



M5STACK

M5PM1 芯片使用手册

V 1.9

深圳明栈信息科技有限公司

目录

一、 概述	1
1 资源	1
2 功能	1
3 自定义固件引脚排列	1
二、 引脚定义	3
三、 寄存器映射	4
四、 关键寄存器详解	12
1 系统寄存器	12
2 GPIO 寄存器	15
3 ADC 寄存器	17
4 PWM 控制寄存器	18
5 系统定时器	19
6 中断与唤醒控制	20
7 按键配置	22
8 NeoPixel 控制模块	23
9 AW8737A PULSE	23
10 NEO 缓存区	24
11 RTC 缓存区	24
五、 附加功能说明	25
1 ADC 功能	25
2 PWM 输出	25
3 PWR_BTN 按键	25
4 LED 指示灯	26
5 低电压保护 (LVP)	26
6 I2C 空闲休眠	26
7 中断唤醒	26
8 IRQ 处理	27
六、 使用案例	28
1 GPIO 唤醒	28
2 RGB	28
3 ADC	29
4 PWM	29
5 TIM	29
附页	30



一、概述

M5PM1 为烧录 M5Stack 自定义电源管理功能固件实现电源端口控制、充电控制、定时唤醒等功能的电源管理芯片。

1 资源

- (1) 5 路可复用 GPIO
- (2) 1 组 I²C 接口
- (3) 32 位定时器
- (4) 32 字节 RTC RAM 保护区

2 功能

- (1) 5 路 GPIO, 扩展功能:
 - 2 路可复用为 12-bit ADC
 - 2 路可复用为 PWM
 - 1 路可复用为 LED 控制 (RGB565)
- (2) GPIO 可编程设置上下拉电阻、开漏/推挽输出、中断极性控制
- (3) 可获取内置温度传感器温度及内部参考电压
- (4) I²C 接口支持 100kHz (默认) / 400kHz 模式, 地址为 0x6E
- (5) 支持脉宽调制 AW8737A 控制音频信号幅度
- (6) 支持一次性驱动最多 32 颗 Neopixel RGB LED

3 自定义固件引脚排列

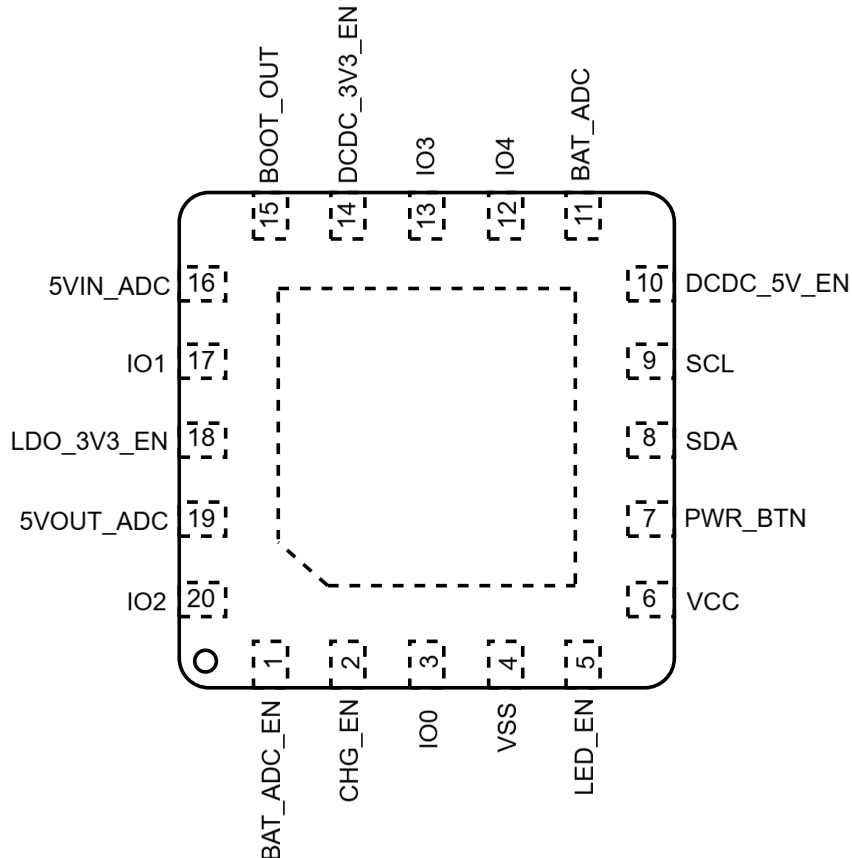


图1 M5PM1 引脚图



表1 自定义固件引脚表

编号	Pin 名称	Pin 类型	上下拉
1	BAT_ADC_EN	推挽输出	无
2	CHG_EN	推挽输出	无
3	IO0	GPIO	无
4	VSS	电源	无
5	LED_EN	推挽输出	无
6	VCC	电源	无
7	PWR_BTN	输入	上拉
8	SDA	I ² C	无
9	SCL	I ² C	无
10	DCDC_5V_EN	推挽输出	无
11	BAT_ADC	ADC	无
12	IO4	GPIO	无
13	IO3	GPIO	无
14	DCDC_3V3_EN	推挽输出	无
15	BOOT_OUT	开漏输出	无
16	5VIN_ADC	ADC	无
17	IO1	GPIO	无
18	LDO_3V3_EN	推挽输出	无
19	5VOUT_ADC	ADC	无
20	IO2	GPIO	无



二、引脚定义

表2 引脚详细定义表

Pin	描述	默认功能	复用功能	备注
IO0	GPIO 端口 0, 支持唤醒	GPIO	Neopixel 输出	唤醒与 IO2 互斥
IO1	GPIO 端口 1	GPIO	ADC1	-
IO2	GPIO 端口 2, 支持唤醒	GPIO	ADC2	唤醒与 IO0 互斥
IO3	GPIO 端口 3, 支持唤醒	GPIO	PWM1	唤醒与 IO4 互斥
IO4	GPIO 端口 4, 支持唤醒	GPIO	PWM2	唤醒与 IO3 互斥
BAT_ADC_EN	电池采样使能, 高电平有效	采样使能	-	默认为高电平
CHG_EN	电池充电使能, 高电平有效	充电使能	-	默认为高电平
DCDC_5V_EN	5V DC/DC 控制, 高电平有效	DC 控制	-	默认为低电平
DCDC_3V3_EN	3.3V DC/DC 控制, 高电平有效	DC 控制	-	默认为高电平
LDO_3V3_EN	3.3V LDO 控制, 高电平有效	LDO 控制	-	默认为高电平
5VIN_ADC	5V 输入 ADC 采集口	ADC	-	分压系数 1:1
5VOUT_ADC	5V 输出 ADC 采集口	ADC	-	分压系数 1:1
BAT_ADC	电池电压 ADC 采集口	ADC	-	分压系数 1:1
PWR_BTN	电源控制按键输入	按键检测	-	默认上拉
LED_EN	状态指示 LED 控制	LED 控制	-	默认为高电平
BOOT_OUT	控制主控 ESP32 BOOT	BOOT	-	默认为高电平
SDA	I ² C 数据线	I ² C	-	开漏模式, 需外部上拉电阻
SCL	I ² C 时钟线	I ² C	-	开漏模式, 需外部上拉电阻

注:

1. 所有 GPIO 输出类型默认都是**开漏模式**, 包括 Neopixel 驱动、PWM 输出等, 如果没有外接上拉电阻, 需要配置为推挽模式才能正常输出。

