



PY32E407 series

32-bit ARM® Cortex®-M4 microcontroller

HAL Library Sample Manual

1 ADC

1.1 ADC_AnalogWatchdog

此样例演示了 ADC 的模拟看门狗功能，当开启看门狗的通道的电压值不在设定的上下限中，会进入看门狗中断。

This example demonstrates the analog watchdog function of ADC. When the voltage value of the channel that opens the watchdog is not within the set upper or lower limits, Will enter watchdog interrupt.

1.2 ADC_DualModeRegsimult

此样例演示了 ADC1 和 ADC2 的同步规则功能。

This example demonstrates the synchronization rule function of ADC1 and ADC2.

1.3 ADC_MultiChannelsSingleConversion_TriggerSW_DMA

此样例演示了 ADC 的多通道 DMA 传输的功能。

This example demonstrates the functionality of multi-channel DMA transmission in ADC.

1.4 ADC_TempSensor_TriggerTimer_IT

此样例演示了 ADC 模块的 Tempsensor 功能和外部触发功能，并通过串口打印出温度值。

This example demonstrates the Tempsensor function and external trigger function of the ADC module, and prints the temperature value through the serial port.

1.5 ADC_Vrebuf

此样例演示了 ADC 的 Vrebuf 功能。

This example demonstrates the Vrebuf function of ADC.

1.6 ADC_Vrefint

此样例演示了 ADC 的 Vrefint 功能，通过 Vrefint 的值，可以反推出 MCU 的供电电压值。

This example demonstrates the Vrefint function of ADC. By using the value of Vrefint, the power supply voltage value of MCU can be inferred

2 COMP

2.1 COMP_CompareGpioVs1_2VCC_IT

此样例演示了 COMP 比较器中断功能，PB1 作为比较器正端输入， $1/2VCC$ 作为比较器负端输入，当 PB1 的电压大于 $1/2VCC$ 电压时，LED 灯亮，小于 $1/2VCC$ 电压时，LED 灯灭。

This example demonstrates the interrupt function of the COMP comparator, with PB1 as the positive input and $1/2VCC$ as the negative input. When the voltage of PB1 is greater than $1/2VCC$ voltage, the LED lights up, and when it is less than $1/2VCC$ voltage, the LED lights out.

2.2 COMP_CompareGpioVs1_2VCC_WakeUpFromStop

此样例演示了 COMP 比较器唤醒功能，PB1 作为比较器正端输入， $1/2VCC$ 作为比较器负端输入，进入 stop 模式后，通过调整 PB1 上的输入电压，产生中断唤醒 stop 模式。

This example demonstrates the COMP comparator wake-up function, with PB1 as the positive input and $1/2VCC$ as the negative input of the comparator. After entering stop mode, the interrupt wake-up stop mode is generated by adjusting the input voltage on PB1.

2.3 COMP_CompareGpioVs1_2VrefintPolling

此样例演示了 COMP 比较器轮询功能，PB1 作为比较器正端输入， $1/2Vrefint2.048V$ 作为比较器负端输入，当 PB1 的电压大于 $1/2Vrefint2.048V$ 电压时，LED 灯亮，小于 $1/2Vrefint2.048V$ 电压时，LED 灯灭。

This example demonstrates the COMP comparator polling function, with PB1 as the positive input and $1/2Vrefint2.048V$ as the negative input. When the voltage of PB1 is greater than $1/2Vrefint2.048V$ voltage, the LED lights up, and when it is less than $1/2Vrefint2.048V$ voltage, the LED lights off.

3 CORDIC

3.1 CORDIC_Sin_DMA

此样例演示了 CORDIC 的 sin 函数计算功能,输入数据和结果都使用 DMA 传输

This sample demonstrates the sin function calculation capability of CORDIC, with both input data and results being transmitted using DMA.

4 CRC

4.1 CRC_Bytes_Stream_7bit_CRC

此样例演示了，用户定义的生成多项式由 HAL_CRC_Init() 配置。同时，设置输入或输出数据均不得反转，使用默认初始值，并指定输入数据类型为字节。

In this example, the user-defined generating polynomial is configured by HAL_CRC_Init(). At the same time, it is set that neither input or output data must be reversed, the default init value is used and it is specified that input data type is byte.

4.2 CRC_CalculateCheckValue

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。

This example demonstrates the CRC verification function. By verifying the data in an array, the obtained verification value is compared with the theoretical verification value. If it is equal, the LED light will be on, otherwise the LED light will be off.

4.3 CRC_Data_Reversing_16bit_CRC

此样例演示了用户定义的生成多项式由 HAL_CRC_Init()配置。同时，输入数据反转功能设置为全字位反转，输出数据反转功能也已启用（只有位级反转选项可用）。计算出 16 位长的 CRC 值，将其存储在 uwCRCValue 变量中，然后与存储在 uwExpectedCRCValue_reversed 变量中的 CRC 预期值进行比较。

In this example, the user-defined generating polynomial is configured by HAL_CRC_Init(). At the same time, input data reversal feature is set to bit reversal on full word. Output data reversal is enabled as well (only bit-level reversal option is available). Additionally, the default init value is discarded and a user-defined one is used instead. The 16-bit long CRC is computed, stored in uwCRCValue variable then compared to the CRC expected value stored in uwExpectedCRCValue_reversed variable.

