第2章 微型计算机的硬件系统（一）2.1~2.3

2.1 微型计算机的硬件共性结构及基本性能指标

2.2 Arm Cortex-M微处理器概述

2.3 CPU内部寄存器与存储器映像

2.1 微型计算机的硬件共性结构及基本性能指标（了解）

微型计算机的名字太多了，简直数不胜数，也反映了微型计算机的不同种类、不同用途，如：个人计算机（Personal Computer，PC）、桌面计算机（Desktop computer）、多媒体应用处理器（Multimedia Application Processor，MAP）、微控制器（Microcontroller，MCU）、单片机（Single Chip Microcomputer，SCM）、工控机（Industrial Personal Computer，IPC）、笔记本电脑（Notebook computer），等等。本节给出微型计算机硬件的共性结构及基本性能指标。

2.1.1 微型计算机的硬件共性结构

微型计算机硬件的共性结构可以简单地表述为：微型计算机是在内部集成了中央处理单元CPU、存储器（RAM/ROM等）、定时器/计数器及多种输入输出（I/O）接口的比较完整的数字处理系统。

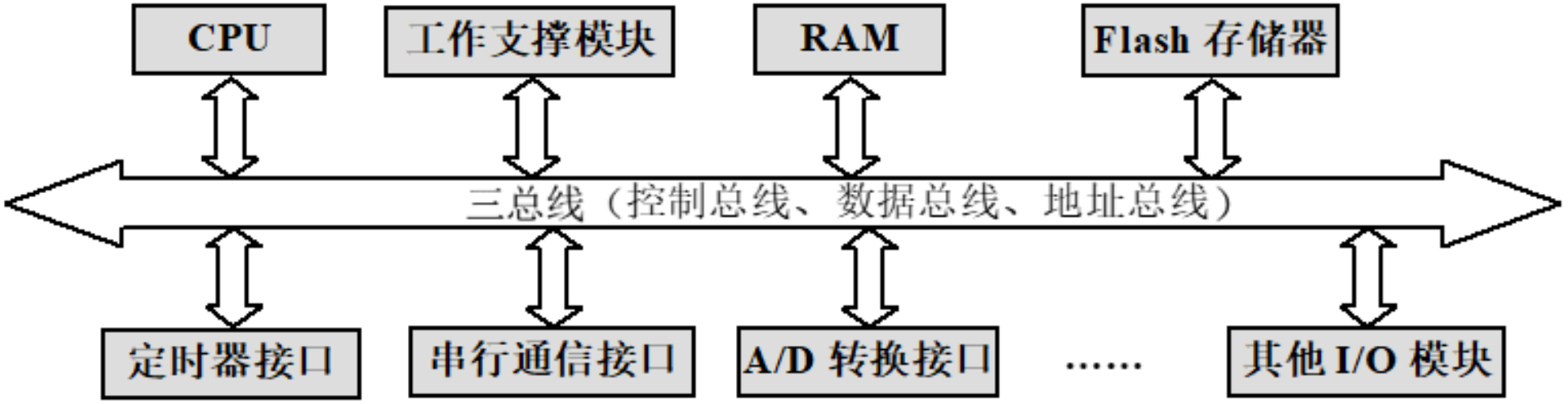


图 2-1 微型计算机的一般硬件组成框图

2.1.2 微型计算机基本性能指标

基本性能指标主要有字长、主频、存储容量、外设扩展能力、软件配置等

1. 字长

字长是计算机内部一次可以处理的二进制数的位数，目前常有32位、64位、128位等

2. 主频

主频是指微型计算机中CPU的时钟频率，在MHz、GHz级别等

3. 存储容量

存储容量是衡量微型计算机中存储能力的一个指标，它包括内存容量和外存容量。内存容量是由CPU的地址总线的位数决定，目前已达到MB、GB级别；外存容量主要是指硬盘容量，目前已达到GB、TB级别。

4. 外设扩展能力

一台微型计算机可配置外部设备的数量以及配置外部设备的类型，对整个系统的性能有重大影响。如显示器的分辨率、多媒体接口功能和打印机型号等，都是外部设备选择中要考虑的问题。

