



应用笔记

PY32T020 的应用 注意事项

前言

PY32T020 系列微控制器采用高性能的 32 位 ARM® Cortex®-M0+内核，宽电压工作范围的 MCU。嵌入最大 32Kbytes Flash 和最大 4Kbytes SRAM 存储器，最高工作频率 48 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 PY32T020 系列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	PY32T020、PY32T020-B

目录

1	SPI 使用注意事项.....	3
2	PWR 使用注意事项.....	3
3	GPIO 配置注意事项	4
4	GPIO 倒灌电流使 MCU 工作.....	5
5	TK 使用注意事项.....	6
6	LED 恒流输出配置注意事项.....	6
7	I2C 从机使用注意事项.....	7
8	ADC 上电校准.....	7
9	ADC 使用注意事项.....	9
10	Flash 使用注意事项.....	10
11	RCC 使用注意事项.....	10
12	Option 操作	10
13	掉电保存.....	13
14	UID 读取.....	13
15	烧录编译.....	13
16	版本历史.....	14

1 SPI 使用注意事项

SPI 模式	收/发模式	SPI 最快速度
从机全双工	收	PCLK/16
从机全双工	发	PCLK/16
主机全双工	收	PCLK/2
主机全双工	发	PCLK/2

- SPI 按 8bit 发送数据时，须将 SPI 寄存器强转成 uint8_t 类型的变量((__IO uint8_t *)SPI->DR);
- 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_APBRSTR2 中的 SPI1RST 来进行对应模块的复位)，重新写入 DR 值；
- SPI 半双工主机接收模式，CPHA=0，CPOL=1，256 分频下，会多一个 CLK；
- 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。

2 PWR 使用注意事项

- MCU 进 Stop 模式时必须使能看门狗；
- 推荐使用硬件看门狗替代软件看门狗，提高程序可靠性；
- 一旦使能看门狗，软件无法关闭，在低功耗下，需使用 RTC 定时唤醒，对看门狗进行喂狗；
- 推荐客户在 Option 中使能看门狗并根据实际情况软件设置看门狗溢出时间；
- MCU 进 Stop 之前需关闭 systick 中断(HAL_SuspendTick());
- A 版本芯片推荐做法：进入 Stop 模式前关闭 BOR 功能。[\(B 版本已修复\)](#)
- A 版本芯片若必须开启 BOR 功能，需要在进入 Stop 模式前开启看门狗，在 Stop 模式下定时唤醒进行喂狗操作。[\(B 版本已修复\)](#)
- 可以通过如下方式唤醒 Stop 模式
 - RTC 闹钟中断
 - TK 达到触发阈值

- IWDG 复位
- GPIO
- COMPx (x为1~2)
- BOR

3 GPIO 配置注意事项

- 当 GPIO 用作触摸时，禁止同时配置为通用输出模式及复用功能模式；
- 初始化 GPIO 等其他的结构体都需要赋值为 0，避免初始值不固定；
- 所有 GPIO 不能有超过-0.3 V 的负压。
- GPIO 输入电压不能高于 VCC+0.3V。
- 所有 GPIO 默认上下拉电阻阻值为 40kΩ，可通过系统配置控制器(SYSCFG)中(Px_IORP) (x = A, B, F)配置为以下四种
 - 00: pull-up/pull-down is open
 - 01: pull-up/pull-down resistance 10KΩ
 - 10: pull-up/pull-down resistance 20KΩ
 - 11: pull-up/pull-down resistance 40KΩ
- 复位期间和复位后，复用功能未被激活，大多数 IO 被配置为模拟模式，如下两类 IO 默认被置于上拉或下拉模式。
 - Debug 引脚中SWCLK默认下拉，SWDIO默认上拉
 - NRST复位引脚默认被置于上拉模式

