# 精简微控制器

本文讨论的是基于Arm Cortex-M0+内核的意法半导体STM32G0x1微控制器，尤其选用STM32G0B1RET为范例，该系列于2021年第七周发布。截至2024年5月，其已成为意法半导体主流产品线中最具性价比，供应情况最好，工艺设计较新的型号，不仅具有堪用的内核，同样具有入门级模拟外设以减轻设计工作，被意法半导体官方称作“适用于各种情形的单一架构”。故认为其在各方面均能显著代表新一代精简内核微控制器作为本文的主要讨论目标，并足以取代老旧过时的F1系列作为更新的典型范本。

有关意法半导体32位微控制器产品线的最新信息及下文所用资料，请参见 <https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html>

## 初步认识

认识每一款芯片的最好方法是通读其数据手册，而在编程时查阅应用手册和参考手册，值得注意的是勘误表记载了设备缺陷并有助于编程。查阅官方文档对初学者尤其关键，可有效避免面向应用的方法论带来的理论根基空虚和错误。STM32G0B1RET6也是如此，阅读其数据手册，尤其是第一页，可以为我们提供几乎关于该单片机的全部简要信息，对学习、选型均大有脾益。本文基于数据手册版本DS13560 Rev 4 。

打开数据表，首先映入眼帘且最引人注目的就是标题：

Arm® Cortex®-M0+ 32-bit MCU, up to 512KB Flash, 144KB RAM, 6x USART, timers, ADC, DAC, comm. I/Fs, 1.7-3.6V

按逗号分隔标题以快速得到如下信息：

* 基于Arm Cortex-M0+ 32位内核；
* 最高提供512KiB[[1]](#footnote-1)闪存，144KiB随机储存器；
* 六个通用同步/异步串行收发器
* 数个计时器
* 模拟到数字转换器
* 数字到模拟转换器
* 数个通讯接口
* 工作电压1.7到3.6V

毫无微控制器基础的读者也无需困惑而垂头丧气，其每一项均将在后文中解释。对于难以理解的英文缩写可以在目录中查询，也应该善于使用搜索引擎查询。继续简要阅读Features，我们将详细了解其每个参数和其意义。

#### 功能

###### 内核

引据设计方对这款处理器的描述：

1. Cortex-M0+ 处理器在所有Cortex-M 处理器中，具有最小占用面积与最低功耗需求。这使得它十分适用于低成本装置，包括智能传感器与混合信号单片系统。
2. Cortex-M0+ 出色的机器码密度大幅降低储存器需求以彻底利用集成于晶圆的闪存以节省储存器成本、降低储存器功耗并提高最大效能。以8或16 位处理器的成本获得32位的处理能力。
3. 开发者可以利用Cortex-M0+处理器内置的低功耗功能，对特定应用的功耗进行优化。其具有三种深度优化的低功耗模式，可以在降低功耗的同时满足性能方面的需求。

做简要总结，可以发现该处理器具有成本低，内存需求小，功耗低等特点。根据上述特征可以窥见该微控制器整体架构设计必定同样精简以追求最大化费效比。

###### 工作温度范围

宽运行温度范围使得该微控制器无论在酷热的工厂还是极寒的北国户外均可以工作。

###### 储存器

闪存的保护功能开启后可避免意外读写，可避免终端用户读出程序以保护知识产权，甚至可以不可逆的彻底熔毁芯片内部熔丝以更强的保护内部程序。闪存的安全加固区域可以让其中代码被执行后被隐藏，直到下次启动时才可恢复，在大型项目中保护用户代码以避免被不可信外部库破坏。双库区则允许闪存被分为两部分，处于其中一个库区的一部分在擦写时处理器仍能正常在另一库区运行，并允许用户实现诸如双启动，空中升级[[2]](#footnote-2)等功能。

静态随机储存器的奇偶校验开启时使用16KiB储存校验数据，将使得用户可用的空间缩减为128KiB，以此换取对内存意外错误（如热噪声，电磁脉冲，宇宙射线轰击）的感知，提升安全攸关应用的可靠性。

###### 专用循环冗余校验计算单元

专用循环冗余校验计算单元，用以加速现代通讯协议中可称为最常见的校验算法。提升该微控制器在现如今“万物互联“场景下的性能。

###### 重置和电源管理

该微控制器支持宽电压输入以工作在不同电压系统和电池直接供电系统中（电池放电中电压会下降）。分离的IO供电使其可作为不同电压域间沟通的桥梁，更加拓展了其适用范围。不仅上电和下电重置可以确保在供电非稳态时微控制器不误触发，欠压重置也可以确保系统欠压逼近临界值时该微控制器不错误动作，大幅提高系统稳定性。四种循序渐进的低功耗模式给予开发者同唤醒开销和睡眠功耗斡旋的更多选择。专用电池供电引脚可以确保微控制器几乎断电的情况下依然可以正常记忆部分参数并保持实时时钟运行。

