# 目录

[目录 1](#_Toc15920623)

[第1章 Cortex-M4基础 2](#_Toc15920624)

[1.1 Cortex-M4介绍 2](#_Toc15920625)

[1.2 STM32F407VGT6介绍 2](#_Toc15920626)

[1.3 STM32F407VGT6最小系统介绍 5](#_Toc15920627)

[1.4 STM32开发环境搭建 6](#_Toc15920628)

[1.5 STM32项目工程创建 6](#_Toc15920629)

# Cortex-M4基础

## Cortex-M4介绍

ARM处理的一些基础知识。

ARM处理器占有量比较大----80%左右。

2008年以前ARM处理没有分系列，ARM7/9/11等等。

2008年以后ARM处理器分为三个系列：A系列、M系列、R系列。

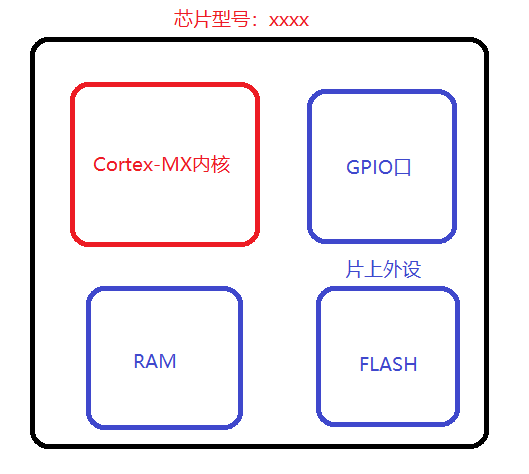
A系列处理器能力较强：消费电子领域。A9/53/72/76等等。搭载的是Linux操作系统。

M系列处理器能力较弱：工控领域、汽车电子、船舶、小型家电。M0/3/4/7。可以搭载单片机操作系统。

R系列处理器军用级，商用不会用到该系列处理器。

M系列处理器----单片机。

M0~7由于ARM公司设计的芯片内核架构不一致产生的芯片内核不同版本。



内核不一致处理能力不一致，M7处理能力属于单片机中最强的，M0处理能力最弱。

18年的产品主导的是M0/3核，M4/7核。19年的产品慢慢都在往M4/7核上转。

ARMv6架构----ARMv7架构----内核架构不一致引起的。

ARM公司只设计芯片内核，不生产一块芯片。生产芯片的主要厂商：ST、NXP(飞思卡尔)、TI、瑞萨。

NXP、瑞萨主要用在汽车电子。TI芯片比较广：无线领域较多：蓝牙，wifi，zigbee等。ST工控领域较多。

## STM32F407VGT6介绍

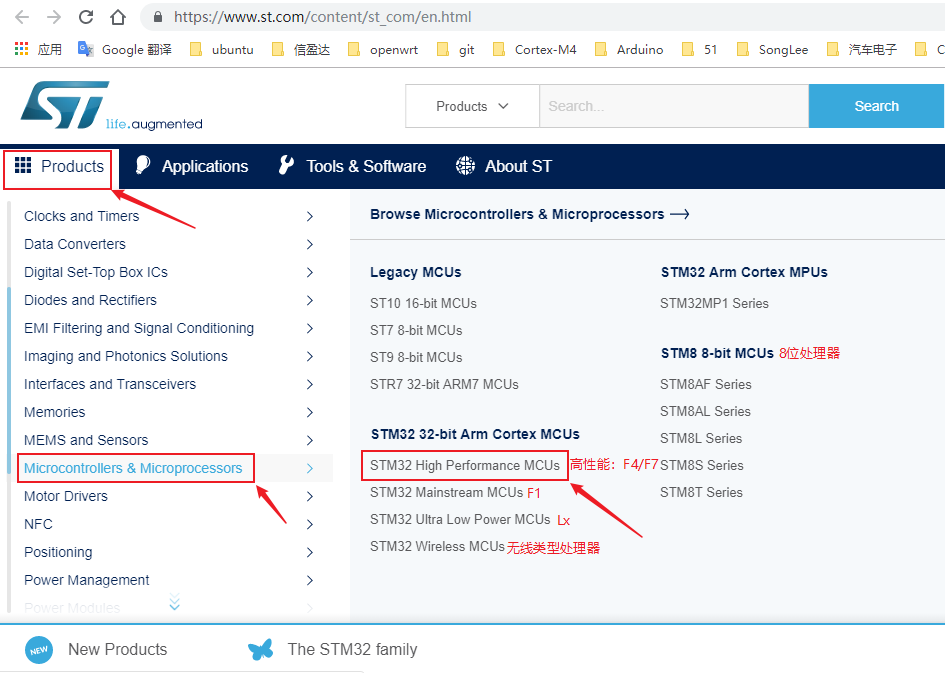
属于Cortex-M4内核架构的ST公司的处理器。学习STM32F407VGT6需要通过数据手册学习。任何一款芯片要知道如何让它工作需要学习这个芯片的数据手册。

