



Open103R 示例程序(基于 HAL 库)说明

程序说明

关于 STM32CubeMx

通过 STM32CubeMX 生成的基于 HAL 库的程序源代码,相较于经典库的代码而言,更加合理和清晰。STM32CubeMX 配置界面是图形化的,参数设置相比于直接修改源代码更为直观。

但是,用户对所需资源进行配置之后,经由 STM32CubeMX 生成的代码并不是直接可用,还需要 在/* USER CODE BEGIN */和/* USER CODE END */之间中加入相应的逻辑。

每次在 STM32CubeMX 对程序的配置进行修改之后,生成的新代码会覆盖旧的代码。但是用户写在/* USER CODE BEGIN */和/* USER CODE END */之间的代码会得到保留。

代码编译环境

目前(2016 年 12 月)STM32CubeMX 可以根据 IDE 生成 EWARM / MDK-ARM / TrueSTUDIO / SW4STM32 等工程。这可以通过 Project → Settings → Toolchain / IDE 进行设置。

我们提供的示例程序都是通过 MDK-ARM (Keil 5)进行编译,并且验证通过。如果打算直接使用 MDK-ARM 工程,那么电脑必须安装 Keil 5(或以上)。连接好下载器之后,打开 Keil,点击的 Build 编译工程,然后点击 Download 按钮,就可以把编译好的程序写入到芯片。下面所有的示例程序 使用过程的说明,均不会对开发环境配置和程序下载进行赘述。如有需要,请参阅相关手册。



目录

程序说明1
关于 STM32CubeMx1
代码编译环境1
目录2
GPIO
EXTI4
TIM
PWM5
USART
USART_DMA5
USART_IT6
USART_printf6
ADC_DMA7
DAC
I2C
SPI
CAN
DS18B209
RTC
MCU_Temperature10
IWDG11
2



WWDG1	1
CRC1	.2
SDIO1	.2
FATFS1	.3
LCD_Display1	.3
LCD_Touch1	.3
USB_FS1	4
CDC_Standalone1	.4
HID_Standalone1	.4
MSC_Standalone1	.5
FreeRTOS1	.5
uCOS_III1	.5



GPIO

描述:

此程序演示如何通过 HAL 库,设置和使用 GPIO。

工程位置:

STM32CubeMX 工程位置: GPIO\GPIO.ioc Keil 工程位置: GPIO\MDK-ARM\GPIO.uvprojx

预期结果:

- 1) 插上 LED JMP 跳线帽(4个)。
- 2) 拨动 5 向摇杆或者按下 USER KEY 键, 会点亮对应的 LED 指示灯。

EXTI

描述:

此程序演示通过外部中断(按键)的方式执行回调函数。

工程位置:

STM32CubeMX 工程位置: EXTI\EXTI.ioc Keil 工程位置: EXTI\MDK-ARM\EXTI.uvprojx

预期结果:

- 1) 插上 LED1 的跳线帽。
- 2) 按下 USER KEY 按键,便可切换 LED1 指示灯的状态。

TIM

描述:

此程序演示通过定时器中断的方式执行回调函数。

工程位置:

STM32CubeMX 工程位置: TIM\TIM.ioc Keil 工程位置: TIM\MDK-ARM\TIM.uvprojx



预期结果:

- 1) 插上 LED JMP 跳线帽(4个)。
- 2) LED1 / LED2 / LED3 / LED4 指示灯闪烁。

PWM

描述:

此程序演示通过 STM32 芯片的 PWM 定时器,对输出电压进行控制。

工程位置:

STM32CubeMX 工程位置: PWM\PWM.ioc Keil 工程位置: PWM\MDK-ARM\PWM.uvprojx

预期结果:

- 1) 插上 LED1 的跳线帽。
- 2) LED 指示灯亮度逐渐变化。

USART

USART_DMA

描述:

此程序演示以直接内存存取(DMA)的方式,通过板载的 USART 接口连续输出信息。

工程位置:

STM32CubeMX 工程位置: USART_DMA\USART_DMA.ioc Keil 工程位置: USART_DMA\MDK-ARM\USART_DMA.uvprojx

- 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB 口。其 中模块的 Rx → 开发板的 Tx,开发板的 Tx → 模块的 Rx。
- 电脑安装串口查看软件,并设置串口:波特率 115200,数据位 8,停止位 1,校验位无, 流控制无。
- 3) 开启串口之后,可以查看到连续的串口信息打印。



USART_IT

描述:

此程序演示以中断的方式,通过板载的 USART 接口接收信息。

工程位置:

STM32CubeMX 工程位置: USART_IT\USART_IT.ioc Keil 工程位置: USART_IT\MDK-ARM\USART_IT.uvprojx

预期结果:

- 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB 口。其 中模块的 Rx → 开发板的 Tx,开发板的 Tx → 模块的 Rx。
- 电脑安装串口查看软件,并设置串口:波特率115200,数据位8,停止位1,校验位无, 流控制无。
- 开启串口之后,电脑端通过串口连续发送 10 个字符,这些字符会通过串口重新打印出 来。

USART_printf

描述:

此程序演示以轮询的方式,通过板载的 USART 接口输出信息。

工程位置:

STM32CubeMX 工程位置: USART_printf\USART_printf.ioc Keil 工程位置: USART_printf\MDK-ARM\USART_printf.uvprojx

- 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB 口。其 中模块的 Rx → 开发板的 Tx,开发板的 Tx → 模块的 Rx。
- 电脑安装串口查看软件,并设置串口:波特率115200,数据位8,停止位1,校验位无, 流控制无。
- 3) 开启串口之后,可以查看到串口信息打印,每隔一秒钟打印一次。



ADC_DMA

描述:

此程序演示通过以直接内存存取(DMA)的方式,连续进行 AD 采集。

工程位置:

STM32CubeMX 工程位置: ADC_DMA\ADC_DMA.ioc Keil 工程位置: ADC_DMA\MDK-ARM\ADC_DMA.uvprojx

预期结果:

- 1) 将 Analog Test Board 模块接入 ADC+DAC (SPI-1) 接口。
- 2) 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB □。其
 中模块的 Rx → 开发板的 Tx, 开发板的 Tx → 模块的 Rx。
- 电脑安装串口查看软件,并设置串口:波特率 115200,数据位 8,停止位 1,校验位无, 流控制无。
- 4) 转动 Analog Test Board 模块上面的两个电位器旋钮,串口会打印读到的 AD 信息:

```
******* ADC DMA Example ******
AD1 value = 3.298V
AD2 value = 1.647V
******* ADC DMA Example *******
AD1 value = 3.298V
AD2 value = 1.647V
```

DAC

描述:

此程序演示通过 STM32 芯片的 DMA 通道输出三角波信号。

工程位置:

STM32CubeMX 工程位置: DAC\DAC.ioc Keil 工程位置: DAC\MDK-ARM\DAC.uvprojx

预期结果:

1) 将 Analog Test Board 模块接入 ADC+DAC (SPI-1) 接口。



- 2) Analog Test Board 模块上的 5V 管脚接到板子上任意 5V 输出管脚上。
- 3) Analog Test Board 模块会发出声音。

120

描述:

此程序演示通过 I2C 接口读写 AT24 EEPROM。

工程位置:

STM32CubeMX 工程位置: I2C-AT24C04\I2C4.ioc Keil 工程位置: I2C-AT24C04\MDK-ARM\I2C-AT24C04.uvprojx

预期结果:

- 1) 将 AT24/FM24 Board 模块接到 I2C1 口上。
- 2) 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB 口。其中模块的 Rx → 开发板的 Tx,开发板的 Tx → 模块的 Rx。
- 3) 电脑安装串口查看软件,并设置串口:波特率→115200,数据位→8,停止位→1,校验 位→无,流控制→无。
- 按下板子的复位按键之后, EEPROM 会被写入数据, 然后被读出数据。数据会通过串口 打印出来。

