



PY32F002A series

32-bit ARM® Cortex®-M0+ microcontroller

LL Library Sample Manual

1 ADC

1.1 ADC_ContinuousConversion_TriggerSW_Vrefint_Init

此样例演示了 ADC 模块的 VCC 采样功能，通过采样 VREFINT,反推 VCC 电压。

1.2 ADC_MultichannelSwitch

此样例演示了 ADC 的多通道切换。

1.3 ADC_SingleConversion_AWD

此样例演示了 ADC 的模拟看门狗功能，PA4 为模拟输入，当 PA4 的电压值大于 1/2 供电电压时，LED 灯会闪烁。

1.4 ADC_SingleConversion_TriggerTimer_IT

此样例演示了 ADC 采集通过 TIM 触发的方式打印通道 4 的电压值，PA4 为模拟输入，每隔 1s 会从串口 PA2/PA3 打印当前的电压值。

1.5 ADC_TempSensor_Init

此样例演示了 ADC 模块的温度传感器功能，程序下载后，串口每隔 200ms 打印一次当前检测的温度值和对应的采样值。

2 COMP

2.1 COMP_CompareGpioVsVrefint_HYST_Init

此样例演示了 COMP 比较器迟滞功能, PA01 作为比较器正端输入, VREFINT 作为比较器负端输入, PA00 作为比较器的输出端口, 通过调整 PA01 上的输入电压, 观测 PA00 引脚上的电平变化。。

2.2 COMP_CompareGpioVsVrefint_IT_Init

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, 运行程序, PA01 输入 1.3V 电压, LED 灯亮。

2.3 COMP_CompareGpioVsVrefint_Polling_Init

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, 通过调整 PA01 上的输入电压, 当检测到比较器输出状态为高时, LED 灯亮, 比较器输出状态为低时, LED 灯灭。

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, 上完电 LED 灯会常亮, 用户点击按钮, LED 灯灭, 进入 stop 模式, 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

此样例演示了 COMP 比较器的 window 功能, 比较器 2 的 Plus 端用比较器 1 的 IO3(PA1)作为输入, VREFINT 作为比较器负端输入, 当 PA1 的电压值大于 1.3V 时, LED 灯亮, 小于 1.1V 时, LED 灯灭。

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_Init

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_Init

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 100ms 翻转一次 LED 引脚电平，运行程序，可以看到 LED 灯闪烁

6.2 GPIO_Toggle_Init

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 100ms 翻转一次 LED 引脚电平，运行程序，可以看到 LED 灯闪烁

7 I2C

7.1 I2C_TwoBoards_MasterTx_SlaveRx_Polling

此样例演示了主机 I2C、从机 I2C 通过轮询方式进行通讯，当按下从机单板的用户按键，再下主机单板的用户按键后，主机 I2C 向从机 I2C 发送"LED ON"数据。当主机 I2C 成功发送数据，从机 I2C 成功接收数据时，主机单板和从机单板 LED2 灯分别亮起。

7.2 I2C_TwoBoard_CommunicationMaster_IT_Init

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。

7.3 I2C_TwoBoard_CommunicationMaster_Polling_Init

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。

7.4 I2C_TwoBoard_CommunicationSlave_IT_Init

此样例演示了从机 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 灯不亮）。

9 LPTIM

9.1 LPTIM_Wakeup

此样例演示了 LPTIM 中断唤醒 stop 模式，500ms 唤醒一次

10 PWR

10.1 PWR_SLEEP_WFI

此样例演示了通过 WFI(wait for interrupt)指令进入 sleep 模式，使用 GPIO 中断唤醒

10.2 PWR_STOP_WFI

此样例演示了通过 WFI(wait for interrupt)指令进入 stop 模式，使用 GPIO 中断唤醒

11 RCC

11.1 RCC_HSE_OUTPUT

此样例演示了时钟输出功能，可输出 HSE 波形。

11.2 RCC_HSI_OUTPUT

此样例演示了时钟输出功能，可输出 HSI 波形。

11.3 RCC_LSI_OUTPUT

此样例演示了将系统时钟设置为 LSI，并通过 MCO 引脚输出系统时钟。

12 SPI

12.1 SPI_TwoBoards_FullDuplexMaster_IT_Init

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_Init

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。

13 TIMER1

13.1 TIM1_InputCapture_Init

此样例演示了 TIM1 的输入捕获功能，配置 PA3 作为输入捕获引脚，PA3 每检测到一个上升沿触发捕获中断在捕获中断回调函数中翻转 LED 灯。

13.2 TIM1_PWM3CH_Init

此样例演示了使用 TIM1 PWM2 模式输出三路频率为 10Hz 占空比分别为 25%、50%、75%的 PWM 波形。

13.3 TIM1_TimeBase_Init

此样例演示了 TIM1 的重装载功能和自动重载预装载功能，在初始化阶段配置重载值为 1000，在 ARR 中断回调函数中重新设置重载值为 500。每次进入中断时翻转 LED。可以通过注释 main.c 中 `LL_TIM_EnableARRPreload(TIM1);` 代码关闭自动重载预装载功能。若关闭自动重载预装载功能，新的重载值将在第一次进入中断后立即生效，则 PA5 前 1 次翻转为 1000ms 后续保持 500ms。若开启自动重载预装载功能，新的重载值将在下一次进入中断后生效，则 PA5 前 2 次翻转为 1000ms 后续保持 500ms。

14 USART

14.1 USART_HyperTerminal_AutoBaud_IT_Init

此样例演示了 USART 的自动波特率检测功能,上位机发送 1 字节的波特率检测字符 0x55,如果 MCU 检测成功,则返回字符:Auto BaudRate Test。

14.2 USART_HyperTerminal_IT_Init

此样例演示了通过 USART 中断收发数据的功能,复位 MCU 并重新运行,PC 端收到字符串:UART Test; PC 端发送 12 个字符,MCU 会反馈同样的 12 个字符给 PC 端。

14.3 USART_HyperTerminal_Polling_Init

此样例演示了通过 USART 轮询收发数据的功能,MCU 复位后会向 PC 端发送"UART Test",PC 端发送 12 个字符,MCU 会反馈同样的 12 个字符给 PC 端。

15 UTILS

15.1 UTILS_ConfigureSystemClock

本样例主要演示如何配置 SYSCLK(系统时钟), HCLK(AHB 时钟), PCLK(APB 时钟)。通过 MCO 输出系统时钟 24MHz。

15.2 UTILS_ReadDeviceInfo

本样例主要读取 DBGMCU->IDCODE 寄存器和 UID 的值。UID Word0 表示 LotNumber, Word1 表示 WaferNumber, Word2 表示 X 和 Y 的坐标。