第一步：学会下载芯片的数据手册

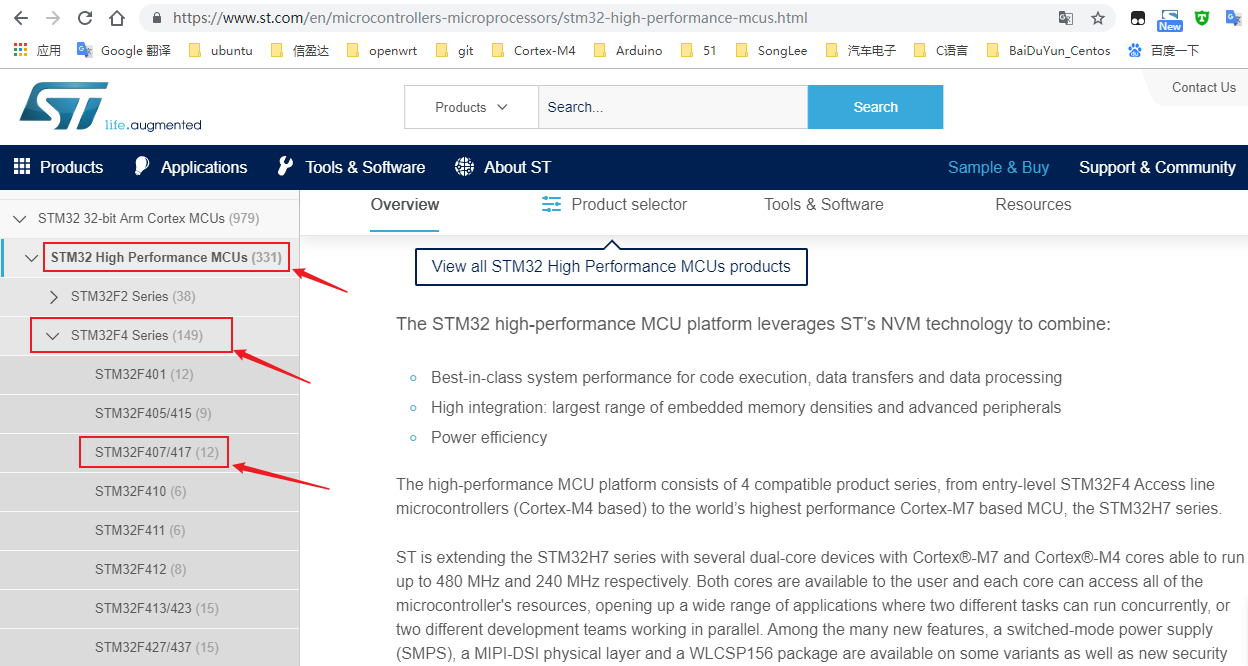
ST的芯片去ST的官网下载。ST官网：<https://www.st.com/content/st_com/en.html>

