PY32L090 series
32-bit ARM® Cortex®-M0+ 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_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.

1.3 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.

1.4 ADC_Vrefbuf

此样例演示了 ADC 的 Vrebuf 功能。

This example demonstrates the Vrebuf function of ADC.

1.5 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，PD0 作为比较器正端输入，1/2VCC 作为比较器负端输入，当PD0 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 PD0 as the positive input and 1/2VCC as the negative input. When the voltage of PD0 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，PD0 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 PD0 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 PD0 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 PD0.

2.3 COMP_CompareGpioVs1_2VREFBUF_Polling

此样例演示了 COMP 比较器轮询功能，PD0 作为比较器正端输入，1/2Vrebuf2.048V 作为比较器负端输入，当 PD0 的电压大于 1/2Vrebuf2.048V 电压时，LED 灯亮，小于 1/2Vrebuf2.048V 电压时,LED 灯灭。

This example demonstrates the COMP comparator polling function, with PD0 as the positive input and 1/2Vrebuf2.048V as the negative input. When the voltage of PD0 is greater than 1/2Vrebuf2.048V voltage, the LED lights up, and when it is less than 1/2Vrebuf2.048V voltage, the LED lights off.

3 CRC

3.1 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.

3.2 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.

3.3 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.

3.4 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.

4 DAC

4.1 DAC_SingleGeneration

此样例演示了 DAC 的软件触发功能,通道 PC6 能够输出 1/2 的供电电压值。

This example demonstrates the software triggering function of DAC, where channel PC6 can output 1/2 of the supply voltage value.

5 DMA

5.1 DMA_SramToSram_IT

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).

6 EXTI

6.1 EXTI_Toggleled_IT

此样例演示了 GPIO 外部中断功能，按键（PA0）引脚上的每一个上升沿都会产生中断，中断函数中 LED 灯会翻转一次。

This example demonstrates the GPIO external interrupt function, where each rising edge on the key (PA0) pin generates an interrupt, and the LED light in the interrupt function flips once.

6.2 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 STOP0 模式；拉高 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 STOP0 mode; After pulling up the PA6 pin, the MCU wakes up and the LED light is in a flashing state.

7 FLASH

7.1 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.

7.2 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.

8 GPIO

8.1 GPIO_FastIO

本样例主要展示 GPIO 的 FAST IO 输出功能。FAST IO 速度可以达到单周期翻转速度。

This sample demonstrates the FAST IO output functionality of GPIO. FAST IO speed can achieve single-cycle toggling speed.

8.2 GPIO_Toggle

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 sample demonstrates GPIO output mode. It configures the LED pin as a digital output and toggles the LED pin level every 250ms. When the program runs, you can observe the LED blinking at a frequency of 2Hz.

9 I2C

9.1 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.

9.2 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 the host I2C communication through DMA, the slave using EEPROM peripheral chip P24C32, press the user button, the host first to the slave to write 15bytes of data for the 0x1 ~ 0xf, and then from the EEPROM 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 “always on” state.

9.3 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.

9.4 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.

9.5 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.

9.6 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.

10 I2S

10.1 I2S_TwoBoards_Com_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.

10.2 I2S_TwoBoards_Com_IT

此样例是对I2S主机与I2S从机以中断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 interrupt. 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.

10.3 I2S_TwoBoards_Com_Polling

此样例是对I2S主机与I2S从机以轮询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 polling. 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.

11 IWDG

11.1 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 1000ms 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 900ms，程序能一直正常运行（LED 灯闪烁），如果喂狗时间 1100ms，程序会一直复位（LED 灯熄灭）。

This example demonstrates the function of IWDG (Independent Watchdog).Set IWDG to count 1000ms 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 900ms, the program can always run normally (LED blink), if the time is 1000ms, the program will always reset (LED off).

12 LCD

12.1 LCD_Display

此样例演示了对单色无源液晶显示器（LCD）的操作。将偏置产生电路配置为内部电阻分压，以使 LCD 全显，并显示“88:88”字样。

This sample demonstrates the operation of a monochrome passive liquid crystal display (LCD). The biasing circuit is configured with internal resistor division to ensure full display on the LCD, showing the text "88:88".

13 LPTIM

13.1 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.

13.2 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.

14 LPUART

14.1 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 9600，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 9600, with data bit 8, stop bit 1, and parity bit None.

14.2 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 9600，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 960000, with data bit 8, stop bit 1, and parity bit None.

14.3 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 9600，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 9600, with data bit 8, stop bit 1, and parity bit None.

15 OPA

15.1 OPA_VoltageFollow

此样例演示了 OPA 的电压跟随功能，PC2 为正端输入，PC1 为负端输入，PC8 为输出，PC8 会输出和 PC2 相同的电压值。

This sample demonstrates the voltage follower functionality of the OPA. PC2 is the positive input, PC1 is the negative input, and PC8 is the output. PC8 will output the same voltage as PC2.