4.4 CRC_UserDefinedPolynomial

此样例演示了 CRC（循环冗余校验），计算单元根据用户定义的生成多项式，为给定的 32 位数据字缓冲区计算 8 位 CRC 代码。

This sample demonstrates CRC (Cyclic Redundancy Check), where the calculation unit calculates an 8-bit CRC code for a given 32-bit data word buffer based on a user-defined generating polynomial.

5 CTC

5.1 CTC_Autotrim

此样例演示了 CTC 使用 LSE 做参考时钟自动校准 HSI48M 时钟的功能。

This example demonstrates the function of CTC using LSE as a reference clock to automatically calibrate the HSI48M clock.

6 DAC

6.1 DAC_SingleGeneration

此样例演示了 DAC 的软件触发功能,通道 PA4 能够输出 1/2 的供电电压值。

This example demonstrates the software triggering function of DAC, where channel PA4 can output 1/2 of the supply voltage value.

7 DMA

7.1 DMA_SramToSram

此样例演示了 DMA 从 SRAM 到 SRAM 传输数据的功能（SRAM 和外设之间传输的样例请参考相关外设样例工程）。

This example demonstrates the function of DMA transferring data from SRAM to SRAM (please refer to the relevant peripheral sample project for the example of transfer between SRAM and peripherals).

8 ESMC

8.1 ESMC_MemoryMapping_QSPI

此样例演示了 ESMC 的 memory mapping 功能，把预先编译好的 bin 文件，下载到 P25Q128H 芯片中，然后把 P25Q128H 地址映射到 0x00000000，主程序跳转到 0x00000000 地址开始执行 bin 文件中的程序，样例中 bin 程序执行的任务是闪烁 LED（PB5）灯。

This example demonstrates the memory mapping function of ESMC. The pre compiled bin file is downloaded to the P25Q128H chip, and then the P25Q128H address is mapped to 0x00000000. The main program jumps to the 0x00000000 address to start executing the program in the bin file. In the example, the bin program executes the task of flashing the LED (PB5) light.

8.2 ESMC_ReadWrite_4IO_DTR_Polling

此样例演示了 ESMC 在间接模式下的 DTR 的 polling 传输功能，对 P25Q128H 的芯片进行擦除，写入数据，DTR 读取数据，然后把读取的数据和写入的数据进行对比，数据正确则 LED 灯亮，否则 LED 灯熄灭。

This example demonstrates the polling transmission function of ESMC's DTR in indirect mode, erasing the chip of P25Q128H, writing data, DTR reading data, and then comparing the read data with the written data. If the data is correct, the LED will ON, otherwise the LED will OFF.

8.3 ESMC_ReadWrite_QSPI_DMA

此样例演示了 ESMC 在间接模式下的 DMA 数据传输功能，对 P25Q128H 的芯片进行擦除，写入数据，读取数据，然后把读取的数据和写入的数据进行对比，数据正确则 LED 灯亮，否则 LED 灯熄灭。

This example demonstrates the DMA data transmission function of ESMC in indirect mode, erasing, writing data, reading data on the P25Q128H chip, and then comparing the read data with the written data. If the data is correct, the LED will ON, otherwise the LED will OFF.

8.4 ESMC_ReadWrite_QSPI_IT

此样例演示了 ESMC 在间接模式下的中断数据传输功能，对 P25Q128H 的芯片进行擦除，写入数据，读取数据，然后把读取的数据和写入的数据进行对比，数据正确则 LED 灯亮，否则 LED 灯熄灭。

This example demonstrates the interrupt data transmission function of ESMC in indirect mode, erasing, writing data, reading data on the P25Q128H chip, and then comparing the read data with the written data. If the data is correct, the LED will ON, otherwise the LED will off.

8.5 ESMC_ReadWrite_QSPI_Polling

此样例演示了 ESMC 在间接模式下的 polling 传输功能，对 P25Q128H 的芯片进行擦除，写入数据，读取数据，然后把读取的数据和写入的数据进行对比，数据正确则 LED 灯亮，否则 LED 灯熄灭。

This example demonstrates the polling transmission function of ESMC in indirect mode, erasing, writing data, reading data on the P25Q128H chip, and then comparing the read data with the written data. If the data is correct, the LED will ON, otherwise the LED will OFF.

8.6 ESMC_XIP_ReadWritePSRAM_4IO

此样例演示了 ESMC 对 PSRAM 的 XIP 模式的读写功能。对 AP1604 进行 XIP 模式写入 1024 个数据，读取回来并且比较。

This example demonstrates ESMC's read and write functionality for XIP mode of PSRAM. Write 1024 data to AP1604 and Read the data back in XIP modes and compare them.

9 EXTI

9.1 EXTI_ToggleLed_IT

此样例演示了 GPIO 外部中断功能，PB6 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。

This example demonstrates the GPIO external interrupt function, where each falling edge on PB6 generates an interrupt, and the LED light in the interrupt function toggle once.

9.2 EXTI_WakeUp_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 STOP1 模式；拉低 PA6 引脚后，MCU 唤醒，LED 灯处于闪烁状态。

This example demonstrates the function of waking up an MCU through the PA6 pin. After downloading the program and running it, the LED light is constantly on; After pressing the user button, the LED light is in a constant dark state and the MCU enters STOP1 mode; After pulling down the PA6 pin, the MCU wakes up and the LED light is in a flashing state.