寻找芯片数据手册：

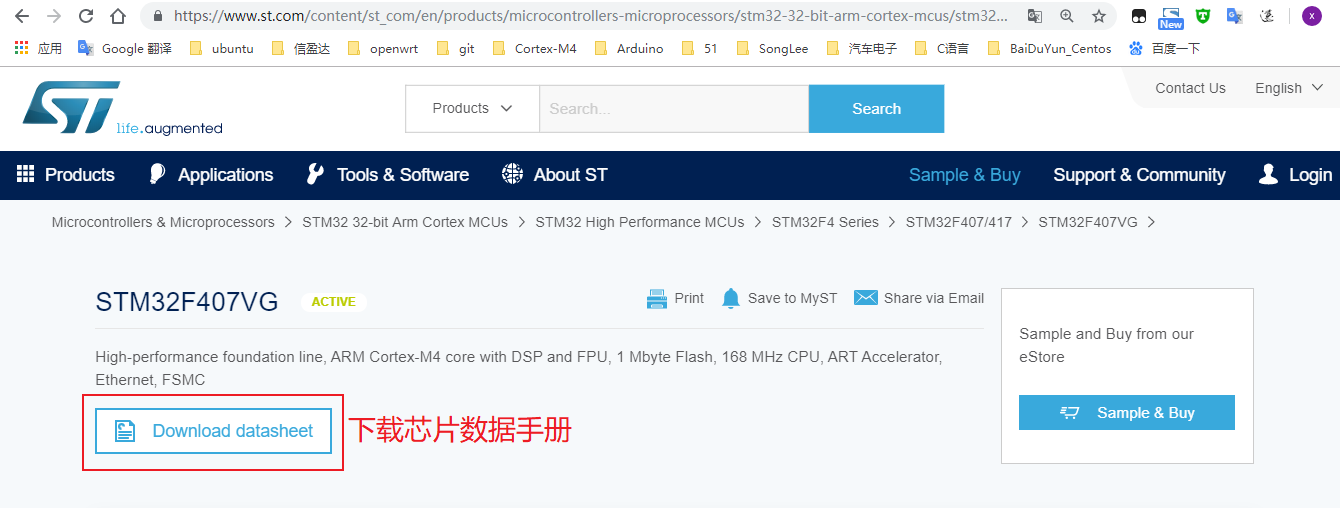
1. 查找路径



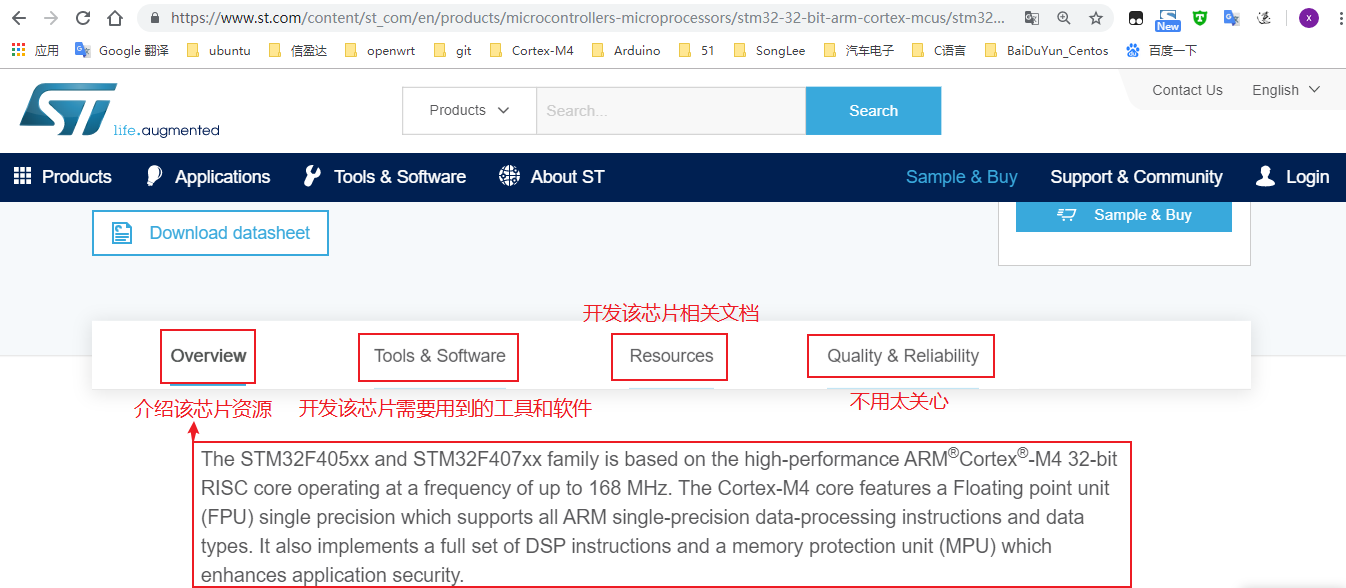
1. 进入路径找到相关芯片



1. 下载芯片数据手册

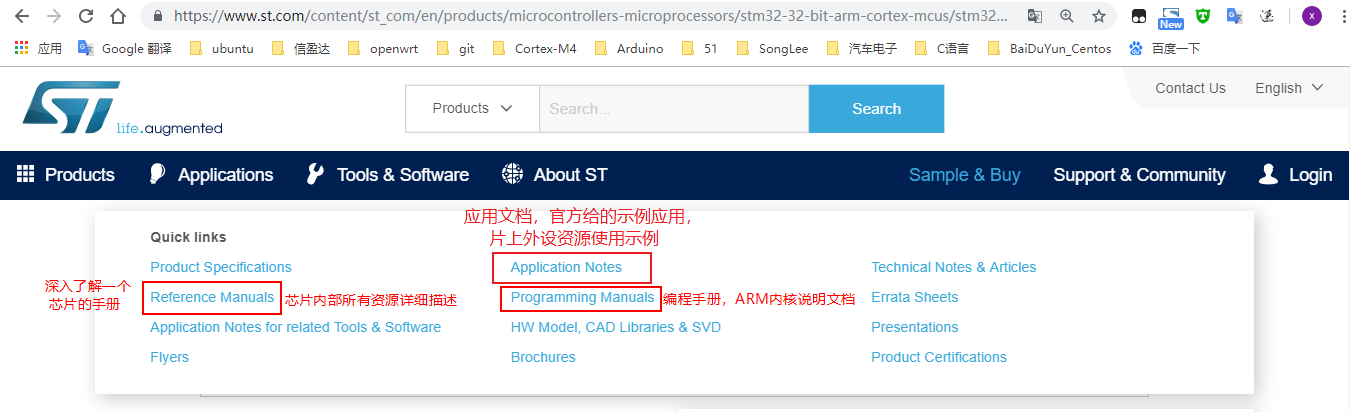


1. 其它资料需要下载



其它资料：工具&相关文档；官网所有的文档都是全英文。

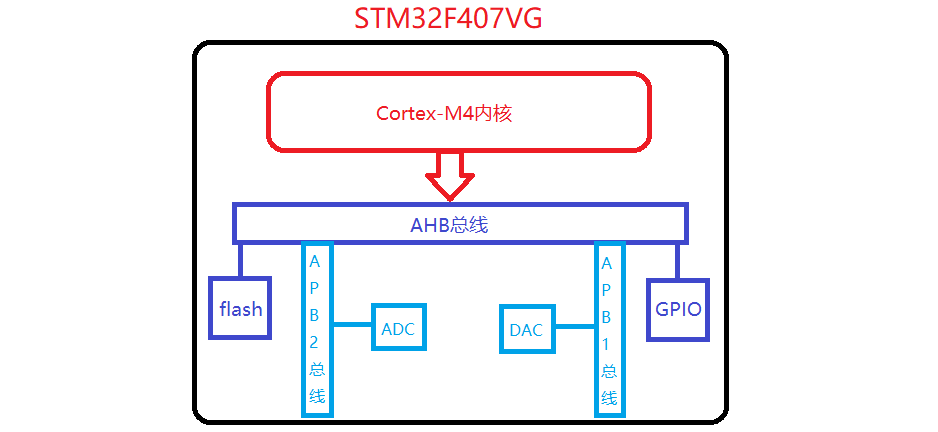
1. 其它文档资料下载