其中Debug引脚设置如下：

option[1:0]	PF3	PF4	PA13	PA14
0/0(default)	SWCLK	SWDIO	GPIO	GPIO
0/1	GPIO	GPIO	SWDIO	SWCLK
1/0	GPIO	SWDIO	GPIO	SWCLK
1/1	SWCLK	GPIO	SWDIO	GPIO

- NRST 复位时，必须确保 NRST 管脚的低电平有效复位脉冲宽度大于 300 μs。
- NRST 管脚通过烧写器修改 option 配置为 GPIO。
- 通过烧写器将 NRST 管脚配置为 GPIO (NRST 管脚生效) 的第一次上电需要保持管脚状态为高，

(NRST 已配置为 GPIO) 第二次开始上电可以引脚为低上电。

- NRST 管脚配置为 GPIO 不影响 SWD 仿真和烧录。
- NRST 管脚悬空的状态下，需要把 NRST 管脚复用为 GPIO，防止出现 IO 口出现低电平的情况导致芯片无法运行。
- MCU 的 VCC 引脚必须连接 VCC 电源进行工作。仅连接 TX、RX、GND 三线时，端口电压会反向供电，导致模块处于异常工作状态，影响其可靠性。正确的连接方式：请确保在连接串口线的同时，为 MCU 的 VCC 引脚提供符合规格的稳定电源。
- 矩阵 LED 扫描亮度不足
 - 矩阵LED扫描，LED点亮的占空比低，更改为分段共阴极方案
 - 换用B版本芯片
- LED 的 COM 脚优先选择 PA11~PA15，每个 IO 高达 80mA 灌电流。

4 GPIO 倒灌电流使 MCU 工作

4.1 注意事项

- VCC 未供电的情况下，IO 倒灌电流使 MCU 工作，可通过软件配置规避。

4.2 操作流程

- 硬件：对应 IO 口需串 100Ω~1KΩ 电阻；
- 上电初始化前需设置对应 IO 输出为开漏模式；
- 延迟 5ms；
- 程序正常初始化。

4.3 代码示例

```
int main(void)
{
    LL_IOP_GRP1_EnableClock(LL_IOP_GRP1_PERIPH_GPIOA);/*Enable the GPIOA clock*/
    /*Configure pin PA1 as open-drain output */
    LL_GPIO_SetPinMode(GPIOA, LL_GPIO_PIN_1, LL_GPIO_OUTPUT_OPENDRAIN);
    LL_mDelay(5);                                     /*Delay for 5ms*/
}
```

5 TK 使用注意事项

- 在电源干扰较大或需 EFT 认证的应用，建议使用 PF5 以外 IO 作为 TK。
- 在 PCB 设计时，开关电源储能电感应尽量远离弹簧按键摆放，间距至少 4cm 以上。
- 外接电容可选，芯片内部集成了 Cmod 电容，建议使用内部电容即可。
- 按规范需串联电阻，以降低射频干扰并提供 ESD 保护，阻值范围为：510Ω~10KΩ，推荐使用 4.3K。
- 触摸功能支持上位机调试，上位机工具可实现触摸数据实时更新、动态扫绘曲线、在线设置灵敏度等。
- 只有本公司专用调试器 PY-LINK 才支持触摸上位机调试，其他调试器如 U_LINK、J_LINK 仅可以进行下载、仿真程序。
- 触摸防水等级要求不同，其硬件设计也需相应调整。详情请参考“PY Touch PCB 设计指南 V1.3”第 5 章节“防水按键设计”。
- 隔空按键对于隔空距离、按键尺寸、触摸走线等都有要求，详情请参考“PY Touch PCB 设计指南 V1.3”第 6 章节“隔空按键设计”。
- 触摸检水的硬件设计可参考“PY Touch PCB 设计指南 V1.3”第 7 章节“触摸检水设计”。
- 内部通道用于检测电源的波动，通常在电压可能会异常波动的场景下使能，如对讲机干扰等。
- 在触摸检水应用中，由于不同水质导电率不一样，实际检测到的水位会有一定的偏差，在应用中需要设置合适的阈值以兼容多种水质。
- 触摸按键引脚严禁外部强制拉高或拉低电平，否则可能因过流导致内部元件损坏，造成触摸功能异常。
- 若下载引脚（如仿真器接口）用作触摸功能，程序下载后请立即断开仿真器，避免其持续驱动引脚引发风险。
- 金属外壳可以作为触摸按键的触点，但注意如果产品为电池供电，PCB 不能被金属外壳全包裹，否则触摸灵敏度会很低。
- 下载引脚(SWD/SWC)复用为 TK/IO 功能后，可使用触摸调试上位机工具 PY-TOUCH LINK 或者烧写器进行下载。
- 不建议在触摸按键引脚连接 TVS 管、二极管、三极管等元器件。