###### 时钟管理

内置的高低速RC振荡器在简化应用中可提供差强人意的系统时钟源，尤其是HSI48[[3]](#footnote-3)振荡器保障了该系列的一个重要功能——无晶振USB通讯。也可使用宽范围的外部高速谐振/振荡器提供更加精确的系统时钟。相应的，外部RTC谐振器提供了更长时间的实时时钟稳定度。高速外部谐振器可由内部电路驱动以起振，也可以使用自行起振的高速外部振荡器。

###### 高速输入输出（引脚）

作为芯片的基本单元，高速的输入输出可保证最大化硬件效能而不至于成为瓶颈之一，也能提供更高的软件定义协议通讯速率。5V耐受的引脚使得该芯片无需电平转换[[4]](#footnote-4)即可同兼容的CMOS芯片通讯。全映射的外部中断更是提供了外部事件输入的极大灵活性。

###### 直接内存访问控制器

可代替CPU进行内存到内存、内存到外设的搬运工作，减轻CPU工作负担，提高总线利用率。灵活的通道映射为开发人员编程提供了方便。

###### 模拟外设（模数转换器、数模转换器和比较器）

限于篇幅该段可不做理解，后文将详细说明。多通道12bit[[5]](#footnote-5)模数转换器不仅支持宽电压输入范围，更支持硬件升采样至16bit以适用于高精度采样应用。双12bit数模转换器支持低功耗和采样并保持模式。可编程轨到轨[[6]](#footnote-6)输入输出低功耗比较器支持在指定电压阈值被触发时发送中断到处理器，也支持作为独立器件工作。这三组器件提供了数字世界到模拟世界的桥梁，使得用户在采集物理量（譬如重量、声强、光强、温度）时无需再外接专用芯片，也使得用户能输出除数字开关量外的连续模拟量。

###### 计时器（看门狗、系统滴答和实时时钟）

计时器在启用时以恒定的时钟速率向前计数直至溢出归零，利用这一特性我们可以完成最简单的定时，也可以利用一个数字阈值（高于某数字则输出，反之不然）生成脉冲宽度调制波驱动诸如步进电机、开关电源等设备。更高的频率允许该定时器以更快的速度计数以提升时间量化精度，更高的位宽则允许该定时器更长时间不溢出以实现更长的单调跨度。看门狗在程序死锁超时未应答时可以自动重置整个系统，这个功能对实时安全应用，譬如汽车尤为有效。系统滴答是唾手可得的、集成于处理器而开销很低的计时器，常用来做程序延迟。实时时钟则提供了程序访问人类历法的入口，同上文所述，该时钟即使断电仍能由电池供电继秒向前计时。同时实时时钟可以设定闹钟以特定时间点/特定周期唤醒或响铃提醒处于正常或低功耗模式下的程序。

###### 通讯接口（USB、PD、FDCAN）

这一部分是高度应用化的介绍，非常宽泛，具体协议请自行搜索，后文讲外设会一并讲解。除了经典接口外，值得注意的是无晶振USB通讯，自带USBPD控制器和最常见的汽车总线CAN。

###### 其他

* 串行线调试接口：我们将使用这个接口在后续的学习中下载和调试程序，这个接口可以使得嵌入式调试就像在自己的计算机上调试普通程序一样。
* 唯一ID：用处很多，读者自行想象。
* 环保封装：环保。

显然，对微控制器的描述大量集中在处理器之外的外设接口上，故在此提出一条关键论断，请谨记心中：**微控制器无论何时焦点都是其互联性而不是算力。**换句话说，微控制器

的价格往往主要构成是外设而不是处理器，学习微控制器就是学习外设。请继续向后翻阅，因为主要内容在第一页已经阐述完毕所以只需快速扫读Description，然后便可以看到处理器的结构框图。结构框图是控制器结构的抽象表达，将贯穿整个微控制器学习和理解过程。

#### 结构框图

1. KiB: Kibibyte, 2^10 即1024个字节。 [↑](#footnote-ref-1)
2. 空中升级（OTA）机制可以让设备在固件正常运行时根据接收数据（如通过Wi-Fi 或蓝牙）进行自我更新。 [↑](#footnote-ref-2)
3. 指Internal 48 MHz RC oscillator [↑](#footnote-ref-3)
4. 5V CMOS和3.3V CMOS在电气上是兼容的，但是5V CMOS的输出端高电平远超3.3V，若接收端不耐受则会损坏引脚。 [↑](#footnote-ref-4)
5. 指转换器将全输入范围量化到数字后，表达该数字范围所需的二进制位数。 [↑](#footnote-ref-5)
6. 指正电源轨道到负电源轨道（这里是地），尤其指可以在两个轨道间满幅度输出。 [↑](#footnote-ref-6)