

前言

PY32F403 系列微控制器采用高性能的 32 位 ARM® Cortex®-M4 内核，支持 FPU 和 DSP 指令。嵌入高达 384 Kbytes Flash 和 64 Kbytes SRAM 存储器，最高工作频率 144 MHz。包含多种不同封装类型多款产品。

本应用笔记将帮助用户了解 PY32F403 各个模块应用的注意事项，并快速着手开发。

表 1. 适用产品

类型	产品系列
微型控制器系列	PY32F403

目录

1	PLL 使用注意事项.....	3
2	ADC 上电校准.....	3
3	ADC 使用注意事项.....	4
4	I2C 配置注意事项.....	4
5	FLASH 使用注意事项.....	5
6	Option 使用注意事项.....	5
7	IWDG 应用注意事项.....	7
8	DMA 使用注意事项.....	8
9	SPI 使用注意事项.....	8
10	I2S 功能注意事项.....	8
11	DBG_MCU 使用注意事项.....	8
12	USB 使用注意事项.....	8
13	TIMER 使用注意事项.....	8
14	ESMC 使用注意事项.....	9
15	PVD 使用注意事项.....	9
16	GPIO 使用注意事项.....	9
17	Standby 模式注意事项 (如若使用该模块, 请与原厂联系).....	9
18	版本历史.....	10

1 PLL 使用注意事项

- PLL 倍频后时钟需大于等于 48 M，时钟源可以是 HSI 或者 HSE。

2 ADC 上电校准

2.1 注意事项

- 当 ADC 的工作条件发生改变时 (V_{CC} 改变是 ADC offset 偏移的主要因素，温度改变次之)，推荐进行再次校准操作。
- 第一次使用 ADC 模块前，必须增加软件校准流程。

2.2 操作流程

- 复位 ADC 模块；
- ADC 模块初始化；
- 校准 ADC。

2.3 代码示例

```
ADC_HandleTypeDef      AdcHandle;
__HAL_RCC_ADC1_CLK_ENABLE();
__HAL_RCC_ADC1_FORCE_RESET();
__HAL_RCC_ADC1_RELEASE_RESET();
AdcHandle.Instance = ADC1;
/* 分辨率 12 位 */
AdcHandle.Init.Resolution      = ADC_RESOLUTION_12B;
/* 对齐方式右对齐 */
AdcHandle.Init.DataAlign      = ADC_DATAALIGN_RIGHT;
/* 扫描方式关闭 */
AdcHandle.Init.ScanConvMode   = ADC_SCAN_DISABLE;
/* 单次模式 */
AdcHandle.Init.ContinuousConvMode = DISABLE;
/* 转换通道数 1 */
AdcHandle.Init.NbrOfConversion = 1;
/* 间断模式不使能 */
AdcHandle.Init.DiscontinuousConvMode = DISABLE;
/* 间断模式短序列长度为 1 */
AdcHandle.Init.NbrOfDiscConversion = 1;
/* 软件触发 */
AdcHandle.Init.ExternalTrigConv = ADC_SOFTWARE_START;
if (HAL_ADC_Init(&AdcHandle) != HAL_OK)
{
```

```

APP_ErrorHandler();
}
if (HAL_ADCEx_Calibration_Start(&AdcHandle) != HAL_OK)
{
APP_ErrorHandler();
}

```

3 ADC 使用注意事项

3.1 ADC 硬件使用注意事项

- ADC 所有通道的电压不能高于 $V_{CC}+0.3V$ (即使 ADC 通道未配置为 AD 功能), 否则 ADC 采样异常。

3.2 ADC 软件使用注意事项

- ADC->SR.Start=1 之前需确保 ADC->SR.EOC=0, 否则 ADC 使能失败;
- CH0 和 CH18、CH19 必须设置为相同的采样周期, 否则会导致 CH18/CH19 采样错误;
- ADC 配置为非连续模式, 注入序列的情况下, 只采样一个通道(JDR1), 多次触发采样, JDR2 会有数据;
- ADC 配置为扫描模式, 注入序列的情况下, 只采样一个通道(JDR1), 多次触发采样, JDR2 会有数据。

3.3 Vreferint 1.2 V

- 在采样 Vreferint 1.2 V 的时候, 通过 ADC 采样时间转换公式算出来的结果至少需要 20 us, 方法如下:
 - a) 降低分辨率;
 - b) 降低 ADC 的时钟频率;
 - c) 提高 ADC 采样周期。

