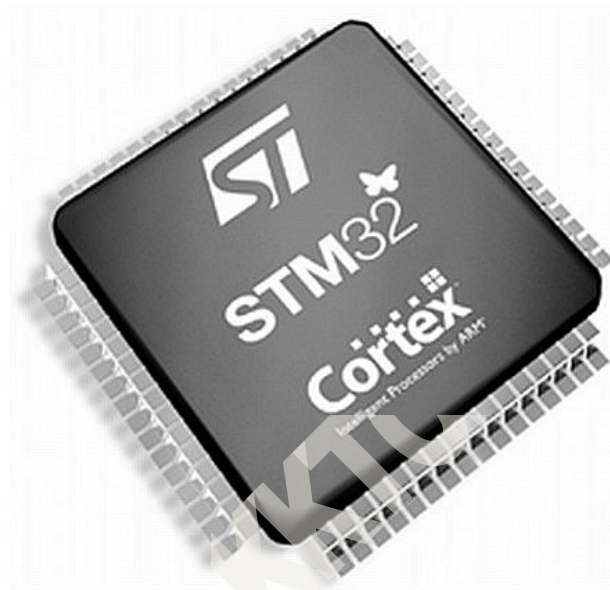


# TCR 嵌入式培训--了解 STM32

## 1.0 STM32 是什么



### 1.1 定义

$$\text{ST} + \text{M} + 32 = \text{STM32}$$

ST: SGS-Thomson 意法半导体

M: Microelectronics 微控制器 微处理器

32: 32 位

-> STM32: 意法半导体所设计制造的 32 位微控制器系列产品

### 1.2 背景

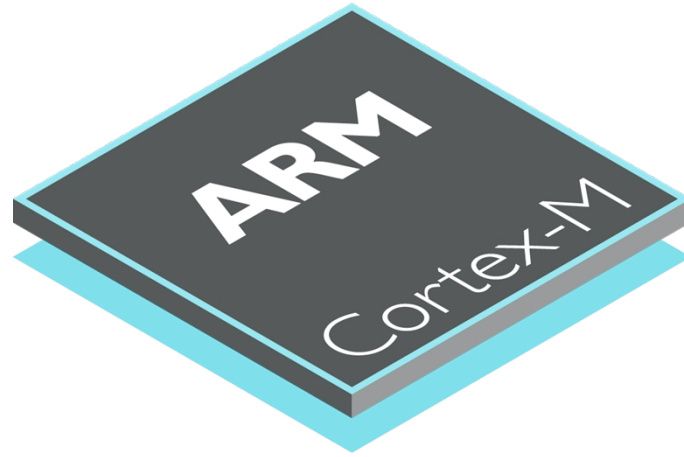
•

RISC -> ARM -> Cortex-M -> STM32

- 

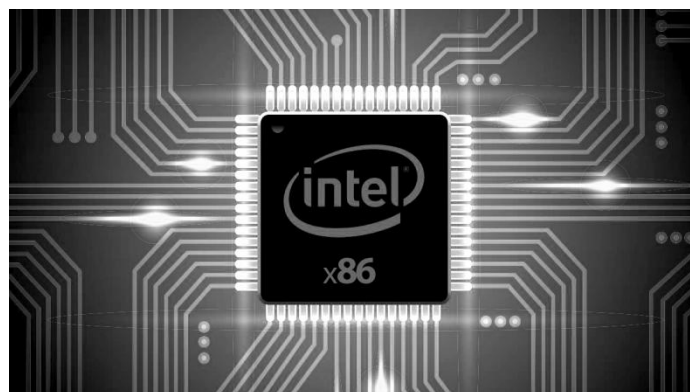
精简指令集 **RISC**: Reduce Instruction Set Computer

复杂指令集 **CISC** : Complex Instruction Set Computer



KKTC

精简指令集并非简单的比复杂指令集简单，而是其高效快速便宜，强调效率和价格，所以，常见的嵌入式设备，手机等等均为精简指令集。接下来的 ARM 架构就是基于精简指令集的。



而复杂指令集，强调指令的多样性和复杂性，常见于电脑和大型服务器，以往的 Intel 英特尔的 x86（IA-32），新起的 x86-64（AMD64）均为复杂指令集。

**ARM:**Advanced RISC Machines 高级精简指令集机器

ARM 处理器是英国 Acorn 有限公司设计的低功耗成本的第一款 RISC 微处理器。但实际上，真正的 ARM 是一个 32 位处理器框架，而 Acorn 自己从不生产处理器，而是搞处理器框架授权的，怎么实现其架构完全取决于客户自己，所以很多时候同为 ARM 架构的厂家互相之间的代码却不能直接套用。STM 就是 Acorn 授权客户之一，所以 STM 大部分产品都是基于 ARM 架构的。