6 LED 恒流输出配置注意事项

- 使用 PB2 作为恒流输出时，必须首先使能 PA0 的恒流功能。
- 使用 PB3 作为恒流输出时，必须首先使能 PA1 的恒流功能。

7 I2C 从机使用注意事项

7.1 操作流程：

每当从机收到 STOP 信号后进行如下操作：

- 关闭 I2C 模块使能；
- 打开 I2C 模块使能；
- 应答使能。

7.2 代码示例：

```
LL_I2C_Disable(I2C1);
LL_I2C_Enable(I2C1);
LL_I2C_AcknowledgeNextData(I2C1, LL_I2C_ACK);
```

8 ADC 上电校准

8.1 注意事项

- 当 ADC 的工作条件发生改变时 (VCC 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。

8.2 操作流程

- 使能 ADC 时钟, ADCEN=1;
- 初始化 ADC;
- ADC 校准。

8.3 代码示例

```
static void APP_AdcConfig()
{
    LL_APB1_GRP2_EnableClock(LL_APB1_GRP2_PERIPH_ADC1);           //使能 ADC1 时钟

    if (LL_ADC_IsEnabled(ADC1) == 0)
    {
        LL_ADC_StartCalibration(ADC1);                                //使能校准
#if (USE_TIMEOUT == 1)
        Timeout = ADC_CALIBRATION_TIMEOUT_MS;
#endif
        while (LL_ADC_IsCalibrationOnGoing(ADC1) != 0)
        {
            #if (USE_TIMEOUT == 1)                                       //检测校准是否超时
                if (LL_SYSTICK_IsActiveCounterFlag())
                {
                    if (Timeout-- == 0)
                    {
                        }
                }
            #endif
            }
        LL_mDelay(1);
    }
}
```