2. LED 为 M5PM1 专有 LED 自控逻辑, 不建议用作其他用途。



三、寄存器映射

表3 寄存器映射表

寄存器名称	类型	地址	位	R/W	默认	描述	复位	下载	关机
Device_ID	System	0x00	[7:0]	R	0x50	设备类型	—	—	—
Device_Model	System	0x01	[7:0]	R	0x20	设备型号	—	—	—
HW_REV	System	0x02	[7:0]	R	0x05	硬件版本号	—	—	—
SW_REV	System	0x03	[7:0]	R	0x06	固件版本号	—	—	—
PWR_SRC	System	0x04	[7:3] Reserved [2:0] VALID	R	—	电源来源位图	—	—	—
WAKE_SRC	System	0x05	[7] Reserved [6:0] FLAGS	R/W	—	唤醒源标志	—	—	—
PWR_CFG	System	0x06	[7:5] Reserved [4] LED CONTROL [3] 5VIN/OUT [2] 3.3V_LDO_EN [1] 3.3V_DCDC_EN [0] CHG_EN	R/W	0x17	电源管理位	0b0001011x 充电状态不会受复位影响	0b0001011x 充电状态不会受复位影响	0b0001011x 充电状态不会受关机影响
HOLD_CFG	System	0x07	[7] Reserved [6] 5VIN/OUT [5] 3.3V LDO [4:0] GPIO4~0	R/W	0x00	相应 bit 设置为 1, 对应的 GPIO、LDO、5VIN/OUT 状态在关机后会保留。进入下载模式, 或触发 Reset (包括 I ² C 看门狗复位、命令复位和用户定时器复位), 该寄存器会复位为 0x00	0x00	0x00	—
BATT_LVP	System	0x08	[7:0]	R/W	0x40	低压阈: 2000 mV + n×7.81 mV	—	—	—



寄存器名称	类型	地址	位	R/W	默认	描述	复位	下载	关机
I2C_CFG	System	0x09	[7:5] Reserved [4] SPD [3:0] SLP_TO	R/W	0x00	—	—	—	—
WDT_CNT	System	0x0A	[7:0]	R/W	0x00	看门狗倒计时(秒), 设置 0 为关闭看门狗功能	—	—	—
WDT_KEY	System	0x0B	[7:0]	W	—	写 0xA5 喂狗/复位	—	—	—
SYS_CMD	System	0x0C	[7:4] KEY(0xA) [3:2] Reserved [1:0] CMD	W	—	系统命令	—	—	—
—	—	0x0D-0x0F				保留	—	—	—
GPIO_MODE	GPIO	0x10	[7:5] Reserved [4:0] GPIO4~0	R/W	0x00	1=输出, 0=输入 (必须要对应的 GPIO_FUNC 设置为 00 才生效)	0x00	0x00	0b000xxxxx 由 GPIO_Power_Hold 寄存器 (0x07) bit0~bit4 决定, 1=保持, 0=状态复位
GPIO_OUT	GPIO	0x11	[7:5] Reserved [4:0] GPIO4~0	R/W	0x00	1=高, 0=低 (必须要对应的 GPIO_FUNC 设置为 00 才生效)	0x00	0x00	0b000xxxxx 由 GPIO_Power_Hold 寄存器 (0x07) bit0~bit4 决定, 1=保持, 0=状态复位
GPIO_IN	GPIO	0x12	[7:5] Reserved [4:0] GPIO4~0	R	—	实时输入值	—	—	—
GPIO_DRV	GPIO	0x13	[7:6] Reserved [5] LED EN [4:0] GPIO4~0	R/W	0x1F	1=开漏, 0=推挽	0x1F	0x1F	0b00xxxxxx 其中 LED EN 不受影响, GPIO0-GPIO4 由寄存器 GPIO_Power_Hold (0x07) bit0~bit4 决定, 1=保持, 0= 状态复位



M5PM1 芯片使用手册

寄存器名称	类型	地址	位	R/W	默认	描述	复位	下载	关机
GPIO_PU/PD_0	GPIO	0x14	[7:6] GPIO3 [5:4] GPIO2 [3:2] GPIO1 [1:0] GPIO0	R/W	0x00	每 2 bit 设置 PULL_NO: 00 PULL_UP: 01 PULL_DOWN: 10	0x00	0x00	0bxxxxxxx 由 GPIO_Power_Hold 寄存器 (0x07) bit0~bit3 决定, 1=保持, 0=状态复位
GPIO_PU/PD_1	GPIO	0x15	[7:2] Reserved [1:0] GPIO4	R/W	0x00	同上; 其余位保留	0x00	0x00	0b000000xx 由 GPIO_Power_Hold 寄存器 (0x07) bit4 决定, 1=保持, 0=状态复位
GPIO_FUNC0	GPIO	0x16	[7:6] GPIO3 [5:4] GPIO2 [3:2] GPIO1 [1:0] GPIO0	R/W	0x00	每 2 bit 设置 GPIO: 00 IRQ: 01 特殊功能: 11 保留: 10	0x00	0x00	0bxxxxxxx 由 GPIO_Power_Hold 寄存器 (0x07) bit0~bit3 决定, 1=保持, 0=状态复位
GPIO_FUNC1	GPIO	0x17	[7:2] Reserved [1:0] GPIO4	R/W	0x00	同上; 其余位保留	0x00	0x00	0b000000xx 由 GPIO_Power_Hold 寄存器 (0x07) bit4 决定, 1=保持, 0=状态复位
GPIO_WAKE_EN	GPIO	0x18	[7:5] Reserved [4:0] GPIO4~0	R/W	0x00	1=对应 GPIOWake 功能使能, 0=对应 GPIOWake 功能关闭 (GPIO1 中断线路和 SDA 冲突, 不能使用 WAKE 功能。GPIO0 和 GPIO2 共用一个中断线路, 两者互斥。GPIO3 和 GPIO4 共用一个中断线路, 两者互斥。)	—	—	—



M5PM1 芯片使用手册

寄存器名称	类型	地址	位	R/W	默认	描述	复位	下载	关机
GPIO_WAKE_CFG	GPIO	0x19	[7:5] Reserved [4:0] GPIO4~0	R/W	0x00	对应 GPIO 1=上升沿唤醒 0=下降沿唤醒	—	—	—
—	—	0x1A-0x1F				保留	—	—	—
VREF_L	ADC	0x20	[7:0]	R	—	MCU ADC (mV) 参考电压低 8 bit	—	—	—
VREF_H	ADC	0x21	[7:0]	R	—	MCU ADC (mV) 参考电压高 8 bit	—	—	—
VBAT_L	ADC	0x22	[7:0]	R	—	电池电压 (mV) 低 8 bit	—	—	—
VBAT_H	ADC	0x23	[7:0]	R	—	电池电压 (mV) 高 8 bit	—	—	—
VIN_L	ADC	0x24	[7:0]	R	—	VIN 电压(mV) 低 8 bit	—	—	—
VIN_H	ADC	0x25	[7:0]	R	—	VIN 电压(mV) 高 8 bit	—	—	—
5VOUT_L	ADC	0x26	[7:0]	R	—	5VOUT 电压 (mV) 低 8 bit	—	—	—
5VOUT_H	ADC	0x27	[7:0]	R	—	5VOUT 电压 (mV) 高 8 bit	—	—	—
ADC_RES_L	ADC	0x28	[7:0]	R	—	ADC 结果低 8 bit	—	—	—
ADC_RES_H	ADC	0x29	[7:4] Reserved [3:0] Data[11:8]	R	—	ADC 结果高 4 bit	—	—	—
ADC_CTRL	ADC	0x2A	[7:4] Reserved [3:1] CH_SEL [0] START	R/W	0x00	描述: START=1 开始转换 (转换完成后自动清 0), CH_SEL 通道选择 (有效通道 1、2、6, 其中 1 和 2 表示 GPIO1 和 GPIO2, 必须要对应的 GPIO_FUNC 设置为 11 才生效, 6 是芯片内部温度采集, 单位是℃)			



M5PM1 芯片使用手册

寄存器名称	类型	地址	位	R/W	默认	描述	复位	下载	关机
—	—	0x2B-0x2F				保留	—	—	—
PWM0_L	PWM	0x30	[7:0] Duty[7:0]	R/W	0x00	PWM0 占空比 低 8 bit	—	—	—
PWM0_HC	PWM	0x31	[7:6] Reserved [5] POL [4] EN [3:0] Duty[11:8]	R/W	0x00	EN=1, 启动 POL=1, 低有效 PWM0 占空比高 8 bit	—	—	—
PWM1_L	PWM	0x32	[7:0] Duty[7:0]	R/W	0x00	PWM1 占空比 低 8 bit	—	—	—
PWM1_HC	PWM	0x33	[7:6] Reserved [5] POL [4] EN [3:0] Duty[11:8]	R/W	0x00	EN=1, 启动 POL=1, 低有效 PWM1 占空比高 8 bit	—	—	—
PWM_FREQ_L	PWM	0x34	[7:0]	R/W	0xF4	PWM 频率低 8 bit	—	—	—
PWM_FREQ_H	PWM	0x35	[7:0]	R/W	0x01	PWM 频率高 8 bit	—	—	—
—	—	0x36-0x37				保留	—	—	—
TIM_CNT_BYTE_0	Timer	0x38	[7:0]	R/W	0x00	定时唤醒计数器 Byte0 (s)	—	—	—
TIM_CNT_BYTE_1	Timer	0x39	[7:0]	R/W	0x00	定时唤醒计数器 Byte1 (s)	—	—	—
TIM_CNT_BYTE_2	Timer	0x3A	[7:0]	R/W	0x00	定时唤醒计数器 Byte2 (s)	—	—	—
TIM_CNT_BYTE_3	Timer	0x3B	[7] Reserved [6:0]	R/W	0x00	定时唤醒计数器 Byte3 (s)	—	—	—