10 FDCAN

10.1 FDCAN_Classic_StandardID_IT

此样例演示了采用 CAN2.0 协议标准帧中断方式与 PCAN-View 的通信功能，MCU 首先自动向 PCAN-View 发送 ID 为 0x111 的 2 帧数据，PCAN-View 接收到数据后，然后手动通过 PCAN-View 向 MCU 发送 ID 为 0x111 的 2 帧数据。

This sample demonstrates the communication function between CAN2.0 protocol standard frame interrupt mode and PCAN-View. MCU automatically sends 2 frames of data with ID 0x111 to PCAN-View at first. After PCAN-View receives the data, MCU manually sends 2 frames of data with ID 0x111 to MCU through PCAN-View.

10.2 FDCAN_Classic_StandardID_Loopback_polling

此样例演示了采用 CAN2.0 协议、标准帧、轮询方式的内部回环通信。

This sample demonstrates internal loopback communication using CAN2.0 protocol, standard frames, polling.

10.3 FDCAN_ExtendID_polling

此样例演示了采用 CANFD 协议扩展帧轮询方式与 PCAN-View 的通信功能，MCU 首先自动向 PCAN-View 发送 64byte 数据 0x0~0x3F，PCAN-View 接收到数据后，然后手动通过 PCAN-View 向 MCU 发送 ID 为 0x1234567F 的 64byte 数据，MCU 会自动将接收到数据通过串口打出。

This example demonstrates the communication function between the CANFD protocol extension frame polling method and the PCAN View. The MCU first automatically sends 64byte data 0x0~0x3F to the PCAN View. After the PCAN View receives the data, it manually sends 64byte data with ID 0x1234567F to the MCU through the PCAN View. The MCU will automatically print the received data through the serial port.

11 FLASH

11.1 FLASH_PageEraseAndWrite

此样例演示了 flash page 擦除和 page 写功能。

This example demonstrates the flash page erase and page write functions.

11.2 FLASH_SectorEraseAndWrite

此样例演示了 flash sector 擦除和 page 写功能。

This example demonstrates the flash sector erase and page write functions.

12 GPIO

12.1 GPIO_IOToggle

此样例演示了 GPIO 输出模式，配置 LED 引脚为数字输出模式，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。

This example demonstrates the GPIO output mode, configuring the LED pin to be in digital output mode, and flipping the LED pin level every 250ms. Running the program, you can see that the LED light flashes at a frequency of 2Hz.

13 I2C

13.1 I2C_TwoBoards_Com_DMA

此样例演示了 I2C 通过 DMA 方式进行通讯，主机先向从机发送 15byte 数据，然后再接收从机发送的 15byte 数据,主机、从机接收数据成功后，主机和从机板上的小灯处于“常亮”状态。

This sample demonstrates I2C communication using DMA. The master device sends 15 bytes of data to the slave device and then receives 15 bytes of data from the slave. When both the master and slave successfully transmit and receive data, the LEDs on both boards will be constantly lit.

13.2 I2C_TwoBoards_Com_DMA_MEM

此样例演示了主机 I2C 通过 DMA 方式进行通讯，从机使用 EEPROM 外设芯片 P24C32，按下 user 按键，主机先向从机写 15bytes 数据为 0x1~0xf，然后再从 EEPROM 中将写入的数据读出，读取成功后，主机板上的小灯处于“常亮”状态。

This sample demonstrates communication between the master device using I2C and the slave device using the EEPROM peripheral chip P24C32. When the user button on the master device is pressed, the master device first writes 15 bytes of data to the slave device, ranging from 0x1 to 0xF. Then it reads the written data from the EEPROM. Once the data is successfully read, the LED on the master board will remain constantly lit.

13.3 I2C_TwoBoards_Com_IT

此样例演示了 I2C 通过中断方式进行通讯，主机先向从机发送 15byte 数据，然后再接收从机发送的 15byte 数据,主机、从机接收数据成功后，主机和从机板上的小灯处于“常亮”状态。

This sample demonstrates I2C communication using interrupt. The master device sends 15 bytes of data to the slave device and then receives 15 bytes of data from the slave. When both the master and slave successfully transmit and receive data, the LEDs on both boards will be constantly lit.

13.4 I2C_TwoBoards_Com_IT_DualAddr

此样例演示了 I2C 通过中断方式进行通讯，主机先向从机发送 15byte 数据，然后再接收从机发送的 15byte 数据,主机、从机接收数据成功后，主机和从机板上的小灯处于“常亮”状态。

This sample demonstrates I2C communication using interrupt. The master device sends 15 bytes of data to the slave device and then receives 15 bytes of data from the slave. When both the master and slave successfully transmit and receive data, the LEDs on both boards will be constantly lit.

13.5 I2C_TwoBoards_Com_Polling

此样例演示了 I2C 通过轮询方式进行通讯，主机先向从机发送 15byte 数据，然后再接收从机发送的 15byte 数据,主机、从机接收数据成功后，主机和从机板上的小灯处于“常亮”状态。

This sample demonstrates I2C communication using polling. The master device sends 15 bytes of data to the slave device and then receives 15 bytes of data from the slave. When both the master and slave successfully transmit and receive data, the LEDs on both boards will be constantly lit.

