Mbps	每秒兆位数
MB	每秒兆字节
MHz	兆赫兹
uA	微安
V	伏特



## 勘误表

本节描述了 EZ-USB® FX2LP™ CY7C68013A/14A/15A/16A Rev. B 版本勘误表。具体包括了错误触发条件，影响范围，可用的处理方法，以及芯片版本可用性。

如果您有任何问题，请联络当地赛普拉斯销售代表。

### Part Numbers Affected

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

### CY7C68013A/14A/15A/16A 资质状态:

生产中

### CY7C68013A/14A/15A/16A 勘误总结:

下表界定了可以使用的 CY7C68013A/14A/15A/16A 家族器件勘误。“X”表示错误属于对应的器件。

Items	CY7C68013A/14A/15A/16A	Silicon Revision	Fix Status
[1.]. Empty FLAG Assertion-空标志激活	X	B	No silicon fix planned currently. Use the workaround.

#### 1. Empty FLAG Assertion- 空标志激活

在从器件 FIFO 异步字宽模式，如果一个单字宽数据从主 USB 向 EP2 传送，并在第一次处理中配置为 OUT 端点，空标志表现错误。如果第一次传送数据多于一个字，错误不会发生。

#### 2. 受影响的参数: 不可用

#### 3. 触发条件

在从器件 FIFO 异步字宽模式，固件启动和初始化后，EP2 OUT 端点空标志表示状态为空。当 EP2 受到数据，状态改为“非空”。无论怎样，当向 EP2 传送的数据为单字，激活 SLRD 并且 FIFOADR 指向任意其他端点，即使 EP2 有一个字的数据（或者是未接触）状态从“非空”变为“空”。这种情况仅在第一次处理为单字且第一次处理后未跟随多字包的状态下发生。

#### 4. 影响范围

外部接口不能看到 EP2 OUT 端点有可用数据，并且结束等待读入数据。

#### 5. 处理办法

可以采用其中一种下列处理办法:

- . 固件初始化后，在主机向 EP2 传送数据之前或之后，向 SLWR 管脚发送一个脉冲，并且 FIFOADR 指向 EP2 以外的其他端点
- . 设置第一个传送数据的长度为大于一个字。
- . EP2 优先从主器件的多个 OUT 端点读取，并向 EP2 写入一个单字数据。
- . 如果可能，在从其他 OUT 节点（不是 EP2）读取之前，向一个 IN 端点写入。

#### 6. 处理办法

当前没有硅片修复计划，请应用提供的处理办法。

## 文档修订记录页

文档标题: CY7C68013A, CY7C68014A, CY7C68015A, CY7C68016A, EZ-USB <sup>®</sup> FX2LP <sup>™</sup> USB 微控制器高速 USB 外设控制器 文档编号: 001-78668				
修订版本	ECN 编号	变更人	提交日期	变更说明
**	3598306	VCS	04/24/2012	新增了数据表
*A	4753787	JNAN	05/04/2015	No Change Sunset ECN
*B	6173563	WHUI	05/14/2018	本文档版本号为 Rev*B, 译自英文版 38-08032 Rev.AA。

## 销售、解决方案和法律信息

### 全球销售和 design 支持

赛普拉斯公司拥有一个由办事处、解决方案中心、原厂代表和经销商组成的全球性网络。如欲查找离您最近的办事处，请访问 [赛普拉斯所在地](#)。

### 产品

Arm® Cortex® 微控制器	<a href="http://cypress.com/arm">cypress.com/arm</a>
汽车级产品	<a href="http://cypress.com/automotive">cypress.com/automotive</a>
时钟与缓冲器	<a href="http://cypress.com/clocks">cypress.com/clocks</a>
接口	<a href="http://cypress.com/interface">cypress.com/interface</a>
物联网	<a href="http://cypress.com/iot">cypress.com/iot</a>
存储器	<a href="http://cypress.com/memory">cypress.com/memory</a>
微控制器	<a href="http://cypress.com/mcu">cypress.com/mcu</a>
PSoC	<a href="http://cypress.com/psoc">cypress.com/psoc</a>
电源管理 IC	<a href="http://cypress.com/pmic">cypress.com/pmic</a>
触摸感应	<a href="http://cypress.com/touch">cypress.com/touch</a>
USB 控制器	<a href="http://cypress.com/usb">cypress.com/usb</a>
无线连接	<a href="http://cypress.com/wireless">cypress.com/wireless</a>