5. 软件配置情况

软件配置情况直接影响微型计算机系统的使用和性能的发挥，通常应配置的软件有：操作系统、计算机语言以及工具软件等，另外还可配置数据库管理系统和各种应用软件。

2.2 Arm Cortex-M微处理器概述（了解）

2.2.1 Arm Cortex系列微处理器系列概述

Arm即Advanced RISC Machines的缩写，既可以认为是一个公司的名称，也可以认为是对一类微处理器的通称，还可以认为是一种技术的名称。

1985年4月26日，第一个Arm原型在英国剑桥的Acorn计算机有限公司诞生，由美国加州SanJoseVLSI技术公司制造。

1990年成立了Advanced RISC Machines Limited（后来简称为Arm Limited，Arm公司）

2019年5月在中国成立了Arm中国 https://www.armchina.com

Arm公司在经典处理器Arm11以后的产品统一改用Cortex命名，并分成A50、A、R和M四类，旨在为各种不同的市场提供服务。

1. Arm Cortex-A50系列处理器

面向高效的低功耗服务器市场领域，将推动手势控制功能，增强现实技术，移动游戏，Web 2.0技术等领域技术的发展。

2. Arm Cortex-A系列处理器

面向尖端的基于虚拟内存的操作系统和用户应用，适用于具有高计算要求、运行丰富操作系统以及提供交互媒体和图形体验的应用领域

3. Arm Cortex-R系列处理器

面向实时系统的应用，适用于智能手机、硬盘驱动器、数字电视、医疗行业、工业控制、汽车电子等

4. Arm Cortex-M系列处理器

Arm Cortex-M系列基于v7M/v6M架构基础的处理器，面向微控制器的应用，是一系列可向上兼容的高能效、易于使用的处理器，这些处理器旨在帮助开发人员满足将来的嵌入式应用的需要。Cortex-M系列针对成本和功耗敏感的MCU和终端应用（如智能测量、人机接口设备、汽车和工业控制系统、大型家用电器、消费性产品和医疗器械）的混合信号设备进行过优化。

Arm Cortex-M中的“M”就是指微控制器（Microcontroller，MCU），在嵌入式人工智能与物联网领域应用广泛，本书就是以该类微型计算机作为蓝本，因此下面单辟一节介绍其中应用广泛的Arm Cortex-M4

2.2 ARM Cortex-M 微处理器概述

2.2.1 ARM Cortex系列微处理器系列概述

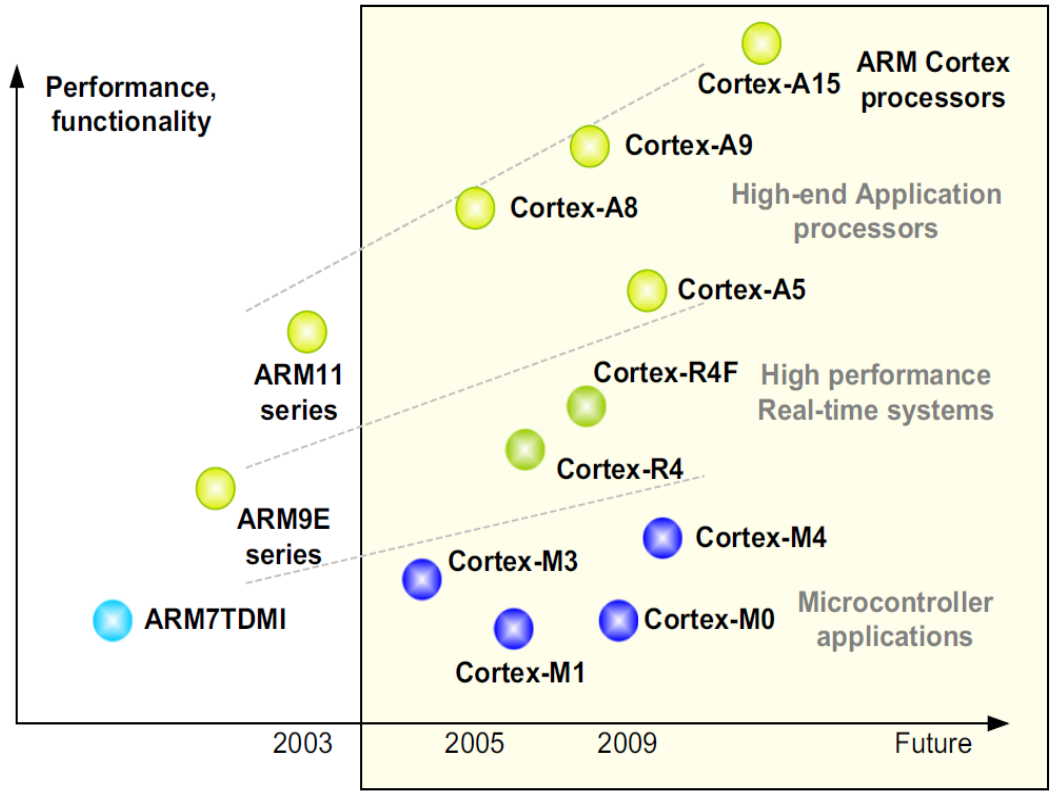


Figure 1.3:
Diversity of processor architecture to three areas in the Cortex processor family.

2.2.2 Arm Cortex-M4微处理器

2010年Arm公司发布了Cortex-M4处理器，单精度浮点单元（FPU）作为内核的可选模块，如果在Cortex-M4内核中包含了FPU，则称它为Cortex-M4F。

主要特点：

（1）32位处理器，内部寄存器、数据总线都为32位；

（2）采用Thumb-2技术同时支持16位与32位指令；（Thumb是Arm架构中的一种16位指令集，而Thumb2则是16/32位混合指令集。）

（3）哈佛总线架构， 32位寻址，最多支持4G存储空间；三级流水线设计；（哈佛总线架构：具有独立的程序指令存储空间和数据存储地址空间）

（4）片上接口基于高级微控制器总线架构技术；

（5）集成NVIC（嵌套向量中断控制器），最多240个中断请求；

（6）可选的MPU（存储器保护单元）具有存储器保护特性；提供时钟嘀嗒，主栈指针，线程栈指针等操作系统特性；

（7）具有多种低功耗特性和休眠模式。

1. M4F内核

Arm Cortex-M4F是一种低功耗、高性能、高速度的处理器，支持Thumb指令集，同时采用Thumb2技术，且拥有符合IEEE 754标准的单精度浮点单元（FPU）