ARM 处理器的三大特点是：耗电少功能强，16 位/32 位双指令集，和合作伙伴众多（所有手机都是 ARM 架构）。

目前 ARM 分为三大系列：Cortex-A，Cortex-R，**Cortex-M**

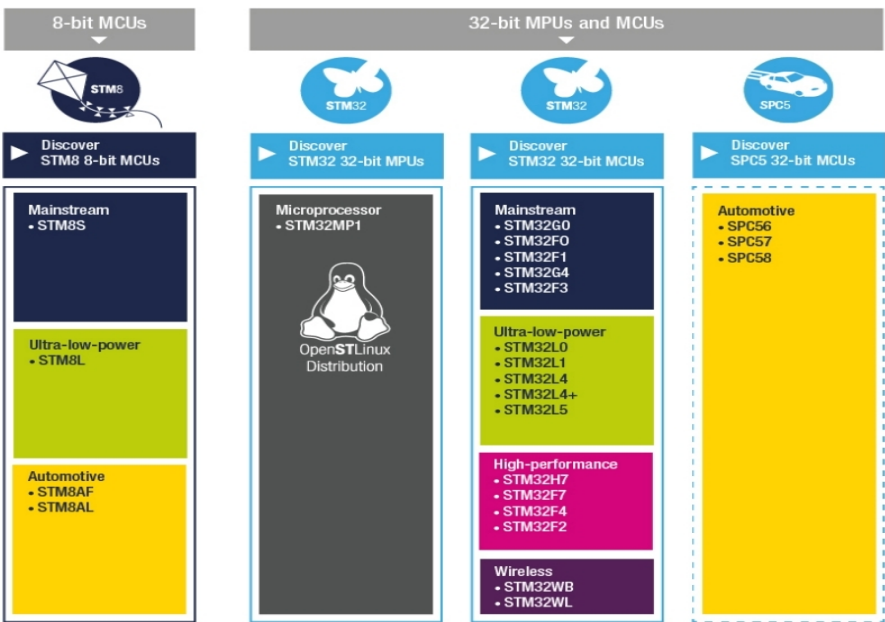
Cortex-A（Application）：适用于应用领域，支持 ARM 和 Thumb 指令集，并支持虚拟地址和内存管

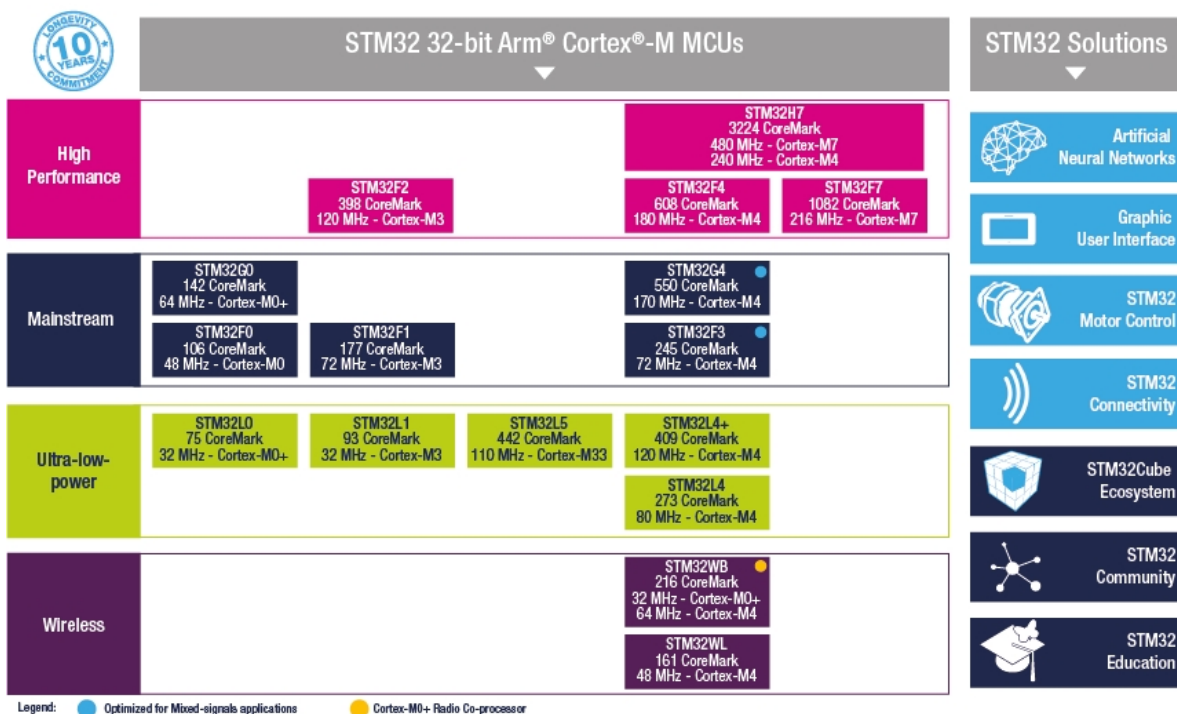
Cortex-R（Real-Time）：适用于高实时性领域，支持 ARM 和 Thumb 指令集，只支持物理地址，并支持内存管理。

Cortex-M（eMbedded）：适用于微处理器领域，只支持 Thumb 指令集。

为我们接下来要讲的就是 STM32 系列就是基于 Cortex-M 架构的

### 1.3 系列





**High-performance MCU:** 高性能微处理器, 是以 CPU 高度集成和多媒体应用为目的开发的, 基于 Cortex-M3/4/7, 频率从 120MHz (F2) 到 400MHz (H7) 不等。该系列均配备 ARTTM 加速器, 能够从 Flash 中 0 延迟加载数据。

