**STM32CubeMX--01--UART**

-------- 转载请注明出处

-------- 更多笔记请访问我的博客：merafour.blog.163.com

-------- 2016-11-24.冷月追风

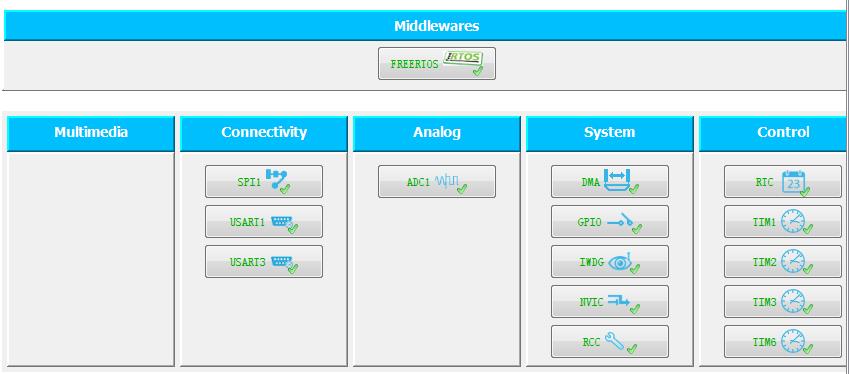
-------- email：merafour@163.com

**注：不特别声明的情况下不区分 UART跟 USART，统称 UART。**

前段时间开发看到别人的代码里边好多 "**HAL**"开头的函数，后来网上查了才知道原来这是 HAL库，区别于我们之前所使用的标准库。HAL通过使用 STM32CubeMX图形化配置，并且自动生成初始化代码。其对底层的封装是标准库所不能比的。

当然，封装有好处，但也有坏处。好处是对于底层硬件细节你不必再那么关心，掌握API即可开发。而坏处在于由于对底层进行抽象中间增加了很多代码，效率降低，代码大小增加。如果你需要写一个实时性比较高的程序，那么你还是得跳过 HAL直接操作底层，这个时候我就会通过 HAL来初始化硬件。

使用 STM32CubeMX的另一个好处是：



STM32CubeMX为我们提供了 FreeRTOS，所以当我们需要使用操作系统的时候我们可以很方便地把 FreeRTOS添加到我们的工程，而我们需要做的只是勾选：



FreeRTOS的 Enabled，后面就可以直接使用 FreeRTOS了。关于 OS我们后面在介绍，今天我们来说说我们的 UART。

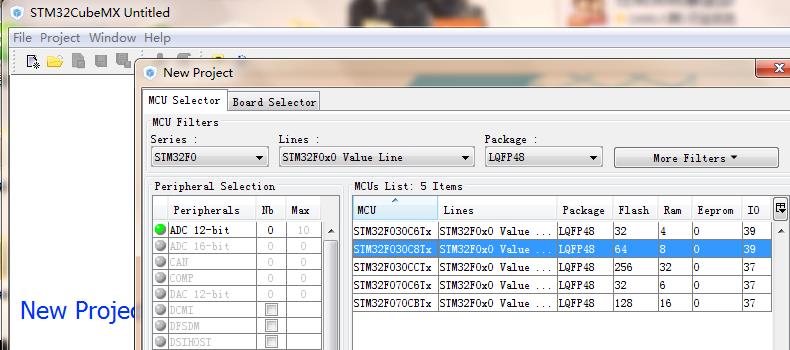
我们都知道，程序开发的第一课都是 "Hello World!"，而单片机开发的第一课都是点亮 LED，但是我们先不说这个，可以在 UART中点一下。

UART是我们做嵌入式开发过程中经常要用到的通讯接口，像现在比较流行的联网功能，通过添加一个 WiFi模块就可以进行互联，而单片机使用 USB的WiFi是吃不消的，这时候串口WiFi就很方便，通过 AT指令就可以操作，并且联网。下面我们就来了解下通过 STM32CubeMX如何使用串口。

