

DATA SHEET



GPM8F3733A/

GPM8F3717A/

GPM8F3709A/

32KB/16KB/8KB

Flash 8051 微控制器

2016 年 09 月 23 日

V1.0 版本

凌通科技股份有限公司保留对此文件修改之权利且不另行通知。凌通科技股份有限公司所提供之信息相信为正确且可靠之信息。但并不保证本文件中绝无错误。请于向凌通科技股份有限公司提出订单前自行确定所使用之技术文件及规格为最新之版本。文件中若有包含他人之专利或著作权之应用，若因贵公司使用本公司产品而涉及第三人之专利或著作权等智慧财产权之应用及配合时，则应由贵公司负责取得同意及授权，本公司仅单纯贩卖产品，上述取得同意及授权非属本公司应为保证之责任，又未经凌通科技股份有限公司之正式书面许可，本公司之所有产品不得使用于医疗器材，生命维持系统及飞航等相关设备。

目录

页	
目录	2
1. 概述	4
2. 特性	4
3. 框图	7
4. 管脚描述	8
4.1. IC 选型	8
4.2. 封装	9
4.2.1. 引脚说明	9
4.2.2. 管脚图	12
5. 功能描述	14
5.1. CPU	14
5.1.1. CPU 特性	14
5.1.2. 算术逻辑单元 (ALU)	14
5.1.3. 累加器 A 寄存器	14
5.1.4. B 寄存器	14
5.1.5. 程序状态字 (PSW)	14
5.1.6. 程序计数器 (PC)	14
5.2. 存储器	16
5.2.1. 介绍	16
5.2.2. 程序存储器分配	16
5.2.3. 数据存储器	17
5.2.4. 存储器相关的 SFR	19
5.2.5. 程序写使能位	19
5.2.6. 数据指针寄存器	19
5.2.7. 堆栈指针	19
5.3. 特殊功能寄存器 (SFR)	22
5.4. 省电模式	26
5.4.1. 介绍	26
5.4.2. HALT 模式	26
5.4.3. STOP 模式	26
5.5. 中断系统	29
5.6. 复位源	41
5.6.1. 介绍	41
5.6.2. 上电复位 (POR)	41
5.6.3. 低电压复位(LVR)	42
5.6.4. 低压检测(LVD)	42
5.6.5. 管脚复位(PAD_RST)	42
5.6.6. 看门狗复位(WDT_RST)	42

5.6.7. 其他复位源.....	43
5.7. 时钟源.....	46
5.8. 低速时钟.....	48
5.9. I/O 端口.....	48
5.10. 定时器模块.....	56
5.10.1. 介绍.....	56
5.10.2. Timer 0/1.....	56
5.10.3. Timer 2.....	63
5.10.4. Timer A/B.....	72
5.11. UART0.....	76
5.11.1. UART0/1: 模式 0 (同步移位寄存器).....	77
5.11.2. UART0/1: 模式 1 (8-Bit UART, 可变波特率, Timer1 时钟源).....	77
5.11.3. UART0/1: 模式 2 (9-Bit UART, 固定波特率).....	77
5.11.4. UART0/1: 模式 3 (9-Bit UART, 可变波特率, Timer1 时钟源).....	78
5.11.5. UART0/1 相关寄存器.....	78
5.12. SPI.....	82
5.13. I2C.....	85
5.13.1. I2C 总线协议.....	85
5.14. ADC.....	89
5.14.1. ADC 控制.....	89
5.14.2. ADC 相关寄存器.....	90
5.15. 电容触摸传感器.....	92
5.15.1. 电容测量方法.....	92
5.15.2. 电容触摸传感器相关寄存器.....	95
5.16. 指令集列表 (按字母顺序).....	102
5.16.1. 算术运算.....	102
5.16.2. 逻辑运算.....	102
5.16.3. 布尔运算.....	103
5.16.4. 数据传送.....	103
5.16.5. 编程跳转.....	104
6. 电气特性.....	105
6.1. 绝对最大额定值.....	105
6.2. DC 特性(VDD = 5V, T _A = 25℃).....	105
6.3. DC 特性 (VDD = 3.3V, T _A = 25℃).....	106
6.4. AC 特性 (T _A = 25℃).....	106
7. 封装/脚位.....	106
7.1. 订购信息.....	106
7.2. 封装信息.....	107
8. 免责声明.....	111
9. 修订记录.....	111

1. 概述

GPM8F3733A/GPM8F3717A/GPM8F3709A 是一款高度集成的微控制器，内置一个 1T 的流水线型 8051 CPU，内置 2KB/1KB/1KB XRAM、256Byte IDM SRAM 和 32KB/16KB/8KB 可编程 Flash。多达 27/25/21 个可编程多功能 I/O、Timer0/1/2、TimerA/B、UART0/1、SPI、I2C、ADC 以及通用型电容触碰传感器。这款微控制器工作电压范围:2.0V~5.5V，工作温度范围:-40℃~85℃。在电源管理单元中，具有 2 种省电模式。此外，该微控制器还拥有两线的片上调试电路，可以实现全速在线调试功能。详细内容会在以下的章节中介绍。

2. 特性

■ CPU

- ◆ 高速高性能的 1T 8051 CPU
 - 和工业标准的 8051 实现 100%的软件兼容
 - 借助于流水线型 RISC 架构，指令执行速度是标准 8051 的 10 倍。
- ◆ 最高时钟频率 32MHz

■ 存储器

- ◆ GPM8F3733A
 - 2K XRAM、256Byte IDM SRAM、32K Flash
- ◆ GPM8F3717A
 - 1K XRAM、256Byte IDM SRAM、16K Flash
- ◆ GPM8F3709A
 - 1K XRAM、256Byte IDM SRAM、8K Flash
- ◆ GPM8F3733A/GPM8F3717A/GPM8F3709A 可编程 Flash.
 - 最少 100K 次的编程/擦除周期
 - 最少 10 年的数据保存时间
 - 1K 字节的页大小

■ 时钟管理

- ◆ 内部振荡器: 32MHz±2% @ 2.0V~5.5V

■ 低速时钟

- ◆ 内部振荡器: 16KHz±50% @ 2.0V~5.5V

■ 电源管理

- ◆ 一个用于省电模式的 STOP 模式
- ◆ 一个仅限于低速时钟操作的 HALT 模式

■ 中断管理

- ◆ GPM8F3733A/GPM8F3717A
 - 多达 16 个内部中断源
 - 多达 5 个外部中断源
 - 多达 14 个键盘中断源
- ◆ GPM8F3709A
 - 多达 16 个内部中断源
 - 多达 4 个内部中断源
 - 多达 10 个键盘中断源

■ 复位管理

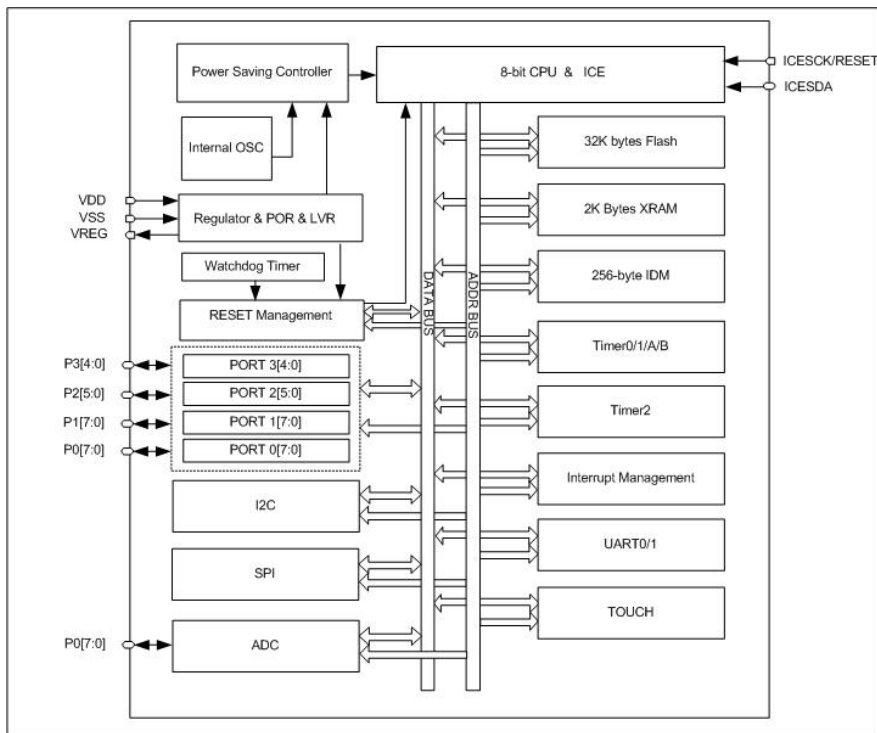
- ◆ 上电复位 (POR)

- ◆ 低电压复位 (LVR)
- ◆ 引脚复位 (PAD_RST)
- ◆ 看门狗复位 (WDT_RST)
- ◆ 软件复位 (S/W_RST)
- ◆ Flash 相关错误复位 (FLASH_ERR_RST)
- 可编程看门狗定时器
 - ◆ 时基发生器
 - ◆ 事件定时器
 - ◆ 系统监控
- I/O 端口
 - ◆ GPM8F3733A/ GPM8F3717A: 最大 27 个多功能双向 I/O; GPM8F3709A 最大 21 个多功能双向 I/O
 - 通过配置相应的寄存器, 每一个 IO 可以分别设置成输入上拉、输入下拉、输出高、输出低或者高阻输入状态。
 - IO 最大灌电流 20mA
 - IO 最大源电流 20mA
 - P0[7: 0]支持最大灌电流 120mA
- 2 个 16-bit 定时器/计数器 (Timer 0/1)
 - ◆ 定时器时钟源可选
 - ◆ 自动载入 8bit 定时器
- 功能强大的 Timer 2, 具有 16bit 的比较/捕捉单元
 - ◆ 定时器时钟源可选
 - ◆ 自动载入 16bit 定时器
 - ◆ 事件捕捉功能
 - ◆ 数字信号发生器
 - ◆ 脉宽调制和测量
- UART0/1
 - ◆ 一个同步模式
 - ◆ 三个异步模式
- SPI
 - ◆ 可编程的主时钟相位和极性
 - ◆ 可编程的主时钟频率
 - ◆ 自动读写功能
 - ◆ 最大 SPI 时钟频率: 16MHz (FOSC /2) @32MHz
- I2C
 - ◆ 可编程的主时钟频率
 - ◆ 最大 I2C 时钟频率: 800KHZ@32MHZ
400KHZ@16MHZ
- 模数转换器
 - ◆ GPM8F3733A/ GPM8F3717A
 - 具有 9 个通道, 12-bit 的解析度
 - ◆ GPM8F3709A
 - 具有 8 个通道, 12-bit 的解析度

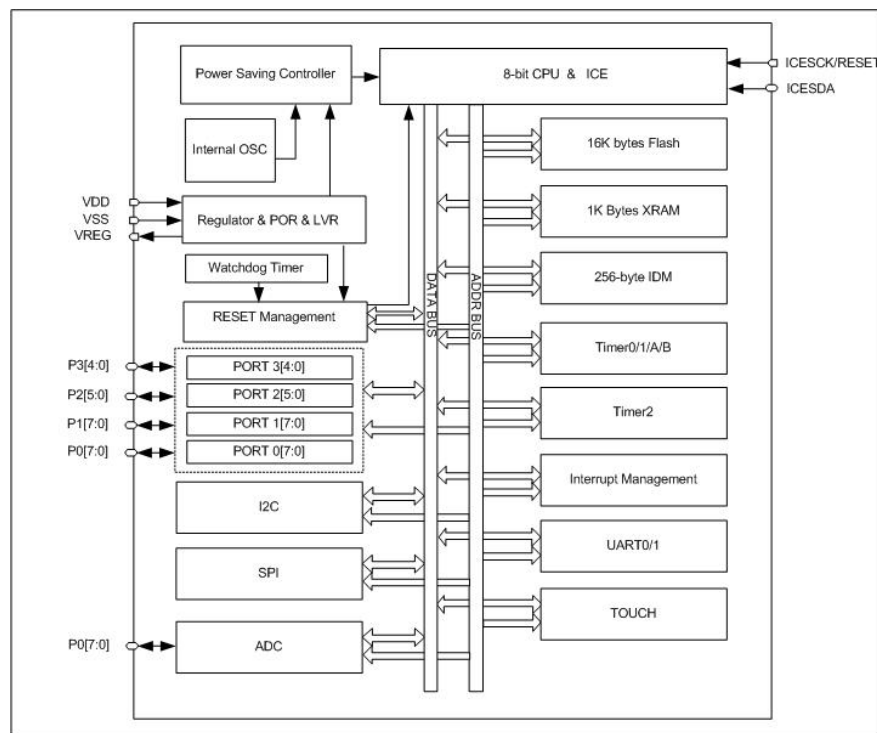
- 电容触摸传感器
 - ◆ GPM8F3733A/ GPM8F3717A
 - 多达 26 个通道的电容触摸传感器
 - ◆ GPM8F3709A
 - 多达 20 个通道的电容触摸传感器
- GPM8F3733A/ GPM8F3717A/ GPM8F3709A
 - 张弛振荡模式（CTS）
 - 一个 16-bit 的定时器 TimerA，用来计数设定的 CTS 扫描周期。
 - 一个 16-bit 的定时器 TimerB，用来计数完成设定的 CTS 扫描周期所需的时间。
 - 电荷转移模式（CTC）
 - 一个 16-bit 的定时器 TimerB，用来计数设定的 CTC 扫描周期。
 - 张弛振荡模式（CTA）
 - 一个 16-bit 的定时器 TimerA，用来计数设定的 CTS 扫描周期。
 - 一个 12 位的 ADC 用来测量 Cs PAD 电压
- 内建低压复位
 - ◆ 触发等级：1.9V、 2.2V、 2.7V、 3.6V
- 内建低压检测
 - ◆ 可编程的等级：2.1V、2.4V、2.9V、3.8V
- 内建调试单元
- 兼容 C 语言开发工具

3. 框图

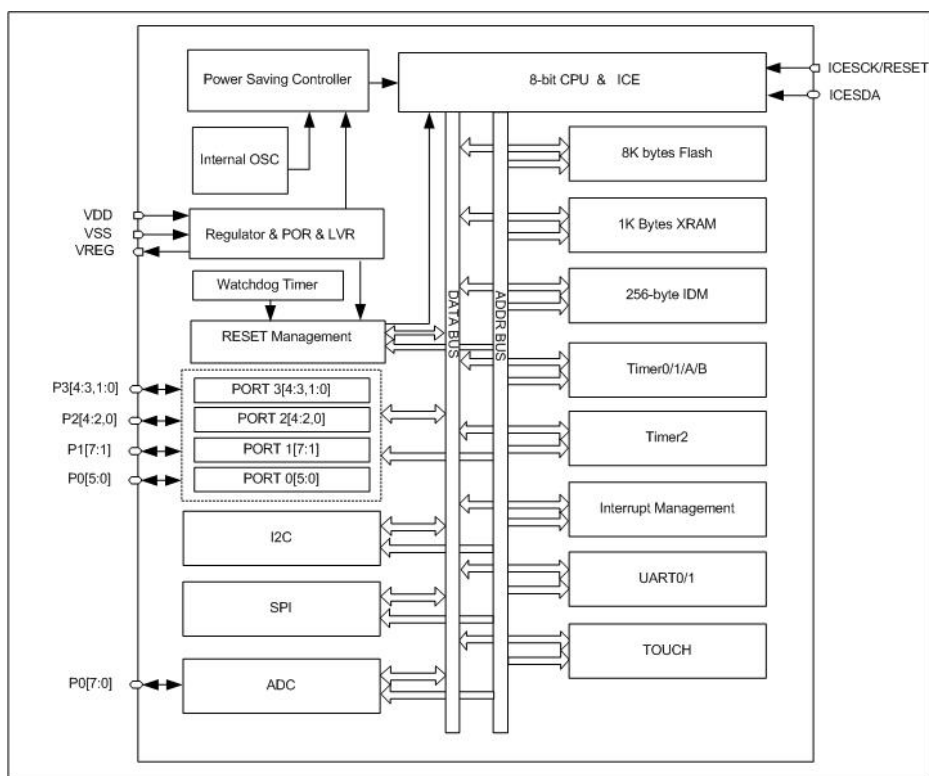
➤ GPM8F3733A



➤ GPM8F3717A



➤ GPM8F3709A



4. 管脚描述

4.1. IC 选型

IC	Device Name	FLASH(Bytes)	XRAM(Bytes)	IRAM(Bytes)	ADC	Touch	I/O/PORT	Package
GPM8F3733A	--	32K	2K	256	8 Channel	26Channel	27	SSOP30
GPM8F3717A	--	16K	1K	256	2 Channel	24Channel	25	SOP28
GPM8F3709A	--	8K	1K	256	1 Channel	20 Channel	21	SOP24
					Channel	12 Channel	13	SOP24

4.2. 封装

4.2.1. 引脚说明

➤ GPM8F3733A

I = 输入, O = 输出, S = 电源管脚

引脚	引脚	类型	描述
P24	1	I/O	Port 2' s bit 4/ Cs Pad(ADC channel 8 input)/ KBI12
P10	2	I/O	Port 1' s bit 0/ AN0 (ADC channel 0 input)/ RXD01/ Touch
P25	3	I/O	Port 2' s bit 5/ KBI13/ Touch
ICESDA/P11	4	I/O	Port 1' s bit 1/ ICE data input/output/ AN1 (ADC channel 1 input) / Touch
ICESCK/RESET	5	I	ICE clock input/ RESET
P12	6	I/O	Port 1' s bit 2/ AN2 (ADC channel 2 input) / TXD01/ Touch
VSS	7	S	Ground
P13	8	I/O	Port 1' s bit 3/ AN3 (ADC channel 3 input) / Touch
VDD	9	S	Power 5V input
P14	10	I/O	Port 1' s bit 4/ AN4 (ADC channel 4 input)/ SPI_TXD/ TXD1/ Touch
P15	11	I/O	Port 1' s bit 5/ AN5 (ADC channel 5 input)/ SPI_CLK/ RXD1/ Touch
P16	12	I/O	Port 1' s bit 6/ AN6 (ADC channel 6 input)/ SPI_RXD/ Touch
P17	13	I/O	Port 1' s bit 7/ AN7 (ADC channel 7 input)/ SPI_CSB/ Touch
P30	14	I/O	Port 3' s bit 0/ RXD00/ INT3/ Touch
P31	15	I/O	Port 3' s bit 1/ TXD00/ INT4/ Touch
P32	16	I/O	Port 3' s bit 2/ I2CCK1/ INT0/ Touch
P00	17	I/O	Port 0' s bit 0/ KBI0/ Touch
P01	18	I/O	Port 0' s bit 1/ KBI1/ Touch
P02	19	I/O	Port 0' s bit 2/ KBI2/ Touch
P03	20	I/O	Port 0' s bit 3/ KBI3/ Touch
P04	21	I/O	Port 0' s bit 4/ KBI4/ Touch
P05	22	I/O	Port 0' s bit 5/ KBI5/ Touch
P06	23	I/O	Port 0' s bit 6/ KBI6/ Touch
P07	24	I/O	Port 0' s bit 7/ KBI7/ Touch
P33	25	I/O	Port 3' s bit 3/ I2CSDA1/ INT1/ Touch
P34	26	I/O	Port 3' s bit 4/ INT2/ CCP0/ Touch
P20	27	I/O	Port 2' s bit 0/ CCP1/ KBI8/ Touch
P21	28	I/O	Port 2' s bit 1/ CCP2/ KBI9/ Touch
P22	29	I/O	Port 2' s bit 2/ I2CCK0/ CCP3/ KBI10/ Touch
P23	30	I/O	Port 2' s bit 3/ I2CSDA0/ KBI11/ Touch

GPM8F3733A/GPM8F3717A/GPM8F3709A

➤ GPM8F3717A

I = 输入, O = 输出, S = 电源管脚

引脚	引脚	类型	描述
P24	1	I/O	Port 2' s bit 4/ Cs Pad(ADC channel 8 input)/ KBI12
ICESDA/P11	2	I/O	Port 1' s bit 1/ ICE data input/output/ AN1 (ADC channel 1 input) / Touch
ICESCK/RESET	3	I	ICE clock input/ RESET
P12	4	I/O	Port 1' s bit 2/ AN2 (ADC channel 2 input) / TXD01/ Touch
VSS	5	S	Ground
P13	6	I/O	Port 1' s bit 3/ AN3 (ADC channel 3 input) / Touch
VDD	7	S	Power 5V input
P14	8	I/O	Port 1' s bit 4/ AN4 (ADC channel 4 input)/ SPI_TXD/ TXD1/ Touch
P15	9	I/O	Port 1' s bit 5/ AN5 (ADC channel 5 input)/ SPI_CLK/ RXD1/ Touch
P16	10	I/O	Port 1' s bit 6/ AN6 (ADC channel 6 input)/ SPI_RXD/ Touch
P17	11	I/O	Port 1' s bit 7/ AN7 (ADC channel 7 input)/ SPI_CSB/ Touch
P30	12	I/O	Port 3' s bit 0/ RXD00/ INT3/ Touch
P31	13	I/O	Port 3' s bit 1/ TXD00/ INT4/ Touch
P32	14	I/O	Port 3' s bit 2/ I2CSCK1/ INT0/ Touch
P00	15	I/O	Port 0' s bit 0/ KBI0/ Touch
P01	16	I/O	Port 0' s bit 1/ KBI1/ Touch
P02	17	I/O	Port 0' s bit 2/ KBI2/ Touch
P03	18	I/O	Port 0' s bit 3/ KBI3/ Touch
P04	19	I/O	Port 0' s bit 4/ KBI4/ Touch
P05	20	I/O	Port 0' s bit 5/ KBI5/ Touch
P06	21	I/O	Port 0' s bit 6/ KBI6/ Touch
P07	22	I/O	Port 0' s bit 7/ KBI7/ Touch
P33	23	I/O	Port 3' s bit 3/ I2CSDA1/ INT1/ Touch
P34	24	I/O	Port 3' s bit 4/ INT2/ CCP0/ Touch
P20	25	I/O	Port 2' s bit 0/ CCP1/ KBI8/ Touch
P21	26	I/O	Port 2' s bit 1/ CCP2/ KBI9/ Touch
P22	27	I/O	Port 2' s bit 2/ I2CSCK0/ CCP3/ KBI10/ Touch
P23	28	I/O	Port 2' s bit 3/ I2CSDA0/ KBI11/ Touch

GPM8F3733A/GPM8F3717A/GPM8F3709A

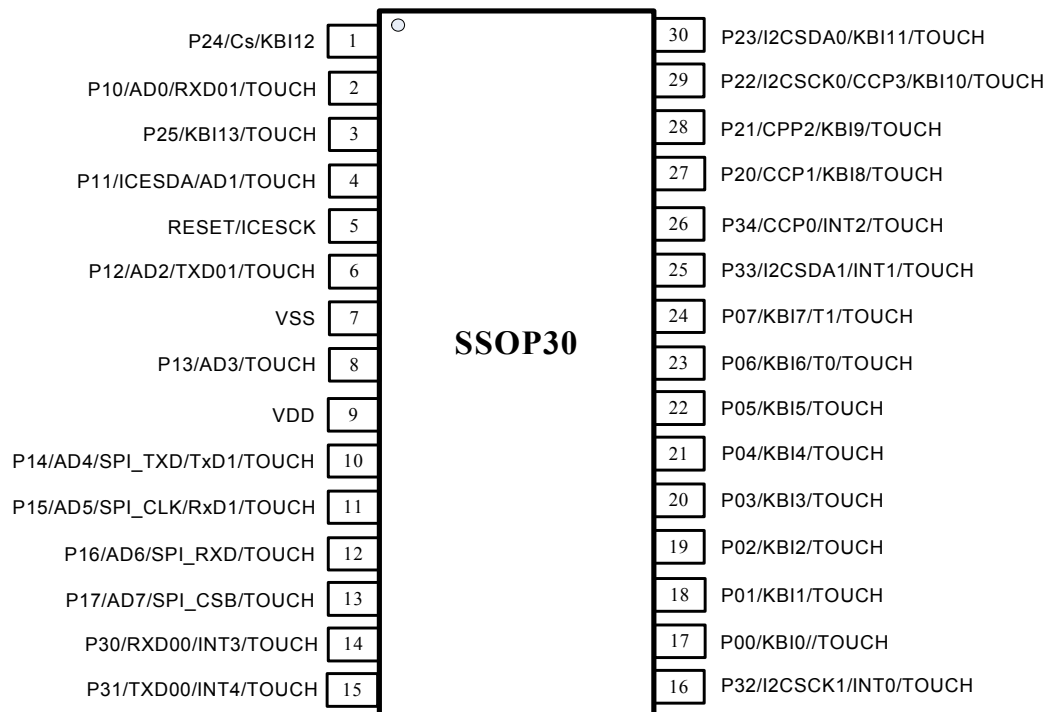
➤ GPM8F3709A

I = 输入, O = 输出, S = 电源管脚

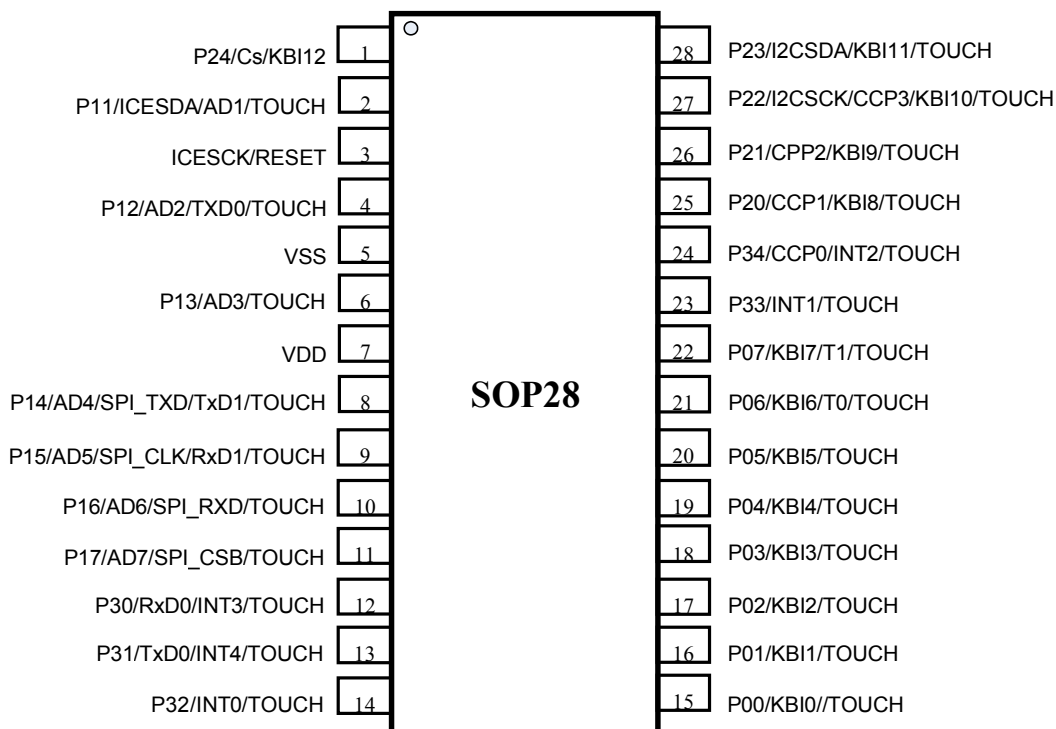
Pin Name	SOP24	SOP16	功能	描述
P24	1	14	I/O	Port 2' s bit 4/ Cs Pad(ADC channel 8 input)/ KBI12
ICESDA/P11	2	15	I/O	Port 1' s bit 1/ ICE data input/output/ AN1 (ADC channel 1 input) / Touch
ICESCK	3	16	I	ICE clock input/ RESET
P12	4	NC	I/O	Port 1' s bit 2/ AN2 (ADC channel 2 input) / TXD01/ Touch
VSS	5	1	S	Ground
P13	6	NC	I/O	Port 1' s bit 3/ AN3 (ADC channel 3 input) / Touch
VDD	7	2	S	Power 5V input
P14	8	NC	I/O	Port 1' s bit 4/ AN4 (ADC channel 4 input)/ SPI_TXD/ TXD1/ Touch
P15	9	NC	I/O	Port 1' s bit 5/ AN5 (ADC channel 5 input)/ SPI_CLK/ RXD1/ Touch
P16	10	NC	I/O	Port 1' s bit 6/ AN6 (ADC channel 6 input)/ SPI_RXD/ Touch
P17	11	NC	I/O	Port 1' s bit 7/ AN7 (ADC channel 7 input)/ SPI_CSB/ Touch
P30	12	3	I/O	Port 3' s bit 0/ RXD00/ INT3/ Touch
P31	13	4	I/O	Port 3' s bit 1/ TXD00/ INT4/ Touch
P00	14	7	I/O	Port 0' s bit 0/ KBI0/ Touch
P01	15	8	I/O	Port 0' s bit 1/ KBI1/ Touch
P02	16	9	I/O	Port 0' s bit 2/ KBI2/ Touch
P03	17	10	I/O	Port 0' s bit 3/ KBI3/ Touch
P04	18	6	I/O	Port 0' s bit 4/ KBI4/ Touch
P05	19	5	I/O	Port 0' s bit 5/ KBI5/ Touch
P33	20	NC	I/O	Port 3' s bit 3/ I2CSDA1/ INT1/ Touch
P34	21	NC	I/O	Port 3' s bit 4/ INT2/ CCP0/ Touch
P20	22	11	I/O	Port 2' s bit 0/ CCP1/ KBI8/ Touch
P22	23	12	I/O	Port 2' s bit 2/ I2CCK0/ CCP3/ KBI10/ Touch
P23	24	13	I/O	Port 2' s bit 3/ I2CSDA0/ KBI11/ Touch

4.2.2. 管脚图

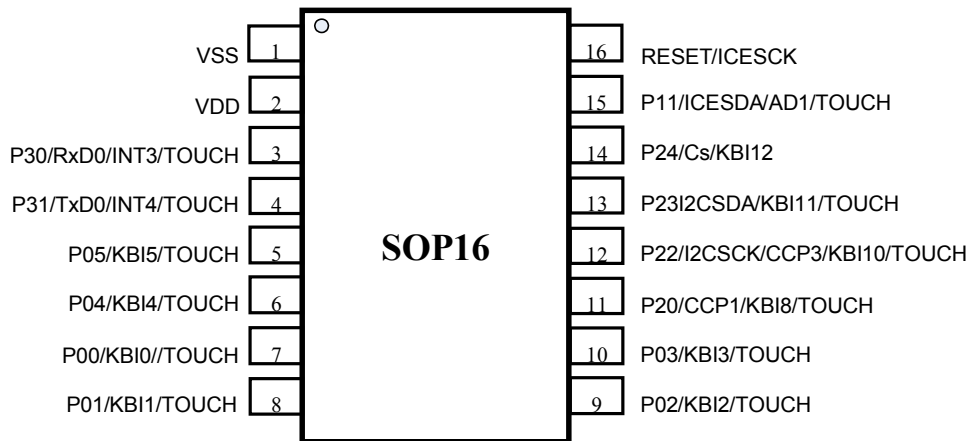
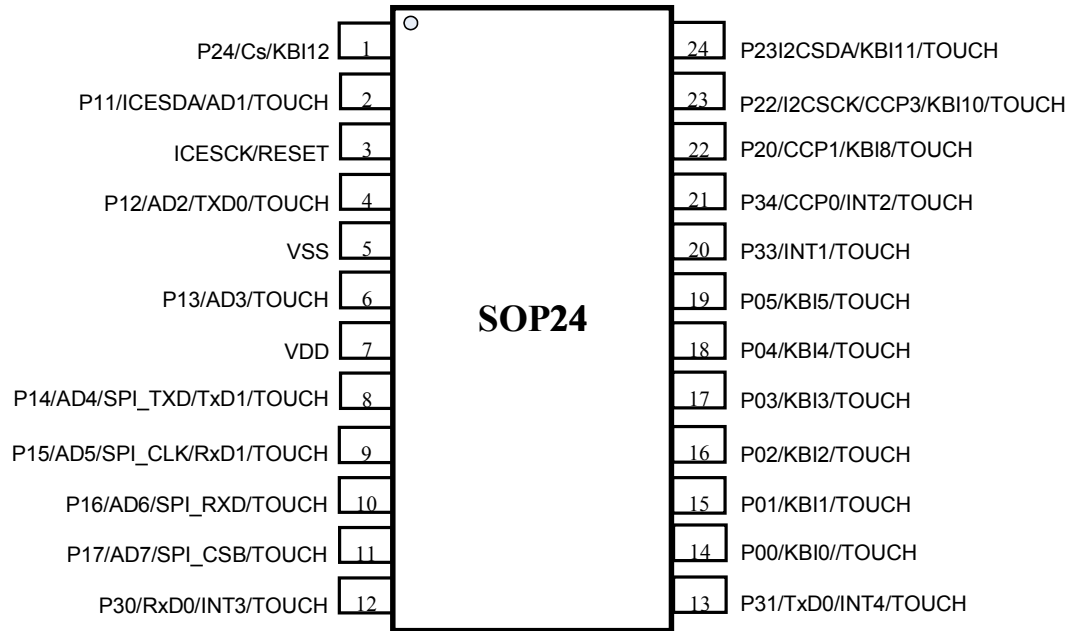
➤ GPM8F3733A



➤ GPM8F3717A



➤ GPM8F3709A



5. 功能描述

5.1. CPU

该款 CPU 具有高速度和高性能。流水线型的结构使它的工作速度比标准 8051 结构快 10 倍。这一特性在低功率应用领域具有巨大的优势，内核可在低于正常工作频率 10 倍的情况下依然保持原有性能。

该款 CPU 完全兼容于工业标准的 8051 微控制器，具有与所有指令助记符和二进制相兼容的特性。强大的结构特性使其可以高性能、高速的执行指令。

5.1.1. CPU 特性

- 与工业 8051 实现 100%软件兼容
- 比标准 8051 快 24 倍的乘法器
- 比标准 8051 快 12 倍的加法器

该款 CPU 完全兼容于工业标准的 8051 微控制器，具有与所有指令助记符和二进制相兼容的特性。强大的结构特性使其可以高性能、高速的执行指令。处理器的算术部分负责进行数据操作，是由一个 8-bit 算术逻辑单元 (ALU)，一个 ACC(0xE0) 寄存器，B(0xF0) 寄存器和 PSW(0xD0) 寄存器构成。

5.1.2. 算术逻辑单元 (ALU)

在一条指令运行期间，ALU 负责完成算术和逻辑运算。典型的算术运算有加法、减法、乘法和除法，其它运算有加 1、减 1、BCD 编码运算的十进制调整以及比较运算。逻辑单元可实现与、或、异或、取反和移位等操作。布尔处理器可以实现位运算，如置位、清零、取反、非置位跳转、置位且清零跳转、移位/进位。

5.1.3. 累加器 A 寄存器

该累加器是一个 8-bit 通用寄存器，可实现数据传送、暂存以及条件判定等操作。

5.1.4. B 寄存器

B 寄存器在乘法和除法运算中使用。在其他情况下，可作为通常的特殊功能寄存器 (SFR)。

5.1.5. 程序状态字 (PSW)

与通用 CPU 的状态寄存器作用类似，PSW 包含了反映 CPU 当前状态的几个位。

5.1.6. 程序计数器 (PC)

程序计数器是一个 16-bit 寄存器，由 PCH 和 PCL 两个 8-bit 寄存器构成。用来表示下一条要执行指令的地址。在复位状态下，程序计数器存放的内容为 0x0000。

ACC			地址: 0xE0		累加器 A 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	ACC[7:0]							
默认	0	0	0	0	0	0	0	0

ACC			地址: 0xE0		累加器 A 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	ACC[7:0]							
默认	0	0	0	0	0	0	0	0

表 5- 1 ACC 寄存器

GPM8F3733A/GPM8F3717A/GPM8F3709A

B			地址: 0xF0		B 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	B[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	B[7:0]	R/W	累加器 B	

表 5- 2 B 寄存器

PSW			地址: 0xD0		程序状态寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CY	AC	—	RS1	RS0	OV	—	P
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述		条件
7	CY	R/W	进位标志		
6	AC	R/W	辅助进位标志		
5	—	R	预留		
4:3	RS[1:0]	R/W	工作寄存器组(R0~R7)选择位		
			RS[1:0]	功能说明	
			00	0 组, 物理地址 0x00-0x07	
			01	1 组, 物理地址 0x08-0x0F	
			10	2 组, 物理地址 0x10-0x17	
			11	3 组, 物理地址 0x18-0x1F	
2	OV	R/W	溢出标志		
1	—	R	预留		
0	P	R/W	奇偶标志		

表 5- 3 PSW 寄存器

5.2. 存储器

5.2.1. 介绍

GPM8F3733A / GPM8F3717A / GPM8F3709A 有 3 个独立的地址空间用于程序存储器和数据存储器。程序存储器是一个片上可重复编程的 32K/16K/8K 字节 Flash。数据存储器分为 2K/1K/1K 的 XRAM 和 256 字节的 IDM，IDM 包含了 128 字节可读写的 SFR。IDM 的上半部分具有和 SFR 相同的地址空间，通过不同的寻址方式来访问。