SPI

描述:

此程序演示通过 SPI 接口读写 AT45DBXX Flash。

工程位置:

STM32CubeMX 工程位置: SPI-AT45DBXX\SPI-AT45DBXX.ioc Keil 工程位置: SPI-AT45DBXX\MDK-ARM\SPI-AT45DBXX.uvprojx

- 1) 将 AT45DBXX DataFlash Board 模块接到 SPI-1 接口上。
- 2) 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB □。其
 中模块的 Rx → 开发板的 Tx, 开发板的 Tx → 模块的 Rx。



- 电脑安装串口查看软件,并设置串口:波特率→115200,数据位→8,停止位→1,校验 位→无,流控制→无。
- 按下板子的复位按键之后,Flash 会被写入数据,然后被读出数据。数据会通过串口打印出来。

CAN

描述:

CAN 回环测试。

工程位置:

STM32CubeMX 工程位置: CAN-LoopBack\CAN.ioc Keil 工程位置: CAN-LoopBack\MDK-ARM\CAN.uvprojx

预期结果:

- 1) 插上 LED1 和 LED2 的跳线帽。
- 2) 如果回环测试成功,则 LED1 点亮,否则 LED2 点亮。

DS18B20

描述:

此程序演示通过 ONE WIRE 接口读取 DS18B20 温度传感器数据。

工程位置:

STM32CubeMX 工程位置: DS18B20\DS18B20.ioc Keil 工程位置: DS18B20\MDK-ARM\DS18B20.uvprojx

- 1) 将 DS18B20 温度传感器插上 ONE WIRE 接口。
- 2) 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB □。其
 中模块的 Rx → 开发板的 Tx, 开发板的 Tx → 模块的 Rx。
- 3) 电脑安装串口查看软件,并设置串口:波特率→115200,数据位→8,停止位→1,校验 位→无,流控制→无。
- 4) 串口将会打印 DS18B20 检测到的温度。



RTC

描述**:**

此程序演示如何通过 HAL 库,设置和使用 RTC。

工程位置:

STM32CubeMX 工程位置: RTC\RTC.ioc Keil 工程位置: RTC\MDK-ARM\RTC.uvprojx

预期结果:

- 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB 口。其 中模块的 Rx → 开发板的 Tx,开发板的 Tx → 模块的 Rx。
- 2) 电脑安装串口查看软件,并设置串口:波特率→115200,数据位→8,停止位→1,校验 位→无,流控制→无。
- 3) 串口将会打印时间信息。

MCU_Temperature

描述:

此程序演示如何通过 Temperature Sensor Channel,读取 MCU 的温度。

工程位置:

STM32CubeMX 工程位置: MCU_Temperature\MCU_Temperature.ioc Keil 工程位置: MCU_Temperature\MDK-ARM\MCU_Temperature.uvprojx

- 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB 口。其 中模块的 Rx → 开发板的 Tx,开发板的 Tx → 模块的 Rx。
- 2) 电脑安装串口查看软件,并设置串口:波特率→115200,数据位→8,停止位→1,校验 位→无,流控制→无。
- 3) 串口将会打印 MCU 温度信息。



IWDG

描述:

此程序演示如何使用 IWDG(独立看门狗)。

工程位置:

STM32CubeMX 工程位置: IWDG\IWDG.ioc Keil 工程位置: IWDG\MDK-ARM\IWDG.uvprojx

预期结果:

- 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB 口。其 中模块的 Rx → 开发板的 Tx,开发板的 Tx → 模块的 Rx。
- 2) 电脑安装串口查看软件,并设置串口:波特率→115200,数据位→8,停止位→1,校验 位→无,流控制→无。
- 3) 每隔一段时间看门狗计数器会更新。如果删除 HAL_IWDG_Refresh 函数,导致计数器没 有及时更新的话,那么程序将会复位。
- 4) 如果及时更新看门狗的话,串口将会打印"IWDG is refreshed",否则程序复位。