总转换时间计算如下:

$$t_{CONV} = \text{采样时间} + (\text{转换分辨率} + 0.5) \times \text{ADC 时钟周期}$$

例如:

当 $\text{ADC_CLK} = 12\text{MHz}$, 分辨率为12位, 且采样时间为 239.5个ADC 时钟周期:

$$t_{CONV} = (239.5 + 12.5) \times \text{ADC 时钟周期} = 252 \times \text{ADC 时钟周期} = 21 \text{ us}$$

4 I2C 配置注意事项

4.1 注意事项

- I2C 在初始化引脚时 IO 口接 GND, BUSY 状态位受 IO 口影响置 1, 导致无法正常使用。软件可在 IO 口初始化后复位一次 I2C 模块。

4.2 操作流程

- 初始化 I2C 对应 IO 口做 SCL、SDA;
- 配置 RCC_APBSTR1 寄存器 I2CRST=1, 再配置 I2CRST=0 复位 I2C 模块;
- 初始化 I2C 模块。

4.3 代码示例

以主机发送为例:

```
GPIO_InitTypeDef GPIO_InitStructure = {0};

__HAL_RCC_I2C1_CLK_ENABLE();
__HAL_RCC_GPIOB_CLK_ENABLE();

GPIO_InitStructure.Pin          = GPIO_PIN_6|GPIO_PIN_7;
GPIO_InitStructure.Mode        = GPIO_MODE_AF_OD;
GPIO_InitStructure.Pull        = GPIO_PULLUP;
GPIO_InitStructure.Speed       = GPIO_SPEED_FREQ_HIGH;
GPIO_InitStructure.Alternate   = GPIO_AF1_I2C1;
HAL_GPIO_Init(GPIOF, &GPIO_InitStructure);

__HAL_RCC_I2C1_FORCE_RESET();
__HAL_RCC_I2C1_RELEASE_RESET();

// 初始化 I2C 模块
.....
```

5 FLASH 使用注意事项

- 系统时钟为 HSI 8M 时, 才可访问并操作 information 区和 Option。
- 高速情况下, 无法读取 information 区域数据, 若需要读取该区域值须在 PLL 倍频前读取。

6 Option 使用注意事项

- 量产时, Option 操作必须在烧写器选项字节中配置, 并把程序中操作 Option 的函数屏蔽;
- 建议客户程序使能写保护, 写保护在 Option 中设置, 具体步骤如图 6-1、图 6-2 所示;

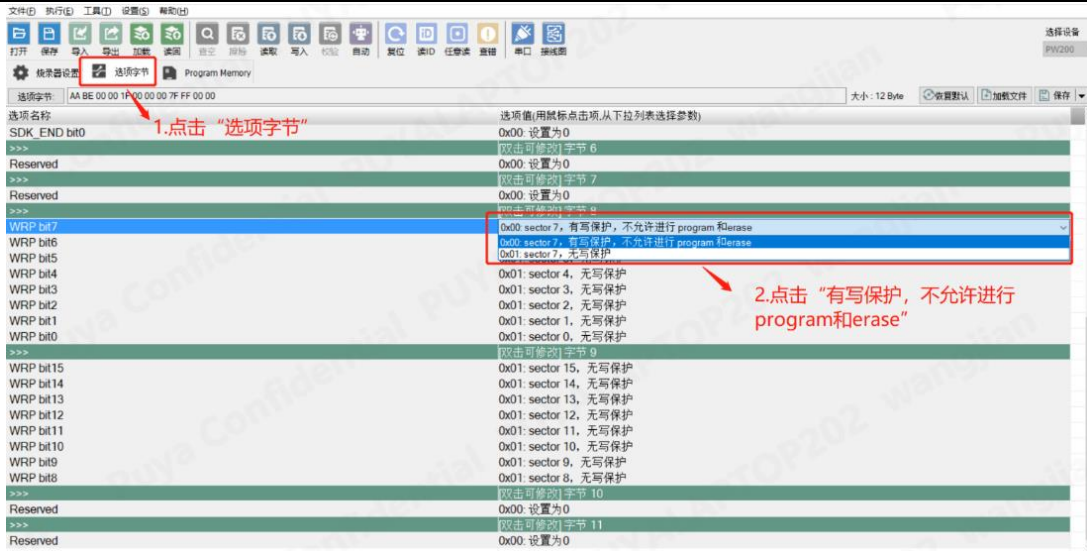


图 6-1 创芯工坊 Option 写保护



图 6-2 轩微操作 Option 写保护

- 烧写器配置 Option 时，需勾选智能复位功能/编程后重启芯片(烧写器均有类似选项需要勾选)，具体步骤如图 6-3、图 6-4 所示。



图 6-3 创芯工坊操作勾选“编程后重启芯片”