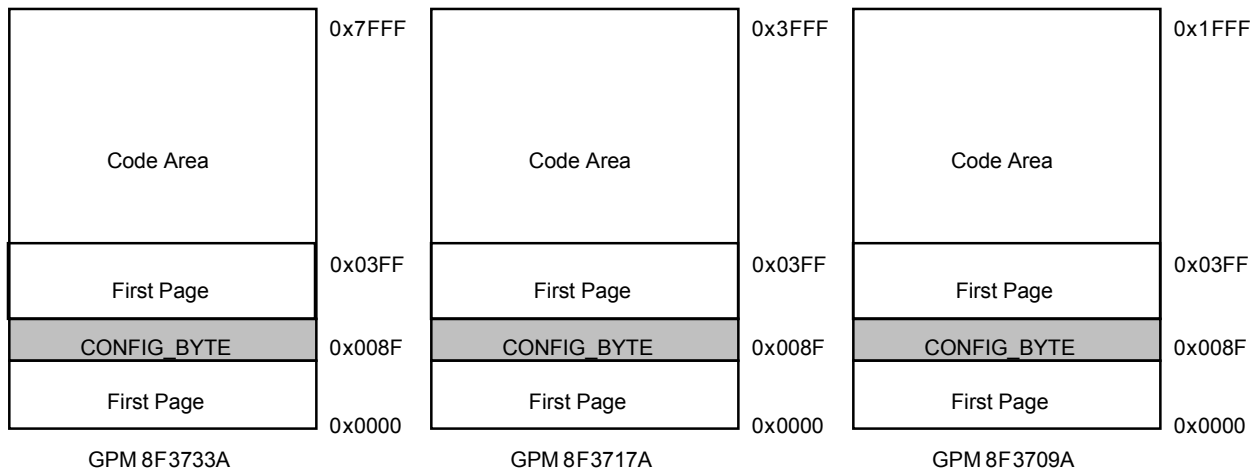
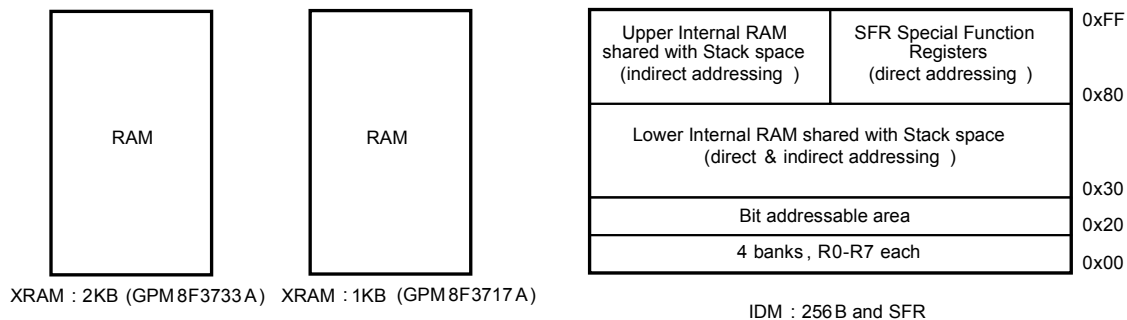


图 5- 1 程序存储区



数据存储器

5.2.2. 程序存储器分配

GPM8F3733A / GPM8F3717A / GPM8F3709A 具有 32K/16K/8K 字节的程序存储器，程序存储器分为 FirstPage 和 NormalPage 两个部分，FirstPage 的地址空间是 0x0000~0x03FF，用来存储 Reset 向量、IRQ 向量、用户密码和 CONFIG_BYTE。CONFIG_BYTE 在程序存储器中的地址是 0x008F，表 5-4 描述了 CONFIG 的定义。如果 CONFIG_BYTE[0] 被设置为 ‘0’，则整个芯片的存储器就被保护起来，任何通过两线串行接口进行页擦除或编程的动作都是不允许的，用户唯一能做的就是擦除整个芯片。用户也可以选择设置 I/O 的初始状态为输入上拉来配置 CONFIG_BYTE[1]为 “0”。

每次复位后，CPU 都要从程序存储器的 0x0000 位置开始执行指令。Flash 存储器可以进行在系统编程，通过 SCK/SDA 接口或者通过软件在 PWE=1 时使用 MOVX 指令实现。Flash 的数据不能实现从 ‘0’ 变为 ‘1’ 的编程，只能通过擦除动作来实现。因此，通常情况下在对 Flash 编程之前，Flash 中的数据应该先被擦除（设置为 0xFF）。出于软件的安全性考虑，用户可通过配置 FL_LEVEL 寄存器来设置 Flash 的级别来限制代码区域，避免不小心的软件改写和擦除。被保护的区域称为 READONLY_PAGE。

5.2.3. 数据存储器

GPM8F3733A / GPM8F3717A / GPM8F3709A 的数据存储器被分为两个部分，第一部分是 2K/1K/1K 的外部 XRAM，第二部分是 256 字节的 IDM RAM，如表 5-1。内部数据存储器 (IDM) 的最低地址区域包含 4 个寄存器组，每个寄存器组包含 8 个寄存器。128 bits (16 bytes) 的位寻址区起始于 0x20。0x30 到 0x7F 这个区域没有特殊定义，用户可以自由使用，在对 IDM 的 0x80 到 0xFF 地址区间进行操作时，使用间接寻址模式时，寻址到的是与堆栈空间共享的数据存储器，使用直接寻址模式时，寻址到的是 SFR 区域。SFR 的存储映射如表 5-8 SFR 存储器映射所示。

CONFIG_BYTE			地址: 0x8F (Flash)		CONFIG_BYTE 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	--	--	--	IOSEL	CODE_UNLOCK
默认	1	1	1	1	1	1	1	1

Bit	功能	类型	描述	条件
7:2	--	R	预留	
1	IOSEL	R/W	IO 初始状态选择位 0: 输入上拉, 1: 浮空	
0	UNLOCK	R/W	程序空间保护使能位 0: 代码锁定, 1: 代码不锁定	

注意: 默认 CONFIG_BYTE 数值是 0xFF

表 5-4 CONFIG_BYTE 描述

FLASHCON			地址: 0xEC		Flash 控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	--	--	--	P_ERASE	PROG
默认	0	0	0	0	0	0	0	0
Key Code	0x8F, 0x32, 0x51							

Bit	功能	类型	描述	条件
7:2	--	R/W	预留	
1	P_ERASE	R/W	Flash 页擦除使能位 0: Flash 页擦除禁用, 1: Flash 页擦除使能	
0	PROG	R/W	Flash 编程使能位 0: Flash 编程禁用, 1: Flash 编程使能	

表 5-5 FLASHCON 寄存器

FL_LEVELS			地址: 0xED		Flash Level 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	FL_LEVEL[4:0]				
默认	1	1	1	1	1	1	1	1

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:5	--	R/W	预留	
4:0	FL_LEVEL[2:0]	R/W	FL_LEVEL 表示有多少个 1K 的 Flash 页被设置成只读，详细如下表。	

表 5- 6 FL_LEVEL 寄存器

FL_LEVEL	注意
0x00	所有页都可读可写
0x01	地址< 0x800 的页只读
0x02	地址< 0xC00 的页只读
0x03	地址< 0x1000 的页只读
0x04	地址< 0x1400 的页只读
0x05	地址< 0x1800 的页只读
0x06	地址< 0x1C00 的页只读
0x07	地址< 0x2000 的页只读
0x08	地址< 0x2400 的页只读
0x09	地址< 0x2800 的页只读
0x0A	地址< 0x2C00 的页只读
0x0B	地址< 0x3000 的页只读
0x0C	地址< 0x3400 的页只读
0x0D	地址< 0x3800 的页只读
0x0E	地址< 0x3C00 的页只读
0x0F	地址< 0x4000 的页只读
0x10	地址< 0x4400 的页只读
0x11	地址< 0x4800 的页只读
0x12	地址< 0x4C00 的页只读
0x13	地址< 0x5000 的页只读
0x14	地址< 0x5400 的页只读
0x15	地址< 0x5800 的页只读
0x16	地址< 0x5C00 的页只读
0x17	地址< 0x6000 的页只读
0x18	地址< 0x6400 的页只读
0x19	地址< 0x6800 的页只读
0x1A	地址< 0x6C00 的页只读
0x1B	地址< 0x7000 的页只读
0x1C	地址< 0x7400 的页只读
0x1D	地址< 0x7800 的页只读
0x1E	地址< 0x7C00 的页只读
0x1F	地址< 0x8000 的页只读

表 5- 7 FL_LEVEL 寄存器描述

注意 1: 黑色: 标准 8051 寄存器, 灰色: 新增寄存器

0xF8	EIP	SYS_CTRL0		SYS_CTRL2	SYS_CTRL3	SYS_CTRL4	SYS_CTRL5	SYS_CTRL6
0xF0	B	ADC_CTRL0	ADC_CTRL1	ADC_DATA	ADC_DATAH	CTS_CTRL7	CTS_CTRL8	CTS_CTRL9
0xE8	EIE			KEYCODE	FLASHCON	FL_LEVEL	EXIPOL	EXIMOD
0xE0	ACC	TMAL	TMAH	TMA_PLOADL	TMA_PLOADH	TMB_CCPL	TMB_CCPH	
0xD8	WDCON	CTS_CTRL0	CTS_CTRL1	CTS_CTRL2	CTS_CTRL3	CTS_CTRL4	CTS_CTRL5	CTS_CTRL6
0xD0	PSW	I2C_CTRL	I2C_STS	I2C_DID	I2C_DATA			
0xC8	T2CON	T2IF	CRCL	CRCH	TL2	TH2	CCEN	I2C_DEBOUNCE
0xC0	SCON1	SBUF1	CCL1	CCH1	CCL2	CCH2	CCL3	CCH3
0xB8	IP		P1_PU	P1_PD	P1_SMTDIS	P1_SR	TMB_PLOADL	TMB_PLOADH
0xB0	P3		P3_PU	P3_PD	P3_SMTDIS	P3_SR	TMBL	TMBH
0xA8	IE		P0_PU	P0_PD	P0_SMTDIS	P0_SR	P0_HS	P0_HSBIA
0xA0	P2		P2_PU	P2_PD	P2_SMTDIS	P2_SR	KBEN_P0	KBEN_P2
0x98	SCON0	SBUF0	SPI_CTRL	SPI_STS	SPI_TXD	SPI_RXD		
0x90	P1	EIF	CCL0	CCH0				WKUEN
0x88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	
0x80	P0	SP	DPL0	DPH0	DPL1	DPH1	DPS	PCON
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F

表 5- 8 SFR 存储器映射

5.2.4. 存储器相关的 SFR

接下来的章节中描述了 8051 内核中与程序、外部和内部存储器相关的 SFR 及其功能。关于标准 SFR 的其它信息请参考相关的外设章节

5.2.5. 程序写使能位

程序写使能 (PWE) 位, 位于 PCON 寄存器的 bit4, 在 MOVX 指令时起作用。当 PWE 位设为逻辑 1 时, “MOVX @DPTR, A” 这条指令会将累加器 A 寄存器中的数据写入由 DPTR 寄存器寻址的程序存储器中。 无论 PWE 位如何, 程序存储器只能通过 MOVX 来读取。

5.2.6. 数据指针寄存器

双数据指针寄存器用来加速数据块的复制。DPTR0 和 DPTR1 位于 4 个 SFR 地址中。设置 SEL 位 (DPS[0]) 来使相应的 DPTR 寄存器有效。如果 SEL=0 选择 DPTR0, 否则选择 DPTR1。

5.2.7. 堆栈指针

8051 有一个 8-bit 堆栈指针称为 SP (0x81), 位于内部 RAM 区域。执行 PUSH 和 CALL 指令时在数据存入之前堆栈指针加 1; 执行 POP, RET 和 RETI 指令时在数据弹出后堆栈指针减 1。换句话说, 它总是指向最后一个有效的堆栈字节。 SP 的访问方式与其它 SFR 寄存器是一样的。

图 5- 2 “PUSH A” 指令的堆栈顺序是执行 PUSH A 的例子, 图 5- 3 “POP PSW” 指令的堆栈顺序是执行 POP PSW 的例子。

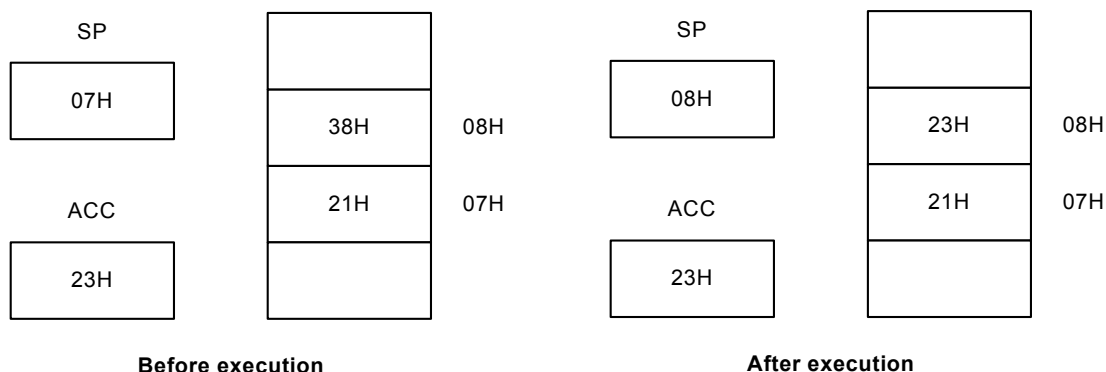


图 5- 2 “PUSH A” 指令的堆栈顺序

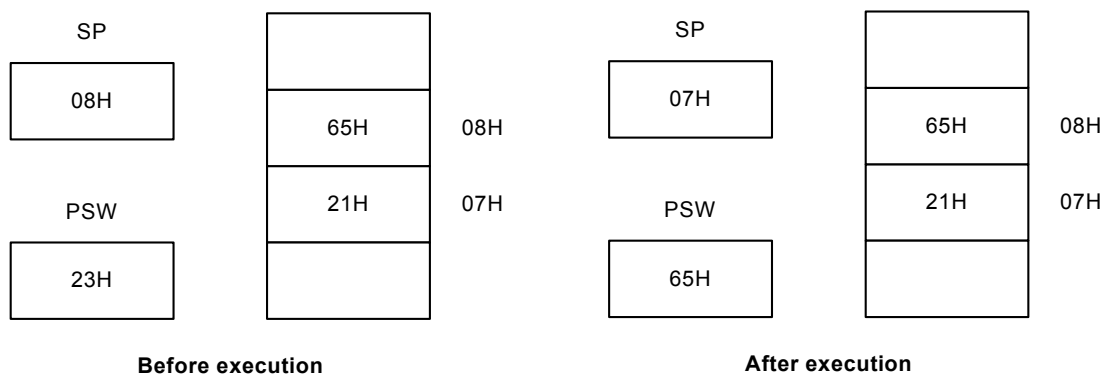


图 5- 3 “POP PSW” 指令的堆栈顺序

PCON			地址: 0x87		电源配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SMOD0	SMOD1	--	PWE	--	--	STOP	--
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	SMOD0	R/W	当时钟源是 Timer1 时, UART0 的波特率位	
6	SMOD1	R/W	当时钟源是 Timer1 时, UART0 的波特率位	
5	--	R/W	预留	
4	PWE	R/W	Flash 编程使能位 0: 在 MOVX 指令期间, 禁止 Flash 写有效 1: 在 MOVX 指令期间, 使能 Flash 写有效	
3: 2	--	R/W	预留	
1	STOP	R/W	停止模式使能位 0: 禁止; 1: 使能	
0	--	R/W	预留	

表 5- 9 PCON 寄存器

GPM8F3733A/GPM8F3717A/GPM8F3709A

DPH0			地址: 0x83		数据指针寄存器 0 高字节			
Bit	7	6	5	4	3	2	1	0
功能	DPTR0[15:8]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	DPTR0[15:8]	R/W	数据指针寄存器 0 高字节	

表 5- 10 DPH0 寄存器

DPL0			地址: 0x82		数据指针寄存器 0 低字节			
Bit	7	6	5	4	3	2	1	0
功能	DPTR0[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	DPTR0[7:0]	R/W	数据指针寄存器 0 低字节	

表 5- 11 DPL0 寄存器

DPH1			地址: 0x85		数据指针寄存器 1 高字节			
Bit	7	6	5	4	3	2	1	0
功能	DPTR1[15:8]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	DPTR1[15:8]	R/W	数据指针寄存器 1 高字节	

表 5- 12 DPH1 寄存器

DPL1			地址: 0x84		数据指针寄存器 1 低字节			
Bit	7	6	5	4	3	2	1	0
功能	DPTR0[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	DPTR1[7:0]	R/W	数据指针寄存器 1 低字节	

表 5- 13 DPL1 寄存器

DPS			地址：0x86		数据指针选择寄存器			
Bit	7	6	5	4	3	2	1	0
功能	ID1	ID0	TSL	–	–	–	–	SEL
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:6	ID[1:0]	R/W	加 1/减 1 功能选择 如表 5- 15 PTR0/DPTR1 操作。	
5	TSL	R/W	翻转切换选择使能位 0: DPTR 相关指令不影响 SEL 位的状态 1: DPTR 相关指令切换 SEL 位	
4:1	–	R/W	预留	
0	SEL	R/W	数据指针选择有效位 如表 5- 15 PTR0/DPTR1 操作。	

表 5- 14 DPS 寄存器

ID1	ID0	SEL=0	SEL=1
0	0	INC DPTR0	INC DPTR1
0	1	DEC DPTR0	INC DPTR1
1	0	INC DPTR0	DEC DPTR1
1	1	DEC DPTR0	DEC DPTR1

表 5- 15 PTR0/DPTR1 操作

SP			地址：0x81		堆栈指针寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SP[7:0]							
默认	0	0	0	0	0	1	1	1

Bit	功能	类型	描述	条件
7:0	SP[7:0]	R/W	堆栈指针	

表 5- 16 SP 寄存器

5.3. 特殊功能寄存器（SFR）

GPM8F3733A / GPM8F3717A / GPM8F3709A 有 111 个特殊功能寄存器。MCU 和外设功能模块通过使用这些 SFR 来控制想要实现的操作。一些 SFR 包含了外设模块的控制和状态位，如：定时器单元、中断控制单元等。一些 SFR 中的有些位是只读的，写入这些位无效，而另一些 SFR 则有 KeyCode 设计，出于软件安全的考虑，在将数据写入这些 SFR 之前，必须先按顺序正确地将 KeyCode 写入 KEYCODE 寄存器中。下表对 SFR 进行了总结，每个 SFR 的详细信息会在每个外设章节中进行介绍。

GPM8F3733A/GPM8F3717A/GPM8F3709A

Addr	Function	Key Code	Reset Value	7	6	5	4	3	2	1	0
0x80	P0		0xFF	Port 0							
0x81	SP		0x07	Stack Pointer							
0x82	DPL0		0x00	Data pointer register DPTR0 — low byte							
0x83	DPH0		0x00	Data pointer register DPTR0 — high byte							
0x84	DPL1		0x00	Data pointer register DPTR1 — low byte							
0x85	DPH1		0x00	Data pointer register DPTR1 — high byte							
0x86	DPS		0x00	ID1	ID0	TSL	--	--	--	--	SEL
0x87	PCON		0x00	SMOD0	SMOD1	--	PWE	--	--	STOP	--
0x88	TCON		0x00	TF1	TR1	TF0	TR0	IE1	--	IE0	--
0x89	TMOD		0x00	--	CT1	M11	M10	--	CT0	M01	M00
0x8A	TL0		0x00	Timer 0 Load value - low byte							
0x8B	TL1		0x00	Timer 1 Load value - low byte							
0x8C	TH0		0x00	Timer 0 Load value - high byte							
0x8D	TH1		0x00	Timer 1 Load value - high byte							
0x8E	CKCON		0x07	WD1	WD0	WDFM	T1M	T0M	--	--	--
0x90	P1		0xff	P17	P16	P15	P14	P13	P12	P11	P10
0x91	EIF		0x00	KBIF	--	--	SPIF	--	INT4F	INT3F	INT2F
0x92	CCL0		0x00	Timer 2 cc0 capture low byte							
0x93	CCH0		0x00	Timer 2 cc0 capture high byte							
0x97	WKUEN		0x00	KB_ WKUEN	WD_ WKUEN	--	INT4_ WKUEN	INT3_ WKUEN	INT2_ WKUEN	INT1_ WKUEN	INT0_ WKUEN
0x98	SCON0		0x00	SM00	SM01	SM02	REN0	TB08	RB08	TI0	RI0
0x99	SBUF0		0x00	UART 0 buffer							
0x9A	SPI_CTRL		0x00	POLARITY	PHASE	CSB_ KEEP	SPI_CLK_SEL[1:0]		SPI_EN	SPI_ MODE	SPI_ START
0x9B	SPI_STS		0x00	SPI_INTEN	--	--	--	--	--	RX_DONE	TX_DONE
0x9C	SPI_TXD		0x00	SPI transmission data							
0x9D	SPI_RXD		0x00	SPI receive data							
0xA0	P2		0xFF	--	--	P25	P24	P23	P22	P21	P20
0xA2	P2_PU		0x00	--	--	P25_PU	P24_PU	P23_PU	P22_PU	P21_PU	P20_PU
0xA3	P2_PD		0x00	--	--	P25_PD	P24_PD	P23_PD	P22_PD	P21_PD	P20_PD
0xA4	P2_SMTDIS		0x00	--	--	P25_ SMTDIS	P24_ SMTDIS	P23_ SMTDIS	P22_ SMTDIS	P21_ SMTDIS	P20_ SMTDIS
0xA5	P2_SR		0x00	--	--	P25_SR	P24_SR	P23_SR	P22_SR	P21_SR	P20_SR
0xA6	KBEN_P0		0x00	KBEN_ P07	KBEN_ P06	KBEN_ P05	KBEN_ P04	KBEN_ P03	KBEN_ P02	KBEN_ P01	KBEN_ P00
0xA7	KBEN_P2		0x00	--	--	KBEN_ P25	KBEN_ P24	KBEN_ P23	KBEN_ P22	KBEN_ P21	KBEN_ P20
0xA8	IE		0x00	EA	ECCP	ET2	ES0	ET1	EX1	ET0	EX0
0xAA	P0_PU		0x00	P07_PU	P06_PU	P05_PU	P04_PU	P03_PU	P02_PU	P01_PU	P00_PU
0xAB	P0_PD		0x00	P07_PD	P06_PD	P05_PD	P04_PD	P03_PD	P02_PD	P01_PD	P00_PD
0xAC	P0_SMTDIS		0x00	P07_ SMTDIS	P06_ SMTDIS	P05_ SMTDIS	P04_ SMTDIS	P03_ SMTDIS	P02_ SMTDIS	P01_ SMTDIS	P00_ SMTDIS

GPM8F3733A/GPM8F3717A/GPM8F3709A

Addr	Function	Key Code	Reset Value	7	6	5	4	3	2	1	0
					SMTDIS		SMTDIS	SMTDIS	SMTDIS	SMTDIS	SMTDIS
0xAD	P0_SR		0x00	P07_SR	P06_SR	P05_SR	P04_SR	P03_SR	P02_SR	P01_SR	P00_SR
0xAE	P0_HS	8F/32/52	0x00	P07_HS	P06_HS	P05_HS	P04_HS	P03_HS	P02_HS	P01_HS	P00_HS
0xAF	P0_HSBIAIS		0x00	--	--	--	--	--	--	SRI	ENI
0xB0	P3		0xFF	--	--	--	P34	P33	P32	P31	P30
0xB2	P3_PU		0x00	--	--	--	P34_PU	P33_PU	P32_PU	P31_PU	P30_PU
0xB3	P3_PD		0x00	--	--	--	P34_PD	P33_PD	P32_PD	P31_PD	P30_PD
0xB4	P3_SMTDIS		0x00	--	--	--	P34_SMTDIS	P33_SMTDIS	P32_SMTDIS	P31_SMTDIS	P30_SMTDIS
0xB5	P3_SR		0x00	--	--	--	P34_SR	P33_SR	P32_SR	P31_SR	P30_SR
0xB6	TMBL		0x00	Timer B value - low byte							
0xB7	TMBH		0x00	Timer B value - high byte							
0xB8	IP		0x00	--	PCCP	PT2	PS0	PT1	PX1	PT0	PX0
0xBA	P1_PU		0x00	P17_PU	P16_PU	P15_PU	P14_PU	P13_PU	P12_PU	P11_PU	P10_PU
0xBB	P1_PD		0x00	P17_PD	P16_PD	P15_PD	P14_PD	P13_PD	P12_PD	P11_PD	P10_PD
0xBC	P1_SMTDIS		0x00	P17_SMTDIS	P16_SMTDIS	P15_SMTDIS	P14_SMTDIS	P13_SMTDIS	P12_SMTDIS	P11_SMTDIS	P10_SMTDIS
0xBD	P1_SR		0x00	P17_SR	P16_SR	P15_SR	P14_SR	P13_SR	P12_SR	P11_SR	P10_SR
0xBE	TMB_LOADL		0x00	TMB_LOADL[7:0]							
0xBF	TMB_LOADH		0x00	TMB_LOADH[7:0]							
0xC0	SCON1		0x00	SM10	SM11	SM12	REN1	TB18	RB18	TI1	RI1
0xC1	SBUF1		0x00	UART 1 buffer							
0xC2	CCL1		0x00	Timer 2 ccl capture low byte							
0xC3	CCH1		0x00	Timer 2 ccl capture high byte							
0xC4	CCL2		0x00	Timer 2 cc2 capture low byte							
0xC5	CCH2		0x00	Timer 2 cc2 capture high byte							
0xC6	CCL3		0x00	Timer 2 cc3 capture low byte							
0xC7	CCH3		0x00	Timer 2 cc3 capture high byte							
0xC8	T2CON		0x00	T2PS	I3FR	--	--	T2R	T2CM	--	T2I
0xC9	T2IF		0x00	CCF3	CCF2	CCF1	CCF0	--	--	--	TF2
0xCA	CRCL		0x00	CRC register - Low byte							
0xCB	CRCH		0x00	CRC register - High Byte							
0xCC	TL2		0x00	Timer 2 Load value - low byte							
0xCD	TH2		0x00	Timer 2 Load value - high byte							
0xCE	CCEN		0x00	CMH3	CML3	CMH2	CML2	CMH1	CML1	CMH0	CML0
0xCF	I2C_DEBOUNCE		0x00	--	--	I2C de-bounce count[5:0]					
0xD0	PSW		0x00	CY	AC	--	RS1	RS0	OV	--	P
0xD1	I2C_CTRL		0x20	MST_STR	MST_STP	MST_NACK	MODE	I2C_CLK_SEL[1:0]		I2C_TRIG	I2C_EN
0xD2	I2C_STS		0x00	SLV_DID_OK	SLV_DAT_0K	SLV_STP_OK	ERR_SDID_IE	I2C_IF_SEL	I2C_INT_EN	NO_ACK	TS_DONE
0xD3	I2C_DID		0x01	DEV_ID[6:0]							

GPM8F3733A/GPM8F3717A/GPM8F3709A

Addr	Function	Key Code	Reset Value	7	6	5	4	3	2	1	0
0xD4	I2C_DATA		0x00	I2C_DATA[7:0]							
0xD8	WDCON	AA, 55	0x00	--	--	--	--	WDIF	--	EWT	RWT
0xD9	CTS_CTRL0		0x00	VTOPSEL[1:0]		AUTO_ADC	VTOP_EN	TRIEN	CTS_ST	CTS_EN[1:0]	
0xDA	CTS_CTRL1		0x00	CTC_DUTY	CTCK_SEL[2:0]			CTC_CNT	--	CTC_RSEL[1:0]	
0xDB	CTS_CTRL2		0x00	LPFSEL[1:0]		VRSEL	DLSEL[1:0]		ISEL[2:0]		
0xDC	CTS_CTRL3		0x00	CHSEL[7:0]							
0xDD	CTS_CTRL4		0x00	CHSEL[15:8]							
0xDE	CTS_CTRL5		0x00	CHSEL[23:16]							
0xDF	CTS_CTRL6		0x00	CHSEL[25:24]	TMBIP	TMAIP	TMB_PLOAD_EN	TMB_EN	TMA_PLOAD_EN	TMA_EN	
0xE0	ACC		0x00	ACC register							
0xE1	TMAL		0x00	Timer A value - low byte							
0xE2	TMAH		0x00	Timer A value - high byte							
0xE3	TMA_PLOADL		0x00	TMA_PLOADL[7:0]							
0xE4	TMA_PLOADH		0x00	TMA_PLOADH[7:0]							
0xE5	TMB_CCPL		0x00	TMB_CCPL[7:0]							
0xE6	TMB_CCPH		0x00	TMB_CCPH[7:0]							
0xE8	EIE		0x00	EKBI	--	EWDI	--	--	EINT4	EINT3	EINT2
0xEB	KEYCODE		0x00	KEYCODE[7:0]							
0xEC	FLASHCON	8F/32/51	0x00	--	--	--	--	--	--	P_ERASE	PROG
0xED	FL_LEVEL	8F/32/51	0x00	--	--	--	FLASH_LEVEL[4:0]				
0xEE	EXIPOL		0x00	DEBOUNCE_TIME[1:0]		--	INT4POL	INT3POL	INT2POL	INT1POL	INT0POL
0xEF	EXIMOD		0x00	--	--	--	INT4MOD	INT3MOD	INT2MOD	INT1MOD	INT0MOD
0xF0	B		0x00	B register							
0xF1	ADC_CTRL0		0x00	ADC_INTEN	ADC_SH_CYCLE[1:0]		ADC_CLK_SEL[2:0]		--	ADCIP	
0xF2	ADC_CTRL1		0x00	ADC_START	--	--	ADC_INTF	ADC_CH_SEL[3:0]			
0xF3	ADC_DATA_L		0x00	ADC output data - low nibble byte				--	--	--	--
0xF4	ADC_DATA_H		0x00	ADC output data - high byte							
0xF5	CTS_CTRL7		0x00	MUPORTSEL[2:0]			MUPINSEL[2:0]			MUTPOL	MUPORTSEL[2:0]
0xF6	CTS_CTRL8		0x00	MUTDEL_SEL[3:0]				--			TSCGF
0xF7	CTS_CTRL9		0x00		TOUCHF	TMBOVF	TMAOVF	TOUCHIP	TOUCHIE	TMBIE	TMAIE
0xF8	EIP		0x00	PKB	PLVD	PWDI	PSPI	PI2C	PINT4	PINT3	PINT2
0xF9	SYS_CTRL0	8F/32/50	0x00	CLK_DIV_SEL	CLK_DIV[1:0]		--	--	--	--	
0xFB	SYS_CTRL2	8F/32/50	0x01	HALT_I_SEL[2:0]			HALT_EN	VREG_SEL[1:0]		IOSC16K_EN	--
0xFC	SYS_CTRL3	8F/32/	0x01	--	LVD_INT_E	LVD_SEL[1:0]		LVD_EN	LVR_SEL[1:0]		LVR_EN

Addr	Function	Key Code	Reset Value	7	6	5	4	3	2	1	0
		50			N						
0xFD	SYS_CTRL4	8F/32/50	0x00	--	WDOG_RST	SW_RST_EN	LVR_RST	LVD_INTF	LVD_STS	ADDR_ERR	ERR_WR
0xFE	SYS_CTRL5	8F/32/50	0x84	--	--	--	TCH_CKEN	SPI_CKEN	UART_CKEN	I2C_CKEN	ADC_CKEN
0xFF	SYS_CTRL6	8F/32/50	0x02	UART0_IF_SEL	UART1_IF_EN	UART0_IF_EN	TO1_CK_SEL		AERR_RSTEN	WDOG_CKEN	--

5.4. 省电模式

5.4.1. 介绍

虽然 GPM8F3733A / GPM8F3717A / GPM8F3709A 是一颗具有强大性能的高速微控制器，但它仍然提供了具有两种先进省电模式的电源管理单元 (PMU)，分别为 HALT 模式和 STOP 模式。当系统不需要工作时，为了减少耗电，就可采用 STOP 模式。关于这两种模式的更多信息，请阅读以下 2 个章节。

5.4.2. HALT 模式

HALT 模式是通过关闭提供给微控制器的时钟来减少耗电的，这样会导致 MCU 停止执行接下来的指令。设置 HALT_EN(SYS_CTRL[2]) 可以进入 HALT 模式。在该模式下，只有低速时钟打开，所以除了看门狗定时器外，其他时钟中断如内部定时器、串行端口都不会工作；HALT 模式下可以通过看门狗中断和其他非时钟的外部中断和键盘中断退出。

5.4.3. STOP 模式

在 STOP 模式下，微控制器最省电。通过关闭 SYSCLK 时钟，芯片可以进入一个全静止状态。整个系统都不工作，定时器停止，也没有任何串行通信。设置了 STOP 位后，处理器将延缓执行指令。退出 STOP 模式有以下几种方式：

采用 INT0~INT4 这类与时钟无关的外部中断或者键盘/按键扫描 IO。当导致退出 STOP 模式的中断发生时，处理器会读取该中断的中断向量并重新开始工作。当执行完中断服务程序后，RETI 指令立即将程序返回到调用 STOP 模式指令的下一条指令处。

当 INT0~INT4 和按键扫描 IO 用作唤醒源时，必须先设置 WKUEN 寄存器，如表 5-20 WKUEN 寄存器所示。这样，处理器将执行以下操作。此外，为了降低系统在正常模式下运行时的功耗。用户可以通过寄存器设置独立的关闭未使用的块的时钟。相关寄存器如下所示。

	系统时钟	外设时钟	低速时钟	唤醒源	唤醒后
RUN 模式	寄存器设置	寄存器设置	寄存器设置	--	--
HALT 模式	关	开	开	1. 所有的唤醒源	执行下一条指令
STOP 模式	关	关	关	1. 外部中断源 2. 键盘扫描中断唤醒源	执行下一条指令

表 5-17 三种运行模式

PCON			地址：0x87		电源配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SMOD0	SMOD1	--	PWE	--	--	STOP	--
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7	SMOD0	R/W	当时钟源是 Timer1 时, UART 的波特率位	
6	SMOD1	R/W	当时钟源是 Timer1 时, UART 的波特率位	
5	---	R/W	预留	
4	PWE	R/W	编程使能位 (PWE) 0: 在 MOVX 指令期间, 禁止 Flash 写有效 1: 在 MOVX 指令期间, 使能 Flash 写有效	
3: 2	---	R/W	预留	
1	STOP	R/W	STOP 模式使能位 0: 禁止 1: 使能	
0	---	R/W	预留	

表 5- 18 PCON 寄存器

SYS_CTRL2			地址: 0xFB		系统控制 2 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	HALT_I_SEL[2:0]			HALT_EN	VREG_SEL[1:0]		IOSC16K_EN	---
默认	0	0	0	0	0	0	0	1
Key Code	0x8F, 0x32, 0x50							

Bit	功能	功能	描述	条件																														
7:5	HALT_I_SEL	R/W	在 halt 模式，校准器操作电流选择 <table><tr><th>HALT_EN</th><th>HALT_I_SEL[2:0]</th><th>I_{max} of VCC_18</th></tr><tr><td>0</td><td>XXX</td><td>30mA</td></tr><tr><td>1</td><td>000</td><td>3uA</td></tr><tr><td>1</td><td>001</td><td>4uA</td></tr><tr><td>1</td><td>010</td><td>5uA</td></tr><tr><td>1</td><td>011</td><td>6uA</td></tr><tr><td>1</td><td>100</td><td>7uA</td></tr><tr><td>1</td><td>101</td><td>8uA</td></tr><tr><td>1</td><td>110</td><td>10uA</td></tr><tr><td>1</td><td>111</td><td>12uA</td></tr></table>	HALT_EN	HALT_I_SEL[2:0]	I _{max} of VCC_18	0	XXX	30mA	1	000	3uA	1	001	4uA	1	010	5uA	1	011	6uA	1	100	7uA	1	101	8uA	1	110	10uA	1	111	12uA	
HALT_EN	HALT_I_SEL[2:0]	I _{max} of VCC_18																																
0	XXX	30mA																																
1	000	3uA																																
1	001	4uA																																
1	010	5uA																																
1	011	6uA																																
1	100	7uA																																
1	101	8uA																																
1	110	10uA																																
1	111	12uA																																
4	HALT_EN	R/W	Halt 模式使能信号. 0 : 当 PCON[1] 设置为 1，系统在 stop 模式下运行 1 : 当 PCON[1] 设置为 1，系统在 halt 模式下运行																															
3:2	VREG_SEL	R/W	校准器输出电源域选择位 <table><tr><th>VREG_SEL[1:0]</th><th>Voltage of VREG</th></tr><tr><td>00</td><td>2.0V</td></tr><tr><td>01</td><td>prohibit</td></tr><tr><td>10</td><td>1.9V</td></tr><tr><td>11</td><td>2.1V</td></tr></table>	VREG_SEL[1:0]	Voltage of VREG	00	2.0V	01	prohibit	10	1.9V	11	2.1V																					
VREG_SEL[1:0]	Voltage of VREG																																	
00	2.0V																																	
01	prohibit																																	
10	1.9V																																	
11	2.1V																																	
1	IOSC16K_EN	R/W	内部 16K 振荡器使能位 0 : 禁止 1 : 使能																															
0	---	R/W	保留																															

表 5- 19 SYS_CTRL2 寄存器

WKUEN			地址：0x97		唤醒使能寄存器			
Bit	7	6	5	4	3	2	1	0
功能	KB_WKUEN	WD_WKUEN	---	INT4_WKUEN	INT3_WKUEN	INT2_WKUEN	INT1_WKUEN	INT0_WKUEN
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	KB_WKUEN	R/W	键盘/按键扫描唤醒使能控制	
6	WD_WKUEN	R/W	看门狗唤醒使能控制	
5	---	R/W	预留	
4	INT4_WKUEN	R/W	INT4 唤醒使能控制，高有效	
3	INT3_WKUEN	R/W	INT3 唤醒使能控制，高有效	
2	INT2_WKUEN	R/W	INT2 唤醒使能控制，高有效	
1	INT1_WKUEN	R/W	INT1 唤醒使能控制，高有效	
0	INT0_WKUEN	R/W	INT0 唤醒使能控制，高有效	

表 5- 20 WKUEN 寄存器

SYS_CTRL5			地址：0xFE		系统控制-5 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	---	---	---	TCH_CKEN	SPI_CKEN	UART_CKEN	I2C_CKEN	ADC_CKEN
默认	1	0	0	0	0	1	0	0
Key Code	0x8F, 0x32, 0x50							

Bit	功能	类型	描述	条件
7:5	---	R/W	保留	
4	TCH_CKEN	R/W	Touch 控制器时钟使能信号 0：禁止 1：使能	
3	SPI_CKEN	R/W	SPI 控制器时钟使能信号 0：禁止 1：使能	
2	UART_CKEN	R/W	USRT 控制器时钟使能信号 0：禁止 1：使能	
1	I2C_CKEN	R/W	I2C 控制器时钟使能信号 0：禁止 1：使能	
0	ADC_CKEN	R/W	ADC 控制器时钟使能信号 0：禁止 1：使能	

表 5-21 SYS_CTRL5 寄存器

GPM8F3733A/GPM8F3717A/GPM8F3709A

SYS_CTRL6			地址：0xFF		系统控制-6 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	UART0_IF_SEL	UART1_IF_EN	UART0_IF_EN	T01_CK_SEL	--	AERR_RSTEN	WD0G_CKEN	--
默认	0	0	0	0	0	0	1	0
Key Code	0x8F, 0x32, 0x50							