其硬件方面支持除法指令，提供中断服务程序（Interrupt Service Routine，ISR）和线程两种模式，具有指令和调试两种状态。

2. 嵌套中断向量控制器

嵌套中断向量控制器（Nested Vectored Interrupt Controller，NVIC）是内建的中断控制器，负责进行中断源的识别、编号、并通知CPU。例如：在STM32L4芯片中，非内核中断源数目为83个，优先等级范围为7-89，其中7等级对应最高中断优先级。NVIC中还包含一个24位倒计时定时器SysTick（本书8.2.2节中给出其编程方法），即使系统在睡眠模式下也能工作，若作为实时操作系统RTOS的时钟，则可以给RTOS在同类内核芯片间移植带来便利。实现中断尾链和迟到功能。

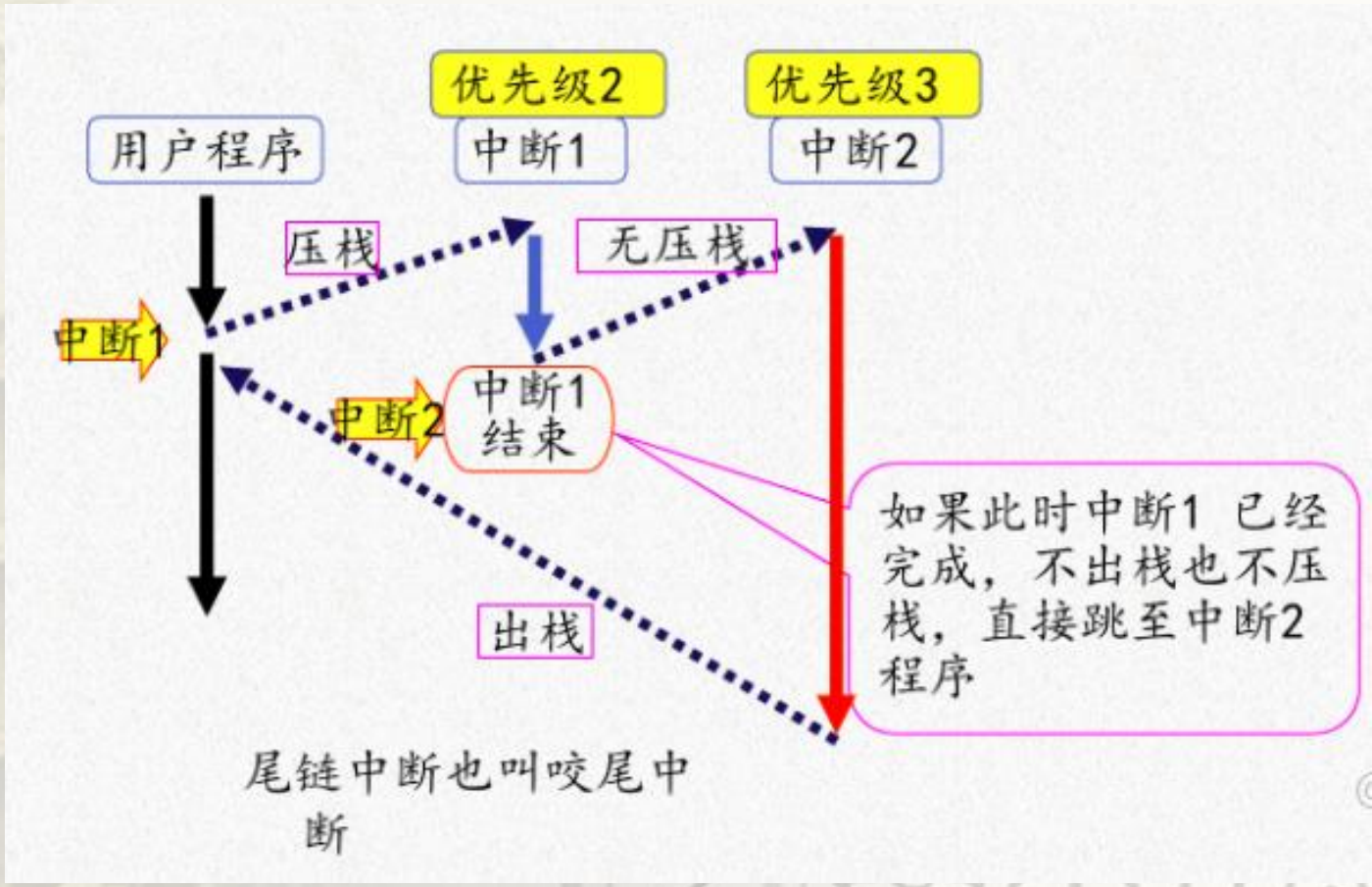
2.2.2 ARM Cortex-M4F微处理器

中断尾链

使用中断嵌套时，必须仔细计算主推栈容量，避免堆栈溢出。

中断尾链就是当上一个异常处理(中断)返回时，Cortex-M4F响应挂起的异常时，为了避免浪费CPU时间，继续使用上一个异常已经入栈好的寄存器数据，两个异常只执行了一次入栈/出栈操作。