寄存器名称	类型	地址	位	R/W	默认	描述	复位	下载	关机
TIM_CFG	Timer	0x3C	[7:4] Reserved [3] ARM [2:0] ACTION	R/W	0x00	ARM=1 计数 (如果 TIM_CNT 为 0, ARM 会被自动清零); ACTION 见表 4	定时器触发时清 0	定时器触发时清 0	0x00
TIM_KEY	Timer	0x3D	[7:0]	W	—	写 0xA5 清零并重载	—	—	—
—	—	0x3E-0x3F				保留	—	—	—
IRQ Status 1	IRQ	0x40	[7:5] Reserved [4:0] GPIO4~0	R/W	0x00	描述: 对应 GPIO 的 bit 等于 1, 表示对应 GPIO 的电平发生变化, 通过寄存器 0x06 获取对应的 GPIO 电平 (除了设置为 IRQ 的 GPIO, 其他 IO 必须要对应的 GPIO_FUNC 设置为 00, GPIO_MODE 设置为 0), 此时设置为 IRQ 的 GPIO 会拉低, 只有对 IRQ Status 清零后才会释放该 GPIO 重新拉高			
IRQ Status 2	IRQ	0x41	[7:6] Reserved [5] 电池移除 [4] 电池插入 [3] 5VINOUT 移除 [2] 5VINOUT 插入 [1] 5V IN 移除 [0] 5V IN 插入	R/W	0x00	描述: 对应 bit 等于 1, 表示对应的事件发生, 此时设置为 IRQ 的 GPIO 会被拉低, 只有对 IRQ Status 清零后才会释放该 GPIO 重新拉高 注意: 1. 电池的插入移除只用在电池充电未使能的时候才有效, 电池充电使能时无效。 2. 5VIN/OUT 的插入移除只有在 5VIN/OUT 设置为 INPUT 时才有效, 设置为 OUTPUT 无效。			
IRQ Status 3	IRQ	0x42	[7:3] Reserved [2] DOUBLE_CLICK [1] WAKEUP [0] SINGLE_CLICK	R/W	0x00	描述: 1. bit0 同时是复位判断位, PWR_BTN 按键复位功能被屏蔽后, 单击 PWR_BTN 才会触发按键单击中断。 2. WAKE_SRC (0x2F) 与 IRQ Status 3 的 bit1 相关, 即如果 WAKE_SRC (0x2F)不清零, IRQ Status 3 bit1 始终为 1。 3. bit2 同时是关机判断位, PWR_BTN 按键双击关机功能被屏蔽后, 双击 PWR_BTN 才会触发按键双击中断。			
IRQ Status 1_Mask	IRQ	0x43	[7:5] Reserved [4:0] GPIO4~0	R/W	0x00	对应的位设置位 1 说明屏蔽该中断	—	—	—



寄存器名称	类型	地址	位	R/W	默认	描述	复位	下载	关机
IRQ Status 2_Mask	IRQ	0x44	[7:6] Reserved [5] 电池移除 [4] 电池插入 [3] 5VINOUT 移除 [2] 5VINOUT 插入 [1] 5V IN 移除 [0] 5V IN 插入	R/W	0x00	对应的位设置位 1 说明屏蔽该中断	—	—	—
IRQ Status 3_Mask	IRQ	0x45	[7:3] Reserved [2] Double click [1] Wakeup [0] Click	R/W	0x00	对应的位设置位 1 说明屏蔽该中断	—	—	—
—	—	0x46-0x47				保留	—	—	—
BTN_Status	BTN	0x48	[7] BTN_EVENT [6:1] Reserved [0] BTN_Status	R	0x00	按键状态	—	—	—
BTN_CFG_1	BTN	0x49	[7] DL_LOCK [6:5] DBL [4:3] LONG [2:1] SINGLE [0] SINGLE_RESET_DIS	R/W	0x2A	按键配置寄存器 1	—	—	—
BTN_CFG_2	BTN	0x4A	[7:1] Reserved [0] DOUBLE_POWEROFF_DIS	R/W	0x00	按键配置寄存器 2	—	—	—
—	—	0x4B-0x4F				保留	—	—	—
NEO_CFG	RGB	0x50	[7] Reserved [6] REFRESH [5:0] LED_CNT	R/W	0x00	NeoPixel 数,刷新控制 32 个灯需要大概 7ms, 此时禁止中断,即 I ² C 通信会被禁止	—	—	—



寄存器名称	类型	地址	位	R/W	默认	描述	复位	下载	关机
—	—	0x51-0x52				保留	—	—	—
AW8737A	PULSE	0x53	[7] REFRESH [6:5] NUM [5:0] GPIO	R/W	0x00	GPIO 有效值 0~4	—	—	—
—	—	0x54-0x5F				保留	—	—	—
NEO_PLXn_L/H	RGB	0x60-0x9F	[7:0]	R/W	0x00	最多 32 × RGB565 像素数据, 共 64 字节	—	—	—
RTC_MEM	RTC	0xA0-0xBF	[7:0]	R/W	0x00	32 Byte RTC 备份 RAM	—	—	—
—	—	0xC0-0xFF				保留	—	—	—

注:

1. RES: 保留位
2. 复位包括按键复位、命令复位、I²C 看门狗复位、定时器复位
3. 关机包括按键关机、命令关机、定时器关机



四、关键寄存器详解

⚠ 寄存器访问限制：I²C 连续读写仅支持特定区块 (0x00 ~ 0x0C、0x10 ~ 0x19、0x20 ~ 0x2A、0x30 ~ 0x35、0x38 ~ 0x3D、0x40 ~ 0x45、0x48 ~ 0x4A、0x50、0x53、0x60 ~ 0x9F、0xA0 ~ 0xBF)，跨区块操作需分次执行。

1 系统寄存器

(1) Device_ID (0x00):

- 权限：R
- 默认值：0x50
- 功能：设备类型

(2) Device_Model (0x01):

- 权限：R
- 默认值：0x20
- 功能：设备型号

(3) HW_REV (0x02):

- 权限：R
- 默认值：0x05
- 功能：硬件版本号

(4) SW_REV (0x03):

- 权限：R
- 默认值：0x06
- 功能：软件版本号

(5) PWR_SRC (0x04):

- 权限：R
- 默认值：无
- 功能：电源来源状态
- 位定义：

[7~3]: 保留

[2] BAT: 电池有效

[1] 5VINOUT: 5VINOUT 有效 (仅当 5V 升压关闭时, 5V 升压打开时, 该 bit 为 0)

[0] 5VIN: 5VIN 有效

注:

1. 电源的多个来源可以同时存在。系统通过检测对应的 ADC 引脚是否有电压来判别当前的电源来源状态。
2. 当开启电池充电功能但电池并未连接时, ADC 测得的电压会出现不稳定的漂浮现象, 导致系统在判定电源来源时会出现不稳定现象, 建议没有电池时关闭电池充电。

**(6) WAKE_SRC (0x05):**

- 权限: R/W
- 默认值: 无
- 功能: 唤醒来源状态
- 位定义:

[7]: 保留

[6] 5V INOUT: 5V INOUT 插入唤醒 (仅当 5V 升压关闭时)

[5] EXT_WAKE: GPIO WAKE 唤醒

[4] CMD_RST: 复位命令唤醒

[3] RSTBTN: 按钮复位唤醒

[2] PWRBTN: 电源按钮唤醒

[1] VIN: 5VIN 插入唤醒

[0] TIM: 定时器唤醒

注: 该寄存器写入只能清除标志位以便下一次唤醒源有效检测, 不能指定唤醒源。

(7) PWR_CFG (0x06):

