



# **PY32F002A series**

## **32-bit ARM® Cortex®-M0+ microcontroller**

### **HAL Library Sample Manual**

## **1 ADC**

### **1.1 ADC\_MultiChannelSwitch**

此样例演示了 ADC 的多通道切换。

### **1.2 ADC\_SingleConversion\_AWD**

此样例演示了 ADC 的模拟看门狗功能。

### **1.3 ADC\_SingleConversion\_TriggerSW\_Vrefint**

此样例演示了 ADC 模块的 VCC 采样功能，通过采样 VREFINT 的值，计算得出 VCC 的值，并通过串口打印出来。

### **1.4 ADC\_SingleConversion\_TriggerTimer\_IT**

此样例演示了通过 TIM1 触发 ADC 模块的通道采样功能，配置 CH0(PA00)为 ADC 的模拟输入通道，TIM1 配置为主模式，TIM1 每产生一次更新事件，触发一次 ADC 采样，采样数据在中断中打印

### **1.5 ADC\_TempSensor**

此样例演示了 ADC 模块的温度传感器功能，串口每隔 500ms 打印当前的温度。

## 2 COMP

### 2.1 COMP\_CompareGpioVsVrefint\_HYST

此样例演示了 COMP 比较器迟滞功能，PA03 作为比较器正端输入，VREFINT 作为比较器负端输入，PA07 作为比较器的输出端口，通过调整 PA03 上的输入电压，观测 PA07 引脚上的电平变化。

### 2.2 COMP\_CompareGpioVsVrefint\_IT

此样例演示了 COMP 比较器中断功能，PA01 作为比较器正端输入，VREFINT 作为比较器负端输入，PA06 作为比较器的输出端口，通过调整 PA01 上的输入电压，观测 PA06 引脚上的电平变化和 LED 翻转。

### 2.3 COMP\_CompareGpioVsVrefint\_Polling

此样例演示了 COMP 比较器轮询功能，PA01 作为比较器正端输入，VREFINT 作为比较器负端输入，PA06 作为比较器的输出端口，通过调整 PA01 上的输入电压，观测 PA06 引脚上的电平变化情况。

### 2.4 COMP\_CompareGpioVsVrefint\_WakeUpFromStop

此样例演示了 COMP 比较器唤醒功能，PA01 作为比较器正端输入，VREFINT 作为比较器负端输入，PA06 作为比较器的输出端口，进入 stop 模式后，通过调整 PA01 上的输入电压，产生中断唤醒 stop 模式。

### 2.5 COMP\_CompareGpioVsVrefint\_Window

此样例演示比较器窗口功能，PA01 作为比较器正端输入，内部通过 COMP1 和 COMP2 的正极相连，PA1 输入大于 VREF 的电压，LED 以 200ms 的间隔进行翻转；PA1 输入小于  $1/4V_{REF}$  的电压，LED 熄灭；PA1 输入的电压小于 VREF 大于  $1/4V_{REF}$  的电压，LED 常亮。

## 3 CRC

### 3.1 CRC\_Computing\_Results

此样例演示了 CRC 校验功能，通过对一个数组里的数据进行校验，得到的校验值与理论校验值进行比较，相等则 LED 灯亮，否则 LED 灯熄灭。

## 4 EXTI

### 4.1 EXTI\_Event

此样例演示了通过 PA6 引脚唤醒 MCU 的功能。下载程序并运行后，LED 灯处于常亮状态；按下用户按键后，LED 灯处于常暗状态，且 MCU 进入 STOP 模式；拉低 PA6 引脚后，MCU 唤醒，LED 灯处于闪烁状态。

### 4.2 EXTI\_IT

此样例演示了 GPIO 外部中断功能，PB2 引脚上的每一个下降沿都会产生中断，中断函数中 LED 灯会翻转一次

## **5 FLASH**

### **5.1 FLASH\_OptionByteWrite\_RST**

此样例演示了通过软件方式将 RESET 引脚改为普通 GPIO。

### **5.2 FLASH\_PageEraseAndWrite**

此样例演示了 flash page 擦除和 page 写功能。

### **5.3 FLASH\_SectorEraseAndWrite**

此样例演示了 flash sector 擦除和 page 写功能。

## 6 GPIO

### 6.1 GPIO\_Toggle

此样例演示了 GPIO 输出模式，配置 LED 引脚(PA5)为数字输出模式，并且每隔 250ms 翻转一次 LED 引脚电平，运行程序，可以看到 LED 灯以 2Hz 的频率闪烁

## 7 I2C

### 7.1 I2C\_TwoBoard\_CommunicationMaster\_IT

此样例演示了主机 I2C 通过中断方式进行通讯，从机使用 PY32F002，按下 user 按键，主机先向从机发送 15 byte 数据，然后再接收从机发送的数据，数据接收后，会保存在 aRxBuffer 数组中，接收数据为 0x1~0xf。

### 7.2 I2C\_TwoBoard\_CommunicationMaster\_Polling

此样例演示了主机 I2C 通过 POLLING 方式进行通讯，从机使用 PY32F002，按下 user 按键，主机先向从机发送 15 byte 数据，然后再接收从机发送的数据，数据接收后，会保存在 aRxBuffer 数组中，接收数据为 0x1~0xf。