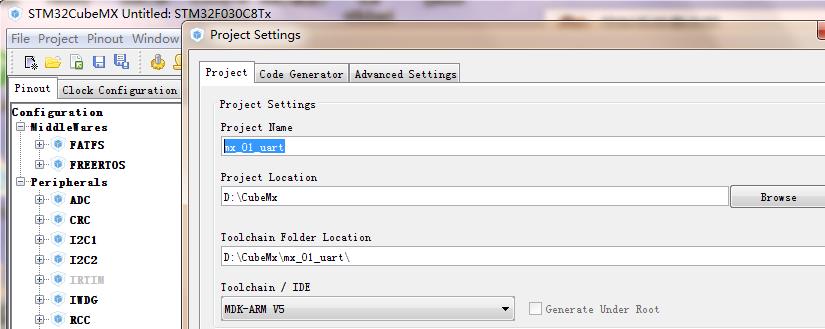
STM32CubeMX 的安装这里就先跳过了。

**1. New Project**

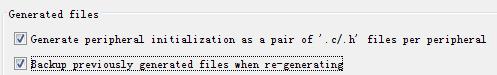
首先我们要通过 STM32CubeMX新建一个工程，根据自己的板子选择芯片：



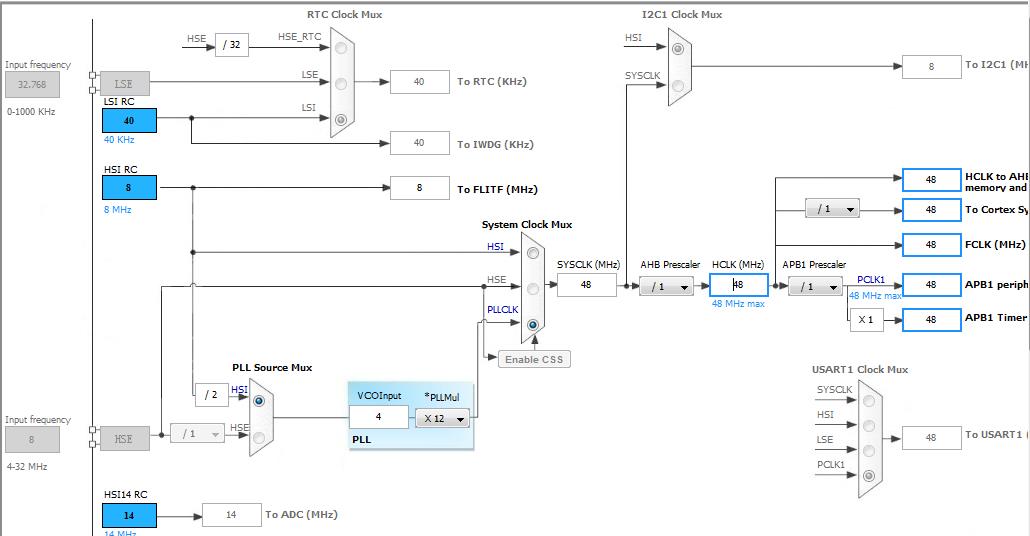
我的芯片为 STM32F030C8T6，选择上面的芯片。保存之后我们进入 Project-->Settings进行设置：



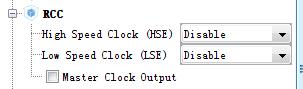
然后我们点击 "Code Generator"标签，勾选：



这个是个人建议，这样会将如 UART，SPI等单独房子 c源文件中，而且会自动备份相应的源码。保存之后我们先配置时钟：



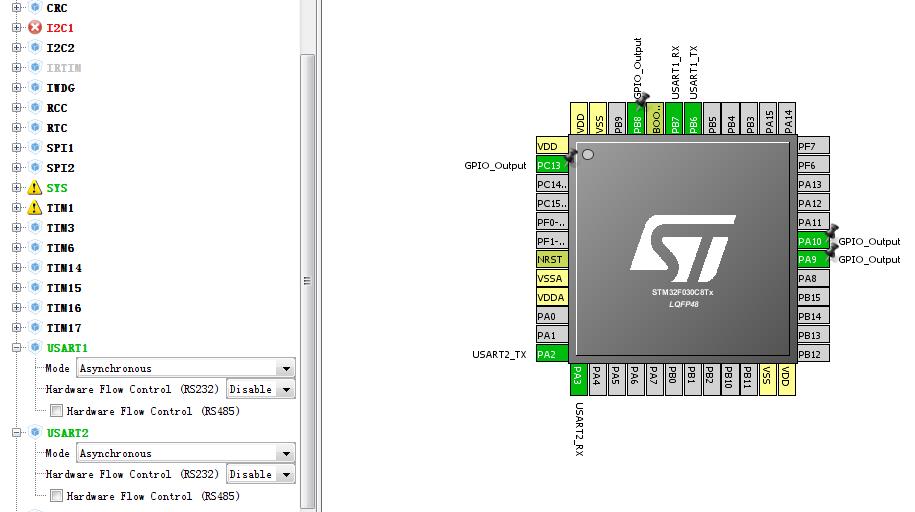
实际上我们直接在 HCLK这里直接输入48，然后敲回车就可以自动配置分频。如果大家要使用外部晶振在 RCC中先设置然后就可以在这里配置：