利用中断尾链，缩短了挂起中断的延迟时间，提高了中断响应速度。

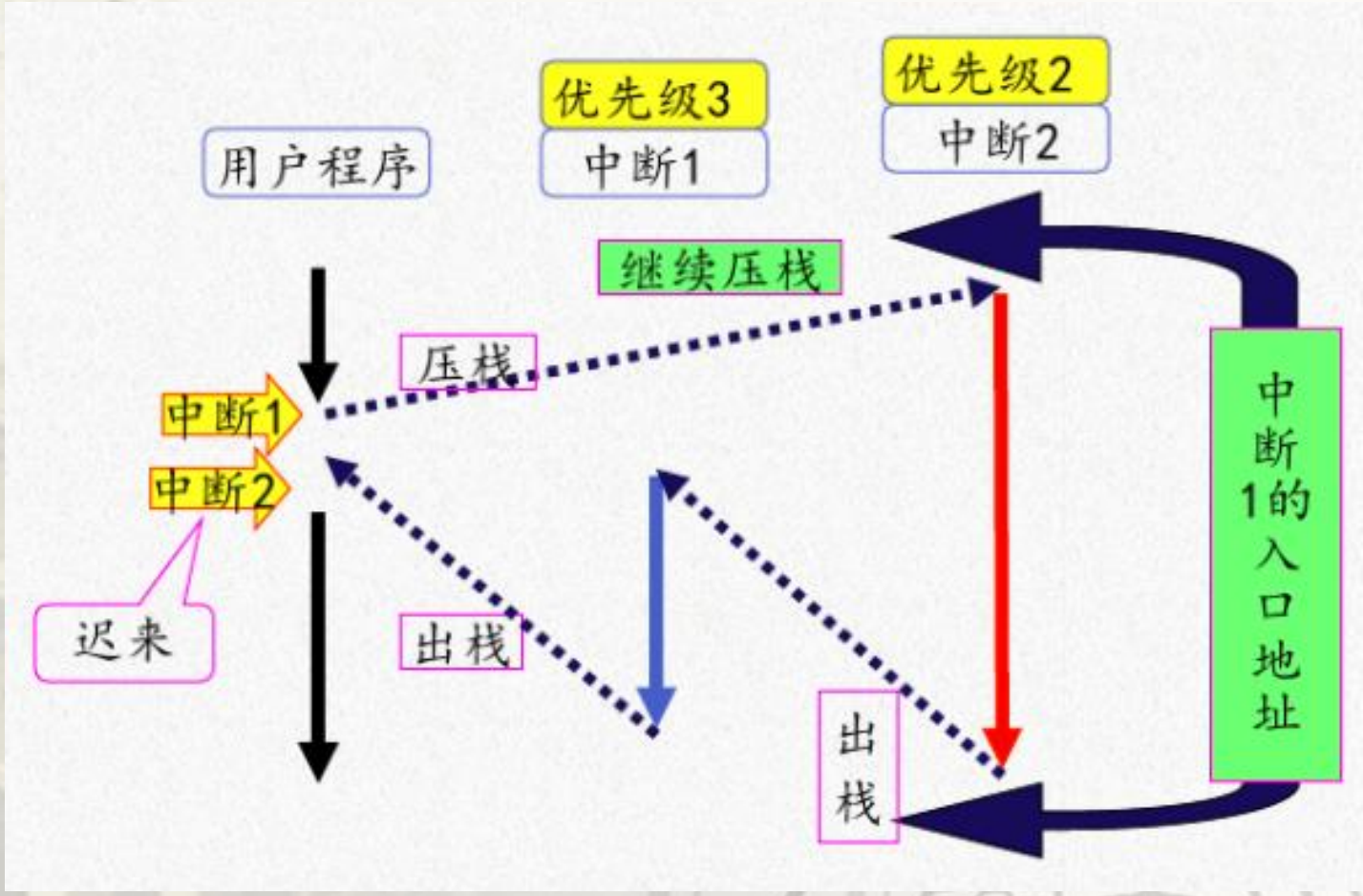


2.2.2 ARM Cortex-M4F微处理器

迟到

当异常产生时，处理器会接受异常请求并开始入栈操作。若在入栈操作期间产生了另外一个更高优先级的异常，则后到的高优先级异常会首先得到服务

利用“迟到”功能缩短了高优先级中断的延迟时间，提高了高优先级中断的响应速度，节省了堆栈控件。



3. 存储器保护单元

存储器保护单元（Memory Protection Unit，MPU）是指可以对一个选定的内存单元进行保护。它将存储器划分为8个子区域，该子区域的优先级均是可自定义的。处理器可以使指定的区域禁用和使能。

4. 调试解决方案

可以对存储器和寄存器进行调试访问，具有SWD或JTAG调试访问接口，或两种都包括。

JTAG：边界扫描测试协议（Joint Test Action Group，JTAG），是由国际联合测试行动组开发，对芯片进行测试的一种方式，可将其用于对MCU的程序进行载入与调试。JTAG能获取芯片寄存器等内容，或者测试遵守IEEE规范的器件之间引脚连接情况。