13.6 I2C_TwoBoards_MasterTxSlaveRxIndefiniteLengthData_IT

此样例演示了通过中断方式，主机发送不定长数据，从机接收不定长数据。主机向从机发送 10 字节的数据（0~9），然后从机接收数据（0~9）并通过串口打印；主机向从机发送 100 字节数据（1~100），然后从机接收数据（1~100）并通过串口打印；主机向从机发送 10 字节的数据（0~9），然后从机接收数据（0~9）并通过串口打印。

This example demonstrates how the host sends variable length data and the slave receives variable length data through interrupt mode. The host sends 10 bytes of data (0-9) to the slave, and then the slave receives the data (0-9) and prints it through the serial port; The host sends 100 bytes of data (1-100) to the slave, and then the slave receives the data (1-100) and prints it through the serial port; The host sends 10 bytes of data (0-9) to the slave, and then the slave receives the data (0-9) and prints it through the serial port.

14 I2S

14.1 I2S_TwoBoards_FullDuplex_DMA

此样例是对 I2S 主机与 I2S 从机以 DMA 方式进行通信的演示, I2S 主机先向 I2S 从机发送数据 0x1~0x10, I2S 从机接收到数据后, 再向 I2S 主机回发数据 0x1~0x10, 当 I2S 主机和 I2S 从机成功接收数据时, 小灯处于常亮状态; 否则小灯处于闪烁状态。

This sample demonstrates communication between the I2S master and I2S slave using DMA. The I2S master sends data 0x1 to 0x10 to the I2S slave. The I2S slave receives the data and sends back data 0x1 to 0x10 to the I2S master. When both the I2S master and I2S slave successfully receive the data, the LED will be constantly on. Otherwise, the LED will be blinking.

14.2 I2S_TwoBoards_FullDuplex_IT

此样例是利用中断对串口外设接口 (SPI) 与外部设备以全双工串行方式进行通信的演示, 主设备提供通信时钟 SCK, 通过 MOSI/MISO 引脚发送/接收数据。从设备通过 MOSI/MISO 引脚接收/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。

This sample is a demonstration of using interrupts to communicate with a serial peripheral interface (SPI) and an external device in full-duplex serial mode. The master device provides the communication clock SCK and sends/receives data through the MOSI/MISO pin. The slave device receives/transmits data through the MOSI/MISO pins. The data is shifted synchronously along the SCK provided by the master to complete full-duplex communication.

14.3 I2S_TwoBoards_FullDuplex_Polling

此样例是利用轮询对串口外设接口 (SPI) 与外部设备以全双工串行方式进行通信的演示, 主设备提供通信时钟 SCK, 通过 MOSI/MISO 引脚发送/接收数据。从设备通过 MOSI/MISO 引脚接收/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。

This sample is a demonstration of using polling to communicate with a serial peripheral interface (SPI) and an external device in full-duplex serial mode. The master device provides the communication clock SCK and sends/receives data through the MOSI/MISO pin. The

slave device receives/transmits data through the MOSI/MISO pins. The data is shifted synchronously along the SCK provided by the master to complete full-duplex communication.

15 IWDG

15.1 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 800ms 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 750ms，程序能一直正常运行（LED 灯闪烁），如果喂狗时间 850ms，程序会一直复位（LED 灯熄灭）。

This example demonstrates the function of IWDG (Independent Watchdog). Set IWDG to count 800ms and then reset. By adjusting the time of refresh the dog each time (code in the main function while loop), it can be observed that if the time is 750ms, the program can always run normally (LED blink), if the time is 850ms, the program will always reset (LED off).

16 LCDC

16.1 LCDC_6800

此样例演示了 LCDC 驱动 6800 LCD 屏的功能。

This sample demonstrates the function of LCDC to drive the LCD screen.

16.2 LCDC_8080

此样例演示了 LCDC 驱动 LCD 屏的功能。

This sample demonstrates the function of LCDC to drive the LCD screen.

17 LPTIM

17.1 LPTIM_Wakeup_WFE

此样例演示了 LPTIM 连续模式事件唤醒 STOP 模式。

This example demonstrates the LPTIM continuous mode event wake-up STOP mode.

17.2 LPTIM_Wakeup_WFI

此样例演示了 LPTIM 连续模式中断唤醒 STOP 模式。

This example demonstrates the LPTIM continuous mode interrupt wake-up STOP mode.

18 LPUART

18.1 LPUART_TwoBoards_Com_DMA

此样例演示了两块电路板之间在 DMA 模式下进行 LPUART 传输，发送/接收在两块电路板之间进行。LPUART 配置为 115200，数据位 8，停止位 1，校验位 None。

This sample demonstrates LPUART transmission between two boards in DMA mode, with transmit/receive taking place between the two boards. The LPUART is configured for 115200, with data bit 8, stop bit 1, and parity bit None.

18.2 LPUART_TwoBoards_Com_IT

此样例演示了两块电路板之间在中断模式下进行 LPUART 传输，发送/接收在两块电路板之间进行。LPUART 配置为 115200，数据位 8，停止位 1，校验位 None。

This sample demonstrates LPUART transmission between two boards in interrupt mode, with transmit/receive taking place between the two boards. The LPUART is configured for 115200, with data bit 8, stop bit 1, and parity bit None.