- 权限: R/W
- 默认值: 0x17 (0b0001 0111)
- 功能: 电源开关控制
- 位定义:

[7:5]: 保留

[4] LED CONTROL: 1=LED EN 输出高电平, 0=LED EN 输出低电平

[3] 5VIN/OUT: 1=5V 升压输出, 0=5V 升压关闭 (可接外部输入电源)

[2] 3.3V_LDO_EN: 1=使能 3.3V LDO,0=关闭 3.3V LDO

[1] 3.3V_DCDC_EN: 1=使能 3.3V DCDC,0=关闭 3.3V DCDC

[0] CHG_EN: 1=使能充电,0=关闭充电

(8) HOLD_CFG(0x07):

- 权限: R/W
- 默认值: 0x00
- 功能: 保持寄存器, 包括电源保持、GPIO 保持
- 位定义:

[7]: 保留

[6] 5vin/out: 5vin/out = 1, 5vin/out 电源关机后保持; 5vin/out = 0, 5vin/out 电源关机后不保持



[5] **ldo_3v3**: ldo_3v3 = 1, ldo_3v3 电源关机后保持; ldo_3v3 = 0, ldo_3v3 电源关机后不保持

[4] **gpio4**: gpio4 = 1, GPIO4 的状态关机后保持; gpio4 = 0, GPIO4 的状态关机后复位

[3] **gpio3**: gpio3 = 1, GPIO3 的状态关机后保持; gpio3 = 0, GPIO3 的状态关机后复位

[2] **gpio2**: gpio2 = 1, GPIO2 的状态关机后保持; gpio2 = 0, GPIO2 的状态关机后复位

[1] **gpio1**: gpio1 = 1, GPIO1 的状态关机后保持; gpio1 = 0, GPIO1 的状态关机后复位

[0] **gpio0**: gpio0 = 1, GPIO0 的状态关机后保持; gpio0 = 0, GPIO0 的状态关机后复位

(9) BATT_LVP (0x08):

- 权限: R/W
- 默认值: 0x40
- 功能: 低压阈值寄存器, 低于低压阈值就会强制关机

低压值计算: $2.0\text{ V} + \text{reg_value} \times 7.81\text{ mV}$

注: 重新开机条件, 满足下面条件之一就可以重新开机

1. 电池电压比配置电压高 100mV
2. 插入 5VIN
3. 插入 5VINOUT (仅当 5V 升压关闭时)

(10) I2C_CFG (0x09):

- 权限: R/W
- 默认值: 0x00
- 功能: I²C 速度配置、空闲休眠配置
- 位定义:

[7:5] **DBL**: 保留

[4] **SPD**: 0=100 k,1=400 k

[3-0] **SLP_TO**: 代表 I²C 多少秒没有通信后, M5PM1 休眠, 设为 0 关闭该功能

注:

1. 空闲休眠功能一旦配置后, M5PM1 不会自动清除该设置, 需由用户手动清除。
2. 配置空闲休眠功能并成功触发后, M5PM1 进入休眠状态, 如果通过 I²C 通讯进行唤醒, 第一次 I²C 通讯仅用于唤醒操作, 会导致该次通讯失败, 后续通讯才能正常进行, 注意第一次通信唤醒之后 300ms 内 M5PM1 未接收到完整的地址仍然会再次进入休眠。

(11) WDT_CNT (0x0A):

- 权限: R/W
- 默认值: 0x00
- 功能: 软件看门狗, 超时复位

注: 看门狗倒计时单位为秒, 设 0 为关闭该功能。

**(12) WDT_KEY (0x0B):**

- 权限：W
- 默认值：无
- 功能：软件看门狗喂狗

注：写 0xA5 清零并重载。

(13) SYS_CMD (0x0C):

- 权限：W
- 默认值：无
- 功能：系统指令寄存器
- 位定义：

[7:4] KEY: 0xA

[3-2]: 保留

[1-0] CMD: 命令, 01=关机, 10=重启, 11=下载

- 写命令: KEY=0xA + CMD (01=关机, 10=重启, 11=下载)

2 GPIO 寄存器**(1) GPIO_MODE (0x10):**

- 权限：R/W
- 默认值：0x00
- 功能：GPIO 模式
- 位定义：

[7:5]: 保留

[4:0]: 对应 GPIO4-GPIO0 方向 (1=输出, 0=输入)

- 生效条件：GPIO_FUNCx 设置为 00。

(2) GPIO_OUT (0x11):

- 权限：R/W
- 默认值：0x00
- 功能：GPIO 输出电平
- 位定义：

[7:5]: 保留

[4:0]: 对应 GPIO4-GPIO0 输出电平 (1=高电平, 0=低电平)

- 生效条件：

GPIO_FUNCx 设置为 00; GPIO_MODE 设置为 1

**(3) GPIO_IN (0x12):**

- 权限：R
- 默认值：无
- 功能：GPIO 输入状态
- 位定义：
 - [7:5]：保留
 - [4:0]：对应 GPIO4-GPIO0 输入电平（1=高电平，0=低电平）

(4) GPIO_DRV (0x13):

- 权限：R/W
- 默认值：0x1F
- 功能：GPIO 输出类型
- 位定义：
 - [7:5]：保留
 - [4:0]：对应 GPIO4-GPIO0 输出类型（1=开漏，0=推挽）

优先级说明：

输出类型配置优先于复用功能，不会随着复用功能的启用或关闭而失效。

例如 GPIO 复用为 PWM 时，若 GPIO_DRV=1（开漏），实际输出仍为开漏模式。

(5) GPIO_PU/PD_0 (0x14), GPIO_PU/PD_1 (0x15):

- 权限：R/W
- 默认值：0x00
- 功能：GPIO 上下拉配置
- 位定义（每 2bit 控制 1 个 GPIO）：
 - 00：无上下拉
 - 01：上拉
 - 10：下拉

(6) GPIO_FUNC0 (0x16), GPIO_FUNC1 (0x17):

- 权限：R/W
- 默认值：0x00
- 功能：GPIO 功能
- 位定义（每 2bit 控制 1 个 GPIO）：
 - 00：标准 GPIO；
 - 01：IRQ 中断
 - 11：复用功能 (NeoPixel/ADC/PWM)；
 - 10：保留

**(7) GPIO_WAKE_EN (0x18):**

- 权限: R/W
- 默认值: 0x00
- 功能: GPIO 唤醒使能
- 位定义:

[7:5]: 保留

[4:0]: 对应 GPIO4-GPIO0 唤醒使能 (1=使能, 0=使能, GPIO1 不支持)

注:

1. 配置 GPIO 的 WAKE 配置不会因为进入下载模式、复位和关机而失效。
2. WAKE 是否启用上下拉, 由 GPIO_PU/PD_0 (0x14) 和 GPIO_PU/PD_1 (0x15) 控制。

(8) GPIO_WAKE_CFG (0x19):

- 权限: R/W
- 默认值: 0x00
- 功能: 唤醒配置
- 位定义:

[7:5]: 保留

[4:0]: 对应 GPIO4-GPIO0 方向边沿配置 (1=上升沿, 0=下降沿, GPIO1 不支持)。

- 生效条件: WAKE_EN 之后才生效

注: 配置 GPIO 的 WAKE 配置不会因为进入下载模式、复位和关机而失效。

3 ADC 寄存器**(1) VREF_L (0x20)、VREF_H (0x21):**

- 权限: R
- 默认值: 无
- 功能: 内部参考电压 VREF

注: MCU ADC 参考电压单位为 mV

(2) VBAT_L (0x22)、VBAT_H (0x23):

- 权限: R
- 默认值: 无
- 功能: 电池电压 VBAT

注: 电池电压单位为 mV

(3) VIN_L (0x24)、VIN_H (0x25):

- 权限: R



- 默认值：无
 - 功能：5V 输入电压 VIN
- 注：5V 输入电压单位为 mV

(4) 5VOUT_L (0x26)、5VOUT_L (0x27):

- 权限： R
 - 默认值：无
 - 功能：5V 输出电压 5VOUT
- 注：5V 输出电压单位为 mV

(5) ADC_RES_L (0x28)、ADC_RES_H (0x29):