SWD：串行线调试（Serial Wire Debug，SWD）技术使用2针调试端口，是JTAG的低针数和高性能替代产品，通常用于小封装微控制器的程序写入与调试。SWD适用于所有ARM处理器，兼容JTAG。

5. 总线接口

Arm Cortex-M4F处理器提供先进的高性能总线（AHB-Lite）接口，包括ICode存储器接口、DCode存储器接口、系统接口和基于高性能外设总线（ASB）的私有外设总线（PPB）。

6. 浮点运算单元

处理器可以处理单精度32位指令数据，结合了乘法和累积指令用来提高计算的精度。此外，硬件能够进行加减法、乘除法以及平方根等运算操作，同时也支持所有的IEEE数据四舍五入模式。拥有32个专用32位单精度寄存器，也可作为16个双字寄存器寻址，并且通过采用解耦三级流水线来加快处理器运行速度。

2.3 CPU内部寄存器与存储器映像（重点）

CPU包含运算器、寄存器和控制器，一般框图中只画出运算器与控制，运算器负责执行算术运算、逻辑运算、移位、地址运算和转换等；寄存器用来暂存指令、数据和地址；控制器负责对指令译码，产生指令所需要的控制信号。在学习过程中，理解寄存器功能十分重要。

一个微型计算机地址线条数决定了CPU的寻找范围，在1.1.4节（微型计算机中的三总线）中，我们知道32位地址总线寻址空间为4GB，这个4GB空间，如何使用，则称为**存储器映像**。

下面首先从一般意义上给出寄存器基础知识及相关基本概念。

2.3.1 寄存器基础知识及相关基本概念

以程序员视角，从底层学习一个CPU，理解其内部寄存器用途是重要一环。计算机所有指令运行均由CPU完成，CPU内部寄存器负责信息暂存，其数量与处理能力直接影响CPU的性能，本小节先从一般意义上阐述寄存器基础知识及相关基本概念，下一小节介绍Arm Cortex-M4微处理器内部寄存器。

从共性知识角度及功能来看，CPU内至少应该有数据缓冲类寄存器、栈指针类寄存器、程序指针类寄存器、程序状态类寄存器及其他功能寄存器。

1. 数据缓冲类寄存器

CPU内数量最多的寄存器是数据缓冲用途的寄存器，名字可用寄存器英文Register的首字母加数字组成，如R0、R1、R2等等，不同CPU其种类不同，例如8086中的通用寄存器有八个，分别是AX，BX，CX，DX，SP，BP，SI，DI。

2. 栈指针类寄存器

在微型计算机的编程中，有全局变量与局部变量的概念。从存储器角度看，对一个具有独立功能的完整程序来说，全局变量具有固定的地址，每次读写都是那个地址。而在一个子程序中开辟的局部变量不是，用RAM中的哪个地址不是固定的，采用“后进先出（Last In First Out，LIFO）”原则使用一段RAM区域，这段RAM区域被称为栈区 。它有个栈底的地址，是一开始就确定的，当有数据进栈或出栈时，地址会自动连续变动（地址变动方向是增还是减，取决于不同计算机），不然就放到同一个存储地址中了，CPU中需要有个地方保存这个不断变化的地址，这就是栈指针（Stack Pointer）寄存器，简称SP。

这里的栈，其英文单词为Stack，在单片微型计算机中基本含义是RAM中存放临时变量的一段区域。现实生活中，Stack的原意是指临时堆放货物的地方，但是堆放的方法是一个一个码起来的，最后放好的货物，必须先取下来，先放的货物才能取，否则无法取。在计算机科学的数据结构学科中，栈是允许在同一端进行插入和删除操作的特殊线性表。允许进行插入和删除操作的一端称为栈顶(top)，另一端为栈底(bottom)；栈底固定，而栈顶浮动；栈中元素个数为零时称为空栈。插入一般称为进栈（PUSH），删除则称为出栈（POP）。栈也称为后进先出表。

3. 程序指针类寄存器

计算机的程序存储在存储器中，CPU中有个寄存器指示将要执行的指令在存储器中位置，这就是程序指针类寄存器。在许多CPU中，它的名字叫做程序计数寄存器（Program Counter ，PC）。在“1.1.3节”中谈及CPU时就指出，PC负责告诉CPU将要执行的指令在存储器的什么地方。

4. 程序运行状态类寄存器

CPU在进行计算过程中，会出现诸如进位、借位、结果为0、溢出等等情况，CPU内需要有个地方把它们保存下来，以便下一条指令结合这些情况进行处理，这类寄存器就是程序运行状态类寄存器。不同CPU其名称不同，有的叫做标志寄存器、有的叫做程序状态字寄存器等等，大同小异。在这类寄存器中，常用单个英文字母表示其含义，例如，N表示有符号运算中结果为负（Negative）、Z表示结果为零（Zero）、C表示有进位（Carry）、V表示溢出（Overflow）等。