Bit	功能	类型	描述	条件															
7	UART0_IF_SEL	R/W	UART IO 接口选择信号 <table><tr><td>UART0_IF_SEL</td><td>UART_TX</td><td>UART_RX</td></tr><tr><td>0</td><td>P31</td><td>P30</td></tr><tr><td>1</td><td>P12</td><td>P10</td></tr></table>	UART0_IF_SEL	UART_TX	UART_RX	0	P31	P30	1	P12	P10							
UART0_IF_SEL	UART_TX	UART_RX																	
0	P31	P30																	
1	P12	P10																	
6	UART1_IF_EN	R/W	UART1 接口使能信号 0 ： 禁止 1 ： 使能																
5	UART0_IF_EN	R/W	UART0 接口使能信号 0 ： 禁止 1 ： 使能																
4	T01_CK_SEL	R/W	Timer0/1 时钟源选择信号. 与 TOM/T1M of CKCON(0x8E) 结合使用 <table><tr><td>TOM /T1M</td><td>T01_CK_SEL</td><td>Timer0/1 Clock</td></tr><tr><td>0</td><td>0</td><td>System Clock / 8</td></tr><tr><td>0</td><td>1</td><td>System Clock / 2</td></tr><tr><td>1</td><td>0</td><td>System Clock / 4</td></tr><tr><td>1</td><td>1</td><td>System Clock / 1</td></tr></table>	TOM /T1M	T01_CK_SEL	Timer0/1 Clock	0	0	System Clock / 8	0	1	System Clock / 2	1	0	System Clock / 4	1	1	System Clock / 1	
TOM /T1M	T01_CK_SEL	Timer0/1 Clock																	
0	0	System Clock / 8																	
0	1	System Clock / 2																	
1	0	System Clock / 4																	
1	1	System Clock / 1																	
3	--	R/W	保留																
2	AERR_RSTEN	R/W	Flash 地址超出范围复位使能 0 ： 禁止 1 ： 使能																
1	WD0G_CKEN	R/W	看门狗控制器时钟使能控制位 0 ： 禁止 1 ： 使能																
0	--	R/W	保留																

表 5-2 SYS_CTRL6 寄存器

5.5. 中断系统

GPM8F3733A / GPM8F3717A / GPM8F3709A 提供了 5/5/4 个外部中断源、16 个内部中断源和 14/14/10 个按键改变中断源。每一个外部中断源都可以单独的设置使能或者禁用、边沿或者电平触发以及触发极性。当按键输入引脚有任何的逻辑改变，键盘改变中断就会发生，这些按键输入引脚需要用户先通过 KBEN_P0(0xA6)和 KBEN_P2(0xA7)来选择。当设置 EIE(0xE8)的 EKBI 位之前，需要先读取 P0 或者 P2 的状态。具有两个中断优先级，在每个系统时钟的上升沿来采样中断请求，通过设置或清除 SFR 中相应的位，每一个中断源可以单独的设置使能或者禁用，IE 寄存器中包含了全局中断系统禁用（0）/使能位（1），该位称作 EA。通常来说，一旦中断事件发生，就会设置相应的标志位。中断标志相关的寄存器如下所示。

如果相关的中断控制位设置为使能中断，就会产生一个中断请求信号，接着 CPU 就会执行服务程序。如果相关的中断控制位不使能，相应标志位还是会被设置，但不会产生中断请求信号。为了防止程序在中断服务程序中锁死，必须清除中断服务程序里的中断标志位。对于任何指令来说，在前一条指令执行期间，中断挂起。在进入中断服务程序之前，系统保存当前的 PC 地址到堆栈最顶部，然后跳转到相应的向量开始执行中断服务。在完成了中断服务程序后，系统从堆栈的顶部取回返回的 PC 地址并接着执行接下来的指令。更多信息请参考相关章节。

GPM8F3733A/GPM8F3717A/GPM8F3709A

➤ GPM8F3733A / GPM8F3717A

中断标志	功能	触发类型	中断标志清除	向量	向量序号	优先级
IE0	设备管脚 INT 0	上升沿/下降沿/低电平/高电平	硬件清除	0x03	0	1
TF0	内部 Timer 0	-	硬件清除	0x0B	1	2
IE1	设备管脚 INT 1	上升沿/下降沿/低电平/高电平	硬件清除	0x13	2	3
TF1	内部 Timer 1	-	硬件清除	0x1B	3	4
TOUCH	TOUCH	-	软件清除 (写 0)	0x23	4	5
ADC	ADC	-	软件清除 (写 0)	0x2B	5	6
TMAIF	内部 Timer A	-	软件清除 (写 0)	0x33	6	7
TMBIF	内部 Timer B	-	软件清除 (写 0)	0x3B	7	8
T2IF	内部 Timer 2	-	软件清除 (写 0)	0x43	8	9
KBIF/SPIF	键盘按键扫描管脚/SPI	-	软件清除 (写 0)	0x4B	9	10
I2CIF	I2C	-	软件清除 (写 0)	0x53	10	11
LVDF	低压检测	-	软件清除 (写 0)	0x5B	11	12
TI0&RI0 TI1&RI1	内部 UART0/1	-	软件清除 (写 0)	0x63	12	13
INT2/3/4F	设备管脚 INT 2/3/4	上升沿/下降沿/低电平/高电平	软件清除 (写 0)	0x6B	13	14
CCP0/1/2/3	Timer 2 捕捉	-	软件清除 (写 0)	0x73	14	15
WDIF	内部看门狗	-	软件清除 (写 0)	0x7B	15	16

➤ GPM8F3709A

中断标志	功能	触发类型	中断标志清除	向量	向量序号	优先级
-	-	-	-	0x03	0	1
TF0	内部 Timer 0	-	硬件清除	0x0B	1	2
IE1	设备管脚 INT 1	上升沿/下降沿/低电平/高电平	硬件清除	0x13	2	3
TF1	内部 Timer 1	-	硬件清除	0x1B	3	4
TOUCH	TOUCH	-	软件清除 (写 0)	0x23	4	5
ADC	ADC	-	软件清除 (写 0)	0x2B	5	6
TMAIF	内部 Timer A	-	软件清除 (写 0)	0x33	6	7
TMBIF	内部 Timer B	-	软件清除 (写 0)	0x3B	7	8
T2IF	内部 Timer 2	-	软件清除 (写 0)	0x43	8	9
KBIF/SPIF	键盘按键扫描管脚/SPI	低电平/高电平	软件清除 (写 0)	0x4B	9	10
I2CIF	内部 I2C	-	软件清除 (写 0)	0x53	10	11
LVDF	低压检测	-	软件清除 (写 0)	0x5B	11	12
TI0&RI0 TI1&RI1	内部 UART0/1	-	软件清除 (写 0)	0x63	12	13
INT2/3/4	设备管脚 INT 2/3/4	上升沿/下降沿/低电平/高电平	软件清除 (写 0)	0x6B	13	14
CCP0/1/2/3	Timer 2 捕捉	-	软件清除 (写 0)	0x73	14	15
WDIF	内部看门狗	-	软件清除 (写 0)	0x7B	15	16

GPM8F3733A/GPM8F3717A/GPM8F3709A

IP			地址：0xB8		中断优先级寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	PCCP	PT2	PS0	PT1	PX1	PT0	PX0
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	--	R/W	预留	
6	PCCP	R/W	Timer2 CC 优先级控制（1：高优先级）	
5	PT2	R/W	Timer2 优先级控制（1：高优先级）	
4	PS0	R/W	UART 优先级控制（1：高优先级）	
3	PT1	R/W	Timer1 优先级控制（1：高优先级）	
2	PX1	R/W	INT1 优先级控制（1：高优先级）	
1	PT0	R/W	Timer0 优先级控制（1：高优先级）	
0	PX0	R/W	INT0 优先级控制（1：高优先级）	

表 5- 23 IP 寄存器

EIP			地址：0xF8		扩展中断优先级控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	PKB	PLVD	PWDI	PSPI	PI2C	PINT4	PINT3	PINT2
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	PKB	R/W	键盘中断优先级控制（1：高优先级）	
6	PLVD	R/W	LVD 优先级控制（1：高优先级）	
5	PWDI	R/W	看门狗优先级控制（1：高优先级）	
4	PSPI	R/W	SPI 优先级控制（1：高优先级）	
3	PI2C	R/W	I2C 优先级控制（1：高优先级）	
2	PINT4	R/W	INT4 优先级控制（1：高优先级）	
1	PINT3	R/W	INT3 优先级控制（1：高优先级）	
0	PINT2	R/W	INT2 优先级控制（1：高优先级）	

表 5- 24 EIP 寄存器

CTS_CTRL6			地址：0xDF		电容触摸传感器控制寄存器 6			
Bit	7	6	5	4	3	2	1	0
功能	CHSEL[25:24]		TMBIP	TMAIP	TMB_PLOAD_EN	TMB_EN	TMA_PLOAD_EN	TMA_EN
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:6	CHSEL[25:24]	R/W	触摸通道选择	
5	TMBIP	R/W	TimerB 优先级控制（1：高优先级）	
4	TMAIP	R/W	TimerA 优先级控制（1：高优先级）	
3	TMB_PLOAD_EN	R/W	TimerB 自动重载功能使能位	
2	TMB_EN		TimerB 使能位	

Bit	功能	类型	描述	条件
1	TMA_PLOAD_EN		TimerA 自动重载功能使能位	
0	TMA_EN		TimerA 使能位	

表 5- 25 CTS_CTRL6 寄存器

CTS_CTRL9			地址: 0xF7		电容触摸传感器控制寄存器 9			
Bit	7	6	5	4	3	2	1	0
功能	--	TOUCHF	TMBOVF	TMAOVF	TOUCHIP	TOUCHIE	TMBIE	TMAIE
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	--	R/W	预留	
6	TOUCHF	R/W	TOUCH 中断标志	
5	TMBOVF	R/W	TimerB 溢出标志	
4	TMAOVF	R/W	TimerA 溢出标志	
3	TOUCHIP	R/W	TOUCH 优先级控制 (1: 高优先级)	
2	TOUCHIE	R/W	使能 TOUCH 中断	
1	TMBIE	R/W	使能 TimerB 中断	
0	TMAIE	R/W	使能 TimerA 中断	

表 5- 26 CTS_CTRL9 寄存器

IE			地址: 0xA8		中断使能寄存器			
Bit	7	6	5	4	3	2	1	0
功能	EA	ECCP	ET2	ES0	ET1	EX1	ET0	EX0
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	EA	R/W	使能全局中断	
6	ECCP	R/W	使能 Timer2 比较/捕捉中断	
5	ET2	R/W	使能 Timer2 中断	
4	ES0	R/W	使能 UART0/1 中断	
3	ET1	R/W	使能 Timer1 中断	
2	EX1	R/W	使能 INT1 中断	
1	ET0	R/W	使能 Timer0 中断	
0	EX0	R/W	使能 INTO 中断	

表 5- 27 IE 寄存器

EIE			地址: 0xE8		扩展中断使能寄存器			
Bit	7	6	5	4	3	2	1	0
功能	EKBI	-	EWDI	--	--	EINT4	EINT3	EINT2
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7	EKBI	R/W	使能键盘脚位中断	
6	--	R/W	预留	
5	EWDI	R/W	使能看门狗中断	
4:3	--	R/W	预留	
2	EINT4	R/W	使能 INT4 中断	
1	EINT3	R/W	使能 INT3 中断	
0	EINT2	R/W	使能 INT2 中断	

表 5- 28 EIE 寄存器

TCON			地址: 0x88		Timer0/1 配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TF1	TR1	TF0	TR0	IE1	--	IE0	--
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	TF1	R/W	Timer1 中断（溢出）标志	
6	TR1	R/W	Timer1 运行控制位 0: 禁用 1: 使能	
5	TF0	R/W	Timer0 中断（溢出）标志	
4	TR0	R/W	Timer0 运行控制位 0: 禁用 1: 使能	
3	IE1	R/W	INT1 中断标志	
2	--	R/W	预留	
1	IE0	R/W	INT0 中断标志	
0	--	R/W	预留	

表 5- 29 TCON 寄存器

EIF			地址: 0x91		扩展中断标志			
Bit	7	6	5	4	3	2	1	0
功能	KBIF	--	--	SPIF	--	INT4F	INT3F	INT2F
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	KBIF	R/W	键盘脚（P0, P2）改变	
6:5	--	R/W	预留	
4	SPIF	R/W	SPI 中断标志	
3	--	R/W	预留	
2	INT4F	R/W	INT4 中断标志	
1	INT3F	R/W	INT3 中断标志	
0	INT2F	R/W	INT2 中断标志	

表 5- 30 EIF 寄存器

GPM8F3733A/GPM8F3717A/GPM8F3709A

T2IF			地址: 0xC9		Timer 2 中断标志寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CCF3	CCF2	CCF1	CCF0	--	--	--	TF2
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	CCF3	R/W	比较 3/捕捉 3 标志, 软件清零	
6	CCF2	R/W	比较 2/捕捉 2 标志, 软件清零	
5	CCF1	R/W	比较 1/捕捉 1 标志, 软件清零	
4	CCF0	R/W	比较 0/捕捉 0 标志, 软件清零	
3:1	--	R/W	预留	
0	TF2	R/W	Timer 2 溢出标志位, 软件清零	

表 5- 31 T2IF 寄存器

EXIPOL			地址: 0xEE		扩展中断极性控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	DEBOUNCE_TIME[1: 0]		--	INT4POL	INT3POL	INT2POL	INT1POL	INT0POL
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	DEBOUNCE_TIME[1: 0]	R/W	外部中断去抖时间选择信号 DEBOUNCE_TIME[1 :0] 时钟周期 00 0 01 16 10 32 11 64	
5	--	R/W	预留	
4	INT4POL	R/W	INT4极性选择 0: 下降沿/低电平; 1: 上升沿/高电平	
3	INT3POL	R/W	INT3极性选择 0: 下降沿/低电平; 1: 上升沿/高电平	
2	INT2POL	R/W	INT2极性选择 0: 下降沿/低电平; 1: 上升沿/高电平	
1	INT1POL	R/W	INT1极性选择 0: 下降沿/低电平; 1: 上升沿/高电平	
0	INT0POL	R/W	INT0极性选择 0: 下降沿/低电平; 1: 上升沿/高电平	

表 5- 32 EXIPOL 寄存器

EXIMODE			地址: 0xEF		扩展中断模式控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	INT4MOD	INT3MOD	INT2MOD	INT1MOD	INT0MOD
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:5	--	R/W	预留	
4	INT4MOD	R/W	INT4 检测模式 0: 边沿检测; 1: 电平检测	
3	INT3MOD	R/W	INT3 检测模式 0: 边沿检测; 1: 电平检测	
2	INT2MOD	R/W	INT2 检测模式 0: 边沿检测; 1: 电平检测	
1	INT1MOD	R/W	INT1 检测模式 0: 边沿检测; 1: 边沿和电平检测	
0	INT0MOD	R/W	INT0 检测模式 0: 边沿检测; 1: 电平检测	

表 5- 33 EXIMOD 寄存器

WDCON			地址: 0xD8		看门狗控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	--	WDIF	--	EWT	RWT
默认	0	0	0	0	0	0	0	0
Key Code	0xAA, 0x55							

Bit	功能	类型	描述	条件
7:4	--	R/W	预留	
3	WDIF	R/W	看门狗中断标志	
2	--	R/W	预留	
1	EWT	R/W	看门狗复位使能 0: 禁用; 1: 使能	
0	RWT	R/W	清除看门狗定时器 0: NA 无效; 1: Reset 清除	

表 5- 34 WDCON 寄存器

SCON0			地址: 0x98		UART0 控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SM00	SM01	SM02	RENO	TB08	RB08	TIO	RI0
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:6	SM0[1:0]	R/W	模式和波特率设置位	
5	SM02	R/W	使能多机通信功能	
4	RENO	R/W	使能串行接收	
3	TB08	R/W	模式 2 和模式 3 下第 9 个发送数据位	
2	RB08	R/W	在模式 0, 这一位无效。	

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
			在模式 1，如果 SM02=0，RB08 是停止位 在模式 2 和模式 3 下，是第 9 个接收的数据位	
1	TI0	R/W	UART0 发送中断标志	
0	RI0	R/W	UART0 接收中断标志	

表 5- 35 SCON0 寄存器

SCON1			地址：0xC0		UART1 控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SM10	SM11	SM12	REN1	TB18	RB18	TI1	RI1
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:6	SM1[1:0]	R/W	模式和波特率设置位	
5	SM12	R/W	使能多机通信功能	
4	REN1	R/W	使能串行接收	
3	TB18	R/W	模式 2 和模式 3 下第 9 个发送数据位	
2	RB18	R/W	在模式 0，这一位无效。 在模式 1，如果 SM02=0，RB18 是停止位 在模式 2 和模式 3 下，是第 9 个接收的数据位	
1	TI1	R/W	UART1 发送中断标志	
0	RI1	R/W	UART1 接收中断标志	

表 5- 36 SCON1 寄存器

KBEN_P0			地址：0xA6		键盘脚 P0 使能寄存器			
Bit	7	6	5	4	3	2	1	0
功能	KBEN_P0[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	KBEN_P0[7:0]	R/W	P0 键盘脚位使能 0：设置KBEN_P0x功能禁用； 1：设置KBEN_P0x功能使能	

表 5- 37 KBEN_P0 寄存器

KBEN_P2			地址：0xA7		键盘脚 P2 使能寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	KBEN_P2[5:0]					
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:6	--	R/W	预留	
7:0	KBEN_P2[7:0]	R/W	P2键盘脚位使能 0: 设置KBEN_P2x功能禁用; 1: 设置KBEN_P2x功能使能	

表 5- 38 KBEN_P2 寄存器

ADC_CTRL0			地址: 0xF1		ADC 控制寄存器 0			
Bit	7	6	5	4	3	2	1	0
功能	ADC_INTEN	ADC_SH_CYCLE[1:0]		ADC_CLK_SEL[2:0]		--		ADCIP
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述		条件
7	ADC_INT_EN	R/W	ADC 中断使能控制位 0：禁止 1：使能		
6:5	ADO_SH_CYCLE	R/W	ADC 采样周期选择位		
			ADC_SH_CYCLE	周期 (ADC_CLK)	
			00	2	
			01	4	
			10	8	
			11	16	
4:2	ADC_CLK_SEL	R/W	ADC 时钟选择控制位		
			ADC_CLK_SEL	ADC_CLK	
			000	系统时钟/ 2	
			001	系统时钟 / 4	
			010	系统时钟 / 8	
			011	系统时钟 / 10	
			100	系统时钟 / 16	
			101	系统时钟 / 20	
			110	系统时钟 / 32	
			111	系统时钟 / 64	
1	---	R/W	保留		
0	ADCIP	R/W	ADC 优先级控制（1：高优先级）		

表 5- 39 ADC_CTRL0 寄存器

ADC_CTRL1			地址: 0xF2		ADC 控制寄存器 1			
bit	7	6	5	4	3	2	1	0
功能	ADC_START	--	--	ADC_INTF	ADC_CH_SEL[3:0]			
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	ADC_START	R/W	ADC 开始传输控制位 0: 空闲的 1: 开始传输	

Bit	功能	类型	描述	条件																				
6:5	-	R/W	保留																					
4	ADC_INTF	R/W	ADC 中断标志. 读: 0 : 空闲/ 正在转换 1 : 转换完成 写: 0 : 清楚 1 : 无影响																					
3:0	ADC_CH_SEL	R/W	ADC 通道选择控制位 <table><tr><td>ADC_CH_SEL</td><td></td></tr><tr><td>0000</td><td>ADC_CH0 (P1[0])</td></tr><tr><td>0001</td><td>ADC_CH1 (P1[1])</td></tr><tr><td>0010</td><td>ADC_CH2 (P1[2])</td></tr><tr><td>0011</td><td>ADC_CH3 (P1[3])</td></tr><tr><td>0100</td><td>ADC_CH4 (P1[4])</td></tr><tr><td>0101</td><td>ADC_CH5 (P1[5])</td></tr><tr><td>0110</td><td>ADC_CH6 (P1[6])</td></tr><tr><td>0111</td><td>ADC_CH7 (P1[7])</td></tr><tr><td>1000</td><td>ADC_CH8 (P2[4], Cs PAD)</td></tr></table>	ADC_CH_SEL		0000	ADC_CH0 (P1[0])	0001	ADC_CH1 (P1[1])	0010	ADC_CH2 (P1[2])	0011	ADC_CH3 (P1[3])	0100	ADC_CH4 (P1[4])	0101	ADC_CH5 (P1[5])	0110	ADC_CH6 (P1[6])	0111	ADC_CH7 (P1[7])	1000	ADC_CH8 (P2[4], Cs PAD)	
ADC_CH_SEL																								
0000	ADC_CH0 (P1[0])																							
0001	ADC_CH1 (P1[1])																							
0010	ADC_CH2 (P1[2])																							
0011	ADC_CH3 (P1[3])																							
0100	ADC_CH4 (P1[4])																							
0101	ADC_CH5 (P1[5])																							
0110	ADC_CH6 (P1[6])																							
0111	ADC_CH7 (P1[7])																							
1000	ADC_CH8 (P2[4], Cs PAD)																							

表 5- 40 ADC_CTRL1 寄存器

I2C_STS			地址: 0xD2		I2C 状态寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SLV_DID_OK	SLV_DAT_OK	SLV_STP_OK	ERR_SDID_IE	I2C_IF_SEL	I2C_INT_EN	NO_ACK	TS_DONE
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	SLV_DID_OK	R/W	从机模式下: 指明是否接到设备 ID 号。 读： 0 : 未有设备 ID 信号 1 : 有设备 ID 信号 写： 0 : 清零 1 : 无影响	
6	SLV_DAT_OK	R/W	从机模式下: 指明数据的接收和传输状态; 读： 0 : 数据正在传输或者空隙状态; 1 : 数据传输完成 写： 0 : 清零 1 : 无影响	
5	SLV_STP_OK	R/W	从机模式下: 指明是否接收到 STOP 信号; 读： 0 : 未有 STOP 信号产生; 1 : 有 STOP 信号产生; 写： 0 : 清零 1 : 无影响	

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
4	ERR_SDID_IE	R/W	从机模式设备 ID 错误中断使能. 0 : 禁止 1 : 使能	
3	I2C_IF_SEL	R/W	I2C 通讯接口选择信号 0 : P2_3 = SDA / P2_2 = SCK 1 : P3_3 = SDA / P3_2 = SCK	
2	I2C_INT_EN	R/W	I2C 中断使能控制位 0 : 禁止 1 : 使能	
1	NO_ACK	R/W	I2C 没有收到应答信号. 读: 0 : 应答信号 1 : 非应答信号 写 : 0 : 清零 1 : 无影响	
0	TS_DONE	R/W	I2C 传输完成标志位. 读: 0 : i2c 闲置或正在传输 1 : i2c 数据传输完成 写: 0 : 清零 1 : 无影响	

表 5- 41 I2C_STS 寄存器

SPI_STS			地址: 0x9B		SPI 状态寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SPI_INTEN	--	--	--	--	--	RX_DONE	TX_DONE
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	SPI_INTEN	R/W	SPI 中断使能 0 : 禁止 1 : 使能	
6:2	--	R/W	保留	
1	RX_DONE	R	从机模式下, SPI 完成数据的接收 0: 闲置 / 忙 1: 完成	
0	TX_DONE	R	主机模式下, SPI 完成数据的传输 0: 闲置 / 忙 1: 完成	

表 5- 42 SPI_STS 寄存器

SYS_CTRL3			地址: 0xFC		系统控制 3 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	LVD_INT_EN	LVD_SEL[1:0]		LVD_EN	LVR_SEL[1:0]		LVR_EN
默认	0	0	0	0	0	0	0	1
Key Code	0x8F, 0x32, 0x50							

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件										
7	--	R/W	保留											
6	LVD_INT_EN	R/W	LVD 中断使能 0：禁止 1：使能											
5:4	LVD_SEL[1:0]	R/W	LVD 电压选择位 <table><tr><td>LVD_SEL[1:0]</td><td>电压</td></tr><tr><td>00</td><td>2.1V</td></tr><tr><td>01</td><td>2.4V</td></tr><tr><td>10</td><td>2.9V</td></tr><tr><td>11</td><td>3.8V</td></tr></table>	LVD_SEL[1:0]	电压	00	2.1V	01	2.4V	10	2.9V	11	3.8V	
LVD_SEL[1:0]	电压													
00	2.1V													
01	2.4V													
10	2.9V													
11	3.8V													
3	LVD_EN	R/W	LVD 使能控制位 0：禁止 LVD 功能 1：使能 LVD 功能											
2:1	LVR_SEL[1:0]	R/W	LVR 电压选择位 <table><tr><td>LVR_SEL[1:0]</td><td>电压</td></tr><tr><td>00</td><td>1.9V</td></tr><tr><td>01</td><td>2.2V</td></tr><tr><td>10</td><td>2.7V</td></tr><tr><td>11</td><td>3.6V</td></tr></table>	LVR_SEL[1:0]	电压	00	1.9V	01	2.2V	10	2.7V	11	3.6V	
LVR_SEL[1:0]	电压													
00	1.9V													
01	2.2V													
10	2.7V													
11	3.6V													
0	LVR_EN	R/W	LVR 使能控制位 0：禁止 LVR 功能 1：使能 LVR 功能											

表 5- 44 SYS_CTRL3 寄存器

SYS_CTRL4			地址：0xFD		系统控制 4 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	WDOG_RST	SW_RST_EN	LVR_RST	LVD_INTF	LVD_STS	ADDR_ERR	ERR_WR
默认	0	0	0	0	0	0	0	0
Key Code	0x8F, 0x32, 0x50							

Bit	功能	类型	描述	条件
7	--	R	保留	
6	WDOG_RST	R/W	看门狗复位指示标志位 读： 0：未发生看门狗复位 1：发生看门狗复位 写： 0：清零 1：无影响	
5	SW_RST_EN	R/W	软件复位使能信号 0：禁止软件复位 1：使能软件复位	
4	LVR_RST	R/W	LVR 指示标志位 读： 0：未发生 LVR 复位 1：发生 LVR 复位 写： 0：清零 1：无影响	

Bit	功能	类型	描述	条件
3	LVD_INTF	R/W	LVD 中断标志位 读: 0 : 未发生 LVD 中断 1 : 发生 LVD 中断 写: 0 : 清零 1 : 无影响	
2	LVD_STS	R	LVD 状态标志位 0 : 未发生 LVD 事件 1 : 发生 LVD 事件	
1	ADDR_ERR	R/W	Flash 地址溢出标志位 读: 0 : Flash 地址未溢出 1 : Flash 地址溢出 写: 0 : 清零 1 : 无影响	
1	ERR_WR	R/W	Flash 非法编程/擦除标志. 编程/擦除超出 flash 空间或者在 flash 的 lock_level 范围内, 此标志位将会变为 1; 读: 0 : 有效的编程/擦除 1 : 非法编程/擦除 写: 0 : 清零 1 : 无影响	

表 5- 45 SYS_CTRL4 寄存器

5.6. 复位源

5.6.1. 介绍

GPM8F3733A/ GPM8F3717A/ GPM8F3709A 有 6 种类型的复位源, 包括上电复位 (POR)、低压复位 (LVR)、Pad 复位 (PAD_RST)、看门狗定时器复位 (WDT_RST)、软件复位 (S/W_RST)、Flash 错误复位 (FLASH_ERR_RST)。图 5- 5 复位源给出了每种复位源的示意图。

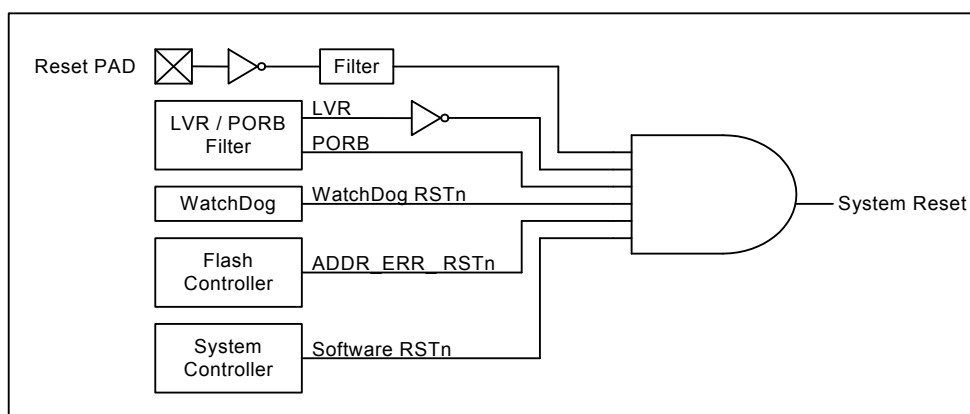


图 5- 5 复位源

5.6.2. 上电复位 (POR)

当 VDD 从 0V 上升时, POR 产生。VDD 上升到一个合适的电平 (~1.5V) 时, 上电复位电路启动上电时序。之后, 系统开始激活并以既定速度进行工作。POR 将复位整个芯片和寄存器。

5.6.3. 低电压复位(LVR)

当电源电压掉至指定的 LVR 触发电压以下时，片内的低压复位 (LVR) 电路将迫使系统进入复位状态。该功能使得 MCU 避免了工作在一个无效的工作电压范围。

通过配置相应的位，可以使能或者禁用低压复位功能。如果 LVR 使能，当芯片工作时，LVR 电路就会监控电源电压。通过设置 LVR_SEL[1:0] 位，LVR 电压可以配置为 1.9V、2.2V、2.7V、3.6V。当电压低于特定的水平时，电压系统就会复位并进入初始状态。

5.6.4. 低压检测(LVD)

通过监测 LVD 标志位，能够使软件及时发现电源电压可能会出现无效的情况。内建的电压检测电路控制 LVD 状态标志和中断标志。当电源电压低于 LVD 电压时，LVD 状态标志置位，当电源电压高于 LVD 电压时，LVD 状态标志被清除。当电源电压低于 LVD 电压时，LVD 中断标志置位，需要软件清零。通过设置 LVD_SEL[1:0] 位，LVD 电压可以设置为 2.1V、2.4V、2.9V、3.8V。

5.6.5. 管脚复位(PAD_RST)

GPM8F3733A/ GPM8F3717A/ GPM8F3709A 提供了一个外部引脚可使系统复位至初始状态。如图 5-6 Pad 复位电路所示，RESET 引脚为高电平有效。当 RESET= VDD 且超过 100us 时，系统将强制进入复位状态，从地址 0x0000 执行指令，所有的寄存器都进入默认状态。

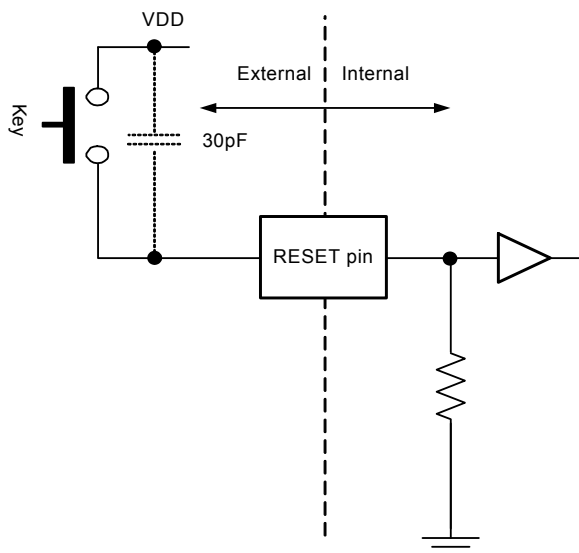


图 5- 6 Pad 复位电路

5.6.6. 看门狗复位(WDT_RST)

当 MCU 进入不确定状态且没有清除看门狗时，片内的看门狗电路就会使芯片进入复位状态。该功能防止 MCU 运行在异常情况下。通过设置 WDCON[1]，可以使能或禁用 WDT。在用户选择的最终值到来之前的任何时间，软件都可以设置复位看门狗定时器 RWT 位 (WDCON[0])。如果在超时之前置位 RWT 的话，看门狗定时器会重新开始计数，否则，看门狗将会复位 CPU。在软件置位后硬件会自动地清除 RWT。当发生复位时，看门狗定时器复位标志 (SYSCON_CTRL4[6]) 将自动置位以表示复位的原因，无论如何必须手动的用软件来清除该位。

WDCON 寄存器是一个时序访问寄存器，用来防止意外写入情况。在写入 WDCON 寄存器之前，0xAA 和 0x55 必须按正确地顺序写入 KEYCODE 寄存器，KEYCODE 位于 0xEB。读该寄存器是不受保护的。基于 16K 的时钟频率，看门狗提供四个时间选项。这四个选项是预先设定的时钟周期个数，由 CKCON[7:6] 的设置来决定。可以通过设置 CKCON[5] 来设置这四个选项是否是快速模式，当 CPU 从 HALT /STOP 模式中醒来时，软件必须要延时大约 100us 后再进入 HALT/STOP 模式，需要再次选择 16K 的 1T 时钟源，图 5- 7 看门狗示意图给出了看门狗模块的示意图。

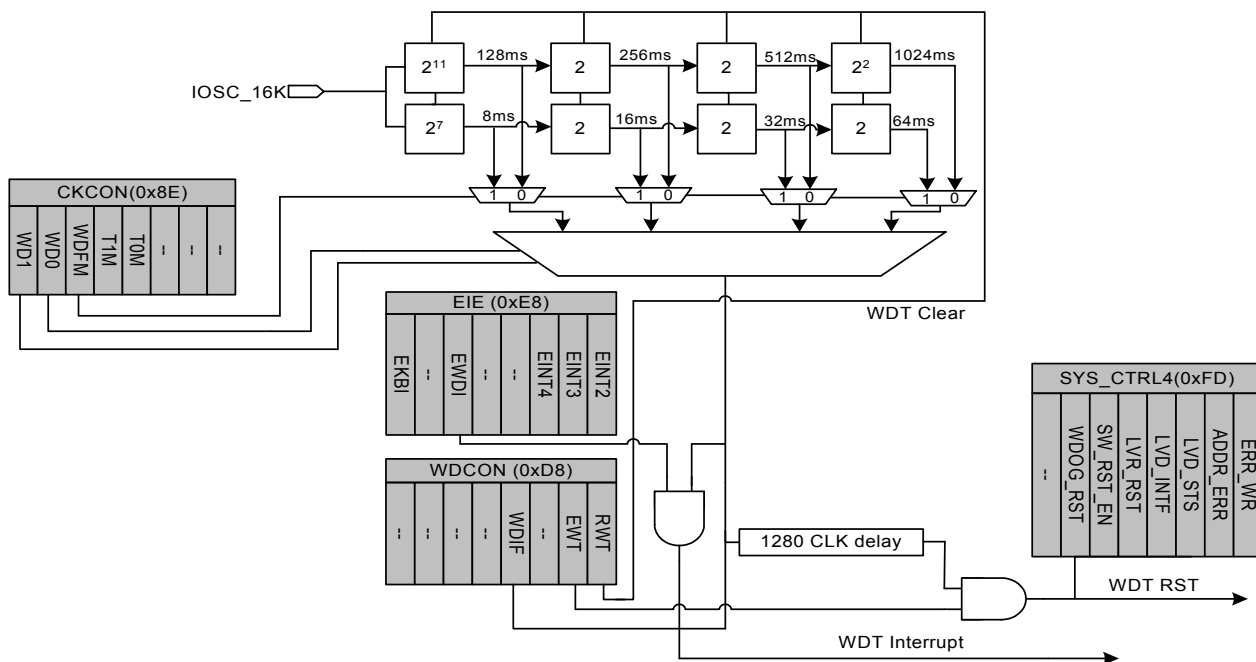


图 5-7 看门狗示意图

5.6.7. 其他复位源

其它复位源包括软件复位 (S/W_RST) 和 Flash 错误复位 (FLASH_ERR_RST)。将 SYS_CTRL4[5] 写为 1，且写入正确的 key code，就会发生软件复位，这些 key code 是 0x8f, 0x32, 0x50，时序没有关系，但在软件复位发生之前必须按顺序将 key code 写入。当 CPU 访问错误地址时，Flash 错误复位就会发生，可以通过 SYS_CTRL4[1] 看到相应的标志位。

SYS_CTRL4			地址：0xFD		系统控制 4 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	—	WDOG_RST	SW_RST_EN	LVR_RST	LVD_INTF	LVD_STS	ADDR_ERR	ERR_WR
默认	0	0	0	0	0	0	0	0
Key Code	0x8F, 0x32, 0x50							

Bit	功能	类型	描述	条件
7	—	R	保留	
6	WDOG_RST	R/W	看门狗复位指示标志位 读： 0：未发生看门狗复位 1：发生看门狗复位 写： 0：清零 1：无影响	
5	SW_RST_EN	R/W	软件复位使能信号 0：禁止软件复位 1：使能软件复位	
4	LVR_RST	R/W	LVR 指示标志位 读： 0：未发生 LVR 复位 1：发生 LVR 复位 写： 0：清零 1：无影响	

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
3	LVD_INTF	R/W	LVD 中断标志位 读: 0 : 未发生 LVD 中断 1 : 发生 LVD 中断 写: 0 : 清零 1 : 无影响	
2	LVD_STS	R	LVD 状态标志位 0 : 未发生 LVD 事件 1 : 发生 LVD 事件	
1	ADDR_ERR	R/W	Flash 地址溢出标志位 读: 0 : Flash 地址未溢出 1 : Flash 地址溢出 写: 0 : 清零 1 : 无影响	
1	ERR_WR	R/W	Flash 非法编程/擦除标志. 编程/擦除超出 flash 空间或者在 flash 的 lock_level 范围内, 此标志位将会变为 1; 读: 0 : 有效的编程/擦除 1 : 非法编程/擦除 写: 0 : 清零 1 : 无影响	

表 5- 46 SYS_CTRL4 寄存器

WDCON			地址: 0xD8		看门狗控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	--	WDIF	--	EWT	RWT
默认	0	0	0	0	0	0	0	0
Key code	0xAA, 0x55							

Bit	功能	类型	描述	条件
7:4	--	R/W	预留	
3	WDIF	R/W	看门狗中断标志	
2	--	R/W	预留	
1	EWT	R/W	看门狗复位使能标志 0: 禁用;1: 使能	
0	RWT	R/W	清除看门狗定时器 0: 无效;1: 清除	

表 5- 47 WDCON 寄存器

KEYCODE			地址: 0xEB		KEYCODE 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	KeyCode 寄存器							
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
0	KEYCODE[7:0]	R/W	KeyCode 寄存器	