18.3 LPUART_TwoBoards_Com_Polling

此样例演示了两块电路板之间在轮询模式下进行 LPUART 传输，发送/接收在两块电路板之间进行。LPUART 配置为 115200，数据位 8，停止位 1，校验位 None。

This sample demonstrates LPUART transmission between two boards in polling mode, with transmit/receive taking place between the two boards. The LPUART is configured for 115200, with data bit 8, stop bit 1, and parity bit None.

19 OPA

19.1 OPA_PGA

此样例演示了 OPA 的可编程增益放大器功能,OPA 的输出端会输出正端 2 倍的电压值。

This example demonstrates the programmable gain amplifier function of OPA, where the output terminal outputs a voltage value twice of the positive terminal.

19.2 OPA_STANDALONE

此样例演示了 OPA 的独立模式功能,OPA 的负端和输出端短接, 输出端会输出和正端一样的电压值。

This example demonstrates the independent mode function of OPA, where the negative and output terminals of OPA are short circuited, and the output terminal outputs the same electrical output as the positive terminal value.

20 PWR

20.1 PWR_LPRUN

此样例演示了进入和退出 lprun 模式。

This example demonstrates entering and exiting the lprun mode.

20.2 PWR_LPSLEEP_WFI

此样例演示了 lpsleep 模式下，GPIO 外部中断唤醒功能。

This sample demonstrates the GPIO external interrupt wake-up feature in lpsleep mode.

20.3 PWR_PVD

此样例演示了 PVD 电压检测功能。当供电电压低于 3.0V 时，LED 会点亮，高于 3.0V 时，LED 灯会熄灭。

This sample demonstrates the PVD (Programmable Voltage Detector) voltage detection functionality. When the supply voltage is lower than 3.0V, the LED will light up. When the supply voltage is higher than 3.0V, the LED will turn off.

20.4 PWR_SLEEP_WFI

此样例演示了 sleep 模式下，GPIO 外部中断唤醒功能。

This sample demonstrates the GPIO external interrupt wake-up feature in sleep mode.

20.5 PWR_STANDBY

此样例演示了 standby 模式下，通过 wakeuppinn 唤醒功能。

This sample demonstrates the wake-up feature using the wakeup pin in standby mode.

20.6 PWR_STOP1_WFE

此样例演示了 stop1 模式下，GPIO 外部事件唤醒功能。

This sample demonstrates the GPIO external event wake-up feature in stop1 mode.

20.7 PWR_STOP1_WFI

此样例演示了 stop1 模式下，GPIO 外部中断唤醒功能。

This sample demonstrates the GPIO external interrupt wake-up feature in stop1 mode.

21 RCC

21.1 RCC_HSE_Output

此样例配置系统时钟为 HSE，并通过 MCO (PA08) 引脚输出。

This sample configures the system clock to use the HSE (High-Speed External) clock source and outputs it through the MCO (PA08) pin.

21.2 RCC_HSI_Output

此样例配置系统时钟为 HSI，并通过 MCO (PA08) 引脚输出。

This sample configures the system clock to use the HSI (High-Speed Internal) clock source and outputs it through the MCO (PA08) pin.

21.3 RCC_LSE_Output

此样例使能 LSE，并通过 MCO (PA08) 引脚输出。

This sample enables the LSE and is output via the MCO (PA08) pin.

21.4 RCC_LSI_Output

此样例使能 LSI，并通过 MCO (PA08) 引脚输出。

This sample enables the LSI and is output via the MCO (PA08) pin.

21.5 RCC_PLL_Output

此样例配置系统时钟为 PLL，并通过 MCO（PA08）引脚输出，PLL 的输入时钟源选择 HSI/2。

This sample configures the system clock to use the PLL (Phase-Locked Loop) clock source with HSI/2 as the input clock source and outputs it through the MCO (PA08) pin.

22 RNG

22.1 RNG_MultiRNG_IT

此样例演示了通过中断方式，产生多个随机数的功能。

This example demonstrates the ability to generate multiple random numbers by interrupt.

22.2 RNG_MultiRNG_Polling

此样例演示了通过轮询方式，产生多个随机数的功能。

This example demonstrates the ability to generate multiple random numbers by polling.

23 RTC

23.1 RTC_AlarmSecond_IT

此样例演示 RTC 的秒中断和闹钟中断功能，每次秒中断，在中断函数中会打印字符 "RTC_IT_SEC"，并且输出实时时间。

This sample demonstrates the second interrupt and alarm interrupt functionality of the RTC. Each time the second interrupt occurs, the interrupt function will print the string "RTC_IT_SEC" and output the current RTC count time.

23.2 RTC_WakeUpAlarm

此样例演示通过 RTC 闹钟中断每隔 1S 将 MCU 从 STOP1 模式下唤醒，每次唤醒会翻转 LED，LED 翻转间隔为 1s。

This example demonstrates waking the MCU from STOP1 mode every 1 second using an RTC alarm clock interrupt. Each wake-up will flip the LED, with an LED flip interval of 1 second.

24 SD

24.1 SDCard_Block_DMA

此样例通过 DMA 方式，演示了 SD 卡擦除、写、读功能。

This sample demonstrates SD card erase, write, read function by DMA mode.

24.2 SDCard_Block_IT

此样例通过中断方式，演示了 SD 卡擦除、写、读功能。

This sample demonstrates SD card erase, write, read function by interrupt mode.