5. 其他功能寄存器

不同CPU中，除了具有数据缓冲、栈指针、程序指针、程序运行状态类等寄存器之外，还有表示浮点数运算、中断屏蔽 等寄存器。

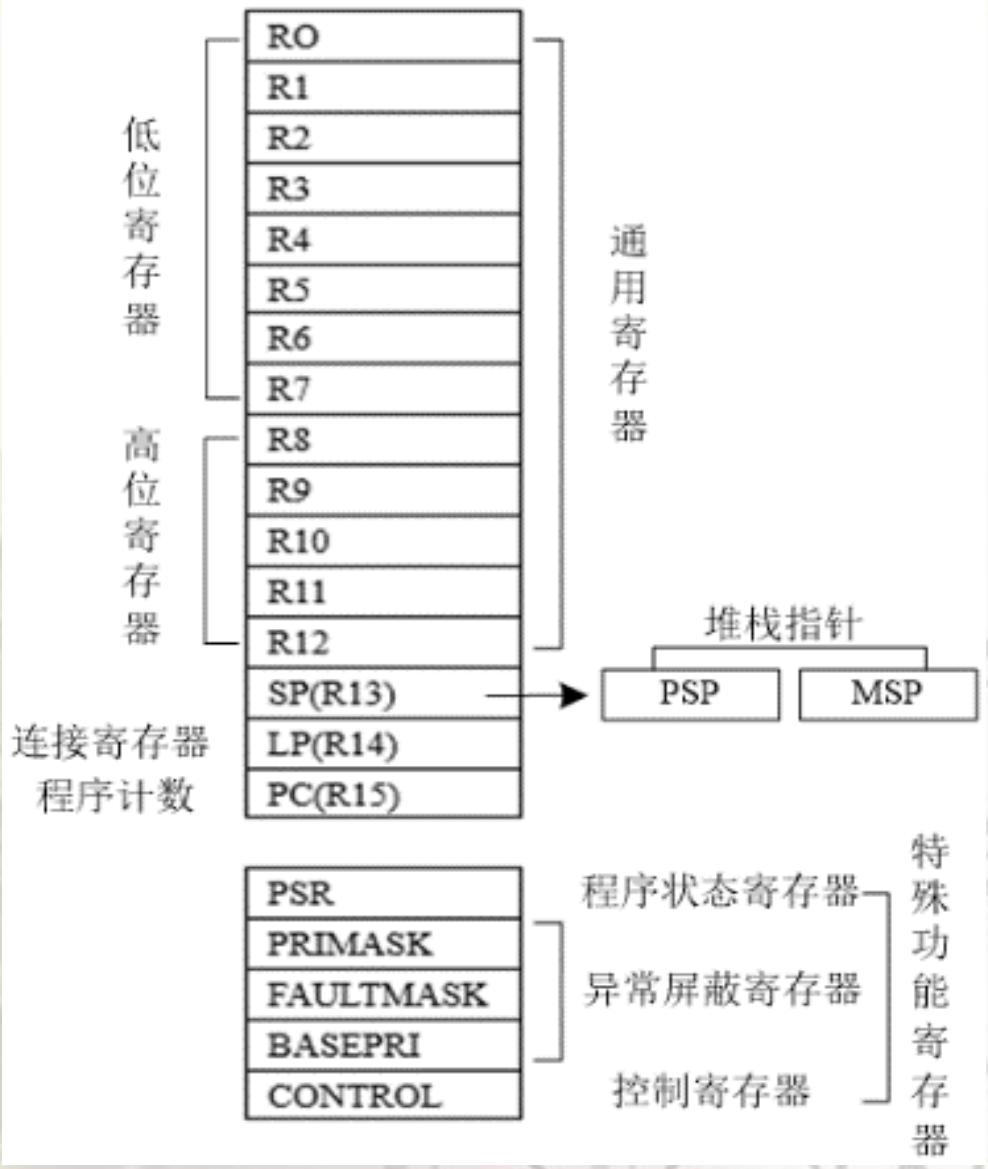
所谓中断屏蔽，就是中断进来也不理它。中断是暂停当前正在执行的程序，先去执行一段更加紧急程序的一种技术，它是计算机中的一个重要概念，将在第8章较为详细的阐述。中断屏蔽标志，就是表示是否允许某种中断进来的标志。

2.3.2 Arm Cortex-M4内部寄存器

这是一个具体的CPU，按照从一般到个别的哲学原理，我们来认识一个具体CPU的内部寄存器，了解其功能，随后第3章学习指令系统将与它们打交道，第4章之后的汇编语言编程也是直接与它们打交道。以下将Arm Cortex-M4内部寄存器主要有：通用寄存器R0~R12，栈指针SP、连接寄存器LR、程序计数寄存器PC、程序状态字寄存器（xPSR）、特殊功能寄存器等。

1. 通用寄存器R0~R12

R0～R12是最具“通用目的”的32位通用寄存器，用于数据缓冲操作。分为两组，一组被称为低位寄存器，R0～R7，它们能够被所有通用寄存器指令访问，另一组被称为高位寄存器，R8～R12，它们能够被所有32位通用寄存器指令访问，而不能被所有的16位指令访问。



2．栈指针SP

寄存器R13被用作栈指针（SP），用于访问RAM中的栈区。在Arm架构中，SP的最低两位被忽略，就是相当于SP的最低两位永远是0，所以SP的值是**4的整数倍**，那么SP指向的RAM地址也是4的整数倍，即是按照4字节对齐的。Arm是32位机，机器字长为32位，4字节对齐即表示栈中的数据存储是按照字对齐的。