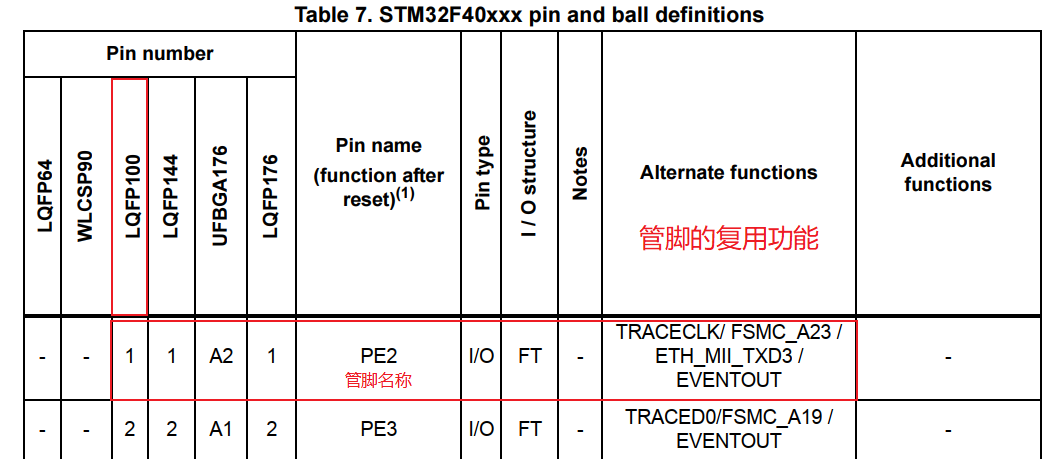


1. 其它软件工具下载在开发环境和项目工程创建的时候会说明。

第二步：学会查看芯片数据手册

通过数据手册的描述章节掌握整个芯片的架构。





## STM32F407VGT6最小系统介绍

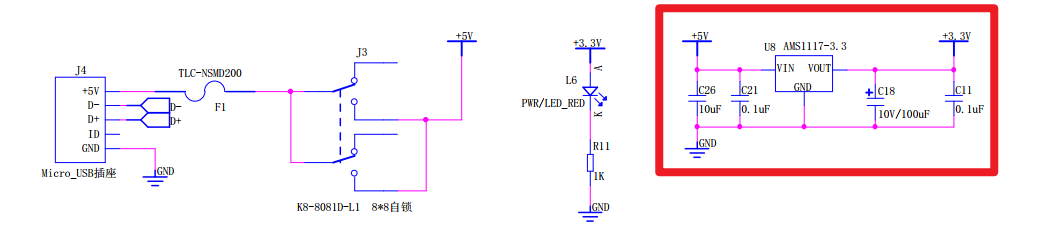
每个单片机的运行都需要一个最小系统。单片机的最小系统都是通用的。

最小系统包括：电源部分、振荡电路、复位电路(硬件复位电路)、微处理器。