16 PWM

16.1 PWM_PWM1

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%，本例程频率为 8000000/50/800=200Hz

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%.The frequency is 8000000/50/800=200Hz

17 PWR

17.1 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.

17.2 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.

17.3 PWR_PVD

此样例演示了 PVD 电压检测功能。当供电电压低于 3.1V 时，LED 会点亮，高于 3.1V 时，LED 灯会熄灭。

This sample demonstrates the PVD (Programmable Voltage Detector) voltage detection functionality. When the supply voltage is lower than 3.1V, the LED will light up. When the supply voltage is higher than 3.1V, the LED will turn off.

17.4 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.

17.5 PWR_STANDBY

此样例演示了 standby 模式下，通过 wakeuppin 唤醒功能。

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

17.6 PWR_STOP0_WFE

此样例演示了在 stop0 模式下，使用 GPIO 事件唤醒。

This example demonstrates using GPIO Event wake-up in stop0 mode.

17.7 PWR_STOP0_WFI

此样例演示了在 stop0 模式下，使用 GPIO 中断唤醒。

This example demonstrates using GPIO interrupt wake-up in stop0 mode.

18 RCC

18.1 RCC_HSEOutput

此样例配置系统时钟为 HSE，并通过 MCO1（PA14）引脚输出。

This sample configures the system clock as HSE and outputs it through the MCO (PA14) pin.

18.2 RCC_HSIOutput

此样例配置系统时钟为 HSISYS，并通过 MCO1（PA14）引脚输出。

This sample configures the system clock as HSISYS and outputs it through the MCO1 (PA14) pin.

18.3 RCC_LSEOutput

此样例使能 LSE，并通过 MCO1（PA14）引脚输出。

This sample enables the LSE and is output via the MCO1 (PA14) pin.

18.4 RCC_LSIOutput

此样例使能 LSI，并通过 MCO（PA14）引脚输出。

This sample enables the LSI and is output via the MCO (PA14) pin.

18.5 RCC_MSIOutput

此样例配置系统时钟为 MSISYS，并通过 MCO1（PA14）引脚输出。

This sample configures the system clock as HSISYS and outputs it through the MCO1 (PA14) pin.

18.6 RCC_PLLOutput

此样例配置系统时钟为 PLL，并通过 MCO1（PA14）引脚输出，PLL 的输入时钟源选择 HSI。

This sample configures the system clock as PLL and outputs it through the MCO1 (PA14) pin. The input clock source for the PLL is set to HSI.

19 RTC

19.1 RTC_Alarm_IT

此样例演示 RTC 的闹钟中断功能。在闹钟中断函数中会打印当前闹钟时间及字符“RTC_IT_ALARMA”。

This sample demonstrates the RTC's alarm interrupt functionality. In the alarm interrupt function, the current alarm time and the character "RTC_IT_ALARMA" will be printed.

19.2 RTC_Calendar

此样例演示 RTC 的日历功能。当前时间和日期更新在全局变量 aShowTime 和 aShowDate 中。

This example demonstrates the calendar function of the RTC. The current time and date are updated in the global variables aShowTime and aShowDate.

19.3 RTC_Tamper

此样例演示外部入侵事件擦除备份寄存器功能。在 TAMP_IN 引脚(PA8)产生下降沿入侵事件可擦除备份寄存器中的数据。

This example demonstrates the function of erasing backup registers by external tamper events. A tamper event generated on the TAMP_IN pin (PA8) with a falling edge can erase the data in the backup registers.

19.4 RTC_TimeStamp

此样例演示 RTC 的时间戳功能。

This example demonstrates the timestamp function of RTC.

19.5 RTC_WakeUpTimer

此样例演示通过 RTC 唤醒定时器中断每隔 1s 将 MCU 从 STOP0 模式下唤醒，每次唤醒会翻转 LED，LED 翻转间隔为 1s。

This example demonstrates waking up the MCU from STOP0 mode every 1 second using an RTC WakeUpTimer interrupt. Each wake-up will flip the LED, with an interval of 1 second between LED flips.

20 SPI

20.1 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.

20.2 SPI_TwoBoards_Com_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.

20.3 SPI_TwoBoards_Com_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.

20.4 SPI_TwoBoards_Com_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.

21 TIM

21.1 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

21.2 TIM1_AutoReloadPreload

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 2.5HZ 的频率翻转

This sample demonstrates base count function of the timer,and show ARR register autoreload function.Example toggle LED in timer update interrupt. Modify in main.c. Set TimHandle.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_ENABLE to enable autoreload,and new ARR value will takes effect on the fourth interrupt generate. Set TimHandle.Init.AutoReloadPreload = TIM_AUTORELOAD_PRELOAD_DISABLE to disable autoreload,and new ARR value will takes effect on the third interrupt generate. After taking effect, the LED lights blinked at a frequency of 2.5HZ.