书上图2-3中，给出的SP有个箭头“→”表示SP有两个名字，分别是：PSP、MSP，主栈指针MSP是复位后缺省使用的栈指针，用于操作系统内核以及异常处理例程（包括中断服务程序）。Handler模式 总是使用主栈指针（MSP），但是也可以配置成“Thread”模式来使用MSP或者进程栈指针（PSP）。

这里的Handler模式也称为处理模式，是执行中断服务程序ISR等异常处理；Thread模式也称为线程模式，是执行普通的用户程序。所有程序的执行均可以看成两个路线：一条为正常执行的线路，有时也称主程序线，就是所谓的线程模式，一条是中断线，也就是正常正常执行的线路被打断，转去执行“中断服务程序”，然后返回正常路线执行

3. 连接寄存器LR

寄存器R14也称作连接寄存器（LR），用于保存函数或子程序调用时的返回地址。LR也被用于异常返回。在其他情况下，可以将R14作为通用寄存器来使用(**用栈来保存返回地址**)。

特别说明：连接寄存器LR存在的价值在于：加快了一级子程序的进入与返回速度。这是因为，返回地址存于内部寄存器LR中，比存于RAM中访问速度快。

4. 程序计数寄存器PC

寄存器R15是程序计数寄存器（PC），指示将要执行的指令在存储器中位置。复位的时候，处理器的硬件机制自动将复位向量值放入PC。如果修改它的值，就能改变程序的执行流。该寄存器的第0位若为0，则指令总是按照字对齐或者半字对齐。

5．程序状态字寄存器（xPSR）

数据位	31	30	29	28	27	26～25	24	23～20	19～16	15～10	9	8～0
APSR	N	Z	C	V	Q				GE[3:0]			
IPSR												异常号
EPSR						ICI/IT	T			ICI/IT		
xPSR	N	Z	C	V	Q	ICI/IT	T			ICI/IT		异常号

程序状态字寄存器在内部分为以下几个子寄存器：APSR、IPSR、EPSR。

（1）应用程序状态寄存器（APSR）：显示算术运算单元ALU状态位的一些信息：

负标志N：若结果最高位为1，相当于有符号运算中结果为负，则置1，否则清0。零标志Z：若结果为0，则置1，否则清0。进位标志C：若有最高位的进位（减法为借位），则置1，否则清0。溢出标志V：若溢出，则置1，否则清0。

（2）中断程序状态寄存器（IPSR）：每次异常完成之后，会实时更新IPSR内异常号

（3）执行程序状态寄存器（EPSR）：T标志位指示当前运行是否是Thumb指令（16位）

6. 特殊功能寄存器（了解）

（1）中断屏蔽寄存器（PRIMASK）
（2）错误屏蔽寄存器（FAULTMASK）
（3）基本优先级屏蔽寄存器（BASEPRI）
（4）控制寄存器（CONTROL）

7. 浮点寄存器(了解)

浮点控制寄存器只在Cortex-M4F处理器中存在，其中包含了用于浮点数据处理与控制的寄存器

由于中断屏蔽、错误屏蔽、基本优先级屏蔽、控制和浮点等寄存器，比较复杂，也不常用，书中不再介绍，需要深入了解的读者，可参阅电子资源中“..\02-Document\《微型计算机原理及应用》补充阅读材料”。

2.3.3 Arm Cortex-M4存储器映像

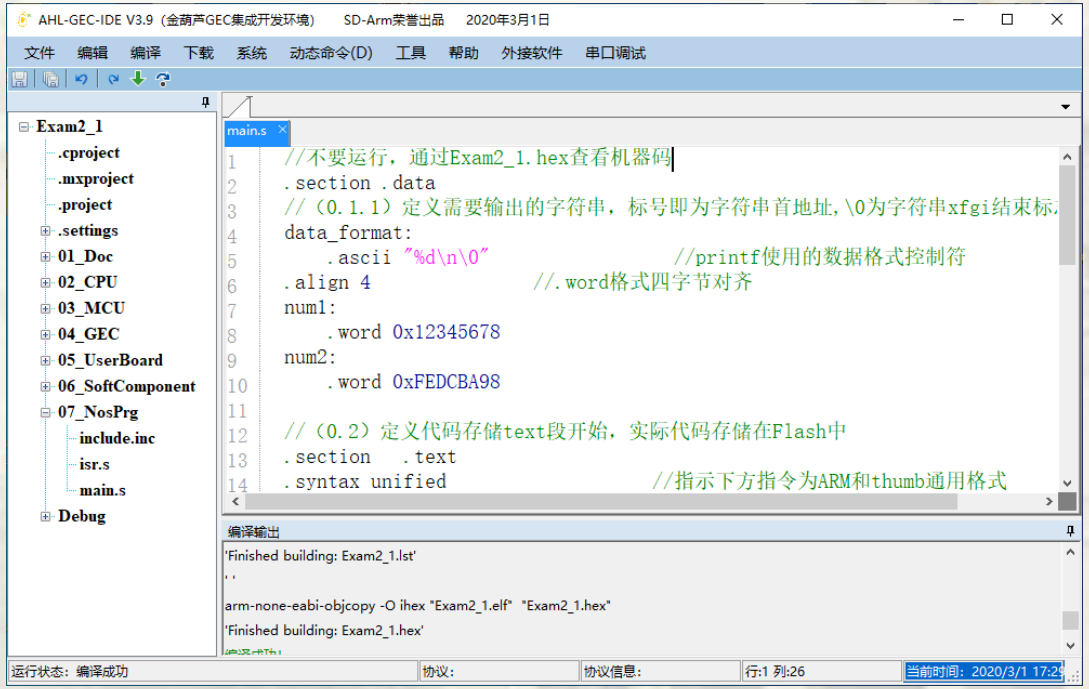
该处理器直接寻址空间为4GB，地址范围是：0x0000_0000～0xFFFF_FFFF。这里所说的存储器映像其含义是指把这4GB空间当做存储器来看待，分成若干个区间，以安排一些实际的物理资源。