24.3 SDCard_Block_Polling

此样例通过轮询方式，演示了 SD 卡擦除、写、读功能。

This sample demonstrates SD card erase, write, read function by polling mode.

25 SPI

25.1 SPI_FullDuplex_ExternalFLASH

此样例演示了主机 SPI 通过轮询方式进行通讯，从机使用 FLASH 外设芯片 P25Q64，按下 user 按键，主机先向从机写 15bytes 数据为 0x1~0xf，然后再从 FLASH 中将写入的数据读出，读取成功后，主机板上的小灯处于“常亮”状态。

This sample demonstrates the host SPI communication through polling, the slave uses FLASH peripheral chip P25Q64, press the user button, the host first to the slave to write 15bytes of data for the 0x1 ~ 0xf, and then from the FLASH will be written to read out the data, read the success of the host board, the small light is in the “always on” state! After successful reading, the small light on the host board is in the “always on” state.

25.2 SPI_TwoBoards_FullDuplex_DMA

此样例是利用 DMA 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，主设备提供通信时钟 SCK，通过 MOSI/MISO 引脚发送/接收数据。从设备通过 MOSI/MISO 引脚接收/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。

This sample is a demonstration of using DMA to communicate with a serial peripheral interface (SPI) and an external device in full-duplex serial mode. The master device provides the communication clock SCK and sends/receives data through the MOSI/MISO pin. The slave device receives/transmits data through the MOSI/MISO pins. The data is shifted synchronously along the SCK provided by the master to complete full-duplex communication.

25.3 SPI_TwoBoards_FullDuplex_IT

此样例是利用中断对串口外设接口 (SPI) 与外部设备以全双工串行方式进行通信的演示，主设备提供通信时钟 SCK，通过 MOSI/MISO 引脚发送/接收数据。从设备通过 MOSI/MISO 引脚接收/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。

This sample is a demonstration of using interrupts to communicate with a serial peripheral interface (SPI) and an external device in full-duplex serial mode. The master device provides

the communication clock SCK and sends/receives data through the MOSI/MISO pin. The slave device receives/transmits data through the MOSI/MISO pins. The data is shifted synchronously along the SCK provided by the master to complete full-duplex communication.

25.4 SPI_TwoBoards_FullDuplex_Polling

此样例是利用轮询对串口外设接口（SPI）与外部设备以全双工串行方式进行通信的演示，主设备提供通信时钟 SCK，通过 MOSI/MISO 引脚发送/接收数据。从设备通过 MOSI/MISO 引脚接收/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。

This sample is a demonstration of using polling to communicate with a serial peripheral interface (SPI) and an external device in full-duplex serial mode. The master device provides the communication clock SCK and sends/receives data through the MOSI/MISO pin. The slave device receives/transmits data through the MOSI/MISO pins. The data is shifted synchronously along the SCK provided by the master to complete full-duplex communication.

26 TIM

26.1 TIM1_6Step

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，下表是换向步骤，比如第一步中的 CH1 和 CH3N 为 1，即设置打开这两个通道的 PWM 输出。

This sample demonstrates advanced timer function 'six-step PWM generation', systick interrupt as COM commutation event trigger source to achieve commutation (brushless motor). The following table shows the commutating steps. For example, CH1 and CH3N in the first step are set to 1, that mean the PWM output of these two channels is set to start

26.2 TIM1_ComplementarySignals_DeadTime

此样例实现了定时器的刹车功能，CH1 和 CH1N 互补 pwm 输出，接收到外部 IO 口的刹车信号（低电平）后，PWM 信号关闭，由于 BDTR.AOE 置位，所以刹车信号取消（高电平）后，继续 pwm 输出，此样例实现了死区功能。

This example realizes the braking function of the timer, CH1 and CH1N complement pwm output, after receiving the brake signal (low level) of the external IO port the PWM signal is turned off. Because the BDTR.AOE is set, so after the brake signal is cancelled (high level), and the pwm output continues. The dead time is inserted in the complementary output of CH1 and CH1N

26.3 TIM1_DMABURST_CCR

此样例演示了在 TIM1 中使用 DMA 连续两次 burst 传输数据的功能，burst 每传输一次更新两个寄存器，CCR1, CCR2, TIM1 的通道 1 和通道 2 输出 PWM 信号，通过逻辑分析仪监测，可看到两个通道都输出周期为 100ms 的 PWM 信号，信号的占空比会从第一次的 20%，第二次 20%，第三次 50%，第四次及后续变为 80%，此时两次 burst 传输完成，并且 CCR1, CCR2 均更新完毕。

This sample demonstrates the use of DMA in TIM1 two consecutive burst transmission of data, burst every transmission to update the two registers, CCR1, CCR2, TIM1 channel 1 and channel 2 output PWM signals, through the logic analyser to monitor the output of the two channels can be seen as a period of 100ms of the PWM signal, the duty cycle of the signal will

be from the first time the 20%, the second 20%, the third 50%, the fourth and subsequent to 80%, when the two burst transmission is complete, and CCR1, CCR2 are updated.