21.3 TIM1_ComplementarySignals

此样例实现了定时器的互补输出功能，三组互补共六路 pwm 输出，此样例没有实现死区功能 CH1 -> PB12CH1N -> PB13CH2 -> PC13CH2N -> PC11CH3 -> PC05CH3N -> PC06

This sample demonstrates complementary output function of the timer,Three sets of complementary outputs total six pwm outputs, this example does not implement the dead zone function

21.4 TIM1_ComplementarySignals_break

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。CH1 -> PB12CH1N -> PB13 刹车输入 -> PA13 通过调整 OCxE,CCxP,OISx,CCxNE,CCxNP,OISxN 的配置，可实现刹车功能的各种应用

This sample demonstrates brake function of the timer,the CH1 and CH1N complementary pwm outputs.After receiving the brake signal (low level)from the external IO port, the PWM signal is turned off. Because BDTR.AOE is set, the pwm output continues after the brake signal is cancelled (high level). This example realizes the dead zone function CH1 -> PB12 CH1N -> PB13 Brake input -> PA13 By adjusting the OCxE, CCxP, OISx, CCxNE, CCxNP, OISxN configuration, which can realize the brake function of a variety of applications

21.5 TIM1_ComplementarySignals_break_it

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。本样例开启了刹车中断，并在刹车中断里翻转 LED 灯通过调整 OCxE,CCxP,OISx,CCxNE,CCxNP,OISxN 的配置，可实现刹车功能的各种应用

This sample demonstrates brake function of the timer,the CH1 and CH1N complementary pwm outputs.After receiving the brake signal (low level)from the external IO port, the PWM signal is turned off. Because BDTR.AOE is set, the pwm output continues after the brake signal is cancelled (high level). This example realizes the dead zone function. This example turns on the brake interrupt and toggle the LED light in the brake interrupt. By adjusting the OCxE, CCxP, OISx, CCxNE, CCxNP, OISxN configuration, which can realize the brake function of a variety of applications

21.6 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。通过调整 OCxE,CCxP,OISx,CCxNE,CCxNP,OISxN 的配置，可实现刹车功能的各种应用

This sample demonstrates brake function of the timer,the CH1 and CH1N complementary pwm outputs.After receiving the brake signal (low level)from the external IO port, the PWM signal is turned off. Because BDTR.AOE is set, the pwm output continues after the brake signal is cancelled (high level). This example realizes the dead zone function. By adjusting the OCxE, CCxP, OISx, CCxNE, CCxNP, OISxN configuration, which can realize the brake function of a variety of applications

21.7 TIM1_DmaBurst_twice

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 每传输一次更新三个寄存器，PSC,ARR,RCR，在更新事件中断中，PA0 会进行翻转，通过逻辑分析仪监测，可看到 PA0 的翻转间隔会从第一次的 400ms，第二次 400ms，第三次 20ms,第四次及后续变为 200us，此时两次 burst 传输完成，并且 PCS,ARR,RCR 均更新完毕。

This sample demonstrates the function to transfer data in TIM1 using DMA in two consecutive bursts.burst updates three registers(PSC,ARR,RCR) per transfer.In the interruption of update event, PA0 will be flipped. Through the monitoring of logic analyzer, it can be seen that the flipping interval of PA0 will change from 400ms for the first time, 400ms for the second time, 20ms for the third time, and 200us for the fourth and subsequent times. At this time, the two burst transmission is completed, and PCS,ARR and RCR are all updated.

21.8 TIM1_EncoderTI2AndTI1

此样例实现了 TIM1 中的编码器计数功能，TI1(PB12)和 TI2(PC13)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(PB12) and TI2(PC13) 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.

21.9 TIM1_ExternalClockMode1

此样例演示了 TIM1 的外部时钟模式 1 功能，选择 ETR(PA11)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 external clock mode 1 function of the TIM1.Select the ETR(PA11) pin as the external clock input source and enable the update interrupt to flip the LED light in the interrupt.