STM32F407VGT6芯片最小系统包括：电源部分、振荡电路、复位电路、微处理器、自举模式。

1. 电源部分

STM32F407VGT6最大承受电压为3.6V，一般使用3.3V电压驱动。设计电路的时候需要给STM32F407VGT6的电压为3.3V。



1. 振荡电路

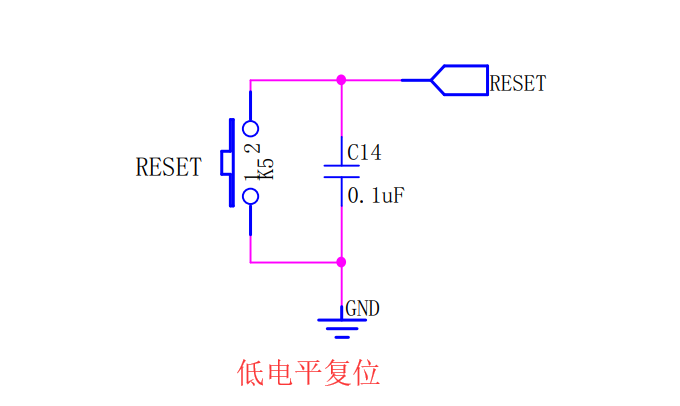
振荡器、晶振；RC振荡器、声表面振荡器。晶振频偏比较小，一般使用这种振荡器。

1. 复位电路

使得CPU复位的电路。复位电路有两种：高电平复位，低电平复位。