26.4 TIM1_EncoderTI2AndTI1

此样例实现了 TIM1 中的编码器计数功能，TI1(PA8)和 TI2(PA9)作为编码器输入引脚，通过 CNT 寄存器可观察到计数器变化，通过 uwDirection 变量可观察到计数器的计数方向，通过打印数据也可观察计数方向和 CNT 寄存器计数值，打印数据 Direction = 0 为向上计数，Direction = 1 为向下计数。

This sample demonstrates encoder count function of the TIM1, TI1(PA8) and TI2(PA9) configured as encoder input pins. The change of the counter can be observed through the CNT register, and the counting direction of the counter can be observed through the uwDirection variable. The counting Direction and CNT register can also be observed by printing data. The printed data Direction = 0 indicates CounterMode:Up, and direction = 1 indicates CounterMode:down.

26.5 TIM1_ExternalClockMode1_TI1F

此样例演示了 TIM1 的外部时钟模式 1 功能，选择 TI1FD(PA8)引脚作为外部时钟输入源，并使能更新中断，在中断中翻转 LED 灯。

This sample demonstrates the external clock mode 1 function of TIM1, selects the TI1FD(PA8) pin as the external clock input source, and enables the update interrupt and toggle the LED light in the interrupt

26.6 TIM1_InputCapture_TI1FP1

此样例演示了在 TIM1(PA8)输入捕获功能，PA8 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。

This sample demonstrates the input capture function of TIM1(PA8), PA8 input clock signal, when TIM1 capture success, will enter the capture interrupt, and toggle the LED in the interrupt

26.7 TIM1_OCToggle

此样例演示了 TIM1 比较模式下的 OC 翻转输出功能，使能 CH1(PA08),CH2(PA09),CH3(PA10),CH4(PA11)四个通道的输出功能，并且当计数器 TIMx_CNT 与 TIMx_CCRx 匹配时输出信号翻转,频率为 10Hz。

This sample demonstrates the OC toggle output function in TIM1 comparison mode, enabling CH1(PA08),CH2(PA09),CH3(PA10),CH4(PA11),four channel output function, then the output signal toggle when the counter TIMx_CNT matches TIMx_CCRx. The frequency is 10Hz

26.8 TIM1_OnePulseOutput

此样例演示了 TIM1 的单脉冲模式，CH2(PA09)引脚上的上升沿，触发计数器开始计数，当计数值与 CCR1 匹配时，CH1(PA08)输出高电平，直到计数器溢出，CH1 再次输出低电平，计数器溢出后，定时器停止工作。

This sample demonstrates the one pulse mode of TIM1.The rising edge on the CH2(PA09) pin triggers the counter to start counting. when the count value matches CCR1,CH1(PA08) outputs a high level. When the counter overflows ,CH1 outputs the low level again. After the counter overflows, the timer stops working.

26.9 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%。

This sample outputs 4 channels PWM, the duty cycle of channel 1 is 20%,channel 2 is 40%, channel 3 is 60%,channel 4 is 80%

26.10 TIM1_SynchronizationEnable

定时器 1 的使能由定时器 3 控制，当定时器 3 计数时，LED 会常亮，当定时器 3 发生更新事件时，更新事件会触发定时器 1，定时器 1 开始计数后，LED 会以 10Hz 的频率进行翻转。

The enable of TIM1 is controlled by TIM3. When TIM3 counts, the LED will be steady on. The update event generated by TIM3 will triggers TIM1, and when TIM1 starts counting, the LED is toggled at a frequency of 10Hz

26.11 TIM_Dither

本例程输出抖动模式下的 PWM 信号

This sample outputs the PWM signal in dithering mode

27 UART

27.1 UART_HyperTerminal_DMA

此样例演示了 UART 的 DMA 方式发送和接收数据，UART 配置为 115200，数据位 8，停止位 1，校验位 None，下载并运行程序后，打印提示信息，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，然后打印结束信息。

This example demonstrates how to use UART to send an amount of data in DMA mode. UART configuration is 115200 baud rate, data bit 8, stop bit 1, check bit None. After download and run the program, Print the prompt message, and then send 12 data through the upper computer, such as 0x1~0xC, the MCU will send the received data to the upper computer again, Then print the end message

27.2 UART_HyperTerminal_IT

此样例演示了 UART 的中断方式发送和接收数据，UART 配置为 115200，数据位 8，停止位 1，校验位 None，下载并运行程序后，打印提示信息，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，然后打印结束信息。

This example demonstrates how to use UART to send an amount of data in interrupt mode. UART configuration is 115200 baud rate, data bit 8, stop bit 1, check bit None. After download and run the program, Print the prompt message, and then send 12 data through the upper computer, such as 0x1~0xC, the MCU will send the received data to the upper computer again, Then print the end message

27.3 UART_HyperTerminal_Polling

此样例演示了 UART 的轮询方式发送和接收数据，UART 配置为 115200，数据位 8，停止位 1，校验位 None，下载并运行程序后，打印提示信息，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，然后打印结束信息。

This example demonstrates how to use UART to send an amount of data in polling mode. UART configuration is 115200 baud rate, data bit 8, stop bit 1, check bit None. After download and run the program, Print the prompt message, and then send 12 data through the upper computer, such as 0x1~0xC, the MCU will send the received data to the upper computer again,

Then print the end message

28 USART

28.1 SCI_HyperTerminal_AutoBaud_IT

此样例演示了 SCI 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。

This sample demonstrates the automatic baud rate detection feature of SCI. When the debugging assistant sends a character 0x7F, the MCU will respond with the string: "Auto BaudRate Test".

