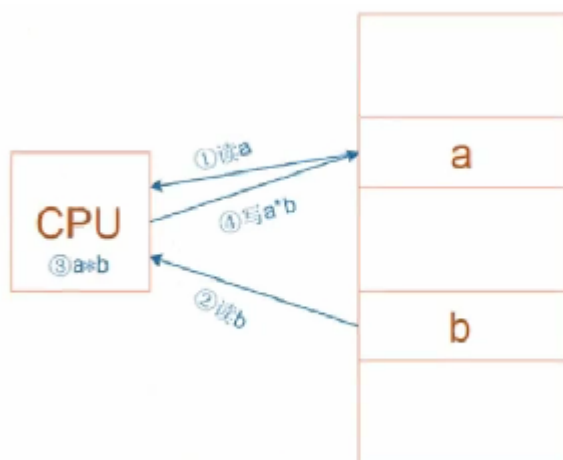


ARM

ARM芯片属于精简指令集计算机(RISC: Reduced Instruction Set Computing), 它所用的指令比较简单, 有如下特点:

- 对内存只有读、写指令
- 对于数据的运算是CPU内部实现
- 使用RISC指令的CPU复杂度小一点, 易于设计



stm32与ARM的关系

Cortex-A 应用程序系列

阿熊话大

A表示应用程序(Application Processor), 该系列有内存管理单元(MMU), 实现虚拟内存, 让每个用户进程都拥有自己独立的地址空间, 这是现代多用户多进程操作系统 (比如Linux、VxWorks) 所必须的。

该系列适用于高端消费电子领域, 比如智能手机、平板电脑、智能电视、路由器等;

Cortex-R 实时控制系列

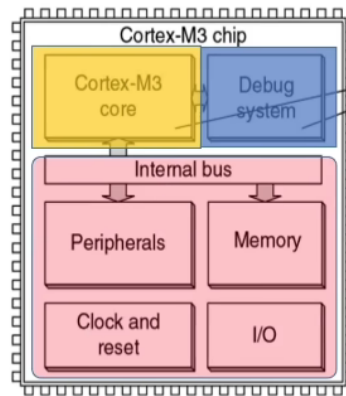
R表示应用程序(Real Time Controller)

该系列适用于高性能、高实时性应用, 比如硬盘/固态驱动控制器、企业网络设备、消费电子领域的蓝光播放器、汽车领域的安全气囊等;

Cortex-M 微控制器系列

M表示应用程序(Microcontroller Processors)

该系列适用于低功耗、高性能且对成本敏感的产品, 比如家电产品、物联网、无人机等;



ARM公司设计

芯片厂商开发

汇编指令

1.4 汇编指令

- 读内存: Load

示例

LDR R0, [R1, #4] ; 读地址"R1+4", 得到的4字节数据存入R0

- 写内存: Store

示例

STR R0, [R1, #4] ; 把R0的4字节数据写入地址"R1+4"

- 加减

ADD R0, R1, R2 ; R0=R1+R2

ADD R0, R0, #1 ; R0=R0+1

SUB R0, R1, R2 ; R0=R1-R2

SUB R0, R0, #1 ; R0=R0-1

- 比较

CMP R0, R1 ; 结果保存在PSR(程序状态寄存器)