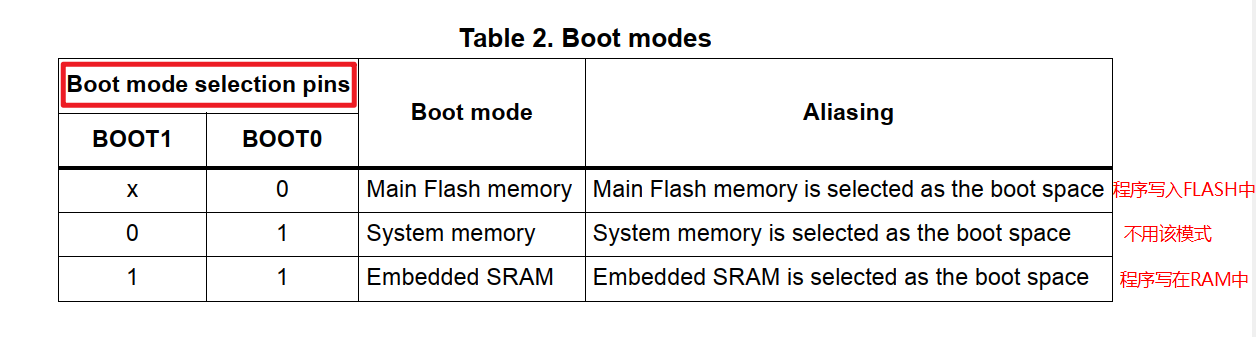
高电平复位：高电平能够让CPU复位的就称为高电平复位。

低电平复位：低电平能够让CPU复位的就称为低电平复位。



当前处理器属于低电平复位。51单片机属于高电平复位。

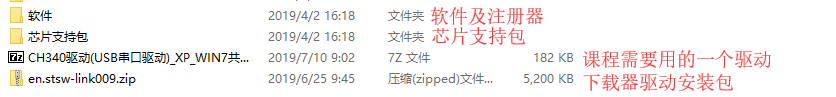
1. 自举模式



## STM32开发环境搭建

市场上开发STM32处理器的软件很多，种类大致有：MDK-ARM平台(KEIL软件)、STM32CubeMX、STM32CubeIDE、IAR、VS+VSGDB。