28.2 SCI_HyperTerminal_DMA

此样例演示了 SCI 的 DMA 方式发送和接收数据，SCI 配置为 115200，数据位 8，停止位 1，校验位 None，下载并运行程序后，打印提示信息，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，然后打印结束信息。

This example demonstrates how to use SCI to send an amount of data in DMA mode. SCI configuration is 115200 baud rate, data bit 8, stop bit 1, check bit None. After download and run the program, print the prompt message, and then send 12 data through the upper computer, such as 0x1~0xC, the MCU will send the received data to the upper computer again, then print the end message.

28.3 SCI_HyperTerminal_IT

此样例演示了 SCI 的中断方式发送和接收数据，SCI 配置为 115200，数据位 8，停止位 1，校验位 None，下载并运行程序后，打印提示信息，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，然后打印结束信息。

This example demonstrates how to use SCI to send an amount of data in interrupt mode. SCI configuration is 115200 baud rate, data bit 8, stop bit 1, check bit None. After download and run the program, print the prompt message, and then send 12 data through the upper computer, such as 0x1~0xC, the MCU will send the received data to the upper computer again, then print the end message.

28.4 SCI_HyperTerminal_Polling

此样例演示了 SCI 的轮询方式发送和接收数据, SCI 配置为 115200, 数据位 8, 停止位 1, 校验位 None, 下载并运行程序后, 打印提示信息, 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, 然后打印结束信息。

This example demonstrates how to use SCI to send an amount of data in polling mode. SCI configuration is 115200 baud rate, data bit 8, stop bit 1, check bit None. After download and run the program, Print the prompt message, and then send 12 data through the upper computer, such as 0x1~0xC, the MCU will send the received data to the upper computer again, Then print the end message

28.5 SCI_TwoBoards_Com_DMA

此样例演示了两块电路板之间在 DMA 模式下进行 SCI 传输, 发送/接收在两块电路板之间进行。SCI 配置为 115200, 数据位 8, 停止位 1, 校验位 None。

This sample demonstrates SCI transmission between two boards in DMA mode, with transmit/receive taking place between the two boards. The SCI is configured for 115200, with data bit 8, stop bit 1, and parity bit None.

28.6 SCI_TwoBoards_Com_IT

此样例演示了两块电路板之间在中断模式下进行 SCI 传输, 发送/接收在两块电路板之间进行。SCI 配置为 115200, 数据位 8, 停止位 1, 校验位 None。

This sample demonstrates SCI transmission between two boards in interrupt mode, with transmit/receive taking place between the two boards. The SCI is configured for 115200, with data bit 8, stop bit 1, and parity bit None.

28.7 SCI_TwoBoards_Com_Polling

此样例演示了两块电路板之间在轮询模式下进行 SCI 传输，发送/接收在两块电路板之间进行。SCI 配置为 115200，数据位 8，停止位 1，校验位 None。

This sample demonstrates SCI transmission between two boards in polling mode, with transmit/receive taking place between the two boards. The SCI is configured for 115200, with data bit 8, stop bit 1, and parity bit None.

28.8 USART_TwoBoards_FullDuplexMaster_DMA

此样例演示了 USART 同步通信功能，USART 设备作为主机、SPI 设备作为从机，采用 DMA 方式通信。在 USART 设备提供同步时钟下，主从机完成全双工通信。

This sample demonstrates USART synchronous communication, with USART devices as masters and SPI devices as slaves, communicating in DMA mode. With synchronous clock provided by USART equipment, master and slave complete full duplex communication.

28.9 USART_TwoBoards_FullDuplexMaster_IT

此样例演示了 USART 同步通信功能，USART 设备作为主机、SPI 设备作为从机，采用中断方式通信。在 USART 设备提供同步时钟下，主从机完成全双工通信。

This sample demonstrates USART synchronous communication, with USART devices as masters and SPI devices as slaves, communicating in interrupt mode. With synchronous clock provided by USART equipment, master and slave complete full duplex communication.

28.10 USART_TwoBoards_FullDuplexMaster_Polling

此样例演示了 USART 同步通信功能，USART 设备作为主机、SPI 设备作为从机，采用轮询方式通信。在 USART 设备提供同步时钟下，主从机完成全双工通信。

This sample demonstrates USART synchronous communication, with USART devices as masters and SPI devices as slaves, communicating in polling mode. With synchronous clock provided by USART equipment, master and slave complete full duplex communication.

29 WWDG

29.1 WWDG_IT

此样例演示了 WWDG 的提前唤醒中断功能，看门狗计数器向下计数到 0x40 时产生中断，中断中喂狗，可以确保看门狗不会复位。

This example demonstrates early wake up interrupt function of the WWDG. When the watchdog counter counts down to 0x40 will generate an interrupt. Refresh the WWDG in interrupt to ensure that the WWDG does not reset.

29.2 WWDG_Window

此样例演示了 WWDG 的窗口看门狗功能，配置 WWDG 的窗口上限（下限固定是 0x3F），程序中通过 delay 延时函数，确保程序是在 WWDG 计数窗口内进行喂狗动作，通过 LED 灯闪烁，可以判断窗口内喂狗并未产生复位。

This example demonstrates the window watchdog function of WWDG. Set the upper limit of the window of WWDG (the lower limit is fixed at 0x3F). The program ensures that the WWDG is refreshed in the WWDG counting window through the delay function, and can judge that the WWDG is refreshed in the window without resetting through the LED light blinking.