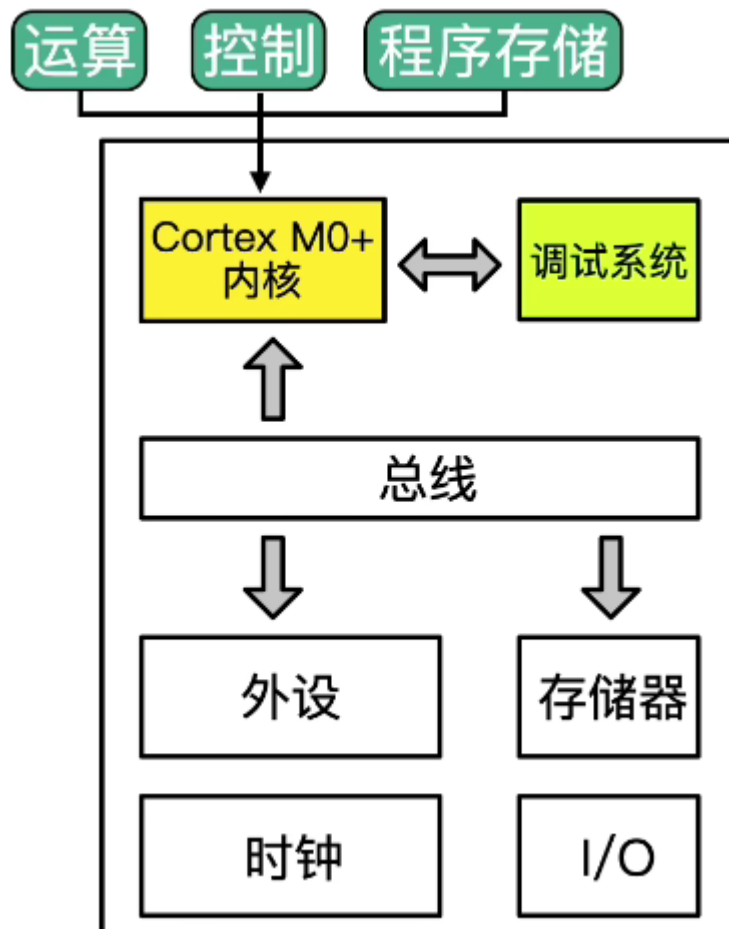
- 跳转

B main ; Branch, 直接跳转

BL main ; Branch and Link, 先把返回地址保存在LR寄存器里再跳转

arm内核

现在的32位单片机
基本上都是采用的
那同样是采用了arm的内核
不同厂家的单片机有什么区别呢
在说这个问题之前
首先说一下内核是什么
以arm的cortex m0加内核为例
cortex m0加可以看作处理器的核心
单片机的运算控制和程序执行
都发生在这个cortex m0加里面



而且在arm的所有内核中
HOTEXM0加有着最小尺寸的柜面积
其硅片面积甚至可以和八位单片机相媲美
而且cos x m0加还有着最低的功耗

那同样采用arm cortex m0加内核的单片机
它们在组成上存在什么差异呢
以STM32和国产的武汉芯源
半导体的cw32为例
可以看一下他们两个的逻辑框图
大体结构基本都差不多
首先他们都是采用了arm公司的M0加内核
然后通过总线与外面通讯
cortex m0加内核和调试系统由arm公司设计
而下面的这些模块由单片机厂商设计
其中外设就是我们所熟知的ADCIPHONCE和SSPI
这些这样就构成了我们看到的单片机
但各个厂商在单片机的设计上会有稍许的不同
比如我们熟知的SSTM32
他的工作电压是2.0到3.6伏
而cw32的工作电压是1.65到5.5伏
拥有更宽的电压范围
还有就是鹰角的电压
如果都是3.3伏供电
STM32的部分引脚可以接受五伏的电压
而cw32是不需要这个功能的
cw32只要把工作电压设置到五伏
它的引脚就能接受五伏电压了
其实只要是采用了arm内核的单片机
在功能上差别一般不会很大
既然功能都差不多
那怎么评价一款单片机的优异呢
此时就得看哪一款单片机抗干扰能力更强了
也就是当芯片在受到外界干扰的时候
还能不能稳定工作
最近网上兴起了
使用电警棍来测试单片机的方法

就是让电警棍在芯片旁边放电
它的放电电压高达3万伏特
以此来判断芯片的抗干扰能力
这是六款不同厂商的单片机
都采用了M0加的内核
PCB和外围电路都是完全一样的
接下来我们让电棍分别干扰这六块电路板
第一块电路板被电击之后直接死机
第二块出现异常
第三块直接不工作了
第四块死机
第五块也死机了
而第六块国产的cw32却可以抗击干扰
虽然使用警棍测试并不是标准的项目
但是也可以从一定程度上说明
芯片的抗干扰能力
当面对3万伏的干扰时
为什么国产新势力的cw32
能保持如此稳定的运行呢
因为它是经验累积的产物
cw32是最近一两年才上市的
它的设计充分借鉴了其他单片机的设计优势
单片机的基本原理是从内置的存储器中
不需程序
在这个过程中
单片机需要记忆程序执行的各种状态和结果
如果遇到强烈的干扰
导致单片机电路丢失数据或者记错
那么MCU的整个运行过程就会被打断
此时一些具备补救保护功能的单片机
会在出错后将单片机复位
虽然复位也属于补救功能

但是单片机重启会导致之前的任务终止
因此为了实现更好的抗干扰性能
单片机的主动防御的优势则会更明显
这就相当于汽车的主被动安全措施一样
主动安全措施直接能避免事故
显然是更高级的处理方式
就拿武汉芯源cw32来说
MCU的防御措施主要分为这三类
首先针对辐射干扰能量的特性
应该尽量减少干扰能量的引入
也就是说电路设计这块需要遵循环路最小原则
尽量在各个维度实现最小的环路面积
因为辐射干扰的能量引入土星
归根结底还是电磁感应
如果MCU没有外围电路或者被关在一个铁笼子里
后面的干扰是很难传递进去的
其次比如说MCU为了控制一个负载
不得不引出一根长线
那通过这根长线接收到的干扰能量不可避免了
只能在MCU的外围增加一些保护器件
让干扰能量不易被传导到MCU
内部的核心敏感区域
最后针对MCU内部设计做相应改造
MCU内部设计也是像一个城市一样
有交通工具
道路码头和管理机构
如果MC又在设计的时候
就特别注意外来干扰能量的引导和防护
那也能够很好的避免和化解干扰问题
所以要提高基于MCU的整个系统的抗干扰性能
是由诸多路径可循的
其中MCU内部的功夫则主要靠CEO设计者的努力

采用arm内核的单片机都是互相兼容的
而最主要差异在抗干扰的表现上
比如新元半导体的cw32
只有着不错的抗干扰能力
这是我对他们的理解
希望对你有帮助