- 权限： R
- 默认值：无
- 功能：通道转换值
- 注：
 1. 组合为 12 位数据 ([11:0]) 当通道选择为 1 或 2 时，组合为 12 位 ADC 结果，范围 0-0xFFF。
 2. 当通道选择为 6 时，组合为芯片内部温度，单位是°C。

(6) ADC_CTRL (0x2A):

- 权限： R/W
- 默认值： 0x00
- 功能： ADC 转换寄存器
- 位定义：
 - [7:4]: 保留
 - [3:1] CH_SEL: 通道选择 (1=GPIO1, 2=GPIO2, 6=内部温度通道)
 - [0] START: 写 1 启动转换 (完成后自动清零)

4 PWM 控制寄存器

(1) PWM0_L (0x30):

- 权限： R/W
- 默认值： 0x00
- 功能： PWM 寄存器 (占空比低 8 位)

(2) PWM0_HC (0x31):

- 权限： R/W
- 默认值： 0x00
- 功能： PWM 寄存器



- 位定义:

- [7:6]: 保留

- [5] POL: 极性 (1=低有效)

- [4] EN: 使能 (1=启动)

- [3:0] Duty[11:8]: 占空比高 4 位 (与 PWM0_L 组合为 12 位占空比)。

(3) PWM1_L (0x32):

- 权限: R/W

- 默认值: 0x00

- 功能: PWM 寄存器 (占空比低 8 位)

(4) PWM1_HC (0x33):

- 权限: R/W

- 默认值: 0x00

- 功能: PWM 寄存器

- 位定义:

- [7:6]: 保留

- [5] POL: 极性 (1=低有效)

- [4] EN: 使能 (1=启动)

- [3:0]: 占空比高 4 位 (与 PWM1_L 组合为 12 位占空比)

(5) PWM_FREQ_L (0x34)、PWM_FREQ_H (0x35):

- 权限: R/W

- 默认值: 0xF4、0x01

- 功能: 配置 PWM 频率, 单位: Hz

5 系统定时器

(1) TIM_CNT_BYTE_0~3 (0x38-0x3B):

- 权限: R/W

- 默认值: 0x00

- 功能: 定时唤醒计数器, 单位: 秒, 范围 0-214748364

(2) TIM_CFG (0x3C):

- 权限: R/W

- 默认值: 0x00

- 功能: 定时器功能配置

- 位定义:



[7:6]: 保留

[3] ARM: 1=启动定时器 (计数到 0 自动清零)

[2:0] ACTION: 定时器动作 (见表 4)

注:

1. 系统关机重新上电会清除 TIM_CFG 寄存器。
2. 置位唤醒标志、系统上电、系统重启、系统关机生效一次就会清除 TIM_CFG 寄存器。

表4 定时器动作真值表

ACTION	功能
0	停止计数器
1	置位唤醒标志
10	系统重启
11	系统上电
100	系统关机

(3) TIM_KEY (0x3D):

- 权限: W
- 默认值: 无
- 功能: 重载定时器 (写 0xA5)

6 中断与唤醒控制

(1) IRQ Status 1 (0x40):

- 权限: R/W
- 默认值: 0x00
- 功能: IRQ 寄存器 1
- 位定义:

[7:5]: 保留

[4:0]: 对应 GPIO4-GPIO0 的电平变化情况, 数值 1 说明该 GPIO 有电平变化

注: 该寄存器用户只能清零, 置 1 由系统自动进行。

(2) IRQ Status 2 (0x41):

- 权限: R/W
- 默认值: 0x00
- 功能: IRQ 寄存器 2
- 位定义: 对应 bit 数值 1 说明该事件发生

[7:6]: 保留

[5] Battery Remove: 电池移除, BAT 电压 $\geq 2400\text{mV}$ \rightarrow BAT 电压 $< 2400\text{mV}$

[4] Battery Add: 电池插入, BAT 电压 $\leq 2400\text{mV}$ \rightarrow BAT 电压 $> 2400\text{mV}$



[3] 5VINOUT Remove: 5VIN/OUT 移除, 5VINOUT 电压 $\geq 2400\text{mV}$ \rightarrow 5VINOUT 电压 $< 2400\text{mV}$

[2] 5VINOUT Add: 5VIN/OUT 插入, 5VINOUT 电压 $\leq 2400\text{mV}$ \rightarrow 5VINOUT 电压 $> 2400\text{mV}$

[1] 5VIN Remove: 5VIN 移除, 5VIN 电压 $\geq 2400\text{mV}$ \rightarrow 5VIN 电压 $< 2400\text{mV}$

[0] 5VIN Add: 5VIN 插入, 5VIN 电压 $\leq 2400\text{mV}$ \rightarrow 5VIN 电压 $> 2400\text{mV}$

注: 该寄存器用户只能清零, 置 1 由系统自动进行。

● 生效条件:

1. 电池插入/移除事件仅在 CHG_EN=0 (充电关闭) 时有效。
2. 5VINOUT 插入/移除事件仅在 5VIN/OUT=0 (INPUT 模式) 时有效。

(3) IRQ Status 3 (0x42):

● 权限: R/W

● 默认值: 0x00

● 功能: IRQ 寄存器 3

● 位定义: 对应 bit 置 1 说明该事件发生

[7:3]: 保留

[2] DOUBLE_CLICK: 按键双击

[1] WAKEUP: 开机

[0] SINGLE_CLICK: 按键单击

注: 该寄存器用户只能清零, 置 1 由系统自动进行。

● 生效条件:

1. bit0 同时是复位判断位, PWR_BTN 按键复位功能被屏蔽后, 单击 PWR_BTN 才会触发按键单击中断。
2. WAKE_SRC (0x2F)与 IRQ Status 3 的 bit1 是相关的, 即如果 WAKE_SRC (0x2F)不清零, IRQ Status 3 bit1 始终为 1。
3. bit2 同时是关机判断位, PWR_BTN 按键双击关机功能被屏蔽后, 双击 PWR_BTN 才会触发按键双击中断。

(4) IRQ Status 1 Mask(0x43):

● 权限: R/W

● 默认值: 0x00

● 功能: IRQ 屏蔽寄存器 1

● 位定义:

[7:5]: 保留

[4:0]: 对应 GPIO4-GPIO0

注: 对应的位置 1 为屏蔽该中断

(5) IRQ Status 2 Mask(0x44):



- 权限： R/W
- 默认值： 0x00
- 功能： IRQ 屏蔽寄存器 2
- 位定义：
 - [5] **Battery Remove**: 电池移除, BAT 电压 $\geq 2400\text{mV}$ \rightarrow BAT 电压 $< 2400\text{mV}$
 - [4] **Battery Add**: 电池插入, BAT 电压 $\leq 2400\text{mV}$ \rightarrow BAT 电压 $> 2400\text{mV}$
 - [3] **5VINOUT Remove**: 5VIN/OUT 移除, 5VINOUT 电压 $\geq 2400\text{mV}$ \rightarrow 5VINOUT 电压 $< 2400\text{mV}$
 - [2] **5VINOUT Add**: 5VIN/OUT 插入, 5VINOUT 电压 $\leq 2400\text{mV}$ \rightarrow 5VINOUT 电压 $> 2400\text{mV}$
 - [1] **5VIN Remove**: 5VIN 移除, 5VIN 电压 $\geq 2400\text{mV}$ \rightarrow 5VIN 电压 $< 2400\text{mV}$
 - [0] **5VIN Add**: 5VIN 插入, 5VIN 电压 $\leq 2400\text{mV}$ \rightarrow 5VIN 电压 $> 2400\text{mV}$

注：对应的位置 1 为屏蔽该中断

(6) IRQ Status 3 Mask (0x45):

- 权限： R/W
- 默认值： 0x00
- 功能： IRQ 屏蔽寄存器 3
- 位定义：
 - [7:3]: 保留
 - [2] **DOUBLE_CLICK**: 按键双击
 - [1] **WAKEUP**: 开机
 - [0] **SINGLE_CLICK**: 按键单击

注：对应的位置 1 为屏蔽该中断

IRQ 注意事项:

1. 无 GPIO 被设置为 IRQ 引脚时, IRQ Status 1 (0x40)、IRQ Status 2 (0x41)和 IRQ Status 3 (0x42)会被清零。
2. 复位和进入下载模式, GPIO 会被复位, 此时没有 GPIO 被设置为 IRQ 引脚, IRQ Status 1 (0x40)、IRQ Status 2 (0x41)和 IRQ Status 3 (0x42)会被清零。此时如果你需要使用 IRQ Status 3 (0x42) Wakeup IRQ, 请不要清除 WAKE_SRC (0x05)寄存器, 直到再次把 GPIO 配置为 IRQ 引脚, 否则无法触发 IRQ Status 3 (0x42) Wakeup IRQ。