注意：一些受保护的寄存器，在写入数据之前，必须先写入正确的 KeyCode 到 KEYCODE 寄存器。

表 5- 48 KEYCODE 寄存器

CKCON			地址：0x8E		时钟控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	WD1	WD0	WDFM	T1M	T0M	--	--	--
默认	0	0	0	0	0	1	1	1

Bit	功能	类型	描述	条件																				
7:6	WD[1:0]	R/W	看门狗超时选择位 如果 WDFM=0: <table><tr><td>WD[1:0]</td><td>超时</td></tr><tr><td>00</td><td>128ms</td></tr><tr><td>01</td><td>256ms</td></tr><tr><td>10</td><td>512ms</td></tr><tr><td>11</td><td>1024ms</td></tr></table> 如果 WDFM=1: <table><tr><td>WD[1:0]</td><td>超时</td></tr><tr><td>00</td><td>8ms</td></tr><tr><td>01</td><td>16ms</td></tr><tr><td>10</td><td>32ms</td></tr><tr><td>11</td><td>64ms</td></tr></table>	WD[1:0]	超时	00	128ms	01	256ms	10	512ms	11	1024ms	WD[1:0]	超时	00	8ms	01	16ms	10	32ms	11	64ms	
WD[1:0]	超时																							
00	128ms																							
01	256ms																							
10	512ms																							
11	1024ms																							
WD[1:0]	超时																							
00	8ms																							
01	16ms																							
10	32ms																							
11	64ms																							
5	WDFM	R/W	看门狗快模式选择位 0: 禁用看门狗快模式; 1: 使能看门狗快模式																					
4	T1M	R/W	系统时钟的除频选择, 用于驱动 Timer1 0: Timer1 使用系统时钟频率的二分频; 1: Timer1 使用系统时钟频率																					
3	T0M	R/W	系统时钟的除频选择, 用于驱动 Timer0 0 : Timer0 使用系统时钟频率的二分频; 1 : Timer0 使用系统时钟频率																					
2:0	---	---	预留																					

表 5- 49 CKCON 寄存器

5.7. 时钟源

GPM8F3733A / GPM8F3717A / GPM8F3709A 有一个 32MHz 的内部振荡器作为时钟源。可以通过系统时钟源的时钟分频器得到不同的频率，总共有 4 种分频选择。时钟源的示意图如图 5-8 时钟源示意图所示。需要注意的是外围设备的时钟在使用之前需要通过设置 SYS_CTRL5 和 SYS_CTRL6 来使能。

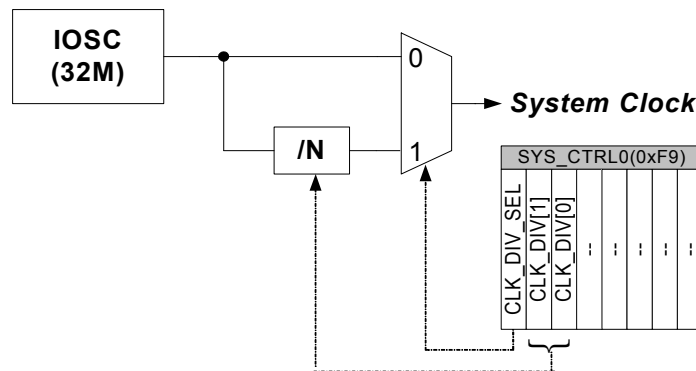


图 5-8 时钟源示意图

SYS_CTRL0			地址：0xF9		系统控制 0 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CLK_DIV_SEL	CLK_DIV[1:0]		--	--	--	--	--
默认	0	0	0	0	0	0	0	0
Key Code	0x8F, 0x32, 0x50							

Bit	功能	类型	描述										
7	CLK_DIV_SEL	R/W	系统时钟选择信号。 0：系统时钟 = 输入时钟源 1：系统时钟 = 输入时钟源/ N										
6:5	CLK_DIV[1 :0]	R/W	系统时钟分频器 <table><tr><td>CLK_DIV</td><td>系统分频</td></tr><tr><td>00</td><td>时钟源/ 2</td></tr><tr><td>01</td><td>时钟源/ 4</td></tr><tr><td>10</td><td>时钟源/ 8</td></tr><tr><td>11</td><td>时钟源/ 16</td></tr></table>	CLK_DIV	系统分频	00	时钟源/ 2	01	时钟源/ 4	10	时钟源/ 8	11	时钟源/ 16
CLK_DIV	系统分频												
00	时钟源/ 2												
01	时钟源/ 4												
10	时钟源/ 8												
11	时钟源/ 16												
4:0	--	R/W	保留										

表 5-22 SYS_CTRL0 寄存器

SYS_CTRL5			地址：0xFE		系统控制 5 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	TCH_CKEN	SPI_CKEN	UART_CKEN	I2C_CKEN	ADC_CKEN
默认	1	0	0	0	0	1	0	0
Key Code	0x8F, 0x32, 0x50							

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:5	—	R/W	保留	
4	TCH_CKEN	R/W	Touch 控制器时钟使能信号 0 : 禁止 1 : 使能	
3	SPI_CKEN	R/W	SPI controller clock enable signal. 0 : 禁止 1 : 使能	
2	UART_CKEN	R/W	USRT 控制器时钟使能信号 0 : 禁止 1 : 使能	
1	I2C_CKEN	R/W	I2C 控制器时钟使能信号 0 : 禁止 1 : 使能	
0	ADC_CKEN	R/W	ADC 控制器时钟使能信号 0 : 禁止 1 : 使能	

表 5- 50 SYS_CTRL5 寄存器

SYS_CTRL6			地址 : 0xFF		系统控制 6 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	UART0_IF_SEL	UART1_IF_EN	UART0_IF_EN	T01_CK_SEL	—	AERR_RSTEN	WDOG_CKEN	—
默认	0	0	0	0	0	0	1	0
Key Code	0x8F, 0x32 , 0x50							

Bit	功能	类型	描述	条件															
7	UART0_IF_SEL	R/W	UART IO 接口选择信号 <table><tr><td>UART0_IF_SEL</td><td>UART_TX</td><td>UART_RX</td></tr><tr><td>0</td><td>P31</td><td>P30</td></tr><tr><td>1</td><td>P12</td><td>P10</td></tr></table>	UART0_IF_SEL	UART_TX	UART_RX	0	P31	P30	1	P12	P10							
UART0_IF_SEL	UART_TX	UART_RX																	
0	P31	P30																	
1	P12	P10																	
6	UART1_IF_EN	R/W	UART1 接口使能信号. 0 : 禁止 1 : 使能																
5	UART0_IF_EN	R/W	UART0 接口使能信号 0 : 禁止 1 : 使能																
4	T01_CK_SEL	R/W	Timer0/1 时钟源选择信号. 组合 TOM/T1M of CKCON(0x8E)使用 <table><tr><td>TOM /T1M</td><td>T01_CK_SEL</td><td>Timer0/1 时钟</td></tr><tr><td>0</td><td>0</td><td>系统时钟 / 8</td></tr><tr><td>0</td><td>1</td><td>系统时钟 / 2</td></tr><tr><td>1</td><td>0</td><td>系统时钟 / 4</td></tr><tr><td>1</td><td>1</td><td>系统时钟 / 1</td></tr></table>	TOM /T1M	T01_CK_SEL	Timer0/1 时钟	0	0	系统时钟 / 8	0	1	系统时钟 / 2	1	0	系统时钟 / 4	1	1	系统时钟 / 1	
TOM /T1M	T01_CK_SEL	Timer0/1 时钟																	
0	0	系统时钟 / 8																	
0	1	系统时钟 / 2																	
1	0	系统时钟 / 4																	
1	1	系统时钟 / 1																	
3	--	R/W	保留																
2	AERR_RSTEN	R/W	Flash 地址溢出复位使能 0 : 禁止 1 : 使能																
1	WDOG_CKEN	R/W	看门狗控制器时钟使能控制位 0 : 禁止 1 : 使能																
0	--	R/W	保留																

表 5- 51 SYS_CTRL6 寄存器

5.8. 低速时钟

GPM8F3733A / GPM8F3717A / GPM8F3709A 为低速计数中断配有一个内部的低速频率振荡器（16KHz）。

5.9. I/O 端口

GPM8F3733A / GPM8F3717A / GPM8F3709A 有四组 I/O 口，包括标准的 Port 0、Port 1、Port 2、Port 3。这些 I/O 引脚与芯片的外设功能引脚是复用的。通常情况下，当复位初始化后，所有 I/O 都用作 Open-Drain 结构的通用输入。使用者可以通过 CONFIG_BYTE[1] 设置选择 I/O 口的初始化状态。所有的 I/O 口都可由 PU 和 PD 寄存器编程设置为拉高或拉低。Port 0 的 PU 和 PD 寄存器由 0xAA 和 0xAB 控制，Port 1 的 PU 和 PD 寄存器由 0xBA 和 0xBB 控制，Port 2 的 PU 和 PD 寄存器由 0xA2 和 0xA3 控制，Port 3 的 PU 和 PD 寄存器由 0xB2 和 0xB3 控制。I/O 端口的读和写可通过操作相应的 SFR 寄存器 P0(0x80)、P1(0x90)、P2(0xA0)、P3(0xB0) 来实现。当 PU 和 PD 同时置 1 时，I/O 端口可根据相应的数据输出高电平或者低电平。表 5-52 模拟管脚真值表给出了引脚的真值表。内置的上拉和下拉电阻为 50KΩ。此外还有用于 P0~P3 (Px_SR, x=0~3 and Px_SMTDIS, x=0~3) 的转态速率控制寄存器和施密特触发器控制寄存器。如果 I/O 端口需要以 25ns 转态速率控制来改变状态的话，每个端口的相应控制位设置为‘1’；在 I/O 默认状态下，只有施密特触发功能而没有转态速率控制。图 5-9 表明了模拟引脚和数字引脚的示意图。

PU	PD	DATA	PAD
0	0	0	输出低
0	0	1	浮空
0	1	0	输出低
0	1	1	输入下拉
1	0	0	输出低
1	0	1	输入上拉
1	1	0	输出低
1	1	1	输出高

表 5-52 数字管脚真值表

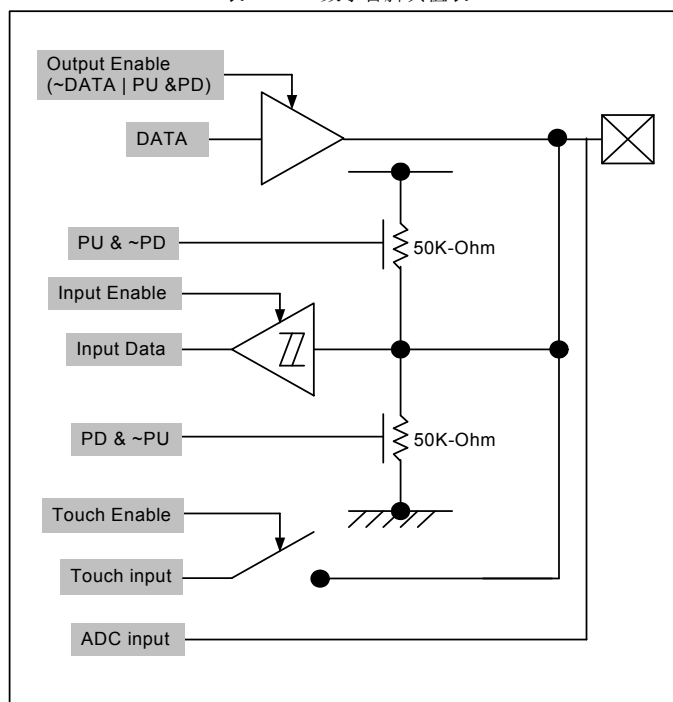


图 5-9 I/O 引脚示意图

GPM8F3733A/GPM8F3717A/GPM8F3709A

CONFIG_BYTE			地址: 0x8F (Flash)		CONFIG_BYTE 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	---	---	---	---	---	---	IOSEL	CODE_UNLOCK
默认	1	1	1	1	1	1	1	1

Bit	功能	类型	描述	条件
7:3	---	R	保留	
2	---	R	保留	
1	IOSEL	R/W	IO 初始化状态选择位 0: 输入上拉 1: 浮空	
0	CODE_UNLOCK	R/W	程序存储保护使能位 0: 程序被锁 1: 程序不被锁	

注意: 默认状态下 CONFIG_BYTE = 0xFF

表 5- 53 CONFIG_BYTE 描述

P0			地址: 0x80		Port0 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P07	P06	P05	P04	P03	P02	P01	P00
默认	1	1	1	1	1	1	1	1

Bit	功能	类型	描述	条件
7:0	P0[7:0]	R/W	Port0	

表 5- 54 P0 寄存器

P1			地址: 0x90		Port1 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P17	P16	P15	P14	P13	P12	P11	P10
默认	1	1	1	1	1	1	1	1

Bit	功能	类型	描述	条件
7:0	P1[7:0]	R/W	Port1	

表 5- 55 P1 寄存器

P2			地址: 0xA0		Port2 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	---	---	P25	P24	P23	P22	P21	P20
默认	1	1	1	1	1	1	1	1

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:6	--	R/W	预留	
5:0	P2[5:0]	R/W	Port20 ~ Port25	

表 5- 56 P2 寄存器

P3			地址: 0xB0		Port3 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	P34	P33	P32	P31	P30
默认	1	1	1	1	1	1	1	1

Bit	功能	类型	描述	条件
7:5	--	R/W	预留	
4:0	P3[4:0]	R/W	Port30~ Port34	

表 5- 57 P3 寄存器

P0_PU			地址: 0xAA		Port0 上拉配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P07_PU	P06_PU	P05_PU	P04_PU	P03_PU	P02_PU	P01_PU	P00_PU
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	P0_PU[7:0]	R/W	Port0 上拉控制位 0: 浮空; 1: 上拉	

表 5- 58 P0_PU 寄存器

P0_PD			地址: 0xAB		Port0 下拉配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P07_PD	P06_PD	P05_PD	P04_PD	P03_PD	P02_PD	P01_PD	P00_PD
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	P0_PD[7:0]	R/W	Port0 下拉控制位 0: 浮空 1: 下拉	

注意: 如果 P0_PU 和 P0_PD 同时被置 '1', P0 将是输出模式

表 5- 59 P0_PD 寄存器

P1_PU			地址: 0xBA		Port1 上拉配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P17_PU	P16_PU	P15_PU	P14_PU	P13_PU	P12_PU	P11_PU	P10_PU
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:0	P1[7:0]_PU	R/W	Port1 上拉控制位 0: 浮空 1: 上拉	

表 5- 60 P1_PU 寄存器

P1_PD			地址: 0xBB		Port1 下拉配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P17_PD	P16_PD	P15_PD	P14_PD	P13_PD	P12_PD	P11_PD	P10_PD
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	P1[7:0]_PD	R/W	Port1 下拉控制位 0:浮空; 1: 下拉	

注意：如果 P1_PU 和 P1_PD 同时被置 ‘1’，P1 将是输出模式

表 5- 61 P1_PD 寄存器

P2_PU			地址: 0xA2		Port2 上拉配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	P25_PU	P24_PU	P23_PU	P22_PU	P21_PU	P20_PU
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:4	--	R/W	预留	
5:0	P2_PU [5:0]	R/W	Port2 上拉控制位 0: 浮空; 1: 上拉	

表 5- 62 P2_PU 寄存器

P2_PD			地址: 0xA3		Port2 下拉配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	P25_PD	P24_PD	P23_PD	P22_PD	P21_PD	P20_PD
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:4	--	R/W	预留	
5:0	P2_PD [5:0]	R/W	Port2 下拉控制位 0: 浮空; 1: 下拉	

注意：如果 P2_PU 和 P2_PD 同时被置 ‘1’，P2 将是输出模式

表 5- 63 P2_PD 寄存器

GPM8F3733A/GPM8F3717A/GPM8F3709A

P3_PU			地址: 0xB2		Port3 上拉配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	P34_PU	P33_PU	P32_PU	P31_PU	P30_PU
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:5	--	R/W	预留	
4:0	P3_PU[4:0]	R/W	Port3 下拉控制位 0: 浮空; 1: 上拉	

表 5- 64 P3_PU 寄存器

P3_PD			地址: 0xB3		Port3 下拉配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	P34_PD	P33_PD	P32_PD	P31_PD	P30_PD
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:5	--	R/W	预留	
4:0	P3_PD[4:0]	R/W	Port3 上拉控制位 0: 浮空; 1: 上拉	

注意: 如果 P3_PU 和 P3_PD 同时被置 '1', P3 将是输出模式

表 5- 65 P3_PD 寄存器

P0_SR			地址: 0xAD		P0 转态速率控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P07_SR	P06_SR	P05_SR	P04_SR	P03_SR	P02_SR	P01_SR	P00_SR
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	P0_SR	R/W	Port0 转态速率控制位 0: 转态速率控制禁用 1: 25ns 的转态速率	

表 5- 66 P0_SR 寄存器

P1_SR			地址: 0xBD		Port1 转态速率控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P17_SR	P16_SR	P15_SR	P14_SR	P13_SR	P12_SR	P11_SR	P10_SR
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:0	P1_SR	R/W	Port1 转态速率控制位 0: 转态速率控制禁用 1: 25ns 的转态速率	

表 5- 67 P1_SR 寄存器

P2_SR			地址: 0xA5		Port2 转态速率控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	P25_SR	P24_SR	P23_SR	P22_SR	P21_SR	P20_SR
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:6	--	R/W	预留	
5:0	P2_SR	R/W	Port2 转态速率控制位 0: 转态速率控制禁用 1: 25ns 的转态速率	

表 5- 68 P2_SR 寄存器

P3_SR			地址: 0xB5		Port3 转态速率控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	P34_SR	P33_SR	P32_SR	P31_SR	P30_SR
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:5	--	R/W	预留	
4:0	P3_SR	R/W	Port3 转态速率控制位 0: 转态速率控制禁用 1: 25ns 的转态速率	

表 5- 69 P3_SR 寄存器

P0_SMTDIS			地址: 0xAC		Port0 施密特触发禁用控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P07_SMTDIS	P06_SMTDIS	P05_SMTDIS	P04_SMTDIS	P03_SMTDIS	P02_SMTDIS	P01_SMTDIS	P00_SMTDIS
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	P0_SMTDIS	R/W	Port0 施密特触发禁用控制位 0: 使能 Port0 施密特触发功能 1: 禁用 Port0 施密特触发功能	

表 5- 70 P0_SMTDIS 寄存器

P1_SMTDIS			地址: 0xBC		Port1 施密特触发禁用控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P17_SMTDIS	P16_SMTDIS	P15_SMTDIS	P14_SMTDIS	P13_SMTDIS	P12_SMTDIS	P11_SMTDIS	P10_SMTDIS
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	秒速	条件
7:0	P1_SMTDIS	R/W	Port1 施密特触发禁用控制位 0: 使能 Port1 施密特触发功能 1: 禁用 Port1 施密特触发功能	

表 5- 71 P1_SMTDIS 寄存器

P2_SMTDIS			地址: 0xA4		Port2 施密特触发禁用控制位			
Bit	7	6	5	4	3	2	1	0
功能	--	--	P25_SMTDIS	P24_SMTDIS	P23_SMTDIS	P22_SMTDIS	P21_SMTDIS	P20_SMTDIS
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:6	--	R/W	预留	
5:0	P2_SMTDIS	R/W	Port2 施密特触发禁用控制位 0: 使能 Port2 施密特触发功能 1: 禁用 Port2 施密特触发功能	

表 5- 72 P2_SMTDIS 寄存器

P3_SMTDIS			地址: 0xB4		Port3 施密特触发禁用控制位			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	P34_SMTDIS	P33_SMTDIS	P32_SMTDIS	P31_SMTDIS	P30_SMTDIS
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:5	--	R/W	预留	
4:0	P3_SMTDIS	R/W	Port3 施密特触发禁用控制位 0: 使能 Port3 施密特触发功能 1: 禁用 Port3 施密特触发功能	

表 5- 73 P3_SMTDIS 寄存器

P0_HS			地址: 0xAE		Port0 大电流控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	P07_HS	P06_HS	P05_HS	P04_HS	P03_HS	P02_HS	P01_HS	P00_HS
默认	0	0	0	0	0	0	0	0
Key Code	0x8F, 0x32, 0x52							

Bit	功能	类型	描述	条件
-----	----	----	----	----

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:0	POX_HS	R/W	Port0 大电流控制位 用户一次只能打开一个 IO 的大电流，两个或者两个以上的都不能保证。 0: 禁用 Port0X 大电流功能 1: 使能 Port0X 大电流功能	

表 5- 74 P0_HS 寄存器

P0_HSBIAS			地址: 0xAF		Port0 大电流接口控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	--	--	--	--	SRI	ENI
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:2	--	R/W	预留	
1	SRI	R/W	大电流 IO 的转态速率控制，典型值是 1.8us。 0: 禁止当 P0 为大电流时的转态速率控制 1: 使能当 P0 为大电流时的转态速率控制	
0	ENI	R/W	Port0 大电流控制位。 0: P0[7:0] 大电流功能无效。 1: P0[7:0] 大电流功能有效	

表 5- 75 P0_HSBIAS 寄存器

5.10. 定时器模块

5.10.1. 介绍

GPM8F3733A / GPM8F3717A / GPM8F3709A 有五个定时器，分别是 Timer0、Timer1、Timer2，TimerA 和 TimerB。除了定时器功能，Timer2 还具有比较、捕捉、重载功能。这五个定时器都是向上计数的 16Bit 的定时器和计数器。下面的章节将详细说明每个定时器的功能。

5.10.2. Timer 0/1

Timer 0 和 Timer1 与标准 8051 的定时器完全兼容。每个定时器包含 2 个 8-bit 寄存器 TH0(0x8C)，TL0(0x8A)，TH1(0x8D) 和 TL1(0x8B)。除了 Mode 3 外 Timer 0 和 Timer 1 的其它 3 种工作模式相同。相关的控制寄存器是 TMOD(0x89)，TCON(0x88) 和 CKCON(0x8E)。

5.10.2.1. Timer 0:模式 0（13-Bit 定时器）

该模式下，Timer 0 寄存器配置为 13-bit 寄存器。当计数从全 ‘1’ 翻转为全 ‘0’ 时，Timer 0 的中断标志 TF0 被置位。当 TR0(TCON[4]) = 1，Timer 0 的计数输入使能。13-bit 寄存器由 TH0 的全部 8 位和 TL0 的低 5 位组成。TL0 高 3 位的状态不定，可以忽略不考虑。图 5- 10 Timer 0 模式 0 示意图给出了模式 0 下 Timer0 的示意图。

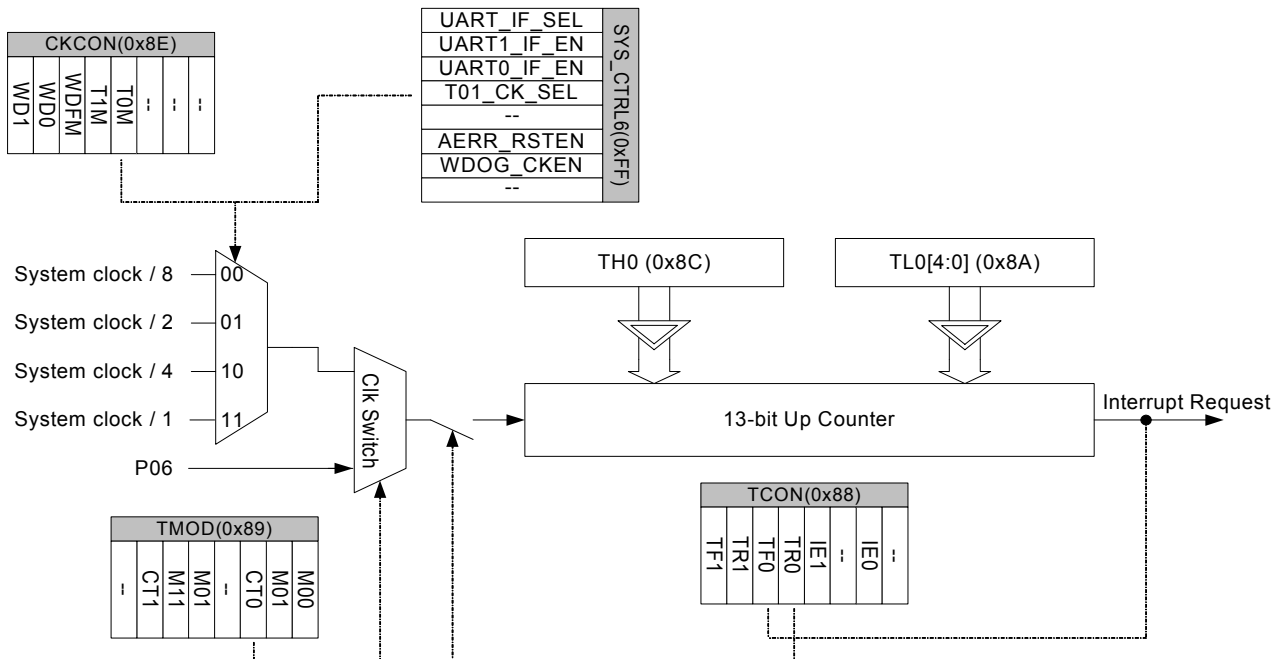


图 5- 10 Timer 0 模式 0 示意图

5.10.2.2. Timer 0:模式 1（16-Bit 定时器）

除了使用全部 16 位定时器寄存器之外，模式 1 与模式 0 相同。图 5- 11 Timer 0 模式 1 示意图给出了模式 1 下 Timer0 的示意图。

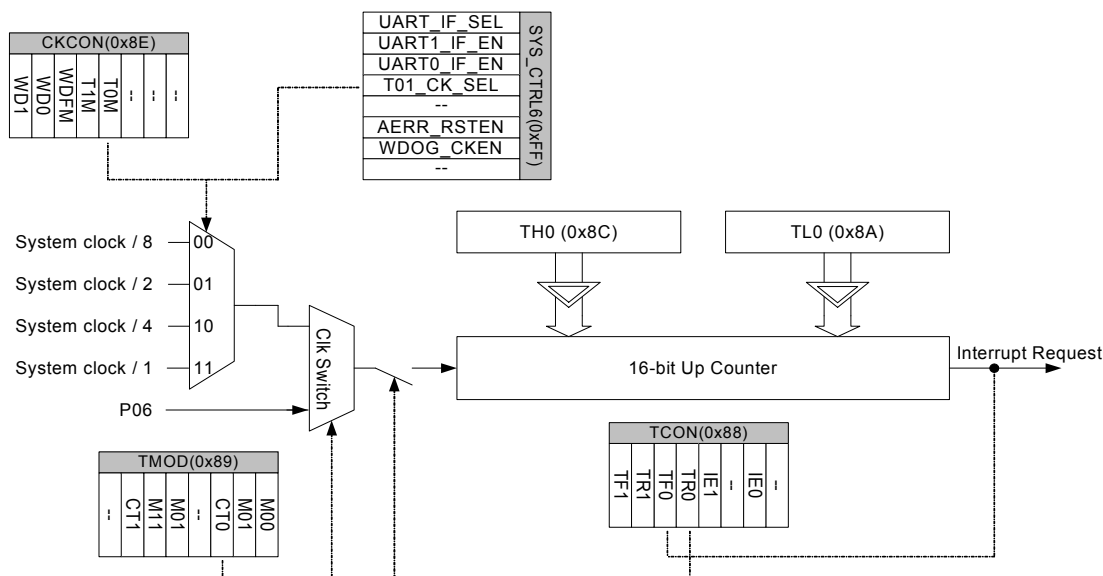


图 5- 11 Timer 0 模式 1 示意图

5.10.2.3. Timer 0:模式 2（带有自动重装功能的 8-Bit 定时器）

Mode 2 下，定时器寄存器配置为自动重载的 8-bit 计数器 (TL0)，如图 5- 12 Timer 0 模式 2 示意图所示。TL0 溢出时，不仅设置了 TF0，而且将 TH0 中的内容重载到 TL0 中，TH0 中的内容通过软件设置。重载不会改变 TH0 的内容。

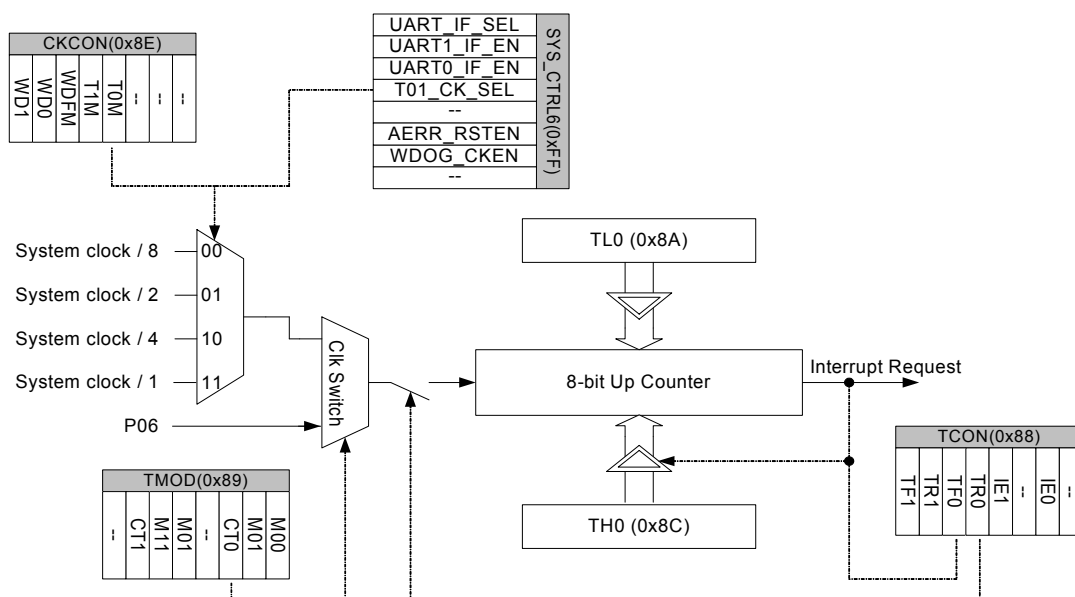


图 5- 12 Timer 0 模式 2 示意图

5.10.2.4. Timer 0:模式 3（两个 8-Bit 定时器）

在 Mode3 下，Timer 0 的 TL0 和 TH0 是两个独立的计数器。

图 5- 13 Timer 0 模式 3 示意图给出了 Mode3 下 Timer 0 的示意图。TL0 使用 Timer 0 的控制位：CT0、TR0 和 TF0。TH0 锁定为定时器功能，使用 Timer 1 的 TR1 和 TF1 标志位并控制 Timer 1 的中断。Mode 3 满足了应用中需要一个额外 8-bit 定时/计数器的需求。当 Timer 0 工作在 Mode3 时，通过切换 Timer 1 到它自己的 Mode 3 来关闭 Timer 1，也可用作串口的波特率发生器，或者不需要 Timer 1 中断的任何应用中。

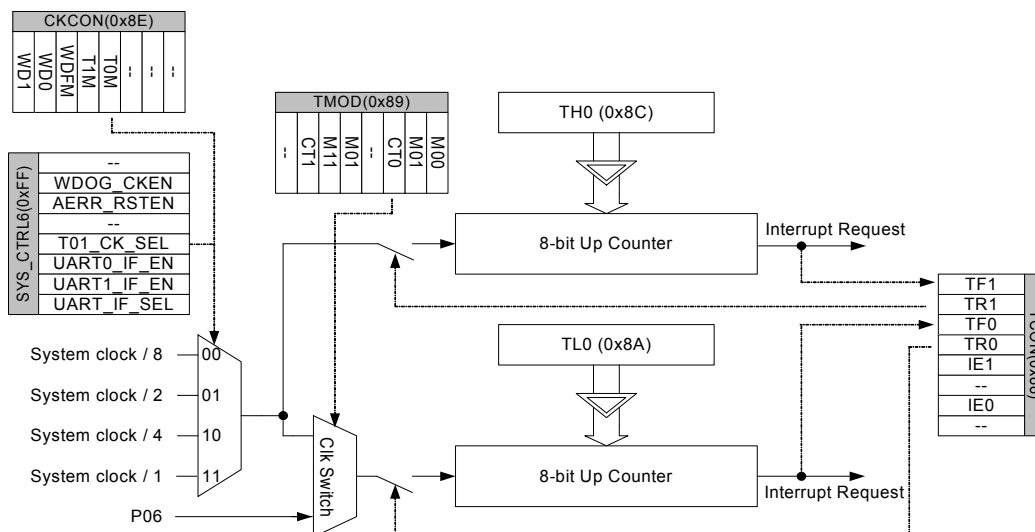


图 5- 13 Timer 0 模式 3 示意图

5.10.2.5. Timer 1:模式 0（13-Bit 定时器）

在该模式下，Timer 1 寄存器配置为 13-bit 寄存器。当计数从全 ‘1’ 翻转为全 ‘0’ 时，Timer1 的中断标志 TF1 被置位。当 TR1 (TCON[6]) = 1，Timer1 的计数输入使能。13-bit 寄存器由 TH1 的全部 8 位和 TL1 的低 5 位组成。TL1 高 3 位的状态不定，可以忽略不考虑。图 5- 14 Timer 1 模式 0 示意图给出了模式 0 下 Timer1 的示意图。

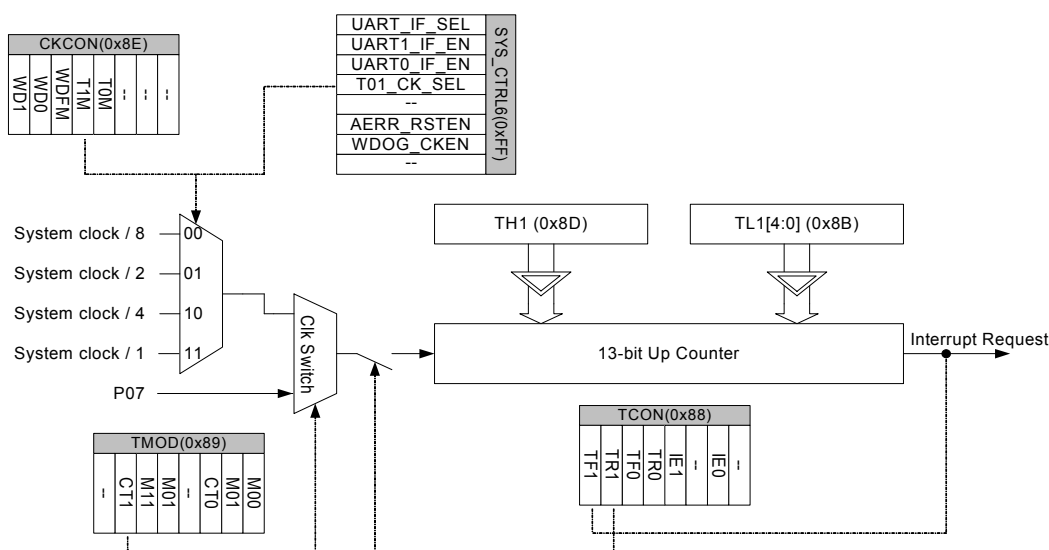


图 5- 14 Timer 1 模式 0 示意图

5.10.2.6. Timer 1:模式 1 (16-Bit 定时器)

除了使用全部 16 位定时器寄存器之外, 模式 1 与模式 0 相同。图 5-15 Timer 1 模式 1 示意图给出了模式 1 下 Timer1 的示意图

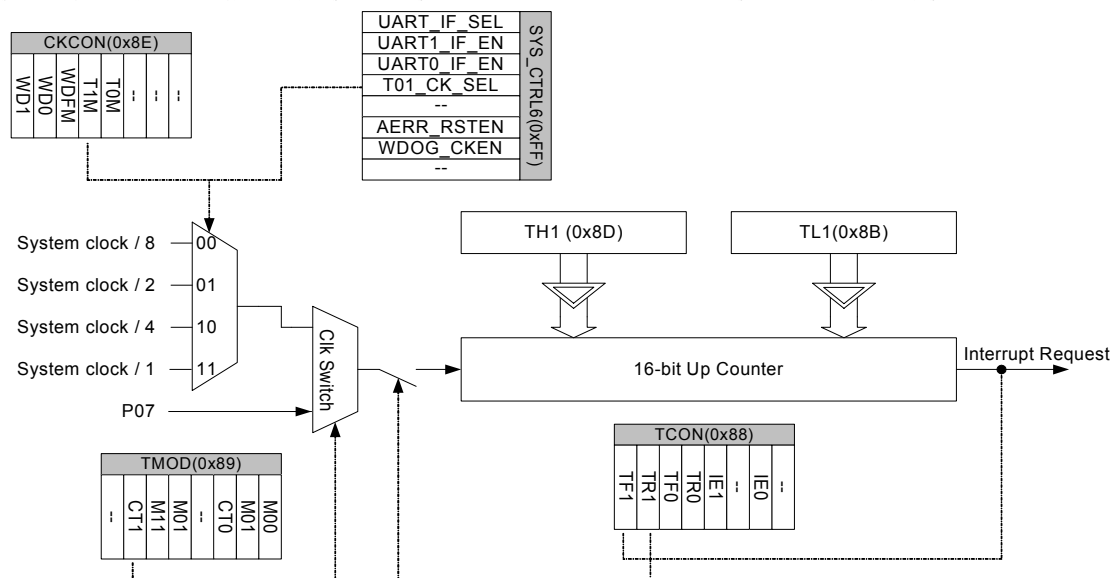


图 5-15 Timer 1 模式 1 示意图

5.10.2.7. Timer 1:模式 2（帶有自动重装功能的 8-Bit 定时器）

模式 2 下, 定时器寄存器配置为自动重载的 8-bit 计数器(TL1), 如图 5-16 Timer 1 模式 2 示意图所示。TL1 溢出时, 不仅设置了 TF1, 而且将 TH1 中的内容重载到 TL1 中, TH1 中的内容通过软件设置。重载不会改变 TH1 的内容。