目前使用MDK-ARM平台进行程序开发。使用MDK-ARM526版本。



第一步：安装开发软件。

第二步：注册软件

第三步：安装芯片支持包

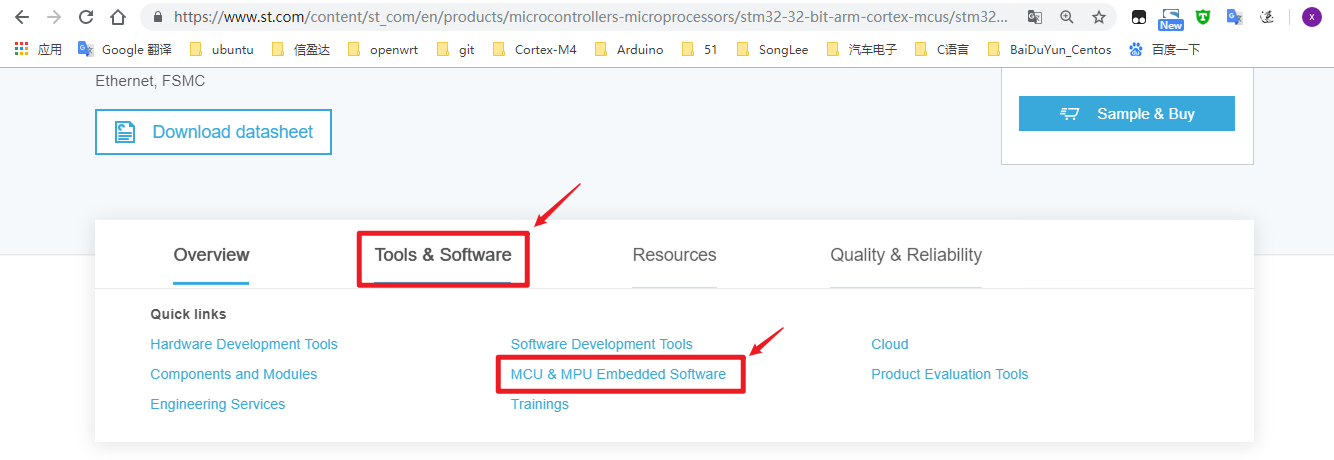
第四步：安装下载器驱动

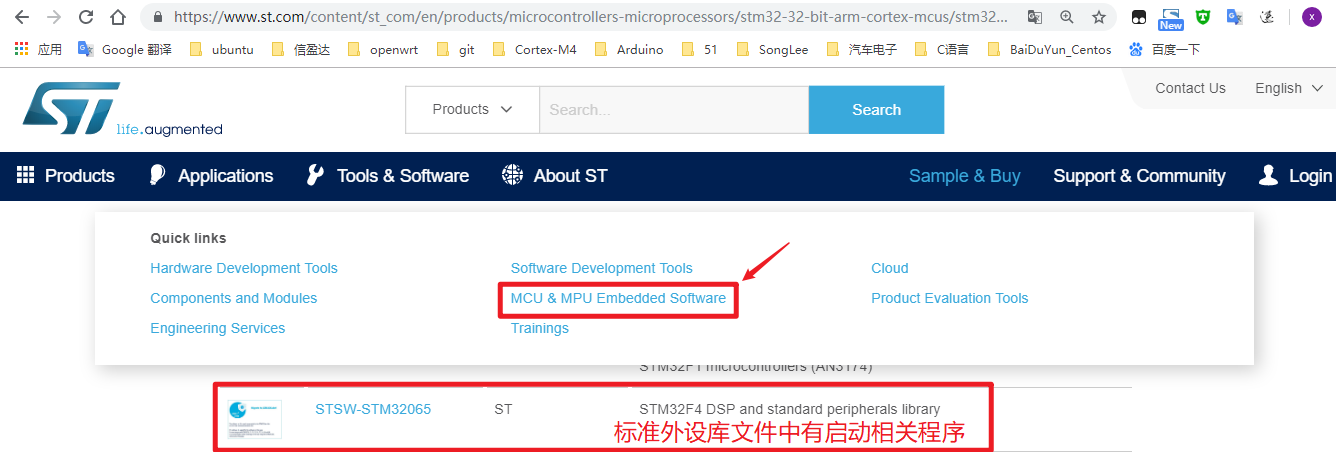
第五步：安装串口驱动

## STM32项目工程创建

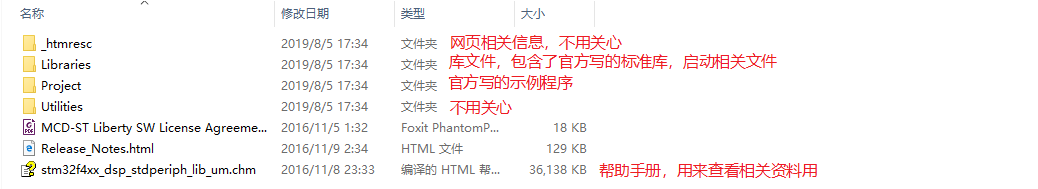
STM32属于ARM处理器，启动有一个启动程序，启动程序代码在官网中。

获取启动程序：

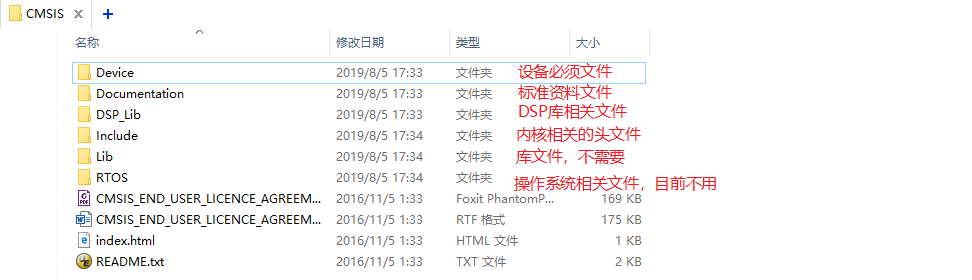




查看一下标准外设库的文件，找到需要的启动文件。









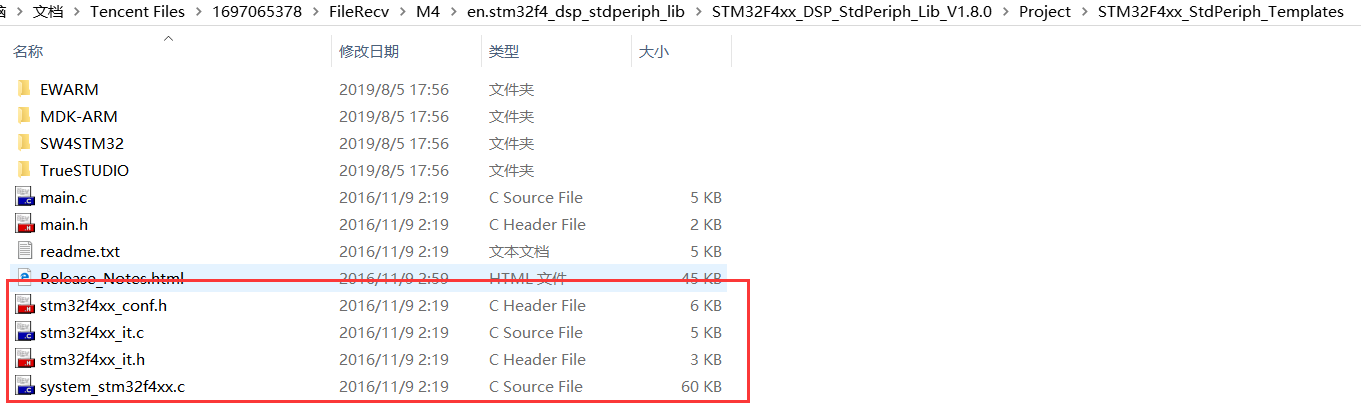
STM32中建立工程主要有三种方式：寄存器版本工程(能够最深入的了解该芯片的工作原理及过程)，库函数版本工程(使用标准官方库，市场上使用率最高)，HAL库版本工程(利用STM32CubeMX/IDE进行开发用)。

以寄存器版本工程为例：

1. 新建一个文件夹用来保存整个工程所有文件，文件夹名称：stm32f407vgt6\_demo
2. 在stm32f407vgt6\_demo文件夹下建立：cmsis文件夹、user文件夹
3. 在cmsis文件夹中建立inc文件夹、src文件夹
4. 在user文件夹中建立inc文件夹、src文件夹
5. 打开软件，新建工程，保存工程(到stm32f407vgt6\_demo文件夹下)，选择芯片型号为：STM32F407VGTx