### PSoC® 解决方案

[PSoC 1](#) | [PSoC 3](#) | [PSoC 4](#) | [PSoC 5LP](#) | [PSoC 6 MCU](#)

### 赛普拉斯开发者社区

[社区](#) | [项目](#) | [视频](#) | [博客](#) | [培训](#) | [组件](#)

### 技术支持

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

© 赛普拉斯半导体公司，2003-2018 年。本文件是赛普拉斯半导体公司及其子公司，包括 Spansion LLC（“赛普拉斯”）的财产。本文件，包括其包含或引用的任何软件或固件（“软件”），根据全球范围内的知识产权法律以及美国与其他国家签署条约由赛普拉斯所有。除非在本款中另有明确规定，赛普拉斯保留在该等法律和条约下的所有权利，且未就其专利、版权、商标或其他知识产权授予任何许可。如果软件并不附随有一份许可协议且贵方未以其他方式与赛普拉斯签署关于使用软件的书面协议，赛普拉斯特此授予贵方属人性质的、非独家且不可转让的如下许可（无再许可）（1）在赛普拉斯特软件著作权项下的下列许可（一）对以源代码形式提供的软件，仅出于在赛普拉斯硬件产品上使用之目的且仅在贵方集团内部修改和复制软件，和（二）仅限于在有关赛普拉斯硬件产品上使用之目的将软件以二进制代码形式的向外部最终用户提供（无论直接提供或通过经销商和分销商间接提供），和（2）在被软件（由赛普拉斯公司提供，且未经修改）侵犯的赛普拉斯专利的权利主张项下，仅出于在赛普拉斯硬件产品上使用之目的制造、使用、提供和进口软件的许可。禁止对软件的任何其他使用、复制、修改、翻译或汇编。

在适用法律允许的限度内，赛普拉斯未对本文件或任何软件作出任何明示或暗示的担保，包括但不限于关于适销性和特定用途的默示保证。没有任何电子设备是绝对安全的。因此，尽管赛普拉斯在其硬件和软件产品中采取了必要的安全措施，但是赛普拉斯并不承担任何由于使用赛普拉斯产品而引起的安全问题及安全漏洞的责任，例如未经授权的使用或使用赛普拉斯产品。此外，本材料中所介绍的赛普拉斯产品有可能存在设计缺陷或设计错误，从而导致产品的性能与公布的规格不一致。（如果发现此类问题，赛普拉斯会提供勘误表）赛普拉斯保留更改本文件的权利，届时将不另行通知。在适用法律允许的限度内，赛普拉斯不对因应用或使用本文件所述任何产品或电路引起的任何后果负责。本文件，包括任何样本设计信息或程序代码信息，仅为供参考之目的提供。文件使用人应负责正确设计、计划和测试信息应用和由此生产的任何产品的功能和安全性。赛普拉斯产品不应被设计为、设定为或授权用作武器操作、武器系统、核设施、生命支持设备或系统、其他医疗设备或系统（包括急救设备和手术植入物）、污染控制或有害物质管理系统中的关键部件，或产品植入之设备或系统故障可能导致人身伤害、死亡或财产损失其他用途（“非预期用途”）。关键部件指，若该部件发生故障，经合理预期会导致设备或系统故障或会影响设备或系统安全性和有效性的部件。针对由赛普拉斯产品非预期用途产生或相关的任何主张、费用、损失和其他责任，赛普拉斯不承担全部或部分责任且贵方不应追究赛普拉斯之责任。贵方应赔偿赛普拉斯因赛普拉斯产品任何非预期用途产生或相关的所有索赔、费用、损失和其他责任，包括因人身伤害或死亡引起的主张，并使之免受损失。

赛普拉斯、赛普拉斯徽标、Spansion、Spansion 徽标，及上述项目的组合，WICED，及 PSoC、CapSense、EZ-USB、F-RAM 和 Traveo 应视为赛普拉斯在美国和其他国家的商标或注册商标。请访问 [cypress.com](http://cypress.com) 获取赛普拉斯商标的完整列表。其他名称和品牌可能由其各自所有者主张为该方财产。