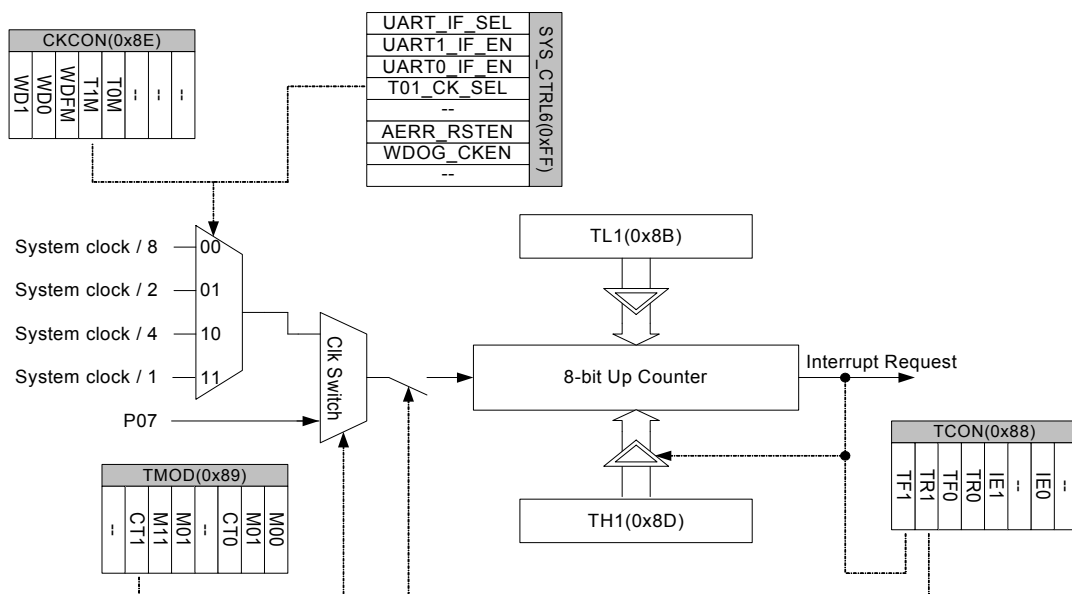


图 5-16 Timer 1 模式 2 示意图

5.10.2.8. Timer 1:模式 3

Timer 1 在 Mode3 时没有定时器功能。等效于设置 TR1=0。

5.10.2.9. Timer0/1 相关寄存器

TH0			地址: 0x8C		Timer0 高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TH0[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TH0[7:0]	R/W	Timer 0 高字节装载值	

表 5- 76 TH0 寄存器

TL0			地址: 0x8A		Timer0 低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TL0[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TL0[7:0]	R/W	Timer 0 低字节转装载值	

表 5- 77 TL0 寄存器

TH1			地址: 0x8D		Timer1 高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TH1[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TH1[7:0]	R/W	Timer 1 高字节装载值	

表 5- 78 TH1 寄存器

TL1			地址: 0x8B		Timer1 低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TL1[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TL1[7:0]	R/W	Timer 1 低字节装载值	

表 5- 79 TL1 寄存器

TMOD			地址: 0x89		Timer0/1 控制模式寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	CT1	M11	M10	--	CT0	M01	M00
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7	--	R/W	预留	
6	CT1	R/W	计数器/定时器选择位 0: 定时器模式, 内部时钟 1: 计数器模式, P07 作为 Timer 1 时钟源	
5:4	M1[1:0]	R/W	timer 1 的模式选择位, 如表 表 5- 81 所示	
3	--	R/W	预留	
2	CT0	R/W	计数器/定时器选择位 0: 定时器模式, 内部时钟 1: 计数器模式, P06 作为 Timer 0 时钟源 Counter mode, 如表 5- 81	
1:0	M0[1:0]	R/W	timer 0 的模式选择位, 如表 5- 81	

表 5- 80 TMOD 寄存器

M1	M0	模式	功能描述
0	0	0	TH0/1 作为 8-bit 的定时器使用, 被 TL0/1 的低五位 32 除频。
0	1	1	16-bit 定时器, TH0/1 和 TL0/1 级联
1	0	2	TL0/1 作为 8-bit 定时器使用, TH0/1 是自动重载值
1	1	3	TL0 用作 8-bit 定时器, 由标准 Timer0 来控制。TH0 用作 8-bit 定时器, 由 Timer 1 的控制位来控制。Timer 1 保持它的计数功能。

表 5- 81 Timer 0 和 Timer 1 的四种模式

TCON			地址: 0x88		Timer0/1 配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TF1	TR1	TF0	TR0	IE1	--	IE0	--
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	TF1	R/W	Timer1 中断(溢出)标志	
6	TR1	R/W	Timer1 运行控制位 0: 禁用; 1: 使能	
5	TF0	R/W	Timer0 中断(溢出)标志	
4	TR0	R/W	Timer0 运行控制位 0: 禁用; 1: 使能	
3	IE1	R/W	INT1 中断标志	
2	--	R/W	预留	
1	IE0	R/W	INT0 中断标志	
0	--	R/W	预留	

表 5- 82 TCON 寄存器

CKCON			地址: 0x8E		时钟控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	WD1	WDO	WDFM	T1M	T0M	--	--	--
默认	0	0	0	0	0	1	1	1

Bit	功能	类型	描述	条件																				
7:6	WD[1:0]	R/W	看门狗超时选择位 If WDFM=0: <table><tr><td>WD[1:0]</td><td>超时</td></tr><tr><td>00</td><td>128ms</td></tr><tr><td>01</td><td>256ms</td></tr><tr><td>10</td><td>512ms</td></tr><tr><td>11</td><td>1024ms</td></tr></table> If WDFM=1: <table><tr><td>WD[1:0]</td><td>超时</td></tr><tr><td>00</td><td>8ms</td></tr><tr><td>01</td><td>16ms</td></tr><tr><td>10</td><td>32ms</td></tr><tr><td>11</td><td>64ms</td></tr></table>	WD[1:0]	超时	00	128ms	01	256ms	10	512ms	11	1024ms	WD[1:0]	超时	00	8ms	01	16ms	10	32ms	11	64ms	
WD[1:0]	超时																							
00	128ms																							
01	256ms																							
10	512ms																							
11	1024ms																							
WD[1:0]	超时																							
00	8ms																							
01	16ms																							
10	32ms																							
11	64ms																							
5	WDFM	R/W	看门狗快模式选择位 0: 禁用看门狗快模式 1: 使能看门狗快模式																					
4	T1M	R/W	系统时钟的除频选择，用于驱动 Timer1 0: Timer1 使用系统时钟频率的二分频 1: Timer1 使用系统时钟频率																					
3	T0M	R/W	系统时钟的除频选择，用于驱动 Timer0 0: Timer0 使用系统时钟频率的二分频 1: Timer0 使用系统时钟频率																					
2:0	---	R/W	预留																					

表 5- 83 CKCON 寄存器

SYS_CTRL6			地址 : 0xFF		系统控制 6 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	UART0_IF_SEL	UART1_IF_EN	UART0_IF_EN	T01_CK_SEL	--	AERR_RSTEN	WDOG_CKEN	--
默认	0	0	0	0	0	0	1	0
Key Code	0x8F, 0x32, 0x50							

Bit	功能	类型	描述	条件												
7	UART0_IF_SEL	R/W	<table><tr><td colspan="3">UART IO 接口选择信号</td></tr><tr><td>UART0_IF_SEL</td><td>UART_TX</td><td>UART_RX</td></tr><tr><td>0</td><td>P31</td><td>P30</td></tr><tr><td>1</td><td>P12</td><td>P10</td></tr></table>	UART IO 接口选择信号			UART0_IF_SEL	UART_TX	UART_RX	0	P31	P30	1	P12	P10	
UART IO 接口选择信号																
UART0_IF_SEL	UART_TX	UART_RX														
0	P31	P30														
1	P12	P10														
6	UART1_IF_EN	R/W	UART1 接口使能信号. 0 : 禁止 1 : 使能													
5	UART0_IF_EN	R/W	UART0 接口使能信号. 0 : 禁止 1 : 使能													

Bit	功能	类型	描述	条件															
4	T01_CK_SEL	R/W	Timer0/1 时钟源选择信号. 与 TOM/T1M of CKCON(0x8E) 结合使用 <table><tr><th>TOM /T1M</th><th>T01_CK_SEL</th><th>Timer0/1 时钟</th></tr><tr><td>0</td><td>0</td><td>系统时钟 / 8</td></tr><tr><td>0</td><td>1</td><td>系统时钟 / 2</td></tr><tr><td>1</td><td>0</td><td>系统时钟 / 4</td></tr><tr><td>1</td><td>1</td><td>系统时钟 / 1</td></tr></table>	TOM /T1M	T01_CK_SEL	Timer0/1 时钟	0	0	系统时钟 / 8	0	1	系统时钟 / 2	1	0	系统时钟 / 4	1	1	系统时钟 / 1	
TOM /T1M	T01_CK_SEL	Timer0/1 时钟																	
0	0	系统时钟 / 8																	
0	1	系统时钟 / 2																	
1	0	系统时钟 / 4																	
1	1	系统时钟 / 1																	
3	--	R/W	预留																
2	AERR_RSTEN	R/W	Flash 地址溢出复位使能 0 : 禁止 1 : 使能																
1	WD0G_CKEN	R/W	看门狗控制器时钟使能控制位 0 : 禁止 1 : 使能																
0	--	R/W	预留																

表 5- 84 The SYS_CTRL6 寄存器

5.10.3. Timer 2

Timer 2 是一个 16-bit 的定时器。其额外的捕捉/重载特性是芯片最强大的外设单元之一。可应用于事件捕捉，如脉冲发生器和脉宽测量等。图 5- 17 Timer 2 比较捕捉功能示意图给出了 Timer 2 的比较和捕捉功能的示意图。

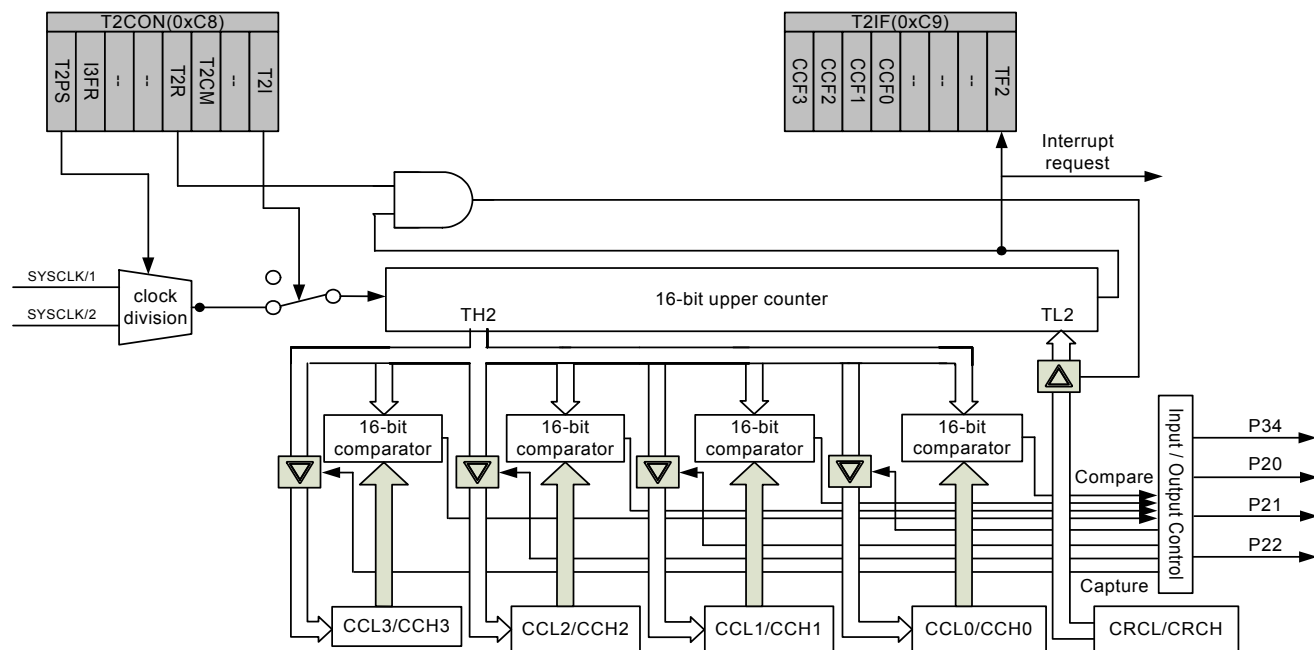


图 5- 17 Timer 2 比较捕捉功能示意图

5.10.3.1. Timer 2 的定时器模式

在该模式下，计数频率来源于振荡器频率。一个预分频器提供了对于振荡器频率 1 分频或者 2 分频的选择。这样，16-bit 定时寄存器(由 TH2 和 TL2 组成)既可以每 1 个时钟周期自加也可以每 2 个时钟周期自加。预分频器的选择由 T2CON 的 T2PS 位设置。

5.10.3.2. Timer 2 的重载功能

Timer 2 的重载模式由 T2CON 的 T2R 位选择。当 Timer 2 的值从全 ‘1’ 翻转为全 ‘0’ 时，不仅 TF2 标志位被设置，而且 16-bit 的 CRC 寄存器中的值被装载入 Timer 2 寄存器中。CRC 中的值预先由软件设置。重载发生在 TF2 标志位被设置的同一个时钟周期里，这样会覆盖计数值 0x0000。

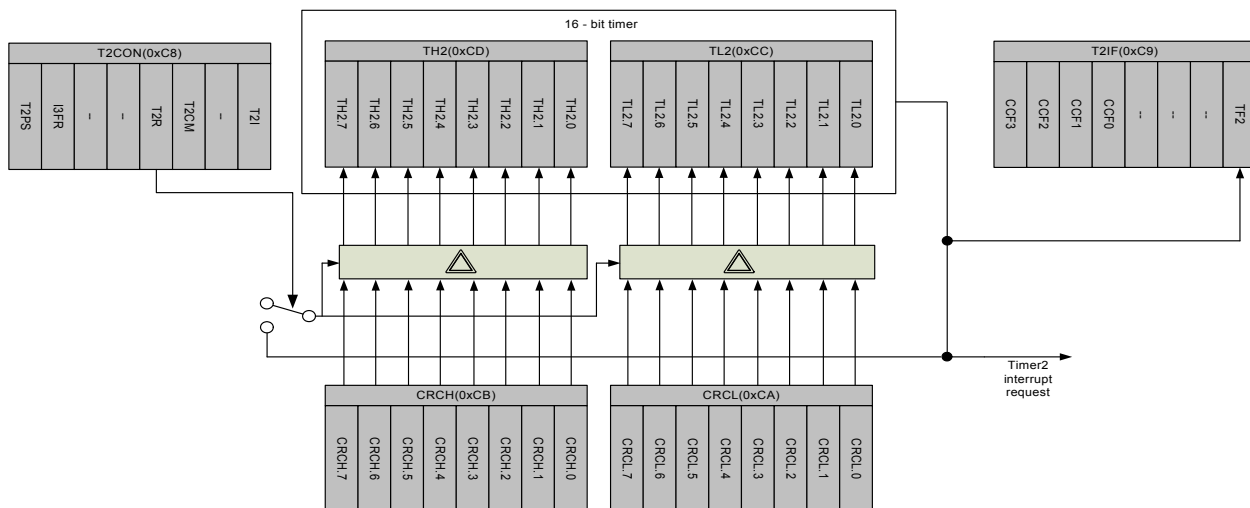


图 5-18 Timer 2 重载功能示意图

5.10.3.3. Timer 2 的比较功能

存储在比较/捕捉寄存器中的 16-bit 值与定时器寄存器中的内容相比较。如果定时器寄存器中的计数值与存储值一致，则在相应的引脚上产生一个合适的输出信号，同时中断请求发生。比较寄存器中的内容可以看作是一个时间标签，在这个时间标签点，会以预定义的方式产生一个特定的输出（无论是正跳变或是负跳变）。这个时间标签的变化会改变引脚上的矩形波输出信号的波形。这种方式—改变周期信号的占空比—可用于脉宽调制，同时也可用于控制持续的任意方波的产生。有两种比较模式可满足各种广泛应用的需求。比较模式 0 和比较模式 1 由 SFR 寄存器 T2CON 的 T2CM 位来选择。在这两种比较模式下，P34 端口上信号的改变与内部比较信号的激活在同一时钟周期内发生。

■ 比较模式 0

在比较模式 0，当定时器和比较寄存器的内容匹配时，输出信号由低电平变高电平，并在定时器溢出时变回低电平。图 5-19 Timer 2 比较模式 0 示意图给出了比较模式的端口寄存器的功能示意图。

■ 比较模式 1

在比较模式 1 下，由软件决定了输出信号的转换。通常用于输出信号与一个固定的周期信号无关的情况下。如果模式 1 使能，当软件写入值到 P34/P20/P22 寄存器时，直至下一次比较匹配时新的数值才会出现在输出引脚上。无论输出信号有新的转换或是保持原有的数值，用户都可以选择这种方式。图 5-20 Timer 2 比较模式 1 示意图给出了 Timer 2 在比较模式 1 下的功能图。

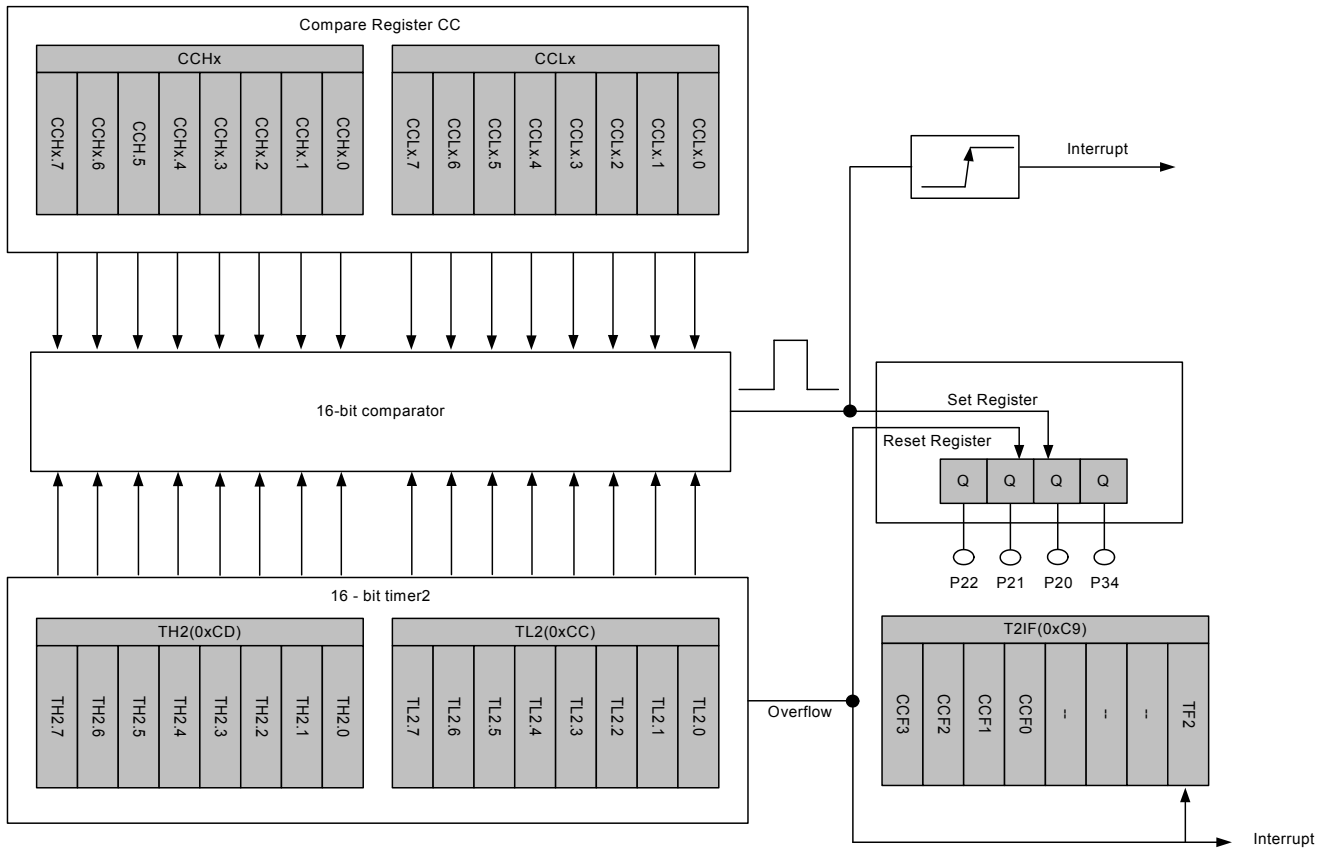


图 5- 19 Timer 2 比较模式 0 示意图

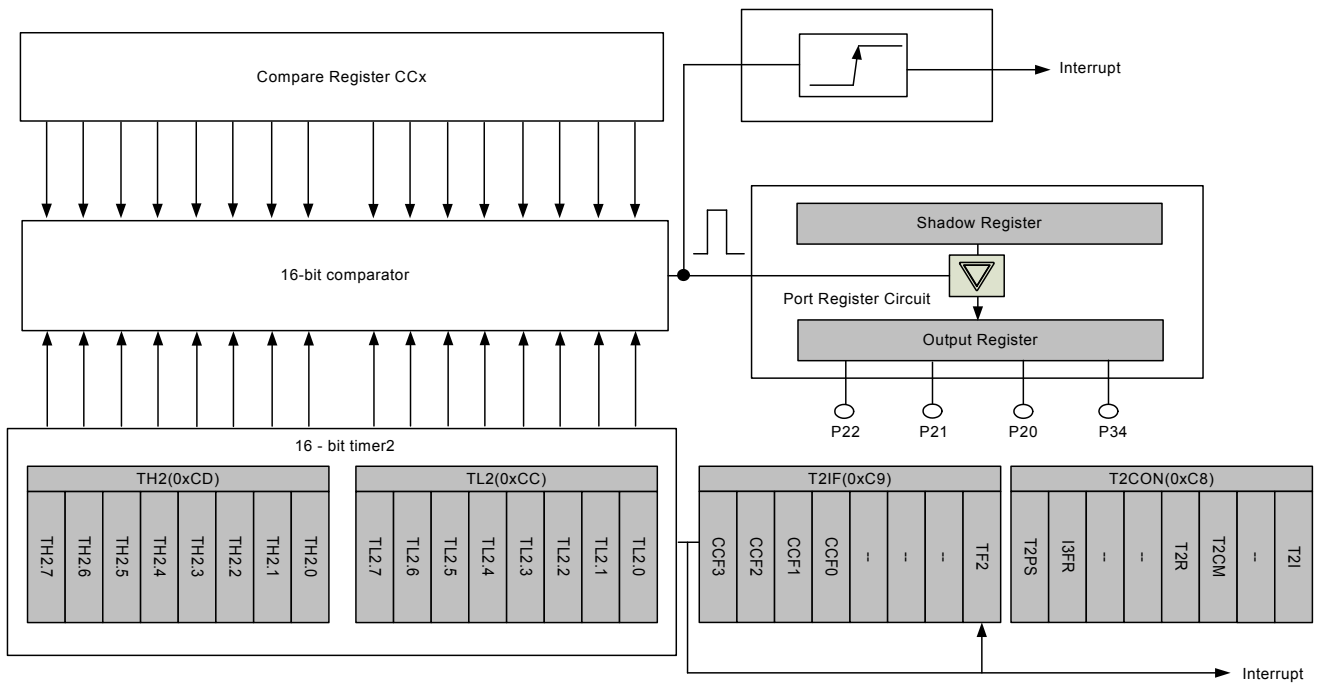


图 5- 20 Timer 2 比较模式 1 示意图

5.10.3.4. Timer 2 的捕捉功能

比较/捕捉寄存器可用来锁存当前 Timer 2 寄存器 TL2 和 TH2 中的 16-bit 值。该功能有两种不同模式。

■ 捕捉模式 0

在模式 0 下，外部事件将 Timer 2 的内容锁存到指定的捕捉寄存器中。在引脚 CAPTURE0 上的一个正跳变或者负跳变，取决于 T2CON 的 I3FR 位。如果 I3FR 标志被清除，负跳变产生捕捉；否则，该引脚上的正跳变产生捕捉。

■ 捕捉模式 1

在模式 1 下，对 16-bit 捕捉寄存器的低字节（CCL）的写操作将会产生捕捉。这种模式允许软件即时地读取 Timer 2 的内容。对捕捉寄存器低字节的写入指令将产生捕捉。写寄存器信号（如写 CRCL）用来启动捕捉。写入专用捕捉寄存器的值与该功能无关。Timer 2 的内容将在写指令的下一个周期里被锁存进捕捉寄存器中。该模式下不会产生中断请求。图 5- 21 给出了 Timer 2 捕捉功能的示意图。

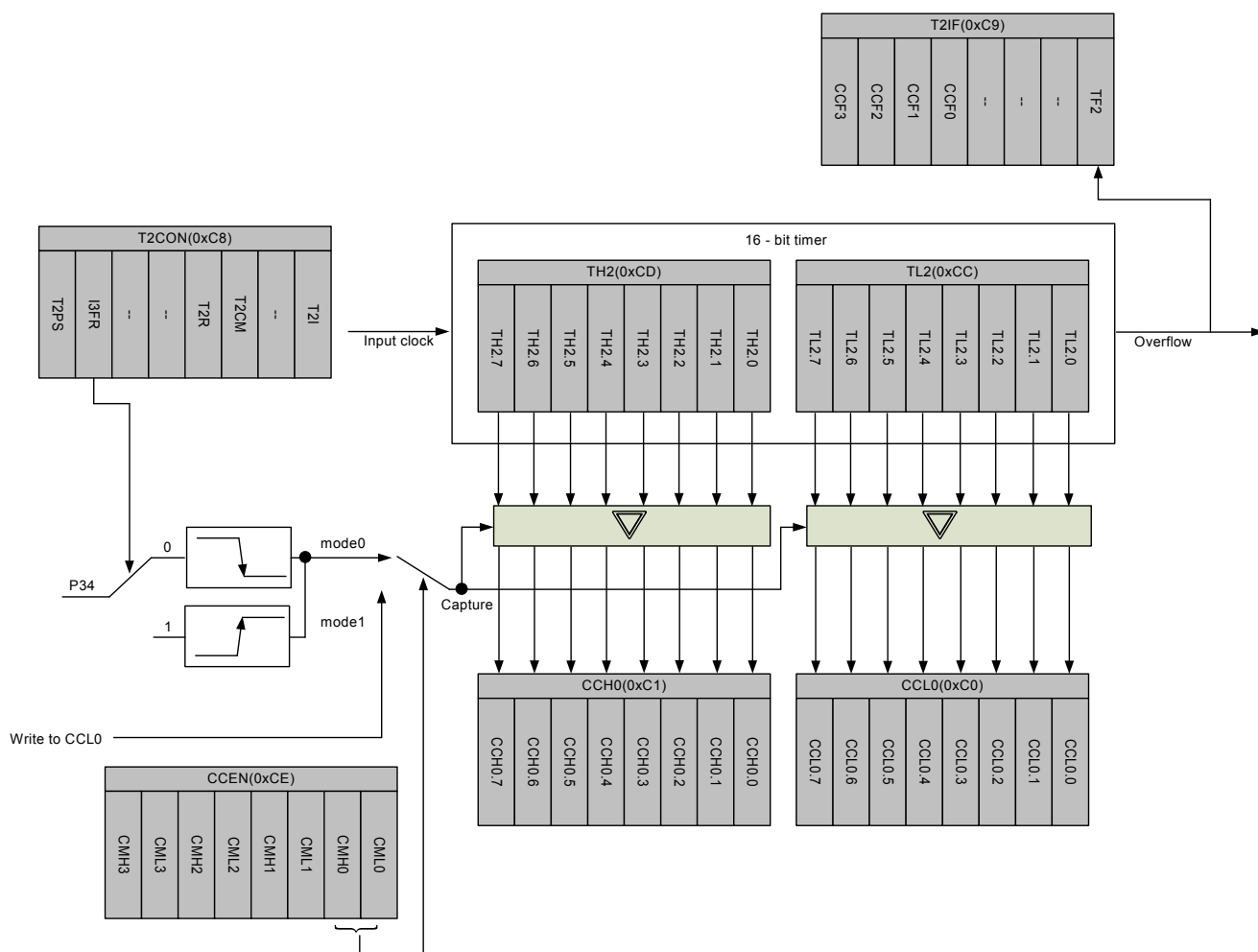


图 5- 21 CCL0 和 CCH0 的 Timer2 捕捉模式示意图

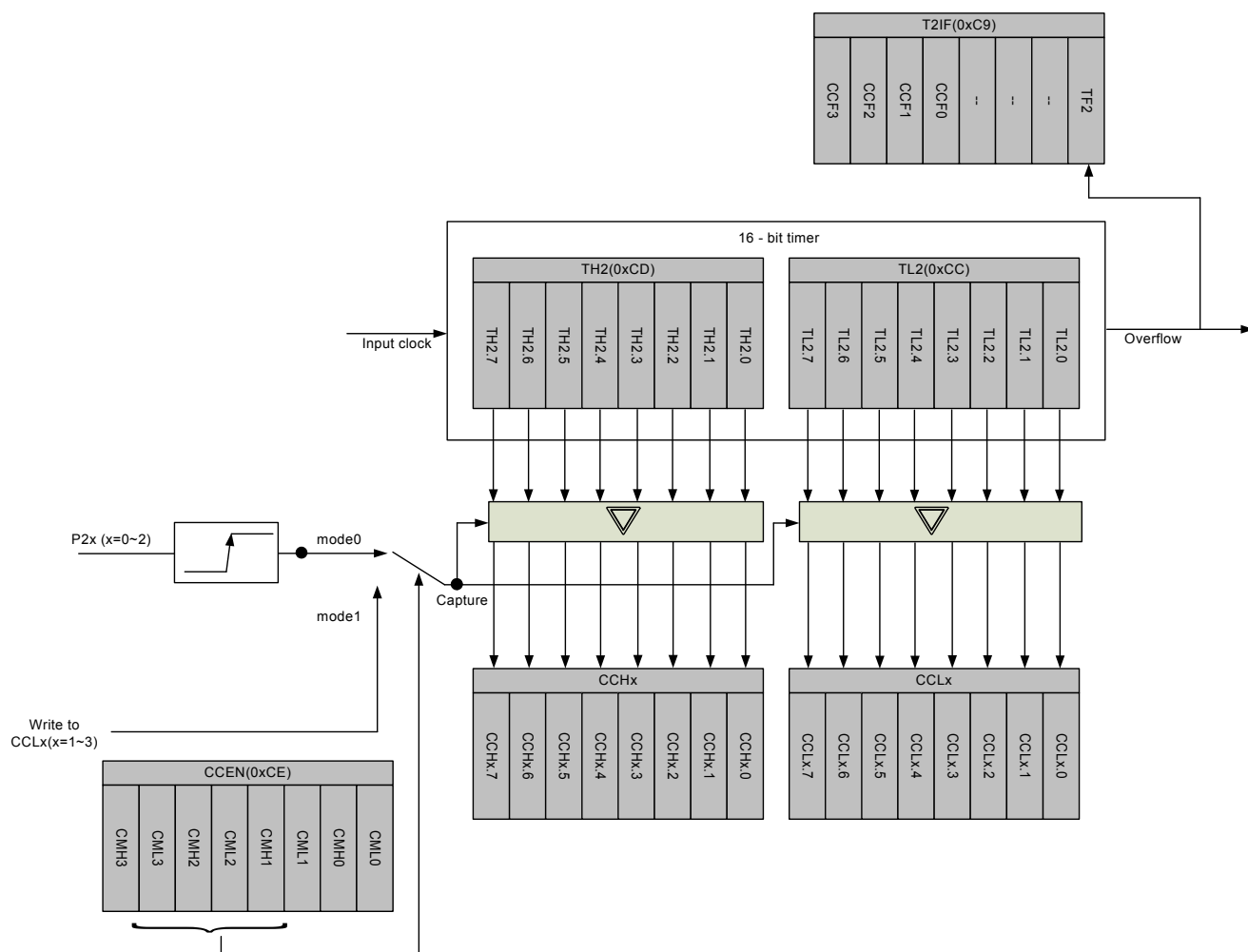


图 5-22 CCLx 和 CCHx (x=1, 2, 3) 的 Timer2 捕捉模式示意图

5.10.3.5. Timer 2 相关寄存器

T2CON			地址：0xC8		Timer2 配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	T2PS	I3FR	---	--	T2R	T2CM	---	T2I
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	T2PS	R/W	预分频选择位 0 : SYSCLK; 1 : SYSCLK/2	
6	I3FR	R/W	捕捉模式 0 边沿有效选择位 捕捉 0 功能: 0: CC 寄存器的捕捉发生在 CAPTURE0 引脚的负跳变 1: CC 寄存器的捕捉发生在 CAPTURE0 引脚的正跳变	

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
5: 4	—	R/W	预留	
3	T2R	R/W	Timer2 自动重载模式使能位	
2	T2CM	R/W	寄存器 CC 的比较模式选择位 0: 选择比较模式 0 1: 选择比较模式 1	
1	—	R/W	预留	
0	T2I	R/W	Timer2 输入选择位 0: 没有输入选择, Timer 2 停止 1: 定时器输入频率: SYSCLK (T2PS=0) SYSCLK/2 (T2PS=1)	

表 5- 85 T2CON 寄存器

CCEN			地址: 0xCE		比较/捕捉使能寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CMH3	CM3	CMH2	CM2	CMH1	CM1	CMH0	CM0
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述			条件
7:6	CM3	R/W	CC3 寄存器的比较/捕捉模式			
			CMH1	CML1	功能	
			0	0	比较/捕捉禁用	
			0	1	在 CAPTURE3 引脚的上升沿捕捉	
			1	0	比较使能	
			1	1	写 CCL3 寄存器时捕捉	
5:4	CM1	R/W	CC2 寄存器的比较/捕捉模式			
			CMH1	CML1	功能	
			0	0	比较/捕捉禁用	
			0	1	在 CAPTURE2 引脚的上升沿捕捉	
			1	0	比较使能	
			1	1	写 CCL2 寄存器时捕捉	
3:2	CM1	R/W	CC1 寄存器的比较/捕捉模式			
			CMH1	CML1	功能	
			0	0	比较/捕捉禁用	
			0	1	在 CAPTURE1 引脚的上升沿捕捉	
			1	0	比较使能	
			1	1	写 CCL1 寄存器时捕捉	

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述			条件
1:0	CM0	R/W	CC0 寄存器的比较/捕捉模式			
			CMH0	CML0	功能	
			0	0	比较/捕捉禁用	
			0	1	在 CAPTURE0 引脚的上升沿捕捉	
			1	0	比较使能	
			1	1	写 CCL0 寄存器时捕捉	

表 5- 86 CCEN 寄存器

T2IF			地址: 0xC9		Timer 2 中断标志寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CCF3	CCF2	CCF1	CCF0	--	--	--	TF2
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	CCF3	R/W	Compare3/Capture3 标志, 软件清零	
6	CCF2	R/W	Compare2/Capture2 标志, 软件清零	
5	CCF1	R/W	Compare1/Capture1 标志, 软件清零	
4	CCF0	R/W	Compare0/Capture0 标志, 软件清零	
3:1	--	R/W	预留	
0	TF2	R/W	Timer 2 溢出标志, 软件清零	

表 5- 87 T2IF 寄存器

CCH0			地址: 0x93		Timer 2 CC0 高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CC0[15:8]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CC0[15:8]	R/W	Timer2 比较/捕捉 - 高字节	

表 5- 88 CCH 寄存器

CCL0			地址: 0x92		Timer 2 CC0 低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CC0[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CC0[7:0]	R/W	Timer2 比较/捕捉 - 低字节	

表 5- 89 CCL 寄存器

GPM8F3733A/GPM8F3717A/GPM8F3709A

CCH1			地址: 0xC3		Timer 2 CC1 高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CC1[15:8]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CC1[15:8]	R/W	Timer2 比较/捕捉 - 高字节	

表 5- 90 CCH1 寄存器

CCL1			地址: 0xC2		Timer 2 CC1 低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CC1[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CC1[7:0]	R/W	Timer2 比较/捕捉 - 低字节	

表 5- 91 CCL1 寄存器

CCH2			地址: 0xC5		Timer 2 CC2 高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CC2[15:8]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CC2[15:8]	R/W	Timer2 比较/捕捉 - 高字节	

表 5- 92 CCH2 寄存器

CCL2			地址: 0xC4		Timer 2 CC2 低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CC2[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CC2[7:0]	R/W	Timer2 比较/捕捉 - 低字节	

表 5- 93 CCL2 寄存器

CCH3			地址: 0xC7		Timer 2 CC3 高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CC3[15:8]							
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:0	CC3[15:8]	R/W	Timer2 比较/捕捉 - 高字节	

表 5- 94 CCH3 寄存器

CCL3			地址: 0xC6		Timer 2 CC3 低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CC3[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CC3[7:0]	R/W	Timer2 比较/捕捉 - 低字节	

表 5- 95 CCL3 寄存器

CRCH			地址: 0xCB		CRC 高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CRC[15:8]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CRC[15:8]	R/W	CRC - 高字节	

表 5- 96 CRCH 寄存器

CRCL			地址: 0xCA		CRC 低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	CRC[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CRC[7:0]	R/W	CRC - 低字节	

表 5- 97 CRCL 寄存器

TH2			地址: 0xCD		Timer 2 高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TH2[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TH2[7:0]	R/W	Timer 2 装载值 - 高字节	

表 5- 98 TH2 寄存器

TL2			地址: 0xCC		Timer 2 低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TL2[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TL2[7:0]	R/W	Timer 2 装载值 - 低字节	

表 5- 99 TL2 寄存器

5.10.4. Timer A/B

Timer A/B 不仅仅用作通用的定时器/计数器，还可以在触摸应用中。Timer A/B 是 16 位的定时器，带有自动重载功能。当使能重载功能后，Timer A/B 从 0xFFFF 到 0x0000 溢出时，TMA/B_PLOAD_H/L 寄存器中 16 位的值会自动重载到 Timer A/B 寄存器中。图 5- 23 和图 5-24 给出了 TimerA 和 TimerB 的框图。