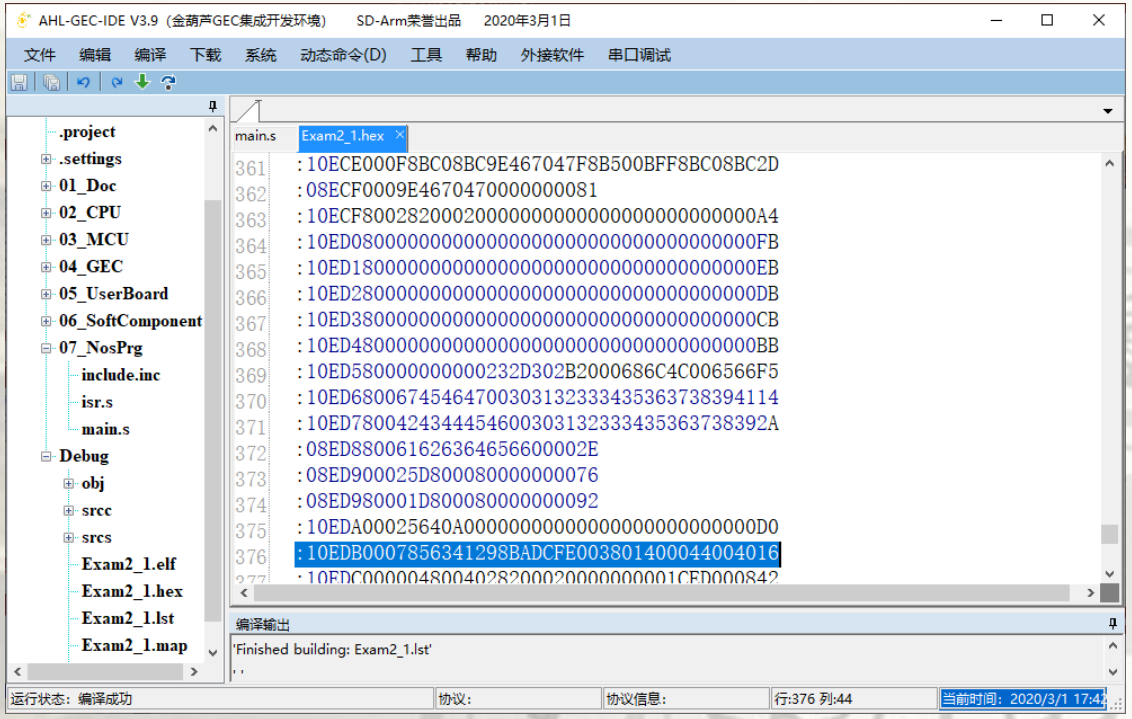
Arm定出的条条框框是粗线条的，它依然允许芯片制造商灵活地分配存储器空间，以制造出各具特色的MCU产品。

系统保留	511MB	0xFFFFFFFF 0xE0100000
私有外部总线-外部	16MB	0XE0FFFFFF 0xE0040000
私有外部总线-内部	256KB	0XE003FFFF 0xE0000000
外部设备	1.0GB	0xDFFFFFFF 0xA0000000
外 RAM	1.0GB	0x9FFFFFFF 0x60000000
外围设备	0.5GB	0x5FFFFFFF 0x40000000
SRAM	0.5GB	0x3FFFFFFF 0x20000000
代码	0.5GB	0x1FFFFFFF 0x00000000

数据存储的小端格式与大端格式：小端格式：字的低字节存储在低地址中，字的高字节存储在高地址中。
大端格式：字的低字节存储在高地址中，字的高字节存储在低地址中。（芯片厂家决定）

利用AHL-GEC-IDE环境打开电子资源中：.. \04-Software\Exam2_1工程
编译查看:0x12345678,存放形式为：偏移地址FAD0～3分别存放：78 56 34 12 （小端格式）





作业3：

1．STM32L431芯片的内部微处理器有哪些寄存器？简述各寄存器的作用。

2．RAM存储区和Flash存储区的访问特点？给出STM32L431芯片的RAM存储区和Flash存储区的大小及地址范围。

3．简述微处理器中存储器映像的含义，给出STM32L431芯片Flash接口模块的存储器映像地址（范围）。

作业提交网址：见群文件

作业文件命名规则(word文档）：学号+姓名

说明：如果手写，可以拍照后插入到word文档提交