WWDG

描述:

此程序演示如何使用 WWDG (窗口看门狗)。

工程位置:

STM32CubeMX 工程位置: WWDG\WWDG.ioc Keil 工程位置: WWDG\MDK-ARM\WWDG.uvprojx

- 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB 口。其 中模块的 Rx → 开发板的 Tx,开发板的 Tx → 模块的 Rx。
- 2) 电脑安装串口查看软件,并设置串口:波特率→115200,数据位→8,停止位→1,校验 位→无,流控制→无。
- 3) 每隔一段时间看门狗计数器会更新。如果删除 HAL_WWDG_Refresh 函数,导致计数器 没有及时更新的话,那么程序将会复位。



4) 如果及时更新看门狗的话,串口将会打印"WWDG is refreshed",否则程序复位。

CRC

描述:

此程序演示如何通过 HAL 库,设置和使用 CRC 校验。

工程位置:

STM32CubeMX 工程位置: CRC\CRC.ioc Keil 工程位置: CRC\MDK-ARM\CRC.uvprojx

预期结果:

- 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB 口。其 中模块的 Rx → 开发板的 Tx,开发板的 Tx → 模块的 Rx。
- 2) 电脑安装串口查看软件,并设置串口:波特率→115200,数据位→8,停止位→1,校验 位→无,流控制→无。
- 3) 如果 CRC 校验成功,串口将会打印"CRC right value",否则打印"CRC wrong value"。

SDIO

描述:

此程序演示如何通过 HAL 库读写 SD 卡。

工程位置:

STM32CubeMX 工程位置: SDIO\SDIO.ioc Keil 工程位置: SDIO\MDK-ARM\SDIO.uvprojx

- 1) 请对 Micro SD 卡进行备份,这个示例程序将会擦除 SD 卡的内容。
- 2) 把 Micro SD 卡插入到 MicroSD Card 模块。
- 3) 把 MicroSD Card 模块连接到 SDIO 接口。
- 4) 通过 USB 串口模块(例如 PL2303 模块)连接板载的 USART1 接口和电脑的 USB □。其
 中模块的 Rx → 开发板的 Tx,开发板的 Tx → 模块的 Rx。
- 5) 电脑安装串口查看软件,并设置串口:波特率→115200,数据位→8,停止位→1,校验 位→无,流控制→无。



6) 串口将会打印写入、读取和擦除 SD 的操作状态。

FATFS

描述:

此程序演示如何通过第三方中间件读写 SD 卡。

工程位置:

STM32CubeMX 工程位置: FATFS\FATFS.ioc Keil 工程位置: FATFS\MDK-ARM\FATFS.uvprojx

预期结果:

- 1) 请对 Micro SD 卡进行备份,这个示例程序将会擦除 SD 卡的内容。
- 2) 把 Micro SD 卡插入到 MicroSD Card 模块。
- 3) 把 MicroSD Card 模块连接到 SDIO 接口。
- 4) 按下板子的复位按键之后,一个 Hello.txt 文件会被写入到 SD 卡。

LCD_Display

描述:

此程序演示如何驱动 3.2 寸 LCD 模块。

工程位置:

STM32CubeMX 工程位置: LCD\LCD.ioc Keil 工程位置: LCD\MDK-ARM\LCD.uvprojx

预期结果:

- 1) 把 3.2inch 320x240 Touch LCD (C)模块插入到 LCD 接口(FSMC+SPI 接口)。
- 2) LCD 上显示字符信息。

LCD_Touch

描述:

此程序演示如何驱动 3.2 寸 LCD 模块,并通过 SPI 接口实现 LCD 触摸功能。



工程位置:

STM32CubeMX 工程位置: TOUCH\TOUCH.ioc Keil 工程位置: TOUCH\MDK-ARM\TOUCH.uvprojx

预期结果:

- 1) 把 3.2inch 320x240 Touch LCD (C)模块插入到 LCD 接口(FSMC+SPI 接口)。
- 首先校准触摸屏,点击3次[+]即可完成校准。之后就会进入触摸屏画板界面。在触摸屏 画板中,你可以在随意画线。

USB_FS

CDC_Standalone

描述:

此程序演示如何使用 CDC 类(Communications Device Class),实现 USB 设备之间的通信。

工程位置:

STM32CubeMX 工程位置: USB_Device\CDC_Standalone\CDC_Standalone.ioc Keil 工程位置: USB_Device\CDC_Standalone\MDK-ARM\CDC_Standalone.uvprojx

预期结果:

- 1) 把 3.2inch 320x240 Touch LCD (C)模块插入到 LCD 接口(FSMC+SPI 接口)。
- 2) 通过 USB 线连接开发板上的 USB 接口和电脑的 USB 接口。
- 3) 打开设备管理器,可以查看到"STMicroelectronics Virtual COM Port"。
- 4) 解压并安装 CDC_Standalone\Software\stsw.zip 串口驱动。
- 5) 电脑安装串口查看软件,这里无需对串口进行设置。因为开发板的串口接收和发送端实际上都是由 USB 进行模拟,因此波特率等一系列设置实际上是没有意义的。
- 6) 电脑端通过串口发送信息,发送的信息会显示在 LCD 上面。

HID_Standalone

描述:

此程序演示如何基于 STM32 设备的人机接口(Human Interface Device)使用 USB 设备。

工程位置:



STM32CubeMX 工程位置: USB_Device\HID_Standalone\HID_Standalone.ioc Keil 工程位置: USB_Device\HID_Standalone\MDK-ARM\HID_Standalone.uvprojx

预期结果:

- 1) 通过 USB 线连接开发板上的 USB 接口和电脑的 USB 接口。
- 2) 打开设备管理器 → 人体学输入设备,可以查看到新增的"USB 输入设备"。
- 3) 拨动 5 向摇杆或者按下 USER KEY 键, 会电脑的光标会发生移动。

MSC_Standalone

描述:

此程序演示如何基于 MSC 类(Mass Storage Class)使用 USB 设备。

工程位置:

```
STM32CubeMX 工程位置: USB_Device\MSC_Standalone\MSC_Standalone.ioc
Keil 工程位置: USB_Device\MSC_Standalone\MDK-ARM\MSC_Standalone.uvprojx
```

预期结果:

- 1) 把带有 SD 卡的 Micro SD Storage Board 模块插入到 SDIO 接口。
- 2) 通过 USB 线连接开发板上的 USB 接口和电脑的 USB 接口。
- 3) 电脑会识别到新增的可移动磁盘。

FreeRT0S

描述:

这些程序都是运行在 STM32cubeMX 软件生成的 FreeRTOS 系统上面的。提供了多达 11 个 RTOS 程序,包括多线程,互斥锁,队列,信号,邮件,定时器等例程。

所有的实验结果都是通过板载的指示灯进行反馈,无需外接其他的器件。具体反馈的信息请 参阅程序源代码的注释。

uCOS_III

描述:

本程序是基于 STM32cubeMX 软件生成的 HAL 库、进行移植的 uCOS III 操作系统。



工程位置:

STM32CubeMX 工程位置: uCOS III\uCOS III.ioc Keil 工程位置: uCOS III\MDK-ARM\uCOS III.uvprojx

预期结果:

- 1) 插上 LED1 的跳线帽。
- 2) LED1 指示灯闪烁。

注意事项:

如果通过 STM32CubeMX 重新生成工程,请注意需要修改 startup_stm32f103xe.s 文件,把所 有的 PendSV_Handler 修改为 OS_CPU_PendSVHandler,而且还要把所有的 SysTick_Handler 修 改为 OS_CPU_SysTickHandler。