7 按键配置

(1) BTN_Status (0x48):

- 权限： R
- 默认值： 无
- 功能： 按键状态
- 位定义：



[7] **BTN_Event**: 按压事件, 1 = 按键按压过、0 表示按键未按压过, 读取后自动清 0, 系统自动置 1

[6:1]: 保留

[0] **BTN_Status**: 按压状态, 1=按下, 0=松开

(2) **BTN_CFG (0x49):**

- 权限: R/W
- 默认值: 0x2A
- 功能: BTN 配置 1
- 位定义:

[7] **DL_LOCK**: DL_LOCK=1, 禁止下载模式

[6:5] **DBL**: 双击, 00 = 125 ms, 01 = 250 ms, 10 = 500 ms, 11 = 1 s

[4:3] **LONG**: 长按, 00 = 1 s, 01=2 s, 10=3 s, 11=4 s

[2:1] **SINGLE**: 单击, 00 = 125 ms, 01 = 250 ms, 10 = 500 ms, 11 = 1 s

[0] **SINGLE_RESET_DIS**: SINGLE_RESET_DIS = 1—禁止单击复位; SINGLE_RESET_DIS = 0—启用单击复位

(3) **BTN_CFG_2 (0x4A):**

- 权限: R/W
- 默认值: 0x00
- 功能: BTN 配置 2
- 位定义:

[7:1]: 保留

[0] **DOUBLE_POWEROFF_DIS**:

DOUBLE_POWEROFF_DIS= 1—禁止双击关机; DOUBLE_POWEROFF_DIS= 0—启用双击复位

8 **NeoPixel 控制模块**

NEO_CFG (0x50):

- 权限: R/W
- 默认值: 0x00
- 功能: RGB 配置
- 位定义:

[7]: 保留

[6] **REFRESH**: 刷新控制 (32 灯约需 7ms, 期间禁用 I²C 中断)

[5:0] **LED_CNT**: 灯数量 (0-32, 0=禁用驱动)

9 **AW8737A PULSE**

PULSE_CTRL (0x53):

- 权限: R/W



- 默认值: 0x00
- 功能: AW8737A 脉冲输出控制
- 位定义:

[7] REFRESH: 1-刷新; 0-不刷新

[6:5] NUM: 取值范围 0~3, 实际输出 0~3 个脉冲

[4:0] GPIO: 取值范围 0~4, 对应硬件 GPIO0 ~ GPIO4。

注:

1. 设置成功后对应 GPIO 为输出模式, 并且 GPIO 输出对应脉冲进行调制。
2. 如果是开漏输出模式则需要有外部上拉, 否则需要先配置对应的 GPIO 为推挽输出模式。

10 NEO 缓存区

NEO_PIXn_L/H (0x60-0x9F):

- 权限: R/W
- 默认值: 0x00
- 功能: RGB 缓冲区

注: 每个灯占 2 字节 (RGB565 格式), 按顺序存储 (PIX0_L=0x36, PIX0_H=0x37, ..., PIX31_H=0x75)。

11 RTC 缓存区

RTC_MEM[0:31] (0xA0-0xBF):

- 权限: R/W
- 默认值: 0x00
- 功能: RTC 缓冲区

注: 32 字节掉电保持 RAM (这里的掉电保持是指 ESP32 掉电保持, M5PM1 掉电不保持)。

五、附加功能说明

1 ADC 功能

(1) 流程：

写入 ADC_CTRL 选择通道并启动 (START=1) → 等待 BUSY=0 → 读取 ADC_D_H/L。

(2) 内部通道 ADC 的使用

- **5VOUT**: 低电压阈值检测、5VOUT 电源插入唤醒、5VOUT 电源移除与插入检测
- **BAT**: 低电压阈值检测、电池移除与插入检测
- **5VIN**: 低电压阈值检测、5VIN 移除与插入检测

注意事项：

1. 5VOUT 有输入和输出功能，5VOUT 升压打开前，需先检测 5VOUT 电压，确保没有电源输入才能打开升压，否则应保持输入。
2. 3 个通道都有特定功能，必须严格对应接入。

2 PWM 输出

占空比 = $(DUTY[11:0] / 0xFFF) \times 100\%$ ，频率由 PWM_FREQ 寄存器设定，数值的高位与低位在两个字节，要一次写两个字节，否则会出现短时间内设置两次不同的占空比或频率的情况。

3 PWR_BTN 按键

单击复位，双击关机，长按主机进入下载模式，具体间隔时间由寄存器 BTN_CFG_1 决定。

(1) 关机

关机后，3.3V DCDC 关闭，LED_EN 关闭，但充电使能不受影响。

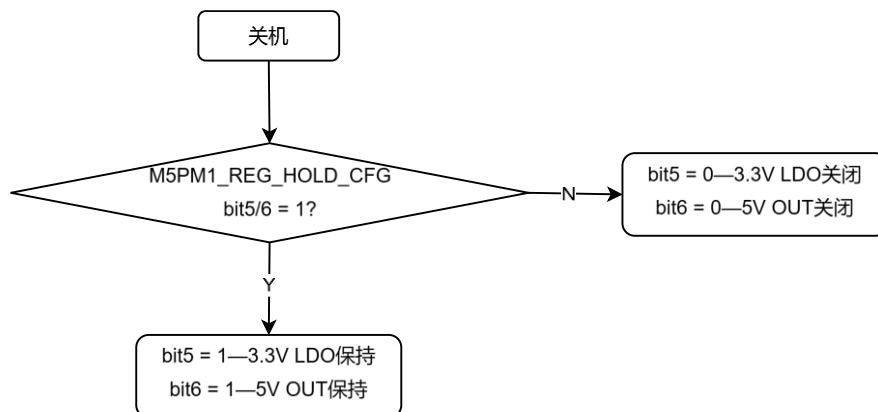


图2 关机流程图

(2) 下载模式

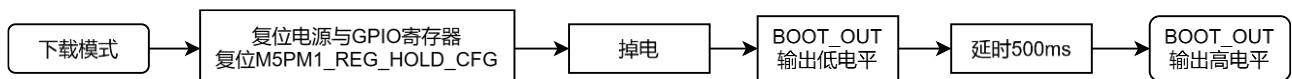


图3 下载模式流程图

(3) 复位

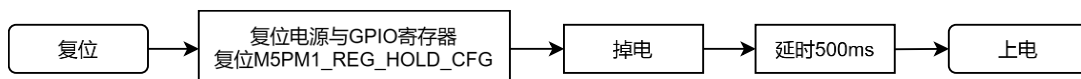


图4 复位流程图

4 LED 指示灯

(1) 流程：

设置 LED 数量 (LED_CFG[5:0]) → 写入 LED_RAM (RGB565 格式) → 触发 REFRESH=1。

(2) 不同情况下 LED 灯指示状态

- **复位：** LED 闪烁一次
- **下载模式：** LED 500ms 闪烁一次
- **按键复位屏蔽且有 GPIO 使能 IRQ 功能：** LED 200ms 闪烁一次
- **按键双击关机屏蔽且有 GPIO 使能 IRQ 功能：** LED 100ms 闪烁一次

5 低电压保护 (LVP)