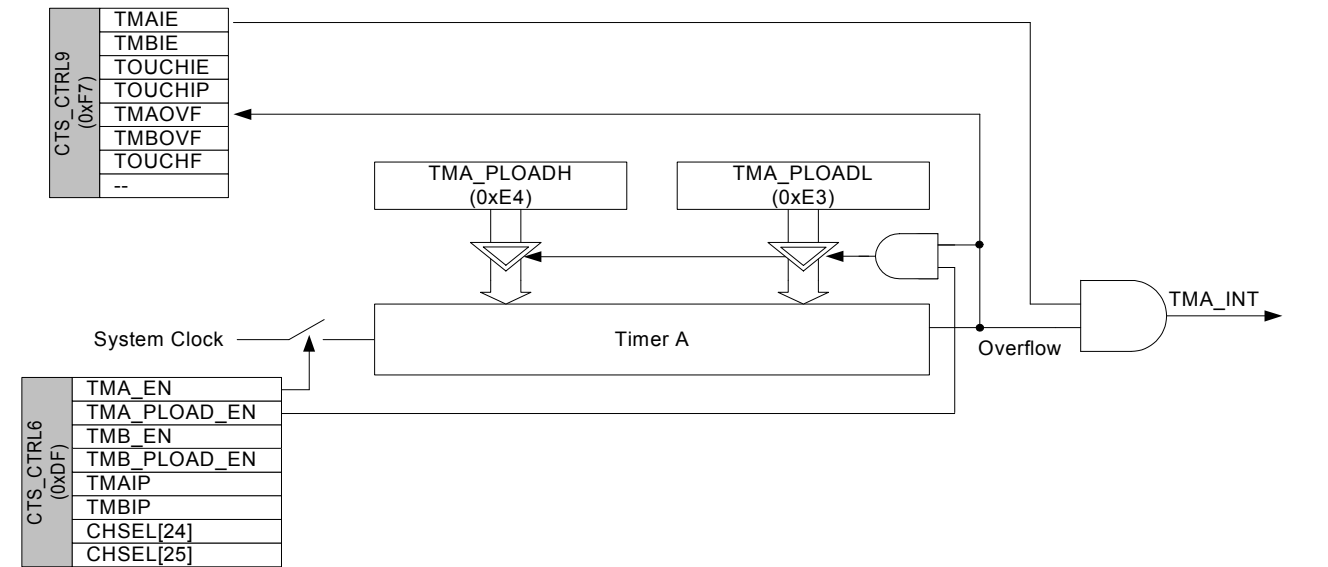


图 5- 23 Timer A 的框图

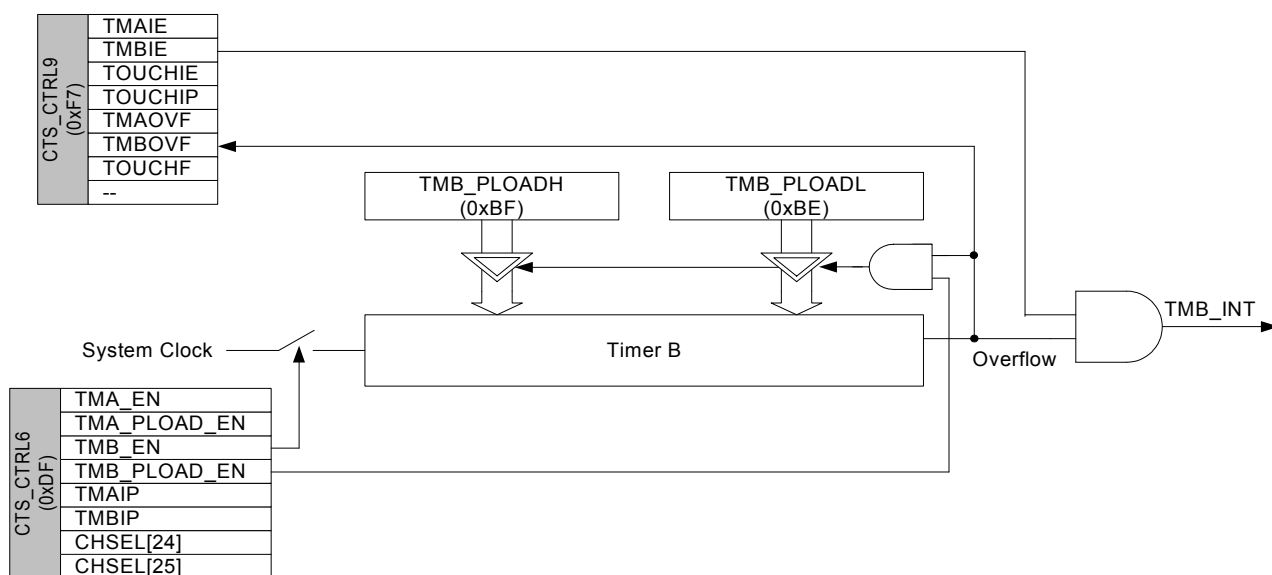


图 5- 24 Timer B 的框图

CTS_CTRL6			地址: 0xF7		电容式触摸感应控制寄存器 - 6			
Bit	7	6	5	4	3	2	1	0
功能	CHSEL[25:24]		TMBIP	TMAIP	TMB_PLOAD_EN	TMB_EN	TMA_PLOAD_EN	TMA_EN
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:6	CHSEL[25:24]	R/W	触摸通道选择, 见表 5- 136 描述	
5	TMBIP	R/W	Timer B 优先级控制 (1: 高优先级)	
4	TMAIP	R/W	Timer A 优先级控制 (1: 高优先级)	
3	TMB_PLOAD_EN	R/W	Timer B 自动重载功能使能位	
2	TMB_EN	R/W	Timer B 使能位	
1	TMA_PLOAD_EN	R/W	Timer A 自动重载功能使能位	
0	TMA_EN	R/W	Timer A 使能位	

表 5- 100 CTS_CTRL6 寄存器

CTS_CTRL9			地址: 0xF7		电容式触摸感应控制寄存器 - 9			
Bit	7	6	5	4	3	2	1	0
功能	--	TOUCHF	TMBOVF	TMAOVF	TOUCHIP	TOUCHIE	TMBIE	TMAIE
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	--	R/W	预留	
6	TOUCHF	R/W	触摸中断标志	
5	TMBOVF	R/W	Timer B 溢出标志	
4	TMAOVF	R/W	Timer A 溢出标志	

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
3	TOUCHIP	R/W	触摸优先级控制(1:高优先级)	
2	TOUCHIE	R/W	使能触摸中断	
1	TMBIE	R/W	使能 Timer B 中断	
0	TMAIE	R/W	使能 Timer A 中断	

表 5- 101 CTS_CTRL9 寄存器

TMA_PLOADH			地址: 0xE4		Timer A 预装载数据高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMA_PLOAD[15:8]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMA_PLOADH	R/W	Timer A 预装载数据高字节	

表 5- 102 TMA_PLOADH 寄存器

TMA_PLOADL			地址: 0xE3		Timer A 预装载数据低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMA_PLOAD[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMA_PLOADL	R/W	Timer A 预装载数据低字节	

表 5- 103 TMA_PLOADL 寄存器

TMB_PLOADH			地址: 0xBF		Timer B 预装载数据高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMB_PLOAD[15:8]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMB_PLOADH	R/W	Timer B 与装载数据高字节	

表 5- 104 TMB_PLOADH 寄存器

TMB_PLOADL			地址: 0xBE		Timer B 预装载数据低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMB_PLOAD[7:0]							
默认	0	0	0	0	0	0	0	0
Bit	功能	类型	描述					条件
7:0	TMB_PLOADL	R/W	Timer B 预装载数据高字节					

表 5- 105 TMB_PLOADL 寄存器

GPM8F3733A/GPM8F3717A/GPM8F3709A

TMAL			地址: 0xE1		Timer A 低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMAL[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMAL[7:0]	R/W	Timer A 装载值-低字节	

表 5- 106 TMAL 寄存器

TMAH			地址: 0xE2		Timer A 高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMAH[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMAH[7:0]	R/W	Timer A 装载值-高字节	

表 5- 107 TMAH 寄存器

TMBL			地址: 0xB6		Timer B 低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMBL[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMBL[7:0]	R/W	Timer B 装载值 - 低字节	

表 5- 108 TMBL 寄存器

TMBH			地址: 0xB7		Timer B 高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMBH[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMBH[7:0]	R/W	Timer B 装载值 - 高字节	

表 5- 109 TMBH 寄存器

5.11. UART0

UART0/1 与标准 8051 UART 有着相同的功能。串行口是全双工的，意味着发送和接收能同时进行。双缓冲式的接收意味着在前一个接收字节从接收寄存器中读出之前，能够开始接收第二个字节。写入 SBUF0/1 将加载发送寄存器，读取 SBUF0/1 是读一个物理上独立的接收寄存器。UART0/1 模块的示意图如图 5- 25 和 5-26. 串行口有 4 种工作模式：1 种同步模式和 3 种异步模式。Mode 2 和 Mode 3 可以进行多机通信，该特点可由设置 SCON0/1 寄存器的 SM02/SM12 位来使能。主机端首先发送一个地址字节来识别目标从机。地址字节不同于数据字节，地址字节的第 9 位是 1，而数据字节的第 9 位是 0。当 SM02/SM12 = 1 时，数据字节不会导致任何从机中断，而地址字节则会中断所有从机。被寻址的从机会清除 SM02/SM12 位，并准备接收后续的数据。没有被寻址的从机保留其 SM02 的设置，并忽略即将传入的数据。

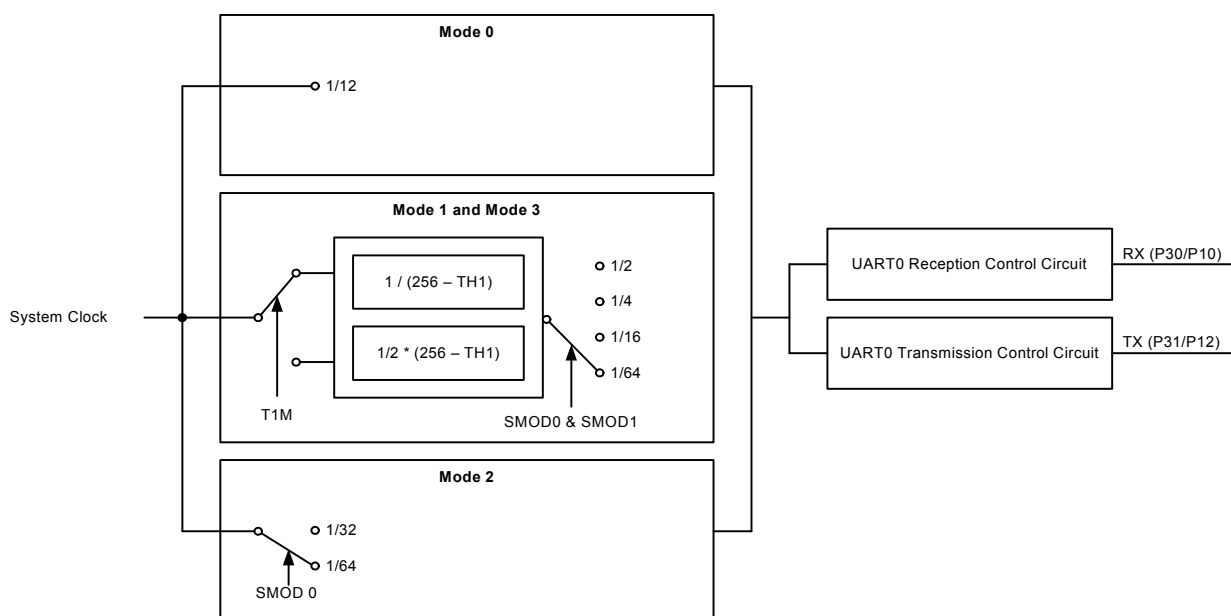


图 5- 25 UART0 模块的示意图

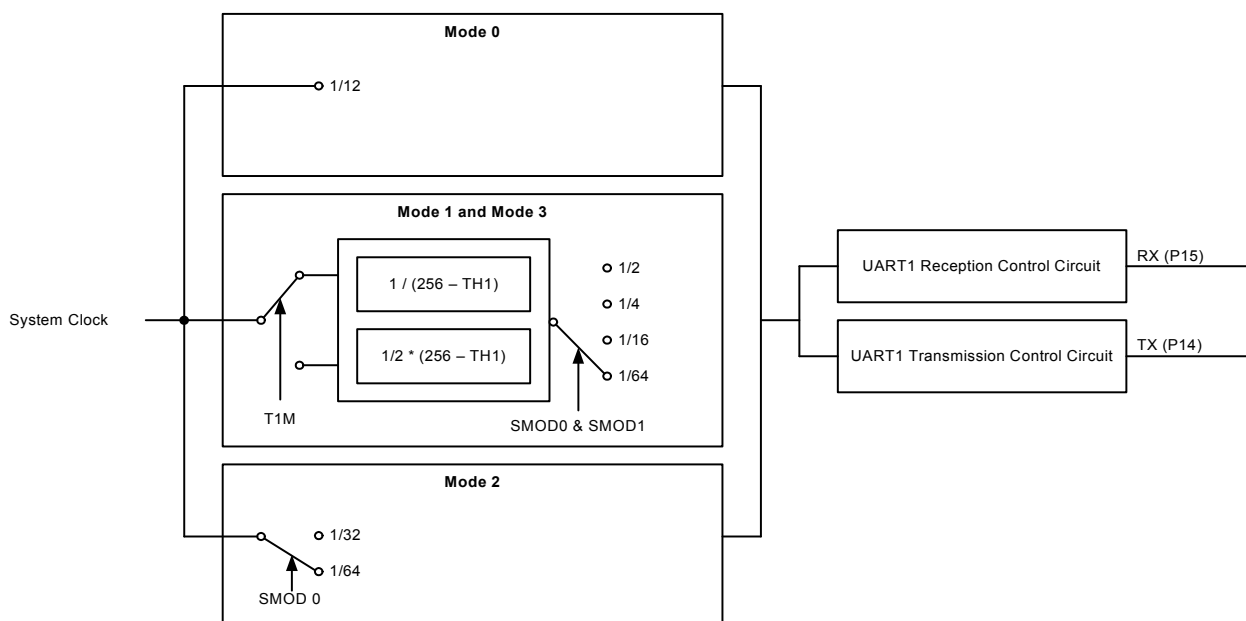


图 5- 26 UART1 模块的示意图

5.11.1. UART0/1: 模式 0(同步移位寄存器)

该模式用作移位寄存器 I/O 控制，而并不是真正的通信。波特率固定在系统时钟频率的 12 分频，TXD 输出是一个移位时钟。8 位数据按照 LSB 顺序传送，通过设置 SCON0/1 的标志 (RI0/RI1 =0 和 REN0/REN1 =1) 对接收端进行初始化。图 5- 27 给出了 UART0/1 模式 0 的发送时序图。

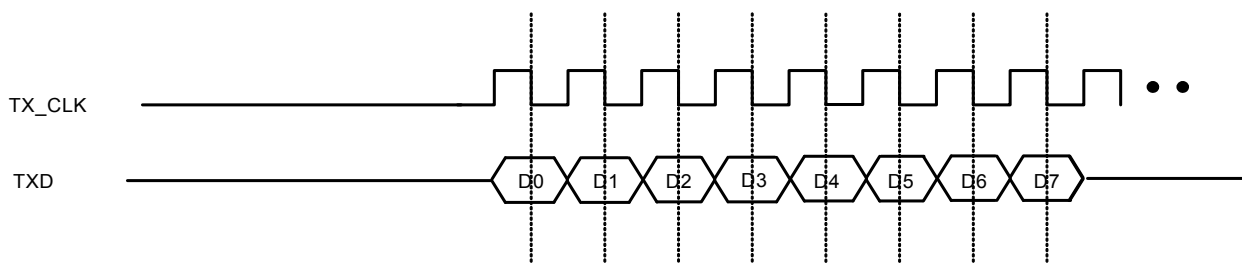


图 5- 27 UART0/1 模式 0 发送时序图

5.11.2. UART0/1: 模式 1(8-Bit UART, 可变波特率, Timer1 时钟源)

在模式 1 下，TXD 作为串行输出。发送 10 个位：一个起始位(总是 0)，8 个数据位(首先是 LSB)和一个停止位 (总是 1)。接收时，起始位用来同步接收，8 位数据可通过读 SBUF0 获得，停止位设置 SFR SCON0/1 的 RB08/RB18 标志位。波特率是可变的，取决于 Timer 1 模式。PCON (0x87) 的 SMOD0 和 SMOD1 位用来设置四种波特率： $T_{1ov}/2$ 、 $T_{1ov}/4$ 、 $T_{1ov}/16$ 、 $T_{1ov}/64$ 。图 5- 28 给出了 UART 模式 1 的发送时序图。

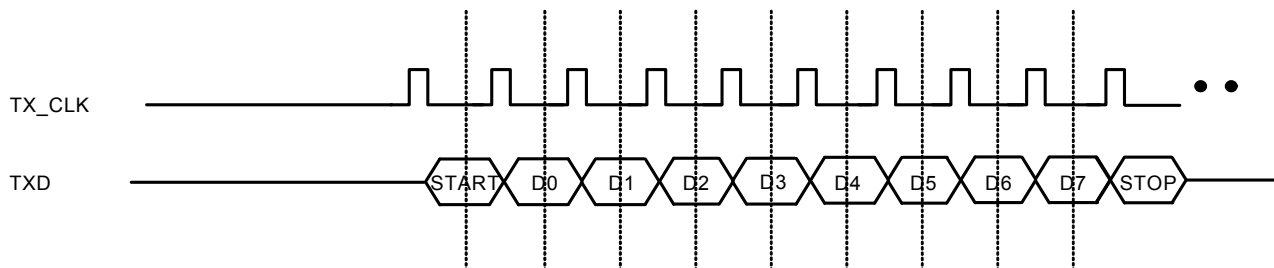


图 5- 28 UART0/1 模式 1 的发送时序图

5.11.3. UART0/1: 模式 2(9-Bit UART, 固定波特率)

该模式与模式 1 类似，有两处不同。波特率固定在系统时钟频率的 1/32 或 1/64，11 位的发送或接收：一个起始位(0)，8 个数据位(首先是 LSB)，一个可编程的第 9 位以及一个停止位(1)。第 9 位用来控制 UART0/1 接口的奇偶性：发送时，SCON0/1 的 TB08/TB18 位作为第 9 位输出；接收时，第 9 位则影响 SCON0/1 的 RB08/RB18。图 5- 29 给出了 UART0/1 模式 2 的发送时序图。

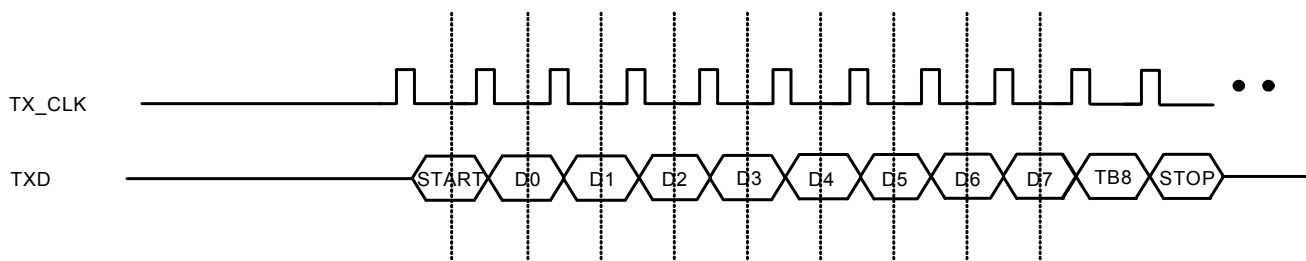


图 5- 29 UART0/1 模式 2 发送时序图

5.11.4. UART0/1: 模式 3(9-Bit UART, 可变波特率, Timer1 时钟源)

模式 2 和模式 3 的唯一区别是波特率在模式 3 下是可变的。当 REN0 =1 时使能数据接收。波特率是可变的且取决于 Timer 1 模式, PCON (0x87) 的 SMOD0 和 SMOD1 位用来设置四种波特率: $T1_{ov}/2$ 、 $T1_{ov}/4$ 、 $T1_{ov}/16$ 、 $T1_{ov}/64$ 。图 5-30 给出了 UART0/1 模式 3 的发送时序图。

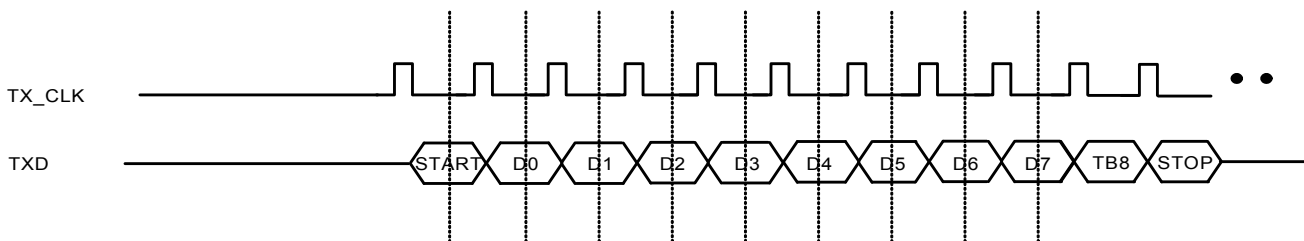


图 5-30 UART0/1 模式 3 的发送时序图

5.11.5. UART0/1 相关寄存器

UART0/1 相关的寄存器有: SBUF0(0x99)、SCON0(0x98)、SBUF1(0xC1)、SCON1(0xC0)、PCON(0x87)、IE(0xA8) 和 IP(0xB8)。UART0/1 数据缓冲器(SBUF0/1)由 2 个独立的寄存器构成: 发送和接收寄存器。写入 SBUF0/1 中的数据被作为为 UART0/1 输出寄存器中的数据并开始发送。读取 SBUF0/1 中的数据, 即读取 UART0/1 接收寄存器。

SBUF0			地址: 0x99		UART0 缓冲寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SBUF0[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
2:0	SBUF0[7:0]	R/W	UART 0 缓存	

表 5-110 SBUF0 寄存器

SCON0			地址: 0x98		UART0 配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SM00	SM01	SM02	REN0	TB08	RB08	TI0	RI0
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:6	SM0[1:0]	R/W	模式和波特率设置位, 如下表所示	
5	SM02	R/W	使能多机通信功能	
4	REN0	R/W	使能串行接收	
3	TB08	R/W	模式 2 和模式 3 下第 9 个发送数据位	
2	RB08	R/W	在模式 0, 这一位无效。 在模式 1, 如果 SM02=0, RB08 是停止位 在模式 2 和模式 3 下, 是第 9 个接收的数据位	
1	TI0	R/W	UART0 发送中断标志	

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
0	RIO	R/W	UART0 接收中断标志	

表 5- 111 SCON0 寄存器

SBUF1			地址: 0xC1		UART1 缓冲寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SBUF1[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
2:0	SBUF1[7:0]	R/W	UART 1 缓存	

表 5- 112 SBUF1 寄存器

SCON1			地址: 0xC0		UART1 配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SM10	SM11	SM12	REN1	TB18	RB18	TI1	RI1
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:6	SM1[1:0]	R/W	模式和波特率设置, 如下表所示	
5	SM12	R/W	使能多机通信功能	
4	REN1	R/W	使能串行接收	
3	TB18	R/W	模式 2 和模式 3 下第 9 个发送数据位	
2	RB18	R/W	在模式 0, 这一位无效。 在模式 1, 如果 SM02=0, RB08 是停止位 在模式 2 和模式 3 下, 是第 9 个接收的数据位	
1	TI1	R/W	UART1 发送中断标志	
0	RI1	R/W	UART1 接收中断标志	

表 5- 113 SCON1 寄存器

SM00/SM10	SM01/SM11	模式	功能	波特率
0	0	0	移位寄存器	SYSCLK/12
0	1	1	8-bit UART	可变
1	0	2	9-bit UART	SYSCLK/32 (SM0D0=0) SYSCLK/64 (SM0D0=1)
1	1	3	9-bit UART	可变

GPM8F3733A/GPM8F3717A/GPM8F3709A

可变：模式 1 and 模式 3 (T1M=0)

SMOD1	SMOD0	波特率
0	0	$T1_{ov}/64$ ($T1_{ov}=SYSCLK/(256-TH1)$)
0	1	$T1_{ov}/16$ ($T1_{ov}=SYSCLK/(256-TH1)$)
1	0	$T1_{ov}/4$ ($T1_{ov}=SYSCLK/(256-TH1)$)
1	1	$T1_{ov}/2$ ($T1_{ov}=SYSCLK/(256-TH1)$)

注意：如果 SMOD1=SMOD0=1，TH1 应该比 0x10 大。

波特率设置举例 (SYSCLK = 32MHz, T1M=0)

比特率	波特率	定时器重载设置 (TH1)	实际比特率	误差 (%)
4800	$T1_{ov}/64$	0x98 (152)	4807.69	0.16%
9600	$T1_{ov}/16$	0x30 (48)	9615.38	0.16%
19200	$T1_{ov}/16$	0x98 (152)	19230.77	0.16%
38400	$T1_{ov}/16$	0xCC (204)	38461.54	0.16%
57600	$T1_{ov}/16$	0xDD (221)	57142.86	-0.79%
115200	$T1_{ov}/4$	0xBB (187)	115942.03	0.64%

波特率设置举例 (SYSCLK = 16MHz, T1M=0)

比特率	波特率	定时器重载设置 (TH1)	实际波特率	误差 (%)
4800	$T1_{ov}/16$	0x30 (48)	4807.69	0.16%
9600	$T1_{ov}/16$	0x98 (152)	9615.38	0.16%
19200	$T1_{ov}/16$	0xCC (204)	19230.77	0.16%
38400	$T1_{ov}/16$	0xE6 (230)	38461.54	0.16%
57600	$T1_{ov}/2$	0x75 (117)	57553.96	-0.08%
115200	$T1_{ov}/2$	0xBB (187)	115942.03	0.64%

波特率设置举例 (SYSCLK = 8MHz)

比特率	波特率	定时器重载设置 (TH1)	实际波特率	误差 (%)
4800	$T1_{ov}/16$	0x98 (152)	4807.69	0.16%
9600	$T1_{ov}/16$	0xCC (204)	9615.38	0.16%
19200	$T1_{ov}/16$	0xE6 (230)	19230.77	0.16%
38400	$T1_{ov}/16$	0xF3 (243)	38461.54	0.16%
57600	$T1_{ov}/2$	0xBB (187)	57971.01	0.64%
115200	$T1_{ov}/2$	0xDD (221)	114285.71	-0.79%

PCON			地址：0x87		电源配置寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SMOD0	SMOD1	--	PWE	--	--	STOP	--
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7	SMOD0	R/W	当时钟源是 Timer1 时, UART 的波特率位	
6	SMOD1	R/W	当时钟源是 Timer1 时, UART 的波特率位	
5	---	R/W	预留	
4	PWE	R/W	编程使能位 0: 在 MOVX 指令期间, 禁止 Flash 写有效 1: 在 MOVX 指令期间, 使能 Flash 写有效	
3: 2	---	R/W	预留	
1	STOP	R/W	停止模式使能位 0: 禁止 1: 使能	
0	---	R/W	预留	

表 5- 114 PCON 寄存器

SYS_CTRL6			地址 : 0xFF		系统控制-6 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	UART0_IF_SEL	UART1_IF_EN	UART0_IF_EN	T01_CK_SEL	---	AERR_RSTEN	WDOG_CKEN	---
默认	0	0	0	0	0	0	1	0
Key Code	0x8F, 0x32 , 0x50							

Bit	功能	类型	描述	条件															
7	UART0_IF_SEL	R/W	UART0 IO 接口选择信号 <table><tr><td>UART0_IF_SEL</td><td>UART_TX</td><td>UART_RX</td></tr><tr><td>0</td><td>P31</td><td>P30</td></tr><tr><td>1</td><td>P12</td><td>P10</td></tr></table>	UART0_IF_SEL	UART_TX	UART_RX	0	P31	P30	1	P12	P10							
UART0_IF_SEL	UART_TX	UART_RX																	
0	P31	P30																	
1	P12	P10																	
6	UART1_IF_EN	R/W	UART1 接口使能信号 0 : 禁止 1 : 使能 <table><tr><td>UART1_TX</td><td>UART1_RX</td></tr><tr><td>P14</td><td>P15</td></tr></table>	UART1_TX	UART1_RX	P14	P15												
UART1_TX	UART1_RX																		
P14	P15																		
5	UART0_IF_EN	R/W	UART0 接口使能信号 0 : 禁止 1 : 使能																
4	T01_CK_SEL	R/W	Timer0/1 时钟源选择信号。 结合 TOM/T1M of CKCON(0x8E) 使用。 <table><tr><td>TOM /T1M</td><td>T01_CK_SEL</td><td>Timer0/1 时钟</td></tr><tr><td>0</td><td>0</td><td>系统时钟/ 8</td></tr><tr><td>0</td><td>1</td><td>系统时钟/ 2</td></tr><tr><td>1</td><td>0</td><td>系统时钟/ 4</td></tr><tr><td>1</td><td>1</td><td>系统时钟/ 1</td></tr></table>	TOM /T1M	T01_CK_SEL	Timer0/1 时钟	0	0	系统时钟/ 8	0	1	系统时钟/ 2	1	0	系统时钟/ 4	1	1	系统时钟/ 1	
TOM /T1M	T01_CK_SEL	Timer0/1 时钟																	
0	0	系统时钟/ 8																	
0	1	系统时钟/ 2																	
1	0	系统时钟/ 4																	
1	1	系统时钟/ 1																	
3	--	R/W	预留																
2	AERR_RSTEN	R/W	Flash 地址溢出复位使能 0 : 禁止 1 : 使能																
1	WDOG_CKEN	R/W	看门狗控制器时钟使能控制位 0 : 禁止 1 : 使能																
0	--	R/W	预留																

表 5- 115 SYS_CTRL6 寄存器

5.12. SPI

GPM8F3733A / GPM8F3717A / GPM8F3709A 内置有串行外设接口 (SPI) 控制器，用来与其它芯片和器件进行通信。SPI 控制器包括 4 种主模式和一个从模式。SPI 有 4 个控制信号，包括 SPI_CSB, SPI_RXD, SPI_CLK, and SPI_TXD, 这四个信号与 P1[7:4] 复用。当 SPI 模块由相应的控制位使能时，这 4 个引脚就不能为 GPIO 了。换句话说，相应的 GPIO 控制寄存器中的任何设置将无效。SPI 有如下几个特点：

- 可编程的主时钟的相位和极性
- 可编程的主 SPI_CLK 时钟频率

在主模式下，移位时钟 (SPI_CLK) 由 SPI 模块产生。有 2 个控制位用来控制时钟的相位和极性。SPI_START 置位 (SPI_CTRL[0]=1, 0x9A) 后发送立即开始。在 8 个 SPI_CLK 周期里，通过 SPI_TXD 引脚 SPI 模块从 MSB 到 LSB 移位发送 8-bit 的数据。编程人员通过设置 SPI_RXD=1 可以从 SPI_RXD 控制寄存器中读取 SPI 数据。如下 4 个图描述了 SPI 主模式在不同工作类型 (极性控制位=“1” 或 “0”，相位控制位=“1” 或 “0”) 下的时序图。相关的寄存器有 SPI_CTRL 寄存器、SPI_STS 寄存器、SPI_TXD 寄存器和 SPI_RXD 寄存器。

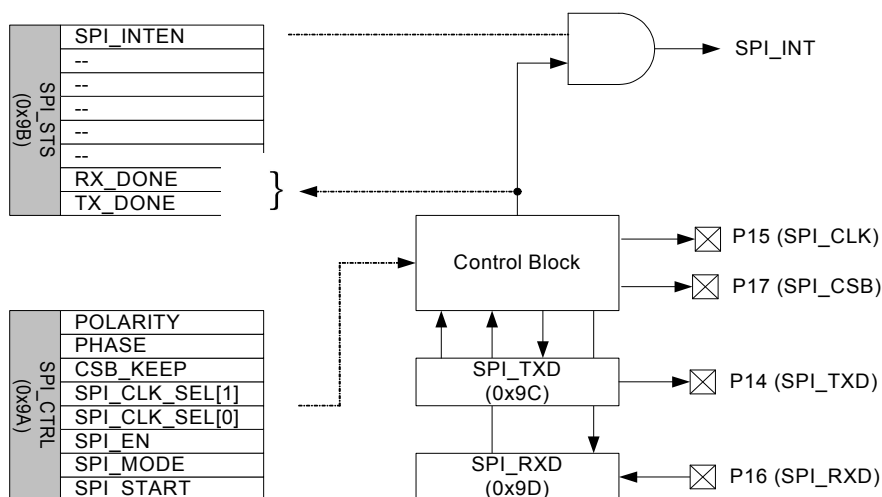


图 5- 31 SPI 控制器框图

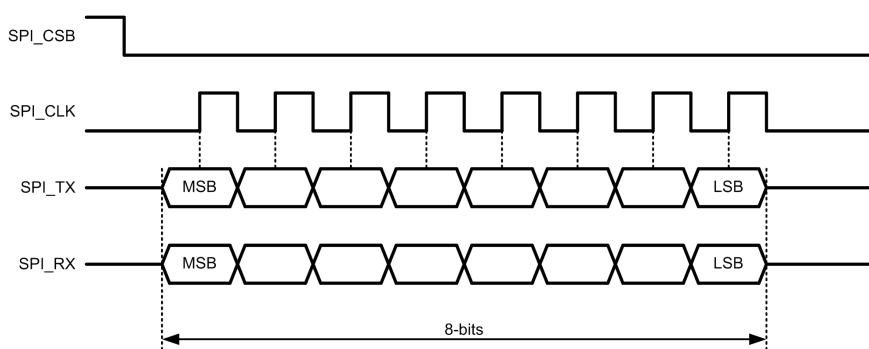


图 5- 32 主模式/从模式，POLARITY=0, PHASE=0

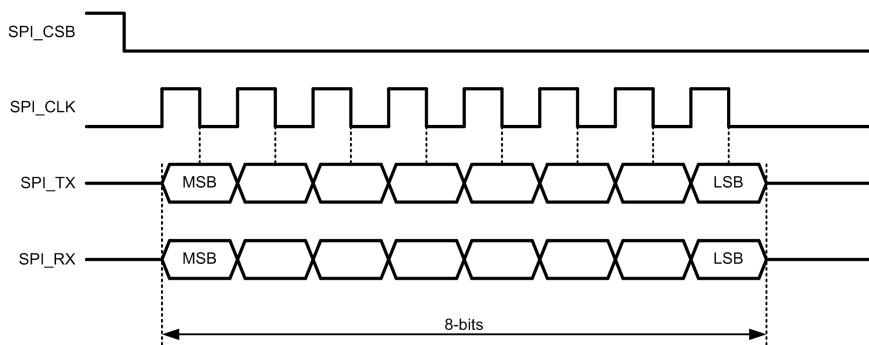


图 5- 33 主模式, POLARITY=0, PHASE=1

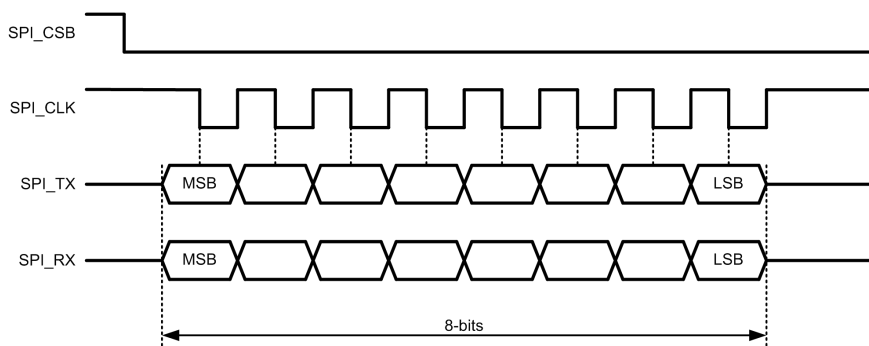


图 5- 34 主模式, POLARITY=1, PHASE=0

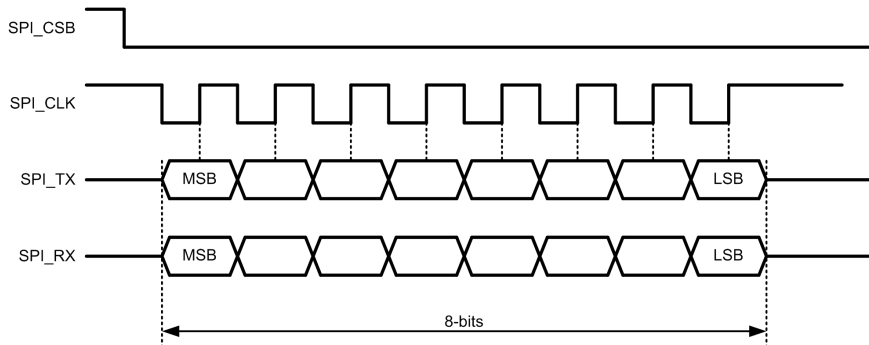


图 5- 35 主机模式, POLARITY=1, PHASE=1

SPI_CTRL			地址: 0x9A		SPI 控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	POLARITY	PHASE	CSB_KEEP	SPI_CLK_SEL[1:0]		SPI_EN	SPI_MODE	SPI_START
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	POLARITY	R/W	SPI_CLK 初始化状态 0: 低态 1: 高态	

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
6	PHASE	R/W	SPI CLK 类型控制 0: 第一个边沿取样 1: 第二个边沿取样	
5	CSB_KEEP	R/W	SPI CSB 保持低	
4:3	SPI_CLK_SEL[1:0]	R/W	SPI 时钟输出选择 00: 系统时钟 / 2 01: 系统时钟 / 4 10: 系统时钟 / 8 11: 系统时钟 / 16	
2	SPI_EN	R/W	P1[7:4]作为 SPI 信号脚使能 P1[5] : SPI_CLK P1[4] : SPI_CSB P1[6] : SPI_RXD P1[7] : SPI_TXD	
1	SPI_MODE	R/W	SPI 操作模式. 0: 主机 1: 从机	
0	SPI_START	R/W	SPI 使能(写)/SPI 忙标志(读)	

表 5- 116 SYSCON1 寄存器

SPI_STS			地址: 0x9B		SPI 状态寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SPI_INTEN	--	--	--	--	--	RX_DONE	TX_DONE
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	SPI_INTEN	R/W	SPI 中断使能 0 : 禁止 1 : 使能	
6:2	-	R/W	预留	
1	RX_DONE	R/W	从机模式下, SPI 完成数据接收 读: 0 : 空闲 / 忙 1 : 完成 写: 0 : 清零 1 : 无效	
0	TX_DONE	R/W	主机模式下, SPI 完成数据接收. 读: 0 : 空闲 / 忙 1 : 完成 写: 0 : 清零 1 : 无效	

