

前言

PY32FL020 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 24Kbytes Flash 和 3Kbytes SRAM 存储器, 最高工作频率 48MHz。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 PY32FL020 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	PY32FL020

目录

1	ADC 上电校准	3
1.1	注意事项	3
1.2	操作流程	3
1.3	代码示例	3
2	ADC 硬件设计注意事项	4
3	ADC 硬件设计注意事项	4
4	ADC VerfBuffer 注意事项	4
5	SPI 最快传输速度	4
6	TIMER 使用 CC 中断注意事项	4
7	LPTIM 连续模式注意事项	4
8	LPTIM 单次模式注意事项	4
9	COMP 硬件设计	5
10	IO 倒灌电流使 MCU 工作	5
10.1	注意事项	5
10.2	操作流程	5
10.3	代码示例	5
11	IWDG 不支持冻结功能	5
11.1	注意事项	5
11.2	操作流程	5
12	版本历史	6

1 ADC 上电校准

1.1 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。

1.2 操作流程

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

1.3 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
#ifdef 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)
                    {
                        }
                    }
            }
        }
    }
    LL_mDelay(1);
}
```

2 ADC 硬件设计注意事项

ADC 不支持采样 VCC/3(内部通道 10)

3 ADC 硬件设计注意事项

ADC 通道电压不能高于 VCC(即使 ADC 通道未配置为 AD 功能),否则 ADC 采样不准

4 ADC VerfBuffer 注意事项

若要使能内部 1.5V VerfBuffer, 软件上必须同时使能 1.2V VREFEN(ADC->CCR.VREFEN=1)

5 SPI 最快传输速度

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

6 TIMER 使用 CC 中断注意事项

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

7 LPTIM 连续模式注意事项

LPTIM 连续模式每次进入 STOP 前必须清 ARRMCF 并需等待 1 个 LSI 时钟周期*PSC 系数 (约需 40us*PSC, 包含程序执行时间)

8 LPTIM 单次模式注意事项

LPTIM 单次模式每次进入 STOP 前必须清 ARRMCF 并需等待 3 个 LSI 时钟周期 (约需 120us, 包含程序执行时间)

9 COMP 硬件设计

当比较器的 VINM 输入信号为内部的模拟电压源时，外部输入通道 VINP 需要加一个电容(1nF)到地

10 IO 倒灌电流使 MCU 工作

10.1 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

10.2 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

10.3 代码示例

```
int main(void)
{
    LL_IOP_GRP1_EnableClock(LL_IOP_GRP1_PERIPH_GPIOA); /*使能 GPIOA 时钟*/
    LL_GPIO_SetPinMode(GPIOA, LL_GPIO_PIN_1, LL_GPIO_OUTPUT_OPENDRAIN);
                                                    /*将 PA1 引脚配置为开漏输出*/
    LL_mDelay(5); /*延迟 5ms*/
}
```

11 IWDG 不支持冻结功能

11.1 注意事项

- IWDG 冻结功能无效，使能以后无法关闭。IWDG 和 STOP 模式同时使用时，需要使用 LPTIM 定时唤醒喂狗。下表是定时唤醒对功耗的影响：

注意：以下运行功耗是在系统时钟为 24M 情况下计算

低功耗运行时间(ms)	stop 模式电流(uA)	唤醒运行时间(ms)	运行模式电流(uA)	平均功耗(uA)
500	1.7	1	1100	3.892215569
1000		1		2.797202797
2000		1		2.248875562
3000		1		2.065978007

11.2 操作流程

- 将所有的唤醒源配置为事件唤醒；
- 进入 STOP 前关闭所有中断并清除相应的标志；
- 开启 LPTIM，使能定时唤醒功能，并确保 LPTIM 定时时间小于 IWDG 溢出时间；(参考第 7、8 章进行配置)

12 版本历史

版本	日期	更新记录
V1.0	2023.6.15	初版
V1.1	2023.7.15	更新 CMP 硬件设计
V1.2	2023.11.10	第 11 章增加 LPTIM 唤醒后的平均功耗 第 7、8 章增加 LPTIM 使用事件唤醒不需要等待时间
V1.3	2023.11.17	修改第 11 章内容
V1.4	2023.11.22	修改第 7、8、11 章 LPTIM 相关内容



Puya Semiconductor Co., Ltd.

IMPORTANT NOTICE

Puya Semiconductor reserves the right to make changes without further notice to any products or specifications herein. Puya Semiconductor does not assume any responsibility for use of any its products for any particular purpose, nor does Puya Semiconductor assume any liability arising out of the application or use of any its products or circuits. Puya Semiconductor does not convey any license under its patent rights or other rights nor the rights of others.