低电压阈值由寄存器 BATT_LVP (0x08)决定，在 5VIN 或 5VINOUT 没有插入时，电池电压低于阈值，会自动关机。

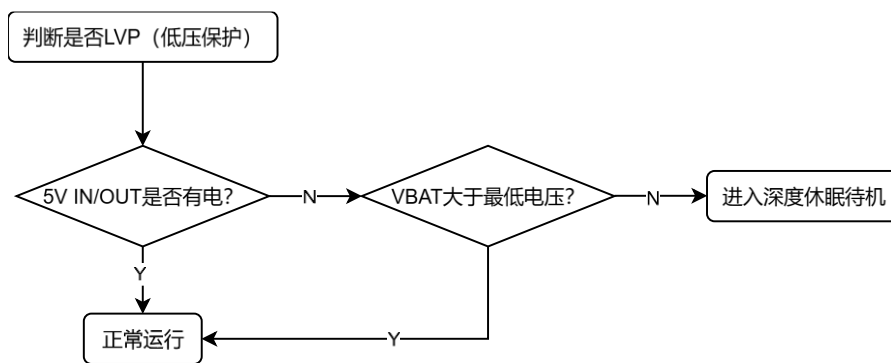


图5 低电压判断流程图

注：没有插入 VIN 的情况下，需要 BAT 电压大于 BATT_LVP 电压+100mV，才能退出低电压待机循环。

6 I²C 空闲休眠

配置 SLEEP[3:0] 可以设置空闲休眠的时间，0=不休眠。

完整休眠操作如下：

- 开启 PWR_BTN 按键外部中断
- 开启 5VIN 外部中断
- 开启 5VINOUT 外部中断
- 开启 SDA 外部中断
- 配置定时器为每 100ms 唤醒一次（此时间间隔不可更改）
- 复位 ADC 外设
- I²C 空闲休眠标志位置 1

注：PWM 功能使能时，I²C 空闲休眠失效；进入下载模式时，I²C 空闲休眠失效。

7 中断唤醒

(1) SDA 唤醒

休眠状态下，若 SDA 上有 I²C 通信信号，M5PM1 会先关闭 SDA 和 PWR_BTN 外部中断，再初始化 I²C 通信配置。

然后，会有一段 300ms 的非阻塞延时，在这期间如果 PWR_BTN 被按下，则认为是 PWR_BTN 唤醒，否则将等待紧接着的下一 I²C 通信信号并判断 I²C 地址是否匹配，若匹配则唤醒成功，否则继续休眠。



(2) PWR_BTN 唤醒

休眠状态下，若 PWR_BTN 被按下，则唤醒成功，会关闭 SDA 和 PWR_BTN 外部中断并重新初始化 I²C 通信配置。

(3) 5VIN 唤醒（同理（2））

(4) 5VINOUT 唤醒（同理（2））

注：上述唤醒基于定时器为每 100ms 唤醒一次，是固件固化流程，不可更改。

8 IRQ 处理

(1) 当 5 个 GPIO 中有配置 IRQ 时，配置为 IRQ 的 IO 会在 IRQ Status 1 (0x40)、IRQ Status 2 (0x41)或 IRQ Status 3 (0x42)非 0 时被拉低，当 IRQ Status 1 (0x40)、IRQ Status 2 (0x41)和 IRQ Status 3 (0x42)都清零时才被释放拉高。

(2) 当 5 个 GPIO 中有配置 IRQ 时，非 IRQ 的 IO(GPIO_FUNC 设置为 00, GPIO_MODE 设置为 0)会被扫描,被扫描的 IO 电平发生变化时，IRQ Status 1 (0x40)对应的位会置 1

(3) 当 5 个 GPIO 中有配置 IRQ 时，会扫描是否发生电源事件,当电源事件发生时，IRQ Status 2 (0x41)对应的位会置 1
电源事件如下：

- Battery Remove: 电池移除，BAT 电压 $\geq 2400\text{mV}$ \rightarrow BAT 电压 $< 2400\text{mV}$
- Battery Add: 电池插入，BAT 电压 $\leq 2400\text{mV}$ \rightarrow BAT 电压 $> 2400\text{mV}$
- 5VINOUT Remove: 5VIN/OUT 移除，5VINOUT 电压 $\geq 2400\text{mV}$ \rightarrow 5VINOUT 电压 $< 2400\text{mV}$
- 5VINOUT Add: 5VIN/OUT 插入，5VINOUT 电压 $\leq 2400\text{mV}$ \rightarrow 5VINOUT 电压 $> 2400\text{mV}$
- 5VIN Remove: 5VIN 移除，5VIN 电压 $\geq 2400\text{mV}$ \rightarrow 5VIN 电压 $< 2400\text{mV}$
- 5VIN Add: 5VIN 插入，5VIN 电压 $\leq 2400\text{mV}$ \rightarrow 5VIN 电压 $> 2400\text{mV}$

(4) 复位或进入下载模式后，GPIO 会被复位，此时没有 GPIO 被设置为 IRQ 引脚，IRQ Status 1 (0x40)、IRQ Status 2 (0x41)和 IRQ Status 3 (0x42)会被清零。如果你需要使用 IRQ Status 3 (0x42) Wakeup IRQ，请不要清除 WAKE_SRC (0x04)寄存器，直到再次把 GPIO 配置为 IRQ 引脚，否则无法触发 IRQ Status 3 (0x42) Wakeup IRQ。



六、使用案例

1 GPIO 唤醒

(1) 上升沿唤醒控制方式

```
pm1.gpioSetMode(M5PM1_GPIO_NUM_0, M5PM1_GPIO_MODE_INPUT); //设置 GPIO0 为输入模式
pm1.gpioSetPull(M5PM1_GPIO_NUM_0, M5PM1_GPIO_PULL_DOWN); //设置 GPIO0 下拉 (如果有外部下拉可以忽略)
pm1.gpioSetWakeEnable(M5PM1_GPIO_NUM_0, true); //使能 GPIO0 唤醒
pm1.gpioSetWakeEdge(M5PM1_GPIO_NUM_0, M5PM1_GPIO_WAKE_RISING); //设置 GPIO0 上升沿唤醒
```

(2) 下降沿唤醒控制方式

```
pm1.gpioSetMode(M5PM1_GPIO_NUM_0, M5PM1_GPIO_MODE_INPUT); //设置 GPIO0 为输入模式
pm1.gpioSetPull(M5PM1_GPIO_NUM_0, M5PM1_GPIO_PULL_UP); //设置 GPIO0 上拉 (如果有外部上拉可以忽略)
pm1.gpioSetWakeEnable(M5PM1_GPIO_NUM_0, true); //使能 GPIO0 唤醒
pm1.gpioSetWakeEdge(M5PM1_GPIO_NUM_0, M5PM1_GPIO_WAKE_FALLING); //设置 GPIO0 下降沿唤醒
```

(3) 注意事项

GPIO0、GPIO2、GPIO3、GPIO4 为支持唤醒引脚，但是 GPIO0 与 GPIO2、GPIO3 与 GPIO4 互斥不能同时使用。

2 RGB

(1) RGB 控制方式

```
pm1.gpioSetFunc(M5PM1_GPIO_NUM_0, M5PM1_GPIO_FUNC_OTHER); //设置 GPIO0 为复用功能
pm1.gpioSetDrive(M5PM1_GPIO_NUM_0, M5PM1_GPIO_DRIVE_PUSHPULL); //设置 GPIO0 为推挽输出模式

m5pm1_rgb_t rgb_red = {255, 0, 0};
m5pm1_rgb_t rgb_green = {0, 255, 0};
m5pm1_rgb_t rgb_blue = {0, 0, 255};
m5pm1_rgb_t rgb_array[3] = { rgb_red, rgb_green, rgb_blue };

pm1.setLeds(&rgb_red, 1, 3, true); //设置 3 个 RGB LED 为红色并刷新
// 适当延时
pm1.setLeds(&rgb_green, 1, 3, true); //设置 3 个 RGB LED 为红色并刷新
// 适当延时
pm1.setLeds(&rgb_blue, 1, 3, true); //设置 3 个 RGB LED 为红色并刷新
// 适当延时
pm1.setLeds(rgb_array, 3, 1, true); //设置 3 个 RGB LED 依次为红色、绿色、蓝色并刷新
// 适当延时
```

(2) 注意事项

- 1) 设置 GPIO0 为复用功能，系统会切换时钟到 24MH 为 RGB 时序刷新做准备，切换时钟会重置 I²C，需要等待一段时间再进行 I²C 通讯，否则通讯会失败。
- 2) 输出方式只以 GPIO_DRV (0x13) 为准，如果没有外部上拉需要设置 GPIO 为推挽才能输出正常的 RGB 时序。
- 3) 输出 RGB 期间中断会关闭，32 个灯刷新需要 7ms 左右，此时不能进行 I²C 通讯，若刷新 RGB 时序，需要等待一段时间再进行通讯。
- 4) 仅 GPIO0 支持 RGB 功能。