表 5- 117 SPI_STS 寄存器

SPI_TXD			地址：0x9C		SPI 传输数据寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SPI_TXD[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	SPI_TXD[7:0]	R/W	SPI 传输数据	

表 5- 118 SPI_TXD 寄存器

SPI_RXD			地址：0x9D		SPI 接收数据寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SPI_RXD[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	SPI_RXD	R/W	SPI 接收数据	

表 5- 119 SPI_RXD 寄存器

5.13. I2C

GPM8F3733A / GPM8F3717A / GPM8F3709A 配置了一个 I2C 接口。I2C 通信仅仅需要两条线来实现（SCK 和 SDA）。为了避免可能出现的混乱、数据丢失和信息阻塞，主机和从机必须预先制定一个协议。通过 I2C 发起数据传输的主机有责任终结传输。

5.13.1. I2C 总线协议

起始条件可以通过 SDA 线，传输一个字节的地址，停止条件可以终止传输。“起始”条件是当 SCK 是高电平时，SDA 线上从高到低的一个跳变。“停止”条件是当 SCK 是高电平时，SDA 线上从低到高的一个跳变。起始和停止条件总是由主机产生。当起始条件产生，I2C 总线处于忙状态。停止条件产生后，I2C 总线处于空闲状态。图 5- 36 给出了起始和停止条件。

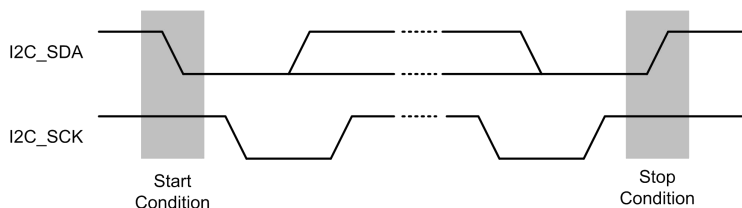


图 5- 36 起始和停止条件

当主机发布了一个起始条件，它应该发送一个从机地址来通知从机。一个字节的地址区包含 7bit 的地址和一个 bit 的传输方向指示符（也就是写或者读）。如果 bit8 是 0，表明是一个写操作（传输操作）；如果 bit8 是 1，表明是一个读数据的请求（接收操作）。SDA 线上的每一个字节都应该是 8bit 长度的数据。每次传输的字节数是不受限制的，起始条件后的第一个字节应该是地址数据。当 I2C 总线工作在主机模式，它可以发送地址数据。每一个字节传输完后，应该跟随一个应答位（ACK）。数据和地址的最高有效位（MSB）应该总是最先传送。完成一个字节的传输后，接收端应该发送一个 ACK 位给发送端。ACK 脉冲应该在 SCL 线的第九个时钟出现。传输一个字节的地址需要 8 个时钟。主机应该产生传输 ACK 位所需的时钟脉冲。图 5- 37 和图 5- 38 给出了 I2C 主机模式的数据传输格式。

在主机模式，数据传输完后，在使能中断的条件下 I2C 将发布一个中断。CPU 接收到中断请求后，应该在清除中断前，写入新的数据到 I2C_DATA 寄存器。在接收模式，接收到一个数据后，在使能中断的条件下 I2C 将发布一个中断。CPU 接收到中断请求后，应该在清除中断前，从 I2C_DATA 读出新的数据。

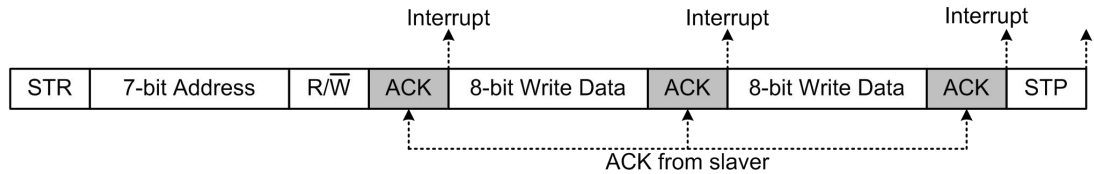


图 5- 37 I2C 主机模式下的写数据

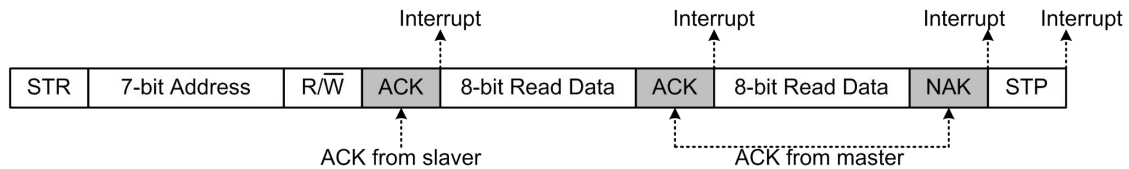


图 5- 38 I2C 主机模式下的读数据

图 5-39 和图 5-40 给出了 I2C 从机模式下的数据传输。

在从机模式下，重新得到数据后（包括地址传输），在使能中断的条件下 I2C 接口将发布一个中断。CPU 接收到中断请求后，在清除中断前，从 I2C_DATA 读取数据。CPU 接收到中断请求后，在从机模式读数据期间，I2C 控制器应该写入一个新数据。

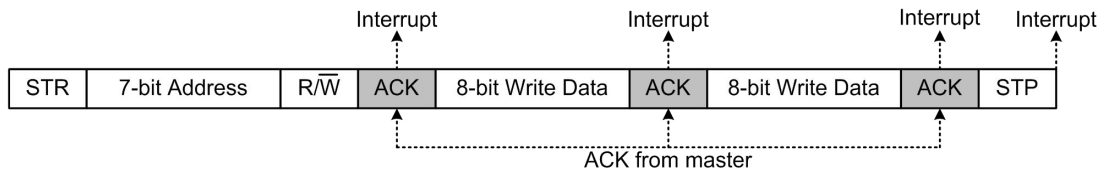


图 5- 39 I2C 从机模式写数据

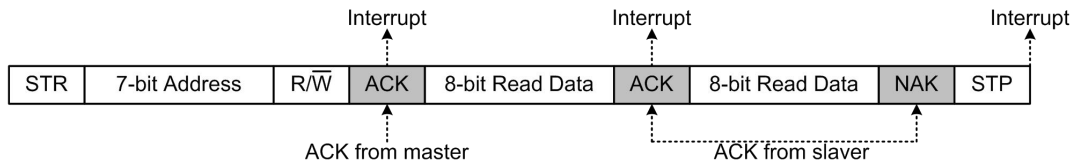


图 5- 40 I2C 从机模式读数据

I2C 相关寄存器

I2C_DEBOUNCE			地址：0xCF		I2C 去抖动计数寄存器			
Bit	7	6	5	4	3	2	1	0
功能	--	--	I2C_DB_CNT[5:0]					
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:6	--	R/W	预留	
5:0	I2C_DEBOUNCE	R/W	I2C 输入 SCK 和 SDA 去抖动计数。	

表 5- 120 I2C_DEBOUNCE 寄存器

I2C_CTRL			地址: 0xD1		I2C 控制寄存器			
Bit	7	6	5	4	3	2	1	0
功能	MST_STR	MST_STP	MST_NACK	MODE	I2C_CLK_SEL[1:0]		I2C_TRIGGER	I2C_EN
默认	0	0	1	0	0	0	0	0

Bit	功能	类型	描述	条件
7	MST_STR	R/W	I2C 发起启动命令设备地址使能位，传输完成后硬件自动清除。 0：禁止 1：使能	
6	MST_STP	R/W	I2C 发起停止命令使能位，传输完成后硬件自动清除。 0：禁止 1：使能	
5	MST_NACK	R/W	I2C 发起无应答使能位。传输完成后硬件自动清除。 0：主机响应 1：主机不响应	
4	MODE	R/W	I2C 操作模式选择控制位。 0: 主机 1: 从机	
3:2	I2C_CLK_SEL[1:0]	R/W	I2C 时钟选择控制位 00：I2C 时钟是系统时钟 / 40 01：I2C 时钟是系统时钟 / 80 10：I2C 时钟是系统时钟 / 320 11：I2C 时钟是系统时钟 / 640	
1	I2C_TRIGGER	R/W	I2C 开始传输触发位，只有主机模式下有效。当 I2C_EN=1，I2C 主机将开始传输或者接收。数据传输后此位可软件或硬件自动清零。 0：禁止 1：使能	
0	I2C_EN	R/W	I2C 使能控制位 0：禁止 1：使能	

表 5- 121 I2C_CTRL 寄存器

I2C_STS			地址: 0xD2		I2C 状态寄存器			
Bit	7	6	5	4	3	2	1	0
功能	SLV_DID_OK	SLV_DAT_OK	SLV_STP_OK	ERR_SDID_IE	I2C_IF_SEL	I2C_INT_EN	NO_ACK	TS_DONE
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	SLV_DID_OK	R/W	表明 i2c 控制器接收设备 ID，仅用于从机模式。 读： 0 : 没有声明设备 ID 1 : 声明设备 ID 写： 0 : 清零 1 : 无效	
6	SLV_DAT_OK	R/W	表明 i2c 控制器已经接收或传输数据，仅用于从机模式。 读： 0 : 数据在传输或空闲 1 : 数据传输完成 写： 0 : 清零 1 : 无效	
5	SLV_STP_OK	R/W	表明 i2c 控制器接收到停止命令，仅用于从机模式。 读： 0 : 没有声明停止命令 1 : 声明停止命令 写： 0 : 清零 1 : 无效	
4	ERR_SDID_IE	R/W	从机模式下，设备 ID 错误中断使能 0 : 禁止 1 : 使能	
3	I2C_IF_SEL	R/W	I2C 接口选择信号 0 : P2_3 = SDA / P2_2 = SCK 1 : P3_3 = SDA / P3_2 = SCK	
2	I2C_INT_EN	R/W	I2C 中断使能控制位。 0 : 禁止 1 : 使能	
1	NO_ACK	R/W	I2C 没有接收到应答信号 读： 0 : 应答 1 : 无应答 写： 0 : 清零 1 : 无效	
0	TS_DONE	R/W	I2C 传输完成标志。 读： 0 : i2c 空闲或正在进行 1 : i2c 完成数据传输 写： 0 : 清零 1 : 无效	

表 5- 122 I2C_STS 寄存器

I2C_DID			地址: 0xD3		I2C 设备 ID 寄存器			
Bit	7	6	5	4	3	2	1	0
功能	DEV_ID							R_W
默认	0	0	0	0	0	0	0	1

Bit	功能	类型	描述	条件
7:1	DEV_ID[6:0]	R/W	I2C 设备 ID 寄存器。在主机模式，设备 ID 通知从机将被连接。在从机模式，辨别来自外部主机发送的 ID。	
0	R_W	R/W	I2C 读 / 写 控制信号，此寄存器用于主机和从机模式。	

表 5- 123 I2C_DID 寄存器

I2C_DATA			地址: 0xD4		I2C 数据寄存器			
Bit	7	6	5	4	3	2	1	0
功能	I2C_DATA[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	I2C_DATA	R/W	I2C 读 / 写数据寄存器。此寄存器用于主机和从机模式。	

表 5- 124 I2C_DATA 寄存器

5.14. ADC

GPM8F3733A / GPM8F3717A / GPM8F3709A 内置了一个模数转换器(ADC)，提供一些其它模拟功能等通用应用。

- 支持编程采样保持时间和 ADC 时钟功能
 - GPM8F3733A / GPM8F3717A
- 9 通道, 12-bit 精度 (11-bit 无丢码) ADC
 - GPM8F3709A
- 8 通道, 12-bit 精度(11-bit 无丢码) ADC

5.14.1. ADC 控制

GPM8F3733A / GPM8F3717A / GPM8F3709A 内置有 9 或者 8 个通道的 12-bit SAR ADC，分别定义为通用线路输入 P10...P17 和 Cs。除了 Cs 其他 8 个通道非常适用于系统电压侦测以及其它一些通用应用。图 5- 41 和图 5-42 给出了相关的时序和方框图。

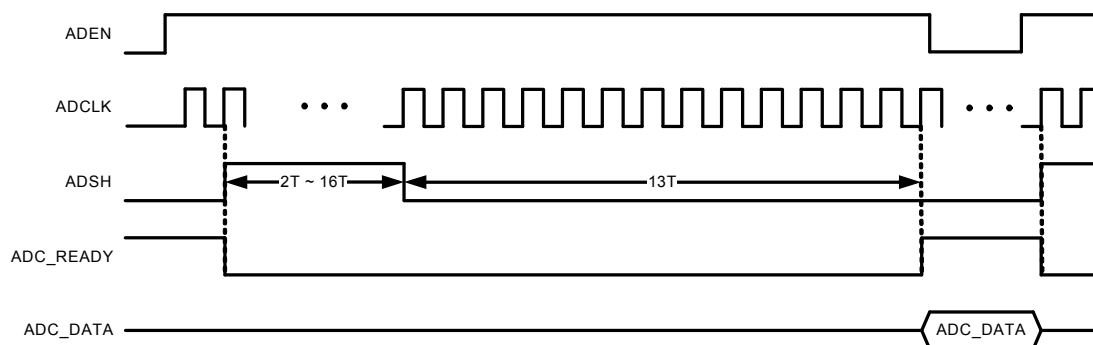


图 5- 41 ADC 时序图

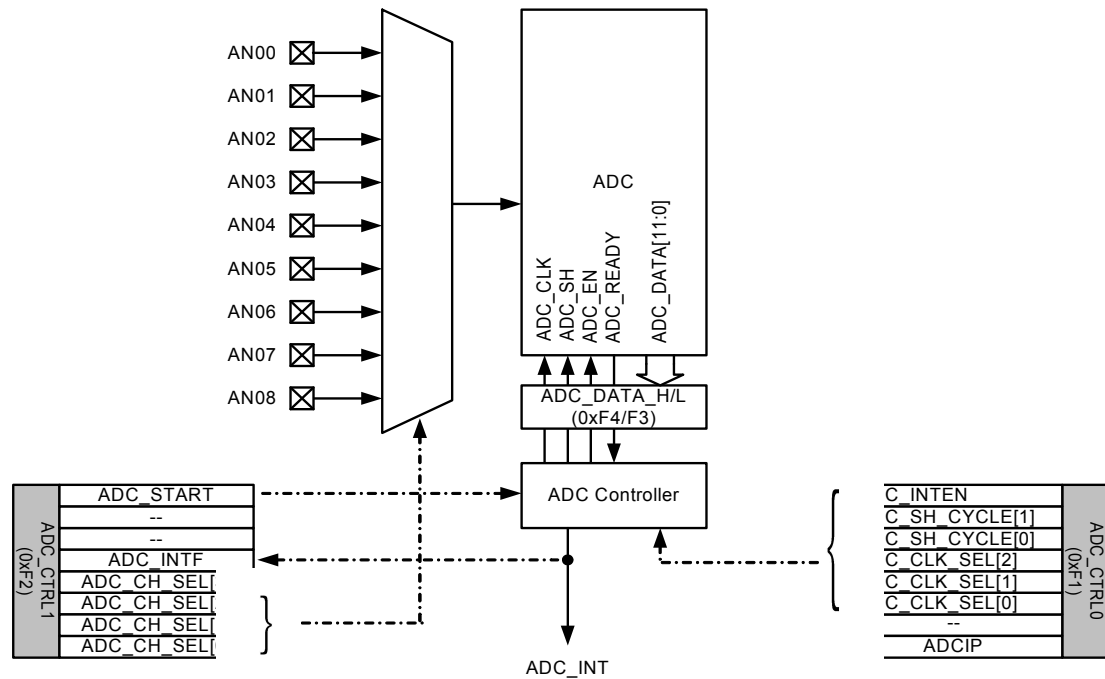


图 5-42 ADC 方框图

5.14.2. ADC 相关寄存器

ADC_CTRL0			地址: 0xF1		ADC 控制寄存器 0			
Bit	7	6	5	4	3	2	1	0
功能	ADC_INTEN	ADC_SH_CYCLE[1:0]		ADC_CLK_SEL[2:0]			--	ADCIP
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述		条件
7	ADC_INT_EN	R/W	ADC 中断使能控制位。 0 : 禁止 1 : 使能		
6:5	ADC_SH_CYCLE	R/W	ADC 采样和保持周期时间选择控制位		
			ADC_SH_CYCLE	Cycle (ADC_CLK)	
			00	2	
			01	4	
			10	8	
			11	16	

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件	
4:2	ADC_CLK_SEL	R/W	ADC 时钟选择控制位		
			ADC_CLK_SEL		ADC_CLK
			000		系统时钟 / 2
			001		系统时钟 / 4
			010		系统时钟 / 8
			011		系统时钟 / 10
			100		系统时钟 / 16
			101		系统时钟 / 20
			110		系统时钟 / 32
111	系统时钟 / 64				
1	--	R/W	预留		
0	ADCIP	R/W	ADC 优先级控制（1：高有效）		

表 5- 125 ADC_CTRL0 寄存器

ADC_CTRL1			地址: 0xF2		ADC 控制寄存器 1			
Bit	7	6	5	4	3	2	1	0
功能	ADC_START	—	—	ADC_INTF	ADC_CH_SEL[2:0]			
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件																				
7	ADC_START	R/W	ADC 开始传输控制位 0: 空闲 1: 开始传输																					
6:5	—	R	预留																					
4	ADC_INTF	R/W	ADC 中断标志. 读 : 0 : 空闲 / 转换 1 : 转换完成 写 : 0 : 清零 1 : 无效																					
3	—	R/W	预留																					
2:0	ADC_CH_SEL	R/W	ADC 通道选择控制位 <table><tr><td>ADC_CH_SEL</td><td></td></tr><tr><td>0000</td><td>ADC_CH0 (P1[0])</td></tr><tr><td>0001</td><td>ADC_CH1 (P1[1])</td></tr><tr><td>0010</td><td>ADC_CH2 (P1[2])</td></tr><tr><td>0011</td><td>ADC_CH3 (P1[3])</td></tr><tr><td>0100</td><td>ADC_CH4 (P1[4])</td></tr><tr><td>0101</td><td>ADC_CH5 (P1[5])</td></tr><tr><td>0110</td><td>ADC_CH6 (P1[6])</td></tr><tr><td>0111</td><td>ADC_CH7 (P1[7])</td></tr><tr><td>1000</td><td>ADC_CH8 (P2[4], Cs PAD)</td></tr></table>	ADC_CH_SEL		0000	ADC_CH0 (P1[0])	0001	ADC_CH1 (P1[1])	0010	ADC_CH2 (P1[2])	0011	ADC_CH3 (P1[3])	0100	ADC_CH4 (P1[4])	0101	ADC_CH5 (P1[5])	0110	ADC_CH6 (P1[6])	0111	ADC_CH7 (P1[7])	1000	ADC_CH8 (P2[4], Cs PAD)	
ADC_CH_SEL																								
0000	ADC_CH0 (P1[0])																							
0001	ADC_CH1 (P1[1])																							
0010	ADC_CH2 (P1[2])																							
0011	ADC_CH3 (P1[3])																							
0100	ADC_CH4 (P1[4])																							
0101	ADC_CH5 (P1[5])																							
0110	ADC_CH6 (P1[6])																							
0111	ADC_CH7 (P1[7])																							
1000	ADC_CH8 (P2[4], Cs PAD)																							

Bit	功能	类型	描述	条件

表 5- 126 ADC_CTRL1 寄存器

ADC_DATA_L			地址: 0xF3		ADC 数据寄存器 - 低字节			
Bit	7	6	5	4	3	2	1	0
功能	ADC_DATA[3:0]				--	--	--	--
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:4	ADC_DATA[3:0]	R	ADC 输出 data[3:0]	
3:0	-	R/W	预留	

表 5- 127 ADC_DATA_L 寄存器

ADC_DATA_H			地址: 0xF4		ADC 数据寄存器 - 高字节			
Bit	7	6	5	4	3	2	1	0
功能	ADC_DATA[11:4]							
默认	0	0	0	0				

Bit	功能	类型	描述	条件
7:0	ADC_DATA[11:4]	R	ADC 输出 data[11:4]	

表 5- 128 ADC_DATA_H 寄存器

5.15. 电容触摸传感器

GPM8F3733A / GPM8F3717A / GPM8F3709A 的 TimerA 和 TimerB 不仅用于定时器还用于电容触摸传感器,其中 TimerA 和 TimerB 都是 16-bit 的向上计数的计数器,电容触摸传感器提供了进行检测电容式感应、手势判断、行为决策的能力,可以提供系统的操作性。

5.15.1. 电容测量方法

电容测量方式有三种: 张弛振荡器 (CTR), CTC 和 CTA 模式。张弛振荡器产生一个用于测量的振荡电压信号,这个信号的频率依赖于接到模块上的目标物体的电容大小。基本的原理描述如下:

- 电容式传感器振荡在一定的频率,这个频率取决于连在感应电极上的电容。
- 当触碰出现在感应电极的附近时,电容式传感器的振荡频率会改变,因为触碰会改变电极的总电容。
- 电容式传感器的振荡频率改变被用来判断触碰是否发生。

图 5- 43 和图 5-46 给出了相关的时序图和示意图。

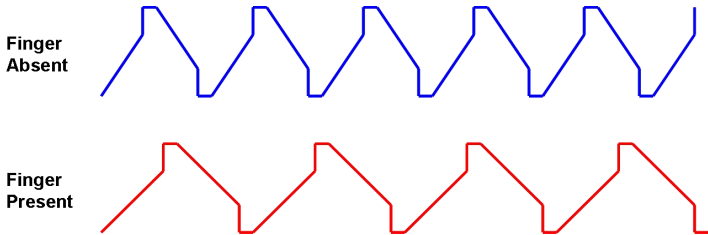


图 5- 43 手指触碰和离开的时序状态图

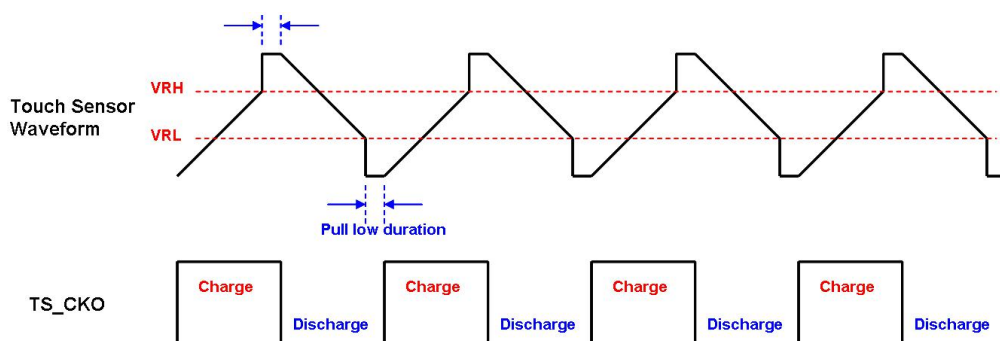


图 5-44 电容传感器输出频率时序图(EN_TRI = “0”)

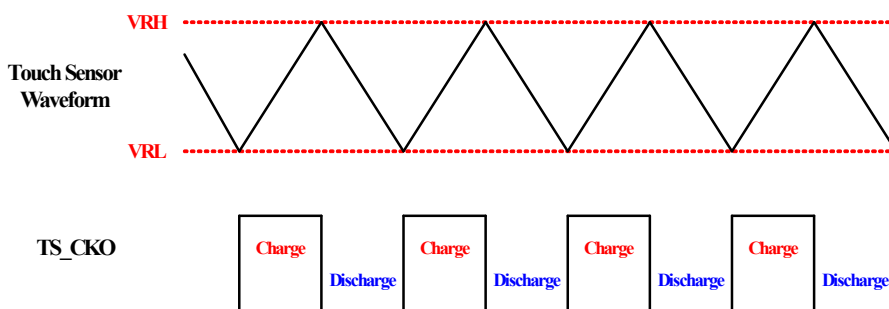


图 5-45 电容传感器三角波输出频率时序图(EN_TRI = “1”)

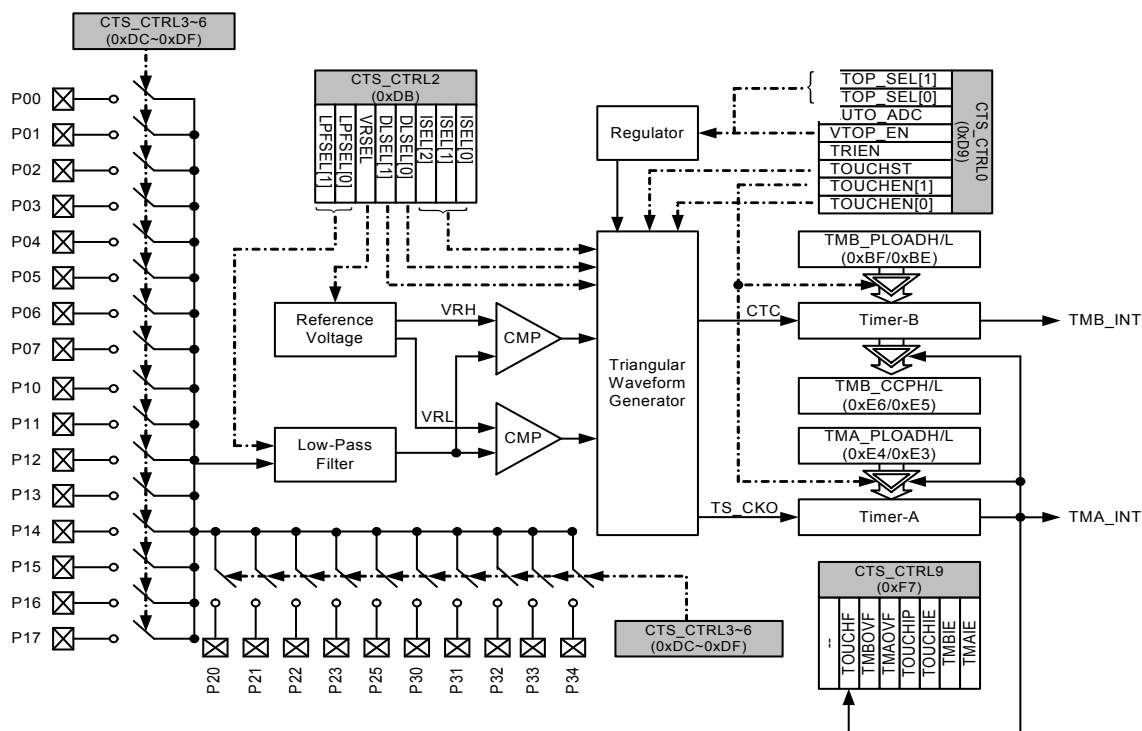


图 5-46 张弛振荡器架构框图

GPM8F3733A/GPM8F3717A/GPM8F3709A

在CTC架构中，在启动命令前，Cs需要连接到电容器上且用户应该设置基准电压。在这种模式下，timerB用于计算CTC_DUTY的周期。图5-47 给出了CTC架构框图。

C_S电容器的材料建议使用NPO, 其值在10nF~47nF 之间。

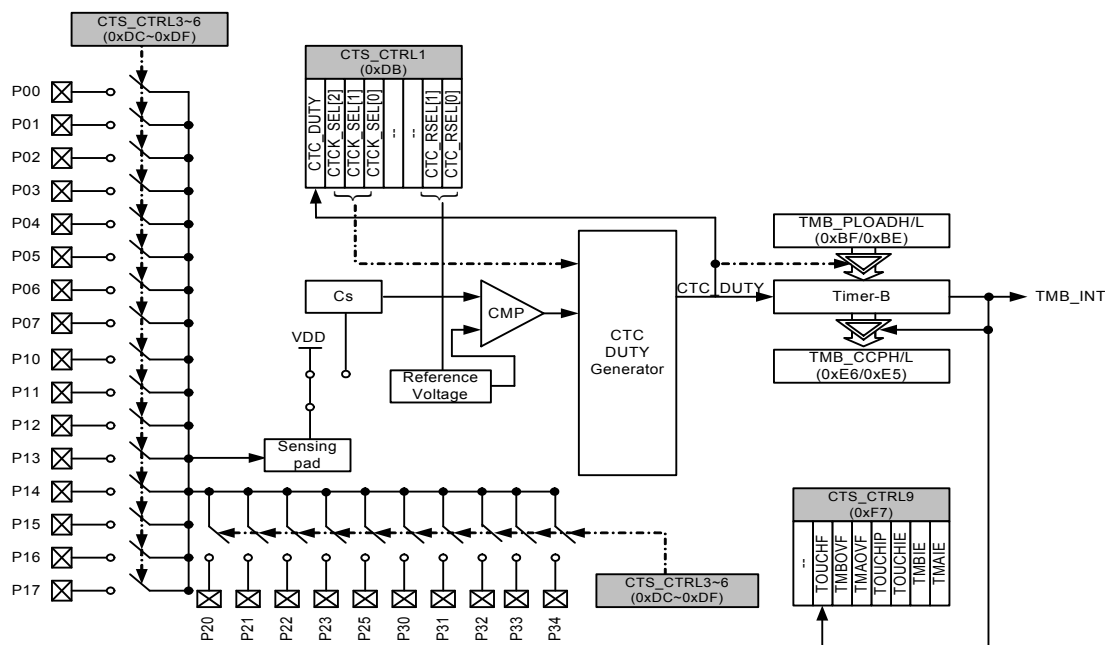


图 5-47 CTC 架构框图

在CTA架构中，在启动命令前，Cs 需要连接到电容器且用户应该设置timerA的周期。图5-48给出了CTA架构框图。Cs电容器的材料建议使用NPO，其值小于1nF。

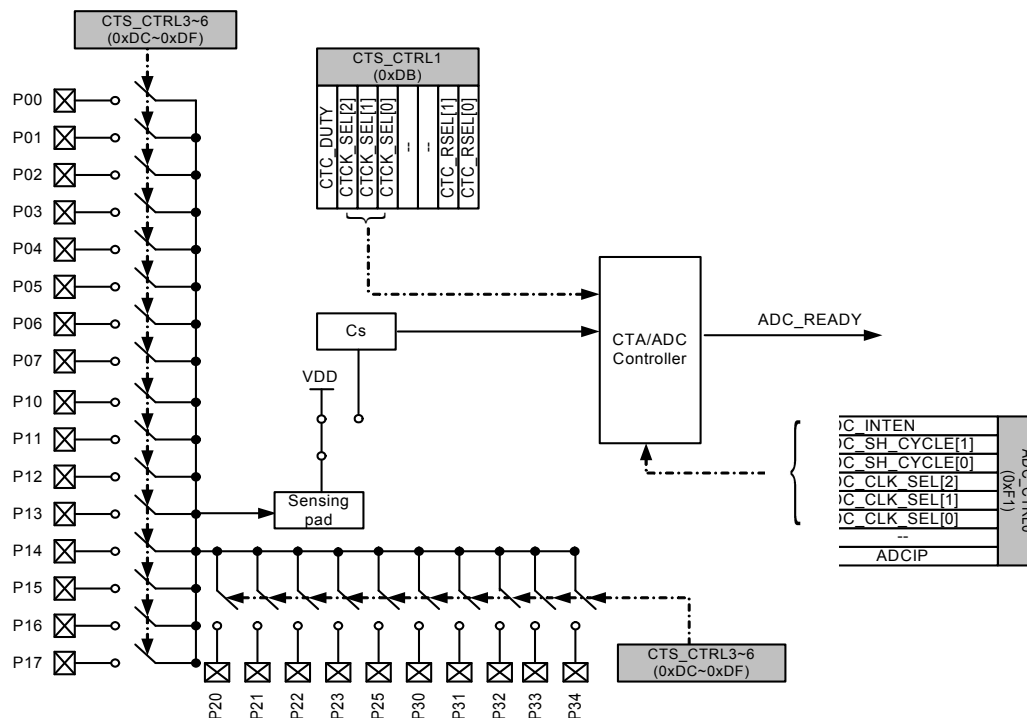


图 5-48 CTA 架构框图

电容式感应架构提供了2种应用：自感式电容感应和互感式电容感应。图5-49给出了一个简单的应用示意图。

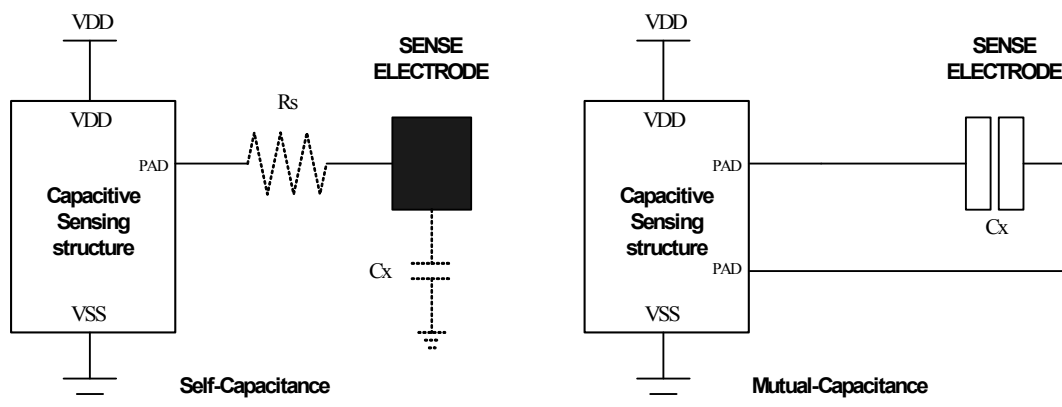


图 5- 49 自感式电容感应和互感式电容感应

5.15.2. 电容触摸传感器相关寄存器

CTS_CTRL0			地址：0xD9		电容触摸传感器控制寄存器 - 0			
Bit	7	6	5	4	3	2	1	0
功能	VTOPSEL[1:0]		AUTO_ADC	VTOP_EN	TRIEN	CTS_ST	CTS_EN[1:0]	
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述		条件
7:6	VTOPSEL[1:0]	R/W	比较器的输出电压		
			VTOPSEL[1:0]	gTouch VDD	
			00	2.2V	
			01	2.7V	
			10	3.6V	
			11	4.3V	
5	AUTO_ADC	R/W	CTA 模式的 ADC 自动转换使能位 0 : 禁止 1 : 使能		
4	VTOPEN	R/W	gTouch 比较器使能 0 : 禁止 1 : 使能		
3	TRIEN	R/W	0 : 禁止三角波模式 1 : 使能三角波模式		
2	CTS_ST	R/W	触摸传感器启动触摸控制 0 : 禁止 1 : 使能		
1:0	CTS_EN[1:0]	R/W	电容式触摸传感器使能位		
			CTS_EN[1:0]		
			00	触摸功能禁止	
			01	RC 模式使能	
			10	CTC 模式使能	
			11	CTA 模式使能	

表 5- 129 CTS_CTRL0 寄存器

GPM8F3733A/GPM8F3717A/GPM8F3709A

CTS_CTRL1			地址: 0xDA		电容触摸传感器控制寄存器 - 1			
Bit	7	6	5	4	3	2	1	0
功能	CTC_DUTY	CTCK_SEL[2:0]			CTC_CNT	--	CTC_RSEL[1:0]	
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件																		
7	CTC_DUTY	R	CTC_DUTY 0: Cs 下拉接地放电 1: 计数高占空比测量电容																			
6:4	CTCK_SEL	R/W	CTC 和 CTA 模式的时钟输入 <table><tr><td>CTCK_SEL[2:0]</td><td>频率</td></tr><tr><td>000</td><td>系统时钟 /2</td></tr><tr><td>001</td><td>系统时钟 /3</td></tr><tr><td>010</td><td>系统时钟 /4</td></tr><tr><td>011</td><td>系统时钟 /5</td></tr><tr><td>100</td><td>系统时钟 /7</td></tr><tr><td>101</td><td>系统时钟 /9</td></tr><tr><td>110</td><td>系统时钟 /11</td></tr><tr><td>111</td><td>系统时钟 /15</td></tr></table>	CTCK_SEL[2:0]	频率	000	系统时钟 /2	001	系统时钟 /3	010	系统时钟 /4	011	系统时钟 /5	100	系统时钟 /7	101	系统时钟 /9	110	系统时钟 /11	111	系统时钟 /15	
CTCK_SEL[2:0]	频率																					
000	系统时钟 /2																					
001	系统时钟 /3																					
010	系统时钟 /4																					
011	系统时钟 /5																					
100	系统时钟 /7																					
101	系统时钟 /9																					
110	系统时钟 /11																					
111	系统时钟 /15																					
3	CTC_CNT	R/W	CTC 计数器模式 0 : 定时器模式. 1 : 计数器模式.																			
2	--	R/W	预留																			
1:0	CTC_RSEL[1:0]	R/W	CTC 比较器的参考电压 <table><tr><td>CTC_RSEL[1:0]</td><td>电压</td></tr><tr><td>00</td><td>0.4*VDD</td></tr><tr><td>01</td><td>0.5*VDD</td></tr><tr><td>10</td><td>0.6*VDD</td></tr><tr><td>11</td><td>0.7*VDD</td></tr></table>	CTC_RSEL[1:0]	电压	00	0.4*VDD	01	0.5*VDD	10	0.6*VDD	11	0.7*VDD									
CTC_RSEL[1:0]	电压																					
00	0.4*VDD																					
01	0.5*VDD																					
10	0.6*VDD																					
11	0.7*VDD																					

表 5- 130 CTS_CTRL1 寄存器

CTS_CTRL2			地址: 0xDB		电容触摸传感器控制寄存器 - 2			
Bit	7	6	5	4	3	2	1	0
功能	LPFSEL[1:0]		VRSEL	DLSEL[1:0]		ISEL[2:0]		
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述		条件
7:6	LPFSEL[1:0]	R/W	比较器的噪音滤波器(1 st order RC filter)		
			LPFSEL[1:0]	带宽	
			00	>200MHz	
			01	>8MHz	
			10	>3MHz	
			11	>1.5MHz	
5	VRSEL	R/W	比较器的参考电压选择		
			VRSEL	VRH	VRL
			0	2/3 *gTouch VDD	1/3 *gTouch VDD
			1	11/12 *gTouch VDD	1/12 *gTouch VDD
4:3	DLSEL[1:0]	R/W	上拉/下拉持续时间		
			DLSEL[1:0]	延后时间	
			00	450ns	
			01	300ns	
			10	150ns	
			11	70ns	
2:0	ISEL[2:0]	R/W	充电/放电电流选择		
			ISEL[2:0]	充电/放电电流	
			000	25uA	
			001	50uA	
			010	100uA	
			011	150uA	
			100	200uA	
			101	300uA	
			110	450uA	
			111	600uA	