图 6-4 轩微操作“智能复位”

7 IWDG 应用注意事项

- IWDG 开启后，只能通过关闭 LSI 来关掉 IWDG。

8 DMA 使用注意事项

- 使用 DMA 传输搬运数据时，需等 DMA 传输完成后才能关闭 DMA，否则需强制 Reset 并重新初始化 DMA 才能正常使用。

9 SPI 使用注意事项

- SCK 频率最大为 PCLK/4;
- SPI 从机发送时，SPI 的 DR 寄存器写入一个值后，未发送出去之前，再次写的值不能覆盖前面写的值；如果想覆盖，需要先复位整个 SPI 模块(通过 RCC_APBSTR2 中的 SPI1RST 以及 RCC_APBSTR1 中的 SPI2RST/SPI3RST 来进行对应模块的复位)，重新写入 DR 值；
- SPI 的 CRC 功能无法使用；
- SPI 半双工主机接收模式，CPHA=0，CPOL=1，256 分频下，会多一个 CLK；
- SPI 发送数据时，必须把 SPI 的 DR 寄存器强制转换为 unit8_t 类型的变量，然后再往 DR 寄存器写数据；
- SPI 通信时，SPI->SR.BSY 位在最后一个时钟期间被清除，Polling 模式时，下一帧数据更新需确保上一帧数据传输完成。

10 I2S 功能注意事项

- 在 8 M 系统时钟下，Fs=32 KHz 和 22.05 KHz 无法使用。

11 DBG_MCU 使用注意事项

- DBG_STANDBY 置位，仿真全速运行时，SWD 会断开。

12 USB 使用注意事项

- USB 使用 DMA 通讯时，同时只能配置一个通道进行 DMA 通讯。
- USB 先使能端点中断，运行一段时间后，禁能端点中断，需要清除 FIFO。

13 TIMER 使用注意事项

- TIMER 中断函数中，清 CC 中断标志，必须等待 TIM_PSC*PCLK，否则会导致清中断标志失败。
- 六步 PWM 输出时，刹车功能无法使用。

14 ESMC 使用注意事项

- 连续读取使用 SS_CLR_RQ, IDLER 位无法置 1;
- DMA 传输时, 一次只能传一个字的数据;
- Buffer 寄存器只支持字读取;
- 实际读取数据长度是 128 bytes 的整数倍;
- 配置为 SPI 模式, 只支持模式 0;
- ESMC 时钟最高只能达到 24 M。

15 PVD 使用注意事项

- 使用 PVD 时, 必须使能 LSI。

16 GPIO 使用注意事项

- 当只有 Vbat 供电时, PC13 不能使用。

17 Standby 模式注意事项 (如若使用该模块, 请与原厂联系)

18 版本历史

版本	日期	更新记录
V1.0	2024.02.01	初版
V1.1	2024.08.07	新增第 4 章, 修改格式
V1.2	2024.10.08	合并第 3、4、5 章内容, 合并第 16、17 章内容, 增加 DMA、PVD、Standby 模块, 修改第 6、9、12 章内容
V1.3	2025.06.05	增加 GPIO 模块内容



Puya Semiconductor Co., Ltd.

声 明

普冉半导体(上海)股份有限公司(以下简称:“Puya”)保留更改、纠正、增强、修改 Puya 产品和/或本文档的权利,恕不另行通知。用户可在下单前获取产品的最新相关信息。

Puya 产品是依据订单时的销售条款和条件进行销售的。

用户对 Puya 产品的选择和使用承担全责,同时若用于其自己或指定第三方产品上的,Puya 不提供服务支持且不对此类产品承担任何责任。

Puya 在此不授予任何知识产权的明示或暗示方式许可。

Puya 产品的转售,若其条款与此处规定不一致,Puya 对此类产品的任何保修承诺无效。

任何带有 Puya 或 Puya 标识的图形或字样是普冉的商标。所有其他产品或服务名称均为其各自所有者的财产。

本文档中的信息取代并替换先前版本中的信息。

普冉半导体(上海)股份有限公司 - 保留所有权利