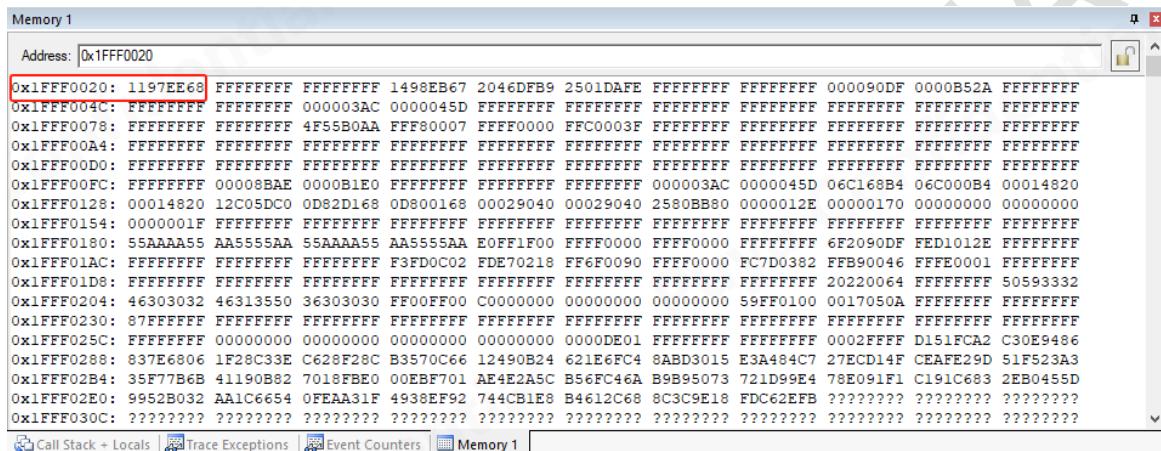
9 ADC 使用注意事项

9.1 ADC 软件配置

- GPIO 直接驱动大功耗器件会影响 ADC 采样结果(例如数码管显示, 建议数码管显示的时候不采样 ADC, 或者在数码管的各个数据线上面串入 10-100 Ω 电阻, 可根据实际情况进行调整)。

9.2 Vreferint 1.2V

- 芯片的 Vreferint 1.2 V 实测值放置在 FLASH 中的 information 区域(0x1FFF0020)(高 16 位是实际值, 低 16 位是反码) 读取 Vreferint 1.2 V 的程序见附录 1。



9.3 Vreferint 1.2V 实测值读取代码示例

```

#define HAL_VREF_INT          (*(uint8_t *)0x1fff0023))
#define HAL_VREF_DEC          (*(uint8_t *)0x1fff0022))

#define vref_int      (*(uint8_t *)HAL_VREF_INT)           //存放参考电压整数部分
#define vref_dec      (*(uint8_t *)HAL_VREF_DEC)           //存放参考电压小数部分
float vref;                                //参考电压值

static uint8_t Bcd2ToByte(uint8_t Value)
{
    uint32_t tmp = 0U;
    tmp = ((uint8_t)(Value & (uint8_t)0xF0) >> (uint8_t)0x4) * 10U;
    return (tmp + (Value & (uint8_t)0x0F));
}

float read_1_2V(void)
{
    uint8_t data_vref_int,data_vref_dec;
    data_vref_int = Bcd2ToByte(HAL_VREF_INT);
    data_vref_dec = Bcd2ToByte(HAL_VREF_DEC);

    //初始化所有外设, flash 接口, systick
    vref = data_vref_int/10;      //计算参考电压
    vref = vref + ((data_vref_int%10)*0.1 + data_vref_dec*0.001);
    return vref;
}

```

10 Flash 使用注意事项

- FLASH 擦写最小操作单位为 Page, 一个 Page 是 128 字节, 起始地址只能 Page 对齐(如起始地址 0x08000000, 0x08000080 等)。
- 每次 Page 写之前必须先 Page 擦。

11 RCC 使用注意事项

- APB 分频系数大时, 执行 (APB 总线上的) 模块复位后, 不能立马对模块寄存器进行读写操作, 如需操作, 则需要增加__NOP()空指令, 空指令数量要大于 APB 分频数, 例如 APB 8 分频, 建议增加 10 个以上的__NOP 指令。
- 配置 LSE 时, 先配置 LSE_DRIVER, 再使能 LSE(LSEON=1)。

12 Option 操作

- 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 12-1、图 12-2 所示;