表 5- 131 CTS_CTRL2 寄存器

CTS_CTRL3			地址: 0xDC		电容触摸传感器控制寄存器 - 3			
Bit	7	6	5	4	3	2	1	0
功能	CHSEL[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CHSEL[7:0]	R/W	触摸通道选择, 如表 5- 136	

表 5- 132 CTS_CTRL3 寄存器

CTS_CTRL4			地址: 0xDD		电容触摸传感器控制寄存器 - 4			
Bit	7	6	5	4	3	2	1	0
功能	CHSEL[15:8]							
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:0	CHSEL[15:8]	R/W	触摸通道选择, 见表 5- 136	

表 5- 133 CTS_CTRL4 寄存器

CTS_CTRL5			地址: 0xDE		电容触摸传感器控制寄存器 - 5			
Bit	7	6	5	4	3	2	1	0
功能	CHSEL[23:16]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	CHSEL[23:16]	R/W	触摸通道选择, 见表 5- 136	

表 5- 134 CTS_CTRL5 寄存器

CTS_CTRL6			地址: 0xDF		电容触摸传感器控制寄存器 - 6			
Bit	7	6	5	4	3	2	1	0
功能	CHSEL[25:24]		TMBIP	TMAIP	TMB_PLOAD_EN	TMB_EN	TMA_PLOAD_EN	TMA_EN
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:6	CHSEL[25:24]	R/W	触摸通道选择, 见表 5- 136	
5	TMBIP	R/W	Timer B 优先级控制 (1: 高有效)	
4	TMAIP	R/W	Timer A 优先级控制 (1: 高有效)	
3	TMB_PLOAD_EN	R/W	Timer B 自动加载功能使能位	
2	TMB_EN	R/W	Timer B 使能位	
1	TMA_PLOAD_EN	R/W	Timer A 自动加载功能使能位	
0	TMA_EN	R/W	Timer A 使能位	

表 5- 135 CTS_CTRL6 寄存器

CHSEL[25:0]	触摸通道	CHSEL[25:0]	触摸通道
00_0000_0000_0000_0000_0001	选择 P00	00_0000_0000_0010_0000_0000_0000	选择 P15
00_0000_0000_0000_0000_0010	选择 P01	00_0000_0000_0100_0000_0000_0000	选择 P16
00_0000_0000_0000_0000_0100	选择 P02	00_0000_0000_1000_0000_0000_0000	选择 P17
00_0000_0000_0000_0000_1000	选择 P03	00_0000_0001_0000_0000_0000_0000	选择 P20
00_0000_0000_0000_0001_0000	选择 P04	00_0000_0010_0000_0000_0000_0000	选择 P21
00_0000_0000_0000_0010_0000	选择 P05	00_0000_0100_0000_0000_0000_0000	选择 P22
00_0000_0000_0000_0100_0000	选择 P06	00_0000_1000_0000_0000_0000_0000	选择 P23
00_0000_0000_0000_1000_0000	选择 P07	00_0001_0000_0000_0000_0000_0000	选择 P25
00_0000_0000_0000_0001_0000_0000	选择 P10	00_0010_0000_0000_0000_0000_0000	选择 P30
00_0000_0000_0000_0010_0000_0000	选择 P11	00_0100_0000_0000_0000_0000_0000	选择 P31
00_0000_0000_0000_0100_0000_0000	选择 P12	00_1000_0000_0000_0000_0000_0000	选择 P32
00_0000_0000_0000_1000_0000_0000	选择 P13	01_0000_0000_0000_0000_0000_0000	选择 P33
00_0000_0000_0001_0000_0000_0000	选择 P14	10_0000_0000_0000_0000_0000_0000	选择 P34

表 5- 136 触摸通道选择

GPM8F3733A/GPM8F3717A/GPM8F3709A

TMA_PLOADH			地址: 0xE4		Timer A 预载数据高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMA_PLOAD[15:8]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMA_PLOADH	R/W	Timer A 预载数据高字节	

表 5- 137 TMA_PLOADH 寄存器

TMA_PLOADL			地址: 0xE3		Timer A 预载数据低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMA_PLOAD[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMA_PLOADL	R/W	Timer A 预载数据低字节	

表 5- 138 TMA_PLOADL 寄存器

TMB_PLOADH			地址: 0xBF		Timer B 预载数据高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMB_PLOAD[15:8]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMB_PLOADH	R/W	Timer B 预载数据高字节	

表 5- 139 TMB_PLOADH 寄存器

TMB_PLOADL			地址: 0xBE		Timer B 预载数据低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMB_PLOAD[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMB_PLOADL	R/W	Timer B 预载数据低字节	

表 5- 140 TMB_PLOADL 寄存器

TMB_CAPDH			地址: 0xE6		Timer B 捕捉数据高字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMB_CAP[15:8]							
默认	0	0	0	0	0	0	0	0

GPM8F3733A/GPM8F3717A/GPM8F3709A

Bit	功能	类型	描述	条件
7:0	TMB_PLOADH	R/W	Timer B 捕捉数据高字节	

表 5- 141 TMB_CAPDH 寄存器

TMB_CAPDL			地址: 0xE5		Timer B 捕捉数据低字节寄存器			
Bit	7	6	5	4	3	2	1	0
功能	TMB_CAP[7:0]							
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:0	TMB_PLOADL	R/W	Timer B 捕捉数据低字节	

表 5- 142 TMB_CAPDL 寄存器

CTS_CTRL7			地址: 0xF5		电容触摸传感器控制寄存器 - 7			
Bit	7	6	5	4	3	2	1	0
功能	MUPORTSEL[2:0]			MUPINSEL[2:0]			MUTPOL	MUTEN
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7:5	MUPORTSEL	R/W	选择互感式电容 IO Port 映射	
			MUPORTSEL[2:0] Port 序号	
			000 IO Port P0	
			001 IO Port P1	
			010 IO Port P2	
			011 IO Port P3	
			1xx 禁止	
4:2	MUPINSEL	R/W	选择互感式电容 IO Pin 映射	
			MUPORTSEL[2:0] Port 序号	
			000 IO Port x Pin 0	
			001 IO Port x Pin 1	
			010 IO Port x Pin 2	
			011 IO Port x Pin 3	
			100 IO Port x Pin 4	
			101 IO Port x Pin 5	
1	MUTPOL	R/W	互感式电容输出波形的极性调整 0 : 负极 1 : 正极	
0	MUTEN	R/W	互感式电容功能控制 0 : 禁止 1 : 使能	

表 5- 143 CTS_CTRL7 寄存器

GPM8F3733A/GPM8F3717A/GPM8F3709A

CTS_CTRL8			地址: 0xF6		电容触摸传感器控制寄存器 - 5			
Bit	7	6	5	4	3	2	1	0
功能	MUTDEL_SEL[3:0]				—	—	—	TSCFG
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件	
7:4	MUTDEL_SEL	R/W	互感式电容输出脚输出 TS_CKO 波形时的延时控制		
			MUTDEL_SEL[3:0]		延时周期
			0000		无延时
			0001		延时 1 个系统时钟.
			0010		延时 2 个系统时钟.
			0011		延时 4 个系统时钟.
			0100		延时 6 个系统时钟.
			0101		延时 8 个系统时钟.
			0110		延时 10 个系统时钟.
			0111		延时 12 个系统时钟.
			1000		延时 14 个系统时钟.
			1001		延时 16 个系统时钟.
			1010		延时 18 个系统时钟.
			1011		延时 20 个系统时钟.
			1100		延时 22 个系统时钟.
			1101		延时 24 个系统时钟.
			1110		延时 26 个系统时钟.
			1111		延时 28 个系统时钟.
3:1	-	R/W	预留		
0	TSCGF	R	读 TS_CKO 信号波形(触摸感应时钟输出)		

表 5- 144 CTS_CTRL5 寄存器

CTS_CTRL9			地址: 0xF7		电容触摸传感器控制寄存器 - 9			
Bit	7	6	5	4	3	2	1	0
功能	—	TOUCHF	TMBOVF	TMAOVF	TOUCHIP	TOUCHIE	TMBIE	TMAIE
默认	0	0	0	0	0	0	0	0

Bit	功能	类型	描述	条件
7	—	R/W	预留	
6	TOUCHF	R/W	触摸中断标志	
5	TMBOVF	R/W	Timer B 溢出标志	
4	TMAOVF	R/W	Timer A 溢出标志	
3	TOUCHIP	R/W	触摸优先级控制(1: 高有效)	
2	TOUCHIE	R/W	使能触摸中断	
1	TMBIE	R/W	使能 Timer B 中断	
0	TMAIE	R/W	使能 Timer A 中断	

表 5- 145 CTS_CTRL9 寄存器

5.16. 指令集列表（按字母顺序）

5.16.1. 算术运算

Mnemonic	Description	Code	Bytes	Cycles
ADD A, Rn	Add register to accumulator	0x28-0x2F	1	1
ADD A, direct	Add direct byte to accumulator	0x25	2	2
ADD A, @Ri	Add indirect RAM to accumulator	0x26-0x27	1	2
ADD A, #data	Add immediate data to accumulator	0x24	2	2
ADDC A, Rn	Add register to accumulator with carry flag	0x38-0x3F	1	1
ADDC A, direct	Add direct byte to A with carry flag	0x35	2	2
ADDC A, @Ri	Add indirect RAM to A with carry flag	0x36-0x37	1	2
ADDC A, #data	Add immediate data to A with carry flag	0x34	2	2
SUBB A, Rn	Subtract register from A with borrow	0x98-0x9F	1	1
SUBB A, direct	Subtract direct byte from A with borrow	0x95	2	2
SUBB A, @Ri	Subtract indirect RAM from A with borrow	0x96-0x97	1	2
SUBB A, #data	Subtract immediate data from A with borrow	0x94	2	2
INC A	Increment accumulator	0x04	1	1
INC Rn	Increment register	0x08-0x0F	1	2
INC direct	Increment direct byte	0x05	2	3
INC @Ri	Increment indirect RAM	0x06-0x07	1	3
DEC A	Decrement accumulator	0x14	1	1
DEC Rn	Decrement register	0x18-0x1F	1	2
DEC direct	Decrement direct byte	0x15	1	3
DEC @Ri	Decrement indirect RAM	0x16-0x17	2	3
INC DPTR	Increment data pointer	0xA3	1	1
MUL A, B	Multiply A and B	0xA4	1	2
DIV A, B	Divide A by B	0x84	1	6
DA A	Decimal adjust accumulator	0xD4	1	3

5.16.2. 逻辑运算

Mnemonic	Description	Code	Bytes	Cycles
ANL A, Rn	AND register to accumulator	0x58-0x5F	1	1
ANL A, direct	AND direct byte to accumulator	0x55	2	2
ANL A, @Ri	AND indirect RAM to accumulator	0x56-0x57	1	2
ANL A, #data	AND immediate data to accumulator	0x54	2	2
ANL direct, A	AND accumulator to direct byte	0x52	2	3
ANL direct, #data	AND immediate data to direct byte	0x53	3	3
ORL A, Rn	OR register to accumulator	0x48-0x4F	1	1
ORL A, direct	OR direct byte to accumulator	0x45	2	2
ORL A, @Ri	OR indirect RAM to accumulator	0x46-0x47	1	2
ORL A, #data	OR immediate data to accumulator	0x44	2	2
ORL direct, A	OR accumulator to direct byte	0x42	2	3
ORL direct, #data	OR immediate data to direct byte	0x43	3	3
XRL A, Rn	Exclusive OR register to accumulator	0x68-0x6F	1	1

Mnemonic	Description	Code	Bytes	Cycles
XRL A, direct	Exclusive OR direct byte to accumulator	0x65	2	2
XRL A, @Ri	Exclusive OR indirect RAM to accumulator	0x66-0x67	1	2
XRL A, #data	Exclusive OR immediate data to accumulator	0x64	2	2
XRL direct, A	Exclusive OR accumulator to direct byte	0x62	2	3
XRL direct, #data	Exclusive OR immediate data to direct byte	0x63	3	3
CLR A	Clear accumulator	0xE4	1	1
CPL A	Complement accumulator	0xF4	1	1
RL A	Rotate accumulator left	0x23	1	1
RLC A	Rotate accumulator left through carry	0x33	1	1
RR A	Rotate accumulator right	0x03	1	1
RRC A	Rotate accumulator right through carry	0x13	1	1
SWAP A	Swap nibbles within the accumulator	0xC4	1	1

5.16.3. 布尔运算

Mnemonic	Description	Code	Bytes	Cycles
CLR C	Clear carry flag	0xC3	1	1
CLR bit	Clear direct bit	0xC2	2	3
SETB C	Set carry flag	0xD3	1	1
SETB bit	Set direct bit	0xD2	2	3
CPL C	Complement carry flag	0xB3	1	1
CPL bit	Complement direct bit	0xB2	2	3
ANL C, bit	AND direct bit to carry flag	0x82	2	2
ANL C, /bit	AND complement of direct bit to carry	0xB0	2	2
ORL C, bit	OR direct bit to carry flag	0x72	2	2
ORL C, /bit	OR complement of direct bit to carry	0xA0	2	2
MOV C, bit	Move direct bit to carry flag	0xA2	2	2
MOV bit, C	Move carry flag to direct bit	0x92	2	3

5.16.4. 数据传送

Mnemonic	Description	Code	Bytes	Cycles
MOV A, Rn	Move register to accumulator	0xE8-0xEF	1	1
MOV A, direct	Move direct byte to accumulator	0xE5	2	2
MOV A, @Ri	Move indirect RAM to accumulator	0xE6-0xE7	1	2
MOV A, #data	Move immediate data to accumulator	0x74	2	2
MOV Rn, A	Move accumulator to register	0xF8-0xFF	1	1
MOV Rn, direct	Move direct byte to register	0xA8-0xAF	2	3
MOV Rn, #data	Move immediate data to register	0x78-0x7F	2	2
MOV direct, A	Move accumulator to direct byte	0xF5	2	2
MOV direct, Rn	Move register to direct byte	0x88-0x8F	2	2
MOV direct1, direct2	Move direct byte to direct byte	0x85	3	3
MOV direct, @Ri	Move indirect RAM to direct byte	0x86-0x87	2	3
MOV direct, #data	Move immediate data to direct byte	0x75	3	3
MOV @Ri, A	Move accumulator to indirect RAM	0xF6-0xF7	1	2

GPM8F3733A/GPM8F3717A/GPM8F3709A

Mnemonic	Description	Code	Bytes	Cycles
MOV @Ri, direct	Move direct byte to indirect RAM	0xA6-0xA7	2	3
MOV @Ri, #data	Move immediate data to indirect RAM	0x76-0x77	2	2
MOV DPTR, #data16	Load 16-bit constant into active DPH and DPL in LARGE mode	0x90	3	3
MOV DPTR, #data24	Load 16-bit constant into active DPH and DPL in Flat mode	0x90	4	4
MOVC A, @A+DPTR	Move code byte relative to DPTR to accumulator	0x93	1	5
MOVC A, @A+PC	Move code byte relative to PC to accumulator	0x83	1	4
MOVX A, @Ri	Move external RAM (8-bit address) to A	XDM	1	3*
		SXDM		3
MOVX A, @DPTR	Move external RAM (16-bit address) to A	XDM	1	2*
		SXDM		2
MOVX @Ri, A	Move A to external XDM (8-bit address)	ODE inside ROM/RAM	1	4*
		Other cases		5*
	Move A to external SXDM (8-bit address)	All cases		3
MOVX @DPTR, A	Move A to external XDM (16-bit address)	CODE inside ROM/RAM	1	3*
		Other cases		4*
	Move A to external SXDM (16-bit address)	All cases		2
PUSH direct	Push direct byte onto IDM stack	0xC0	2	3
POP direct	Pop direct byte from IDM stack	0xD0	2	2
XCH A, Rn	Exchange register with accumulator	0xC8-0xCF	1	2
XCH A, direct	Exchange direct byte with accumulator	0xC5	2	3
XCH A, @Ri	Exchange indirect RAM with accumulator	0xC6-0xC7	1	3
XCHD A, @Ri	Exchange low-order nibble indirect RAM with A	0xD6-0xD7	1	3

5.16.5. 编程跳转

Mnemonic	Description	Code	Bytes	Cycles
ACALL addr11	Absolute subroutine call	0x11-0xF1	2	4
LCALL addr16	Long subroutine call	0x12	3	4
RET	Return from subroutine	0x22	1	4
RETI	Return from interrupt	0x32	1	4
AJMP addr11	Absolute jump	0x01-0xE1	2	3
LJMP addr16	Long jump	0x02	3	4
SJMP rel	Short jump (relative address)	0x80	2	3
JMP @A+DPTR	Jump indirect relative to the DPTR	0x73	1	5
JZ rel	Jump if accumulator is zero	0x60	2	4
JNZ rel	Jump if accumulator is not zero	0x70	2	4
JC rel	Jump if carry flag is set	0x40	2	3
JNC	Jump if carry flag is not set	0x50	2	3
JB bit, rel	Jump if direct bit is set	0x20	3	5
JNB bit, rel	Jump if direct bit is not set	0x30	3	5
JBC bit, direct rel	Jump if direct bit is set and clear bit	0x10	3	5
CJNE A, direct rel	Compare direct byte to A and jump if not equal	0xB5	3	5

GPM8F3733A/GPM8F3717A/GPM8F3709A

CJNE A, #data rel	Compare immediate to A and jump if not equal	0xB4	3	4
CJNE Rn, #data rel	Compare immediate to reg. and jump if not equal	0xB8-0xBF	3	4
CJNE @Ri, #data rel	Compare immediate to ind. and jump if not equal	0xB6-0xB7	3	5
DJNZ Rn, rel	Decrement register and jump if not zero	0xD8-0xDF	2	4
DJNZ direct, rel	Decrement direct byte and jump if not zero	0xD5	3	5
NOP	No operation	0x00	1	1

6. 电气特性

6.1. 绝对最大额定值

Characteristics	Symbol	Limit			Unit	Condition
		Min.	Typ.	Max.		
Storage Temperature	T_{STG}	-65	-	150	° C	Flash memory blank status
		-40	-	150	° C	Flash memory programming already performed
Operation Temperature	T_{OP}	-40	-	85	° C	
VDD Total MAX Current	I_{VDDM}	-	-	160mA		
VSS Total MAX Current	I_{VSSM}	-	-	180mA		

6.2. DC 特性(VDD = 5V, T_A = 25°C)

Characteristics	Symbol	Limit			Unit	Condition
		Min.	Typ.	Max.		
Operating Voltage	VDD	2.0	5	5.5	V	
Operating Current	I_{OP}	-	7	8	mA	$F_{CPU} = 32MHz @ 5.5V$, no load
		-	6	7	mA	$F_{CPU} = 16MHz @ 5.5V$, no load
Standby Current	I_{STBY}	-	-	5.0	uA	VDD = 5.5V, LVR disabled
		-	-	9.0	uA	VDD = 5.5V, LVR enabled
Input High Level	V_{IH}	0.7VDD	-	-	V	VDD = 5.0V
Input Low Level	V_{IL}	-	-	0.3VDD	V	VDD = 5.0V
Output High Level	V_{OH}	0.8VDD	-	-	V	$I_{OH} = -20mA$ at VDD = 5.0V
Output Low Level	V_{OL}	-	-	0.2VDD	V	$I_{OL} = 20mA$ at VDD = 5.0V
Input Pull High Resistor	R_{PH}	30	50	70	K Ω	VDD = 5.0V
Input Pull High Resistor	R_{PL}	30	50	70	K Ω	VDD = 5.0V
Low Voltage Reset	V_{LVR}	1.9/2.2/2.7/3.6 × (1-5%)	1.9/2.2/2.7/3.6	1.9/2.2/2.7/3.6 × (1+5%)	V	
Low Voltage Detect	V_{LVD}	2.1/2.4/2.9/3.8 × (1-5%)	2.1/2.4/2.9/3.8	2.1/2.4/2.9/3.8 × (1+5%)	V	
Flash operation voltage	V_{FLASH}	V_{LVR}	-	-	V	

GPM8F3733A/GPM8F3717A/GPM8F3709A

6.3. DC 特性 (VDD = 3.3V, TA = 25°C)

Characteristics	Symbol	Limit			Unit	Condition
		Min.	Typ.	Max.		
Operating Current	I _{OP}	–	6	7	mA	F _{CPU} = 32MHz @ 3.6V, no load
		–	5	6	mA	F _{CPU} = 16MHz @ 3.6V, no load
Standby Current	I _{STBY}	–	–	4.0	uA	VDD = 3.6V, LVR disabled
		–	–	8.0	uA	VDD = 3.6V, LVR enabled
Input High Level	V _{IH}	0.7VDD	–	–	V	VDD = 3.3V
Input Low Level	V _{IL}	–	–	0.3VDD	V	VDD = 3.3V
Output High Level	V _{OH}	0.8VDD	–	–	V	I _{OH} = -10mA at VDD = 3.3V
Output Low Level	V _{OL}	–	–	0.2VDD	V	I _{OL} = 10mA at VDD = 3.3V
Input Pull High Resistor	R _{PH}	30	50	70	K Ω	VDD = 3.3V
Input Pull High Resistor	R _{PL}	30	50	70	K Ω	VDD = 3.3V
Low Voltage Reset	V _{LVR}	1.9/2.2/2.7/3.6	1.9/2.2/2.7/3.6	1.9/2.2/2.7/3.6	V	
		× (1-5%)		× (1+5%)		
Low Voltage Detect	V _{LVD}	2.1/2.4/2.9/3.8	2.1/2.4/2.9/3.8	2.1/2.4/2.9/3.8	V	
		× (1-5%)		× (1+5%)		
Flash operation voltage	V _{FLASH}	V _{LVR}	–	–	V	

6.4. AC 特性 (TA = 25°C)

Characteristics	Symbol	Limit			Unit	Condition
		Min.	Typ.	Max.		
INOSC Frequency	F _{OSC}	32 × (1-2%)	32	32 × (1+2%)	MHz	VDD = 2.0~5.5V

7. 封装/脚位

7.1. 订购信息

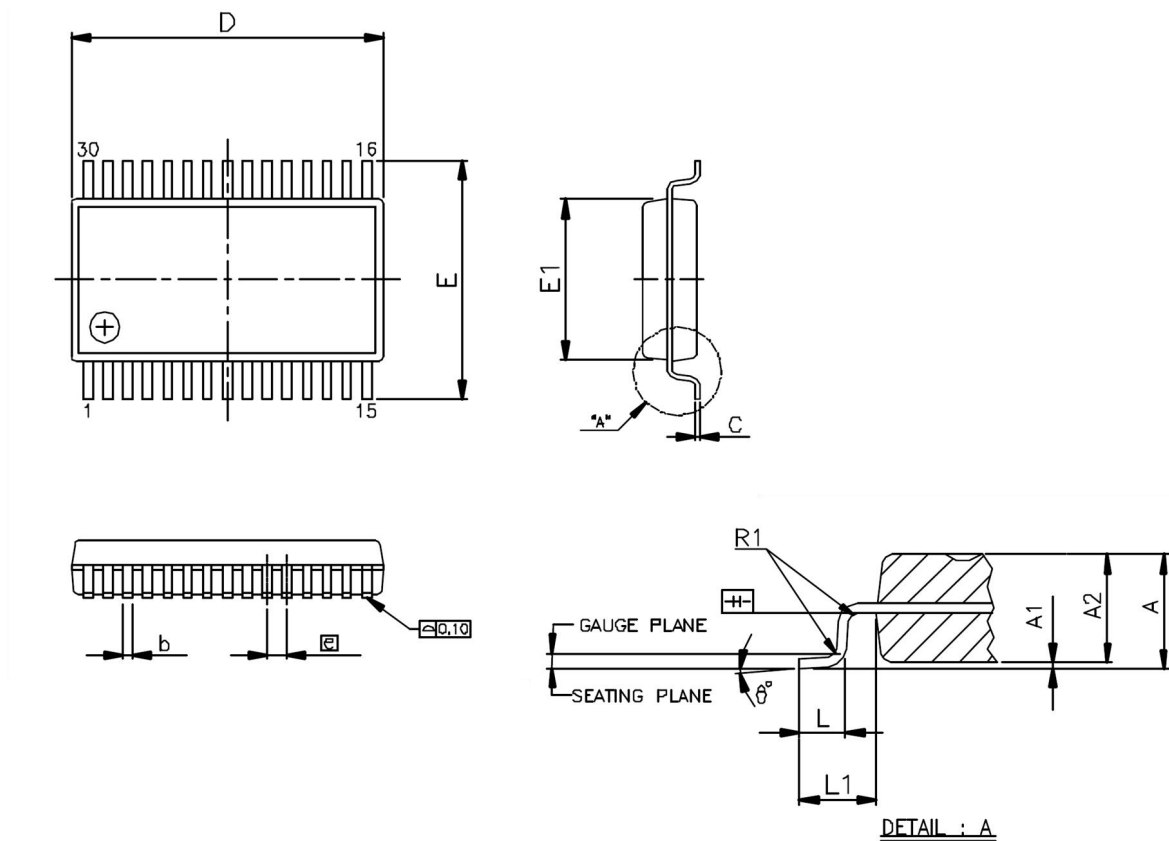
Product Number	Package Type	Packing Information
GPM8F3733A - nnnA - HG09x	Green Package	Tube
GPM8F3717A - nnnA - HS09x	Green Package	Tube
GPM8F3717A - nnnA - HS05x	Green Package	Tube
GPM8F3709A - nnnA - HS10x	Green Package	Tube
GPM8F3709A - nnnA - HS03x	Green Package	Tube

Note:

1. nnn: code number from 000 to 999. If you want to run mass production code, please apply a new code from our sales assistant first. And if no nnnA number, it means a chip without code inside. Ex: GPM8F3733A-HG091 is a standard chip.
2. x: package serial number from 0 to 9. It will depend on the top site mark.
3. Product naming rule: ex: GPM8F3733A-
nnnA-HG09x.

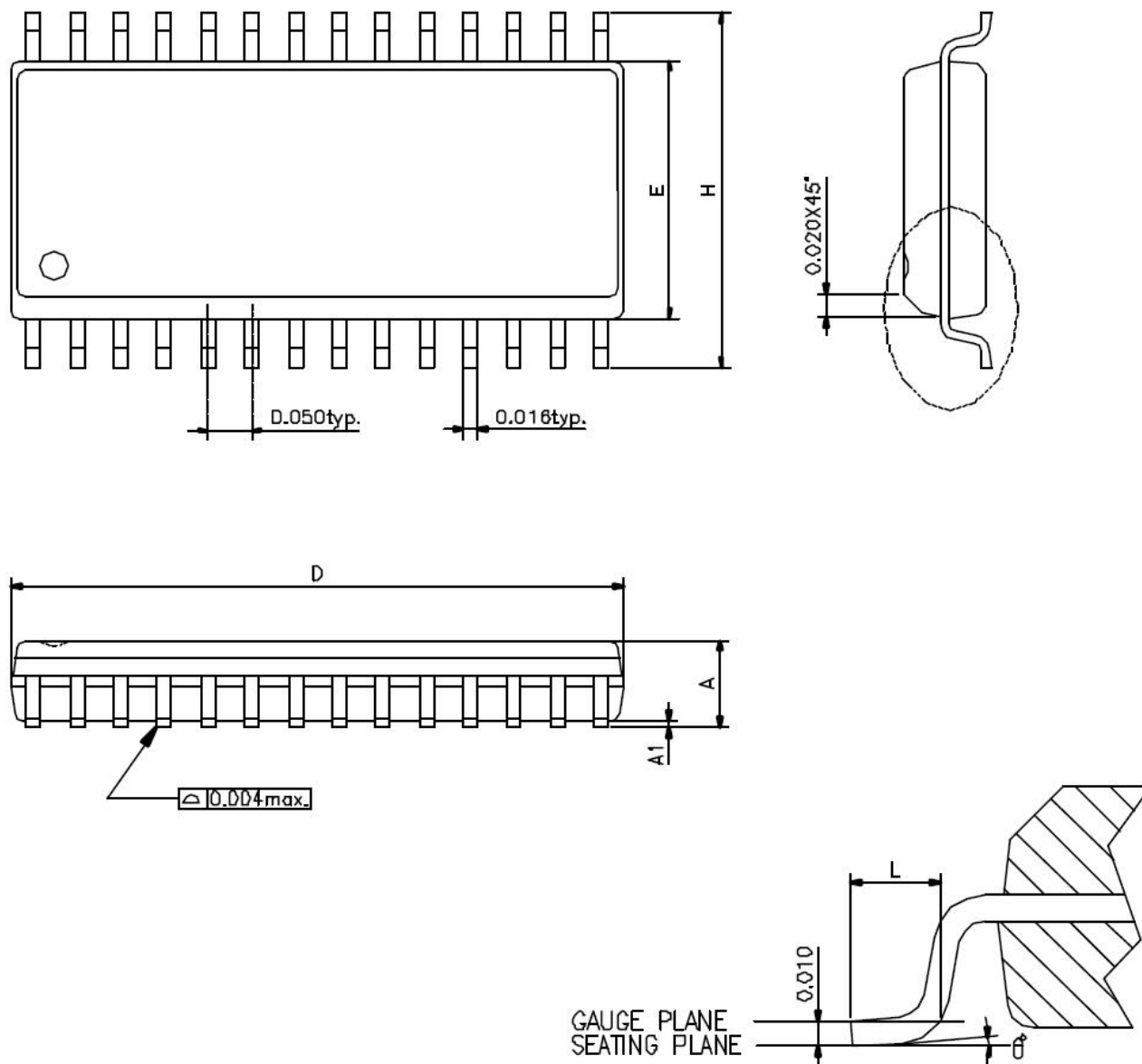
7.2. 封装信息

SSOP 30



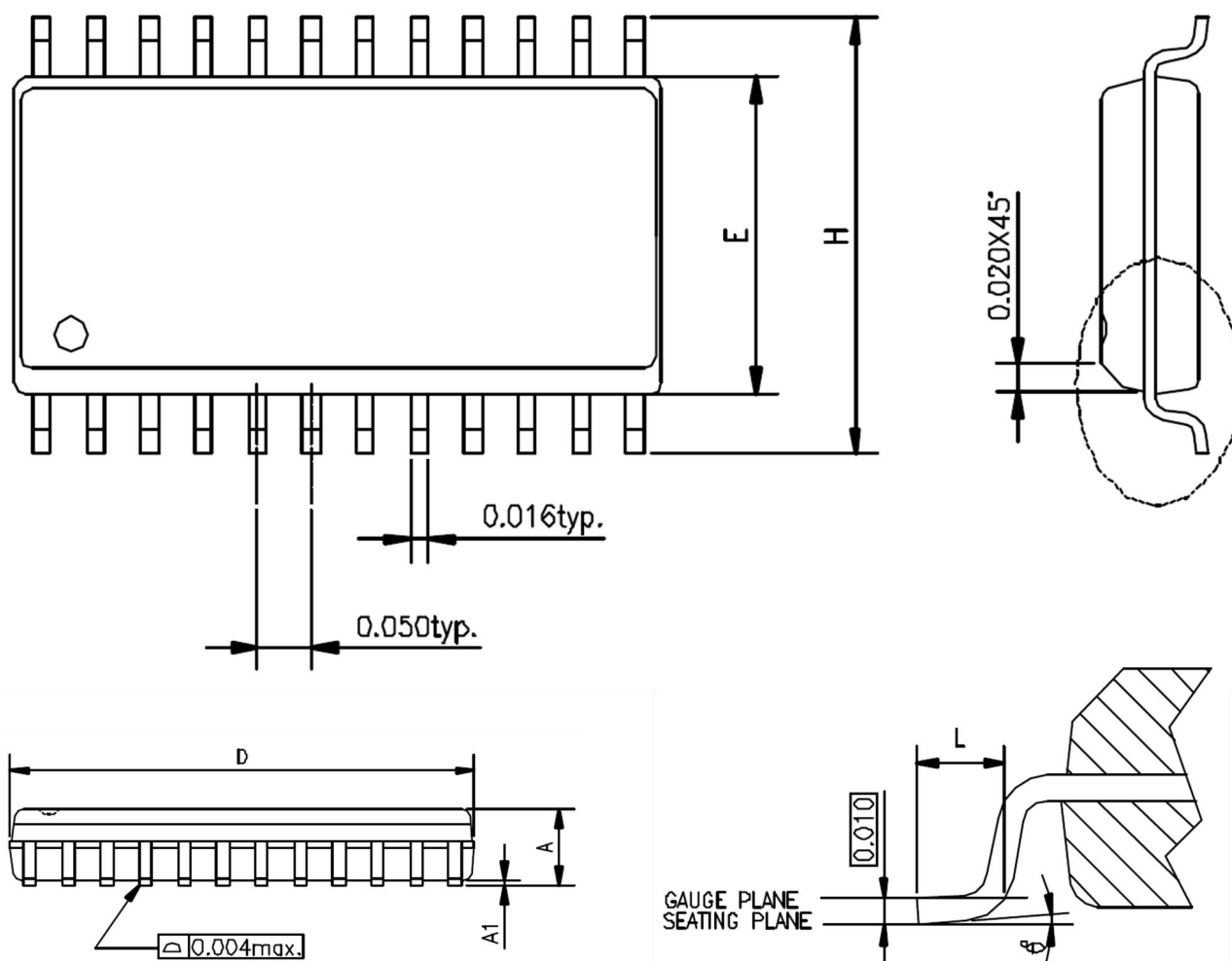
Symbol	Millimeter		
	Min.	Nom.	Max.
A	--	--	2.0
A1	0.05	--	--
A2	1.65	1.75	1.85
b	0.22	--	0.38
c	0.09	--	0.21
D	9.90	10.20	10.50
E	7.40	7.80	8.20
E1	5.00	5.30	5.60
$\square e$	0.65 BSC		
L	0.55	0.75	0.95
L1	1.25 REF		
R1	0.09	--	--
θ°	0°	4°	8°

SOP 28



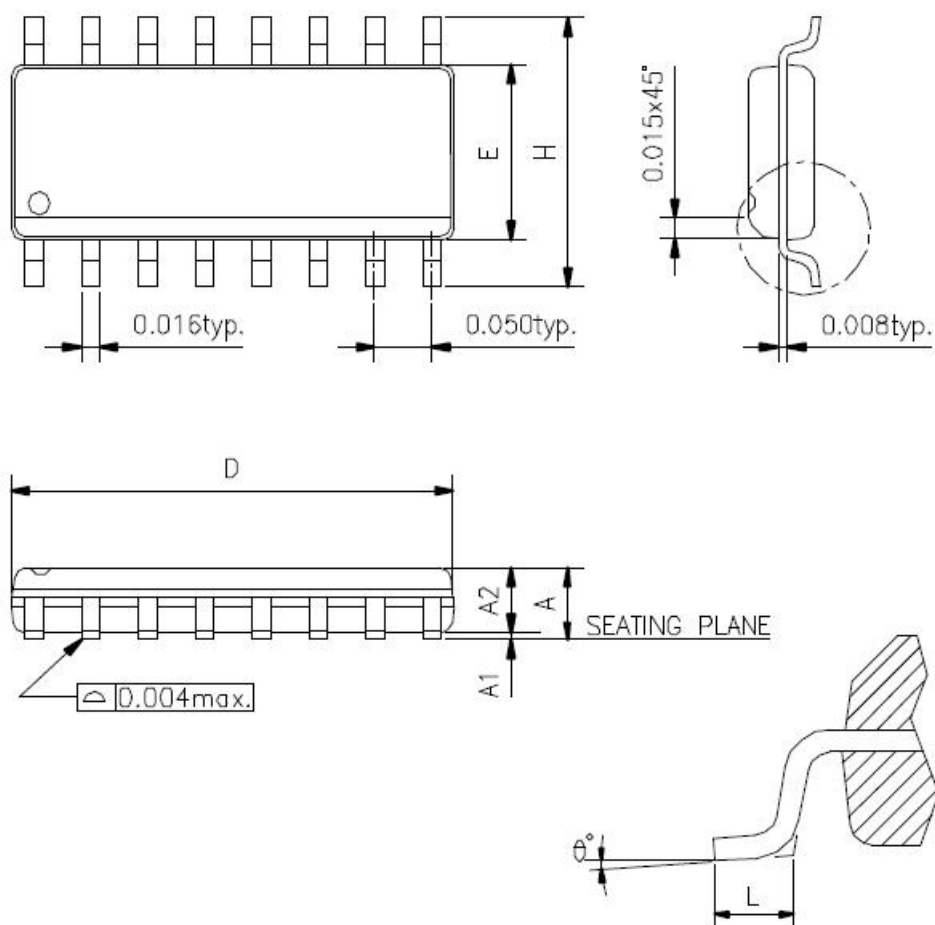
Symbol	INCH		
	Min.	Nom.	Max.
A	0.093	--	0.104
A1	0.004	--	0.012
D	0.697	--	0.713
E	0.291	--	0.299
H	0.394	--	0.419
L	0.016	--	0.050
θ°	0°	--	8°

SOP 24



Symbol	INCH		
	Min.	Nom.	Max.
A	0.093	0.099	0.104
A1	0.004	--	0.012
D	0.599	0.600	0.614
E	0.291	0.295	0.299
H	0.394	0.406	0.419
L	0.016	0.035	0.050
θ°	0°	-	8°

SOP 16



Symbol	INCH		
	Min.	Nom.	Max.
A	0.053	---	0.069
A1	0.004	---	0.010
D	0.386	---	0.394
E	0.150	---	0.157
H	0.228	---	0.244
L	0.016	---	0.050
θ °	0°	-	8°



8. 免责声明

本文件所载信息据信均为正确无误。

凌通科技所出售之集成电路仅受销售条件中所规定担保及专利赔偿条款之规范。凌通科技对于本文件所载之信息或本文件所述芯片无侵害他人专利之情事，不负任何明示、法定默示或叙述性之担保。另外，不论为何目的使用，凌通科技不保证其市场性及适合性。凌通科技保留随时暂停生产或变更规格及价格之权利，而无须为任何通知。兹此敬告本文件阅读者于发出订单前应确认本文件所载资料表及其他信息符合现况。本文件所述产品系拟为一般商业应用之使用。涉及特殊环境或可靠性要求之应用，例如军事设备或医疗维持生活设备等，未经凌通科技就此等应用目的另行处理时，特别不建议之。敬请注意本文件所述应用电路仅系供参考之用。

9. 修订记录

日期	版本号	描述	页数
2016 年 9 月 23 日	1.0	初版	111