6. 拷贝启动相关文件到cmsis文件夹下的inc和src中
   1. 进入en.stm32f4\_dsp\_stdperiph\_lib\STM32F4xx\_DSP\_StdPeriph\_Lib\_V1.8.0\Libraries\CMSIS\Include下拷贝所有文件到inc中
   2. 进入en.stm32f4\_dsp\_stdperiph\_lib\STM32F4xx\_DSP\_StdPeriph\_Lib\_V1.8.0\Libraries\CMSIS\Device\ST\STM32F4xx\Include下，拷贝所有文件到inc中
   3. 拷贝一个汇编文件到src中，具体拷贝哪一个查看stm32f4xx.h文件，里面有介绍；通过查看stm32f4xx.h文件，找到需要的文件是startup\_stm32f40\_41xxx.s

汇编文件路径：en.stm32f4\_dsp\_stdperiph\_lib\STM32F4xx\_DSP\_StdPeriph\_Lib\_V1.8.0\Libraries\CMSIS\Device\ST\STM32F4xx\Source\Templates\arm

* 1. 在路径：en.stm32f4\_dsp\_stdperiph\_lib\STM32F4xx\_DSP\_StdPeriph\_Lib\_V1.8.0\Project\STM32F4xx\_StdPeriph\_Templates下面有相关的文件需要拷贝，将头文件放到inc中，将源文件放到src中



1. 配置工程的虚拟文件夹以及添加文件

操作软件中的三色品字形图标按钮。

1. 配置软件的编译器，魔术棒。
2. 编写最简单的程序代码进行验证