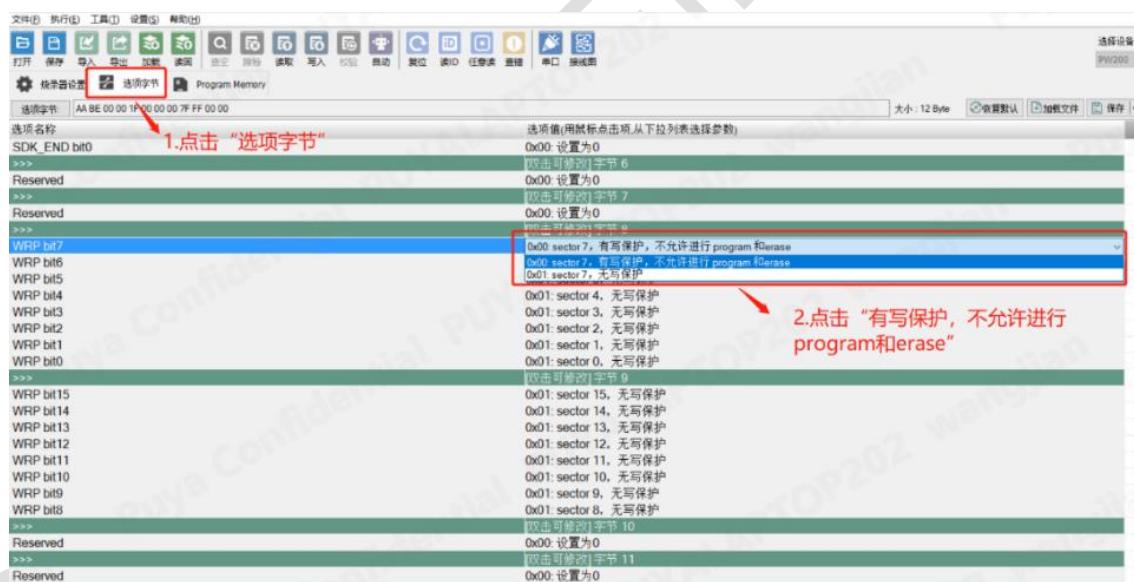


图 12-1 创芯工坊操作 Option 写保护



图 12-2 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 12-3、图 12-4 所示。



图 12-3 创芯工坊操作勾选“编程后重启芯片”



图 12-4 轩微操作“智能复位”

13 掉电保存

- 由于欠压检测只有复位功能，而没有中断模式，只能使用 ADC 定时检测芯片的供电电压，当检测到掉电动作时将用户数据存入 USER OTP memory。

14 UID 读取

14.1 注意事项

- 用户可通过 MCU 内置 16 字节 UID(地址范围 0x1FFF0000~0x1FFF000F)，结合自定义算法生成产品唯一标识，适用于高安全性滚码系统、设备认证及防伪验证等应用场景。

14.2 代码示例

```
/**  
 * @brief 读取 MCU 完整 16 字节 UID  
 * @param uid_buffer: 存储 UID 的缓冲区(必须至少 16 字节)  
 * @retval 无  
 */  
void Read_Full_UID(uint8_t *uid_buffer)  
{  
    volatile uint8_t *uid_base = (volatile uint8_t *)0x1FFF0000;  
  
    for (uint8_t i = 0; i < 16; i++) {  
        uid_buffer[i] = uid_base[i];  
    }  
}
```

15 烧录编译

- 当在 Keil 中编译外设例程时出现"找不到对应的驱动文件"错误，这通常是由于文件路径过长导致的。可通过优化项目存储位置改善。操作方法如下：
 - 将整个开发包移至浅层目录，例如 C:\Projects\ 或 D:\Keil_Projects\
 - 避免使用桌面、文档等默认用户文件夹（路径通常包含用户名，增加长度）
 - 路径中避免使用特殊字符

16 版本历史

版本	日期	更新记录
V1.0	2024.03.20	初版
V1.1	2024.05.11	添加 GPIO 配置注意事项
V1.2	2025.04.17	增加 Stop、TK、LED、I2C 章内容
V1.3	2025.06.04	添加 NRST 复位注意事项
V1.4	2025.06.23	增加 ADC、IWDG、Flash、RCC、SPI、GPIO 章节内容
V1.5	2025.07.10	1、合并 SPI 章内容 2、添加 Stop 章节内容
V1.6	2025.09.12	修改文档格式
V1.7	2025.12.22	1、增加 GPIO 配置注意事项 2、增加 Option 操作章节 3、合并进入 Stop 模式章节 到 PWR 章节 4、合并 LSI 章节到 RCC 章节 5、增加 TK、GPIO、掉电保存、烧录编译、UID 读取章节注意事项



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