### 7.3 I2C\_TwoBoard\_CommunicationSlave\_IT

此样例演示了主机 I2C 通过中断方式进行通讯，从机使用 PY32F002，按下 user 按键，主机先向从机发送 15 byte 数据，然后再接收从机发送的数据，数据接收后，会保存在 aRxBuffer 数组中，接收数据为 0x1~0xf。



## 8 IWDG

### 8.1 IWDG\_RESET

此样例演示了 IWDG 看门狗功能，配置看门狗重载计数值，计数 1s 后复位，然后通过调整每次喂狗的时间（main 函数 while 循环中代码），可以观察到，如果每次喂狗时间小于 1s 钟，程序能一直正常运行（LED 灯闪烁），如果喂狗时间超过 1s 钟，程序会一直复位（LED 灯熄灭）。

## 9 LPTIM

### 9.1 LPTIM\_WakeUp

此样例演示了 LPTIM 中断唤醒 stop 模式，200ms 唤醒一次。

## **10 PWR**

### **10.1 PWR\_SLEEP\_WFI**

此样例演示了 sleep 模式下，GPIO 外部中断唤醒功能

### **10.2 PWR\_STOP\_WFI**

此样例演示了 stop 模式下，GPIO 外部中断唤醒功能

## **11 RCC**

### **11.1 RCC\_HSE\_OUTPUT**

此样例配置系统时钟为 HSE，并通过 MCO（PA01）引脚输出

### **11.2 RCC\_HSI\_OUTPUT**

此样例配置系统时钟为 HSI，并通过 MCO（PA01）引脚输出

### **11.3 RCC\_LSI\_OUTPUT**

此样例使能 LSI，并通过 MCO（PA01）引脚输出。

## 12 SPI

### 12.1 SPI\_TwoBoards\_FullDuplexMaster\_IT

此样例是对 串口外设接口(SPI)与外部设备以全双工串行方式进行通信 的演示,此接口设置为主模式,为外部从设备提供通信时钟 SCK。主机通过 MOSI 引脚发送数据,从 MISO 引脚接收从机的数据,数据以主机提供的 SCK 沿同步被移位,完成全双工通信。

### 12.2 SPI\_TwoBoards\_FullDuplexSlave\_IT

此样例是对 串口外设接口(SPI)与外部设备以全双工串行方式进行通信 的演示,此接口设置为从模式,通信时钟 SCK 由外部主设备提供。从机通过 MOSI 引脚接收主机数据,从 MISO 引脚发送数据,数据以主机提供的 SCK 沿同步被移位,完成全双工通信。

## 13 TIM1

### 13.1 TIM1\_6StepPWM

此样例是对高级定时器功能“六步 PWM 的产生”的演示，通过 systick 中断作为 COM commutation 事件的触发源，实现（无刷电机的）换向

### 13.2 TIM1\_ARR

此样例实现了定时器的基本计数功能，以及演示了 ARR 自动重载功能，样例在定时器重载中断中翻转 LED 灯 修改 main.c 中的配置 `TimHandle.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_ENABLE`;使能自动重载功能，新的 ARR 值在第四次进中断时生效，配置 `TimHandle.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_DISABLE`;禁止自动重载功能，新的 ARR 值在第三次进中断时生效,生效后，LED 灯以 1.875HZ 的频率翻转

### 13.3 TIM1\_InputCapture

此样例演示了 TIM1 输入捕获功能，PA3 输入时钟信号，TIM1 捕获成功后，会进入捕获中断，每进一次中断，翻转一次 LED

### 13.4 TIM1\_PWM

本例程输出 4 路 PWM，通道 1 的占空比为 20%，通道 2 为 40%，通道 3 为 60%，通道 4 为 80%本例程周期为  $8000000/(50+1)/800=196\text{Hz}$

## 14 TIM16

### 14.1 TIM16\_ARR

此样例演示了在 TIM16 中基本计数功能,并使能了更新中断,每次重装 ARR 值时会产生一次更新中断,并在中断中翻转 LED 灯,LED 灯会进行翻转。

## 15 USART

### 15.1 USART\_HyperTerminal\_AutoBaud\_IT

此样例演示了 USART 的自动波特率检测功能，调试助手发送一个字符 0x7F，MCU 反馈字符串：Auto BaudRate Test。

### 15.2 USART\_HyperTerminal\_IT

此样例演示了 USART 的中断方式发送和接收数据，USART 配置为 115200，数据位 8，停止位 1，校验位 None,下载并运行程序后,上位机通过 USART 会接收到 0x1-0xC,然后通过上位机下发 12 个数据,例如 0x1~0xC,则，MCU 会把接收到的数据再次发送到上位机，后续 USART 循环上面的收发

### 15.3 USART\_HyperTerminal\_Polling

此样例演示了 USART 的 POLLING 方式发送和接收数据，USART 配置为 115200，数据位 8，停止位 1，校验位 None,下载并运行程序后，通过 USART 会接收到 0x1-0xC,然后通过上位机下发 12 个数据,例如 0x1~0xC,则，MCU 会把接收到的数据再次发送。