21.10 TIM1_ExternalClockMode1_TI1F

此样例演示了 TIM1 的外部时钟模式 1 功能，选择 TI1FD(PB12)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(PB12) pin as the external clock input source, and enables the update interrupt and toggle the LED light in the interrupt

21.11 TIM1_ExternalClockMode2

此样例演示了 TIM1 的外部时钟模式 2 功能，选择 ETR(PA11)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 2 function of TIM1, selects the ETR(PA11) pin as the external clock input source, and enables the update interrupt and toggle the LED light in the interrupt

21.12 TIM1_InputCapture_TI1FP1

此样例演示了在 TIM1(PB12)输入捕获功能, PB12 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(PB12), PB12 input clock signal, when TIM1 capture success, will enter the capture interrupt,and toggle the LED in the interrupt

21.13 TIM1_InputCapture_XORCh1Ch2Ch3

此样例演示了在 TIM1 输入捕获功能，PB12 或 PC13 或 PC5 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, input clock signal into PB12 or PC13 or PC5 will generate the capture interrupt after TIM1 capture successfully.Toggle the LED once per interruption

21.14 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(PB12),CH2(PC13),CH3(PC05),CH4(PA01)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 100KHz

This sample demonstrates the OC toggle output function in TIM1 comparison mode, enabling CH1(PB12),CH2(PC13),CH3(PC05),CH4(PA01) four channel output function, then the output signal toggle when the counter TIMx_CNT matches TIMx_CCRx. The frequency is 100KHz

21.15 TIM1_OnePulseOutput

此样例演示了 TIM1 的单脉冲模式，CH2(PC13)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(PB12)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，本例程脉冲宽度计算 (TIM1_ARR-TI1_CCR1)/CLK= (65535-16383) /8000000=6.144ms

This sample demonstrates the one pulse mode of TIM1.The rising edge on the CH2(PC13) pin triggers the counter to start counting. when the count value matches CCR1,CH1(PB12) outputs a high level. When the counter overflows ,CH1 outputs the low level again. After the counter overflows, the timer stops working. This example pulse width calculation (TIM1_ARR-TI1_CCR1)/CLK= (65,535-16383)/ 8,000,000 =6.144ms

21.16 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 8000000/50/800=200Hz

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%.The period is 8000000/50/800=200Hz

21.17 TIM1_SynchronizationEnable

定时器 1 的使能由定时器 2 控制，当定时器 2 计数时，LED 会常亮，当定时器 2 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 5Hz 的频率进行翻转

The enable of TIM1 is controlled by TIM2. When TIM2 counts, the LED will be steady on. The update event generated by TIM2 will triggers TIM1, and when TIM1 starts counting, the LED is toggled at a frequency of 5Hz

21.18 TIM1_TIM2_Cascade

此样例实现了 TIM1 和 TIM2 级联成 48 位计数器，TIM2 做主机，TIM2 的计数溢出信号作为 TIM1 的输入时钟，通过配置 TIM1 和 TIM2 的重载寄存器值，(在 TIM1 中断回调函数中）实现 LED 灯以 0.5Hz 频率闪烁。

This example realizes the cascade of TIM1 and TIM2 into a 48-bit counter, with TIM2 as the host.The count overflow signal of TIM2 acts as the input clock of TIM1.By configuring the reloaded register values of TIM1 and TIM2, the LED is toggled at 0.5Hz (in the TIM1 interrupt callback function).

21.19 TIM1_Update_DMA

此样例演示了在 TIM1 中使用 DMA 传输数据的功能，通过 DMA 从 SRAM 中搬运数据到 ARR 寄存器，实现 TIM1 周期变化，在 TIM1 第一次溢出后，PA0 会翻转，此时翻转间隔为 400ms，DMA 开始搬运数据到 TIM1_ARR，第一次 PA0 翻转间隔为 400ms，第二次翻转间隔为 100ms，第三次翻转间隔为 200ms，第四次翻转间隔为 300ms，此时 DMA 搬运结束，后续翻转间隔均为 300ms

This sample demonstrates the function of using DMA to transfer data in TIM1, carrying data from SRAM to ARR register by DMA to achieve TIM1 cycle change. After the first overflow of TIM1, PA0 will toggle, at this time the toggle interval is 400ms. DMA starts to carry data to TIM1_ARR, the first PA0 toggle interval is 400ms, the second toggle interval is 100ms, the third toggle interval is 200ms, the fourth toggle interval is 300ms, at this time the DMA carrying ends, the subsequent toggle interval are 300ms

21.20 TIM1_Update_IT

此样例演示了在 TIM1 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 灯会以 5Hz 的频率进行翻转。

This sample demonstrates basic count function of the TIM1 and enable update interrupt.Each time an update interrupt is generated, the ARR value is reloaded and the LED light is toggled in the interrupt.The LED light is toggled at a frequency of 5Hz.

22 UART

22.1 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

22.2 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

22.3 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

23 USART

23.1 SCI_HyperTerminal_AutoBaund_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".

23.2 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

23.3 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

23.4 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

23.5 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.

24 WWDG

24.1 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 generates an interrupt.Refresh the WWDG in interrupt to ensure that the WWDG does not reset.

24.2 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.