配置好时钟之后就保存。我在使用过程中发现 STM32CubeMX容易闪退，所以修改之后及时保存。

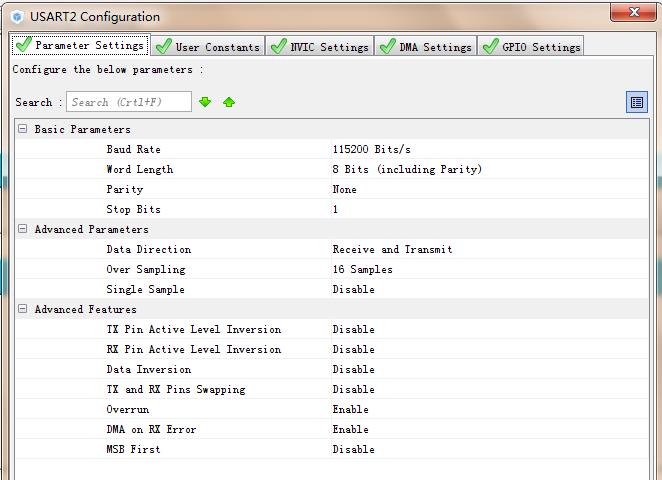
**2. 配置 UART**

接下来我们配置 UART，



当我们使用一个外设的时候会在右侧显示出占用的引脚，这很容易发现端口冲突。另外，UART1默认使用 PA9，PA10，当我把 PA9跟 PA10占用的时候，再使用 UART1会自动映射到 PB7，PB7，即能够自动解决冲突。PC13是我所使用的板子上的一个 LED。PB8是因为我所使用的板子上 UART接了一个外设，需要通过 PB8拉低将外设停止工作便于我们测试UART。

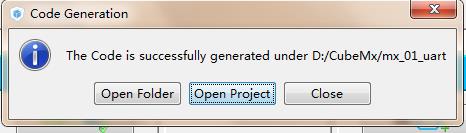
下面我们来对串口进行配置。



设置波特率之后我们的UART就可以工作了。

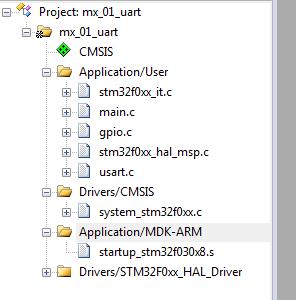
E34E42C935964B249F5CB4CAA83AEFAB.jpg

点击最右边这个按钮生成我们的代码，生成之后我们会看到：



直接打开工程，会自动调用我们安装好的编译器。

生成工程后打开我们可以看到器工程视图：



HAL下面就是我们的 HAL库。main函数为：

|  |
| --- |
| int main(void) {   /\* USER CODE BEGIN 1 \*/   /\* USER CODE END 1 \*/   /\* MCU Configuration----------------------------------------------------------\*/   /\* Reset of all peripherals, Initializes the Flash interface and the Systick. \*/   HAL\_Init();   /\* Configure the system clock \*/   SystemClock\_Config();   /\* Initialize all configured peripherals \*/   MX\_GPIO\_Init();   MX\_USART2\_UART\_Init();   MX\_USART1\_UART\_Init();   /\* USER CODE BEGIN 2 \*/   /\* USER CODE END 2 \*/   /\* Infinite loop \*/   /\* USER CODE BEGIN WHILE \*/   while (1)   {   /\* USER CODE END WHILE \*/   /\* USER CODE BEGIN 3 \*/   }   /\* USER CODE END 3 \*/ } |

然后我们增加串口测试代码：

|  |
| --- |
|  |

编译之后下载就可以测试了。

需要注意的是，HAL\_UART\_Transmit的第四个参数Timeout不能随便给，否则很容易挂掉。原因我想跟它的 \_\_HAL\_LOCK函数有关：

|  |
| --- |
| #define \_\_HAL\_LOCK(\_\_HANDLE\_\_) \                                 do{ \                                     if((\_\_HANDLE\_\_)->Lock == HAL\_LOCKED) \                                     { \                                        return HAL\_BUSY; \                                     } \                                     else \                                     { \                                        (\_\_HANDLE\_\_)->Lock = HAL\_LOCKED; \                                     } \                                   }while (0)   #define \_\_HAL\_UNLOCK(\_\_HANDLE\_\_) \                                   do{ \                                       (\_\_HANDLE\_\_)->Lock = HAL\_UNLOCKED; \                                     }while (0) |

虽说轮训的方式接收数据不存在抢占，但发送设置超时确实多接收几个数据就接收不到了。

正如你所看到的，我们值添加了几行代码就使用了串口，这就是使用 MX的便捷。不过凡事都是利弊共存的。

**3. LED**

下面我们来增加LED的功能。

我们只需将主循环改为：

|  |
| --- |
|  |

然后再定义 "uint16\_t flash=0;"，编译下载就可以看到 LED在闪烁了，而且发数据的时候闪烁频率加快，原因是收到数据 HAL\_UART\_Receive函数提前返回。

参考:

[STM32CubeMX](http://www.st.com/content/st_com/en/products/development-tools/software-development-tools/stm32-software-development-tools/stm32-configurators-and-code-generators/stm32cubemx.html): <http://www.st.com/content/st_com/en/products/development-tools/software-development-tools/stm32-software-development-tools/stm32-configurators-and-code-generators/stm32cubemx.html>

[STM32cube中文网](http://www.stm32cube.com/): <http://www.stm32cube.com/>

[STM32CubeMX介绍、下载与安装](http://blog.csdn.net/ybhuangfugui/article/details/52225736): <http://blog.csdn.net/ybhuangfugui/article/details/52225736>

[在Linux下使用STM32CubeMX开发和调试STM32](http://www.xlgps.com/article/387805.html): <http://www.xlgps.com/article/387805.html>