**Mainstream MCU:** 主流微控制器, 是专门为对价格比较挑剔的产品开发的, 成本要求甚至苛刻到一片 1\$ 以下, 空间也非常受限。在这个类别下, 有基于 Cortex-M0/3/4 框架的, 频率从 48MHz (F0) 到 72MHz (F1/F3) 不等。

**Wireless MCU:** 无线微控制器, 是 STM32 微控制器家族新的双核成员, 集成 2.4GHz 的通讯模块, 可以用于无线和蓝牙应用。其中基于 Cortex-M0+ 的核心 (网络处理器) 是用于通讯管理, 而另外一个基于 Cortex-M4 的核心 (应用处理器) 是用于主要的嵌入式开发。

**Ultra Low-Power MCU:** 超低功耗微控制器, 该系列为低功耗应用而生, 用于电池驱动的设备, 从整体降低电池消耗来延迟电池使用时间。在这个系列中, 我们可以找到针对低价客户的基于 Cortex-M0 的微控制器, 和基于 Cortex-M4F 的带有 *动态电压调节 (DVS)* 功能的微控制器, 能够根据频率变化来最优化内部 CPU 电压。

## 2.0 为什么选择 STM32, 抛弃 Arduino ?

这是个艰难的选择，我们有无数的理由选择 Arduino，放弃 STM32，比如 Arduino 简单易学开发迅速支持强大 BUG 少，STM32 复杂难懂开发缓慢支持受限 BUG 多，但是很多时候我们不得不承认，现实就是，几乎没有真的用 Arduino 做产品开发的，而最流行的专业的嵌入式开发，STM32 就是绕不开的一条路。说白了：**Arduino 就是玩具，给兴趣爱好者玩的；STM32 才是真正的嵌入式，远远比 Arduino 强大几个量级，是开发电子产品的必备技能。**

下面我们再详细的列举一下抛弃 Arduino，选择 STM32 的理由

#### 1. STM32 在硬件上比 Arduino 更强大

Features	STM32F103	ATMEGA328
Clock Frequency	72 Mhz	16 Mhz
I2C Buses	2	1
SPI Buses	2	1
CAN Bus	Yes	No
Analog Channel	10	8
PWM Channel	15	6
USART Buses	3	1
GPIO's	32	24
On Board RTC	Yes	No
Architecture	ARM Cortex M3 32 bit	AVR RISC 8 bit
ADC Resolution	12 bit	10 bit
Quantization Level	4096	1024
Flash Memory	64KB	32KB
SRAM	20KB	2KB
Debugging	Serial, JTAG	Serial
PWM Resolution	16 bit	10bit
Price	110	115

2. STM32 有专门的低功耗系列，电量消耗是产品最重要的指标之一，消耗甚至比 Arduino Nano 在深度睡眠模式还要低很多？！

3. 没有高级封装的函数供你使用，也许很多人会说，这不正是 STM32 的缺陷吗？Arduino 正是因为有这些函数，所以开发迅速，简单吗？是这样，没错。但是，正是因为 STM32 足够底层，足够复杂，所以相对应的，把所有的控制权和自由都交给了工程师，可以做到所有 Arduino 能做到的事情和其所有不能做到的事情。

4. 可以跑真正的操作系统，比如 FreeRTOS，多任务处理不再是梦？！

5. 专业应用的广泛。STM32 甚至可以直接跑 AI，而 Arduino 不行（只能外接模块），STM32 可以运行 IoT 系统，而 Arduino 虽然可以，但是是没法发挥 IoT 真正的实力的。

6. STM32 是 ARM 下的一个分支，而所有的手机芯片基本都是 ARM 架构的，所以学会 STM32 对 ARM 下的其他分支开发，比如手机硬件开发等，都大有帮助，或者说是入门的必经之路

7. “我是搞 STM32 开发的”听起来就很帅~~，而“我是玩 Arduino 项目的”则有些黯然失色。

## 2.0 STM32 上手难度如何？

这部分我就直说了吧，难，非常难，资料少不说，数不尽的坑一定要自己全部踩过一遍才能熟悉开发环境。光是配置开发环境就花了我 2 周，就算要想做个最简单的 LED 流水灯，点亮个 LED，也是要深入理解 STM32 的各种底层，芯片引脚设置才行的，这也是新手最劝退的地方，你能想象，至少要枯燥无聊地花一周阅读 300 页的手册才能点亮一个简单的 LED 吗？是不是有点兴致全无？但是，一旦越过这个门槛，你就会打开 STM32 的大门，真正进入嵌入式的殿堂，深入地理解计算机内部的任何细节，无论软件还是硬件上的，你将会感觉自己身处一片智慧的汪洋什么都不懂，但同时又即将成为这片汪洋的新君主，感觉能通过计算机做到任何事情！！



以上图片及文字叙述均来自@Alchemic Ronin 于 RoboMaster 论坛发表的文章

本文档仅供学习，不得用于商业用途

Author:Chen