3 ADC

(1) ADC 使用示例

```
pm1.gpioSetFunc(M5PM1_GPIO_NUM_1, M5PM1_GPIO_FUNC_OTHER); //设置 GPIO1 为复用功能
pm1.gpioSetFunc(M5PM1_GPIO_NUM_2, M5PM1_GPIO_FUNC_OTHER); //设置 GPIO2 为复用功能
uint16_t gpio1_adc_value = 0;
uint16_t gpio2_adc_value = 0;
uint16_t temp_value = 0;
uint16_t vref_value = 0;
pm1.analogRead(M5PM1_ADC_CH_1, &gpio1_adc_value); //读取 GPIO1 的 ADC 值
pm1.analogRead(M5PM1_ADC_CH_2, &gpio2_adc_value); //读取 GPIO2 的 ADC 值
pm1.readTemperature(M5PM1_ADC_CH_TEMP, &temp_value); //读取 MCU 内部温度值
pm1.readVref(&vref_value); //读取参考电压以校准 ADC 值
uint16_t gpio1_volt = (gpio1_adc_value * vref_value) / 4096; //计算 GPIO1 的实际电压值
uint16_t gpio2_volt = (gpio2_adc_value * vref_value) / 4096; //计算 GPIO2 的实际电压值
```

(2) 注意事项

- 1) ADC 通道输入电压不允许超出 3.3V 否则 ADC 的模块会出错。
- 2) 实际电压通过参考电压 VREF 进行计算。

4 PWM

(1) PWM 时示例

```
pm1.gpioSetFunc(M5PM1_GPIO_NUM_3, M5PM1_GPIO_FUNC_OTHER); //设置 GPIO3 为复用功能
pm1.gpioSetFunc(M5PM1_GPIO_NUM_4, M5PM1_GPIO_FUNC_OTHER); //设置 GPIO4 为复用功能
pm1.gpioSetDrive(M5PM1_GPIO_NUM_3, M5PM1_GPIO_DRIVE_PUSH_PULL); //设置 GPIO3 为推挽输出模式
pm1.gpioSetDrive(M5PM1_GPIO_NUM_4, M5PM1_GPIO_DRIVE_PUSH_PULL); //设置 GPIO4 为推挽输出模式
pm1.setPwmFrequency(20000); //设置 PWM 频率为 20MHz
pm1.setPwmDuty(M5PM1_PWM_CH_0, 50, false, true); //设置 GPIO3 的 50% 占空比 PWM 波
pm1.setPwmDuty(M5PM1_PWM_CH_1, 50, false, true); //设置 GPIO4 的 50% 占空比 PWM 波
```

(2) 注意事项

- 1) 两个 PWM 通道是一个定时器控制，所以两个通道的控制频率相同。
- 2) 占空比范围：0-100%。
- 3) 如果没有外部上拉需要配置 GPIO 为推挽才能正常输出 PWM。

5 TIM

(1) 系统重启

```
pm1.timerSet(10, M5PM1_TIM_ACTION_REBOOT); // 10s 后重启
```

(2) 系统上电

```
pm1.timerSet(10, M5PM1_TIM_ACTION_POWERON); // 10s 后开机，如果系统关机定时到可自动开机
```

(3) 系统关机

```
pm1.timerSet(10, M5PM1_TIM_ACTION_POWEROFF); // 10s 后关机
```



附页

固件变更记录		
版本	日期	变更描述
HW:2 / SW:1	2025-06-30	初始版本
HW:3 / SW:2	2025-07-23	<ol style="list-style-type: none"> 1. 根据硬件修订, 修改 I2C 端口和 Bootout 端口, 修改 ADC 分压系数, 修改 CHG_EN 为推挽并修改控制逻辑为高开启低关闭。 2. GPIO 默认的输出类型改为开漏。 3. 系统定时器 TIM_CNT 改为 31 位并调整寄存器分布。 4. 添加 5VINOUT 唤醒 (仅当 5V 升压关闭时) 。
HW:4 / SW:3	2025-09-01	<ol style="list-style-type: none"> 1. 32-bit 定时器最大秒数限制为 214748364 秒。 2. 添加 GPIO_Power_Hold 寄存器 (0x33), 相应的 bit 设置为 1, 对应的 GPIO 或 LDO 状态在关机后会保留。 3. 关机时, GPIO 会恢复为默认的输入无上下拉 (如果 GPIO_Power_Hold 寄存器相应的 bit 设置为 1, 对应的 gpio 状态在关机后会保留), 电源状态会恢复为默认状态 (充电控制不会影响, 如果 GPIO_Power_Hold 寄存器相应的 bit 设置为 1, 对应的 Ido 状态在关机后会保留) 。 4. 进入下载模式时, GPIO_Power_Hold 寄存器会恢复为 0, I2C 看门狗会停止执行, I2C 空闲休眠会停止执行, 用户定时器会停止执行, GPIO 会恢复为默认的输入无上下拉, 电源状态会恢复为默认状态。 5. 进入复位模式时 (包括按键复位、命令复位、I2C 看门狗复位和用户定时器复位), GPIO_Power_Hold 寄存器会恢复为 0, GPIO 会恢复为默认的输入无上下拉, 电源状态会恢复为默认状态。 6. 添加 LED 提示: <ol style="list-style-type: none"> a. 复位时, LED 会闪一下; b. 下载模式, LED 灯会以 500ms 间隔闪烁; c. 按键复位屏蔽, 且有 GPIO 使能了 IRQ 功能, LED 灯会以 200ms 间隔闪烁; d. 按键双击关机屏蔽, 且有 GPIO 使能了 IRQ 功能, LED 灯会以 100ms 间隔闪烁。 7. 添加 IRQ Status 3 (0x23), bit0 是 rst 中断, 按键复位屏蔽后, 单击 PWR_BTN 会触发该中断。bit1 是 wakeup 中断, 开机或复位时 WAKE_SRC (0x2F) 相应的位会置 1, bit1 也会置 1 (注意 WAKE_SRC (0x2F) 与 IRQ Status 3 的 bit1 是相关的, 即如果 WAKE_SRC (0x2F) 不清零, IRQ Status 3 bit1 始终为 1)。bit2 是 btn_dl_click 中断, 按键双击关机屏蔽后, 双击 PWR_BTN 会触发该中断。 8. 添加 IRQ Mask, 可以屏蔽按需屏蔽对应的中断。 9. 充电默认打开。 10. BTN_CFG 添加 bit0:SINGLE_RESET_DIS, SINGLE_RESET_DIS = 1 禁止单击复位; SINGLE_RESET_DIS = 0 启用单击复位。 11. 添加寄存器 BTN_CFG_2(0x31), bit0:DOUBLE_POWEROFF_DIS= 1 禁止双击关机; DOUBLE_POWEROFF_DIS= 0 启用双击复位。
HW:5/ SW:4	2025-09-17	<ol style="list-style-type: none"> 1. 增加 AW8737A 脉宽调制功能。 2. 更新固件版本到 V4。
HW:5 / SW:5	2025-11-04	<ol style="list-style-type: none"> 1. 添加 AW8737A 脉宽调制输出功能。 2. 添加 DCDC_5V 保持功能。 3. 看门狗修改为默认关闭状态。
HW:5 / SW:6	2025-12-13	<ol style="list-style-type: none"> 1. 更新寄存器映射 (Register Map) 。 2. 新增 BTN_Status (0x48) 寄存器。 3. 移除 UID 寄存器, 新增 Device_ID 与 Device_Model。



固件变更记录		
版本	日期	变更描述
HW:5 / SW:S	2026-01-06	1. 修改定时器功能，定时器配置生效一次就自动清除，关机之后再次开启定时器配置清除。 2. 修改 USB、5VOUT 的无效阈值调整为 4V。

文档变更记录		
版本	日期	变更描述
1.0	2025-06-30	初始版本
1.1	2025-07-23	硬件修订: 3, 固件主版本: 2
1.2	2025-08-04	修订部分描述错误
1.3	2025-09-01	硬件修订: 4, 固件主版本: 3
1.4	2025-09-10	对关键寄存器详解进行了补充
1.5	2025-09-17	硬件修订: 5, 固件主版本: 4
1.6	2025-11-04	硬件修订: 5, 固件主版本: 5
1.7	2025-12-13	硬件修订: 5, 固件主版本: 6, 增加使用案例
1.8	2026-01-06	硬件修订: 5, 固件主版本: S
1.9	2026-01-22	修订部分描述错误