第 2 章
嵌入式系统硬件体系结构

本章要点

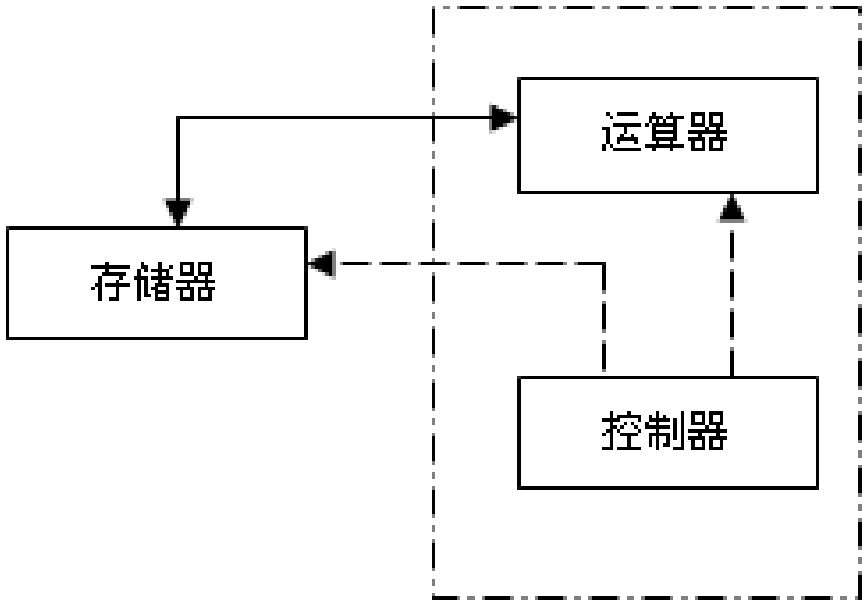
- 学习完本章读者将掌握如下内容：
- 1、嵌入式硬件的相关基础知识
- 2、嵌入式硬件平台基本组成
- 3、 ARM 系列微处理器简介

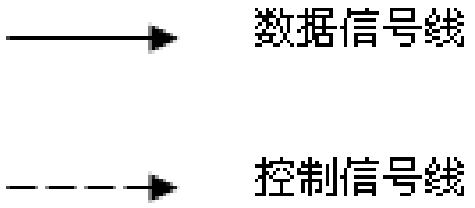
2.1 相关基础知识

2.1.1、嵌入式微处理器

1、嵌入式微处理器的组成

中央微处理器，简称 CPU，它是计算机中最重要的一个部分，它决定嵌入式系统的主要功能特性。CPU 又由运算器和控制器两大部分组成。





（虚线框内为 CPU）

2、微处理器的重要指标

(1) 主频、倍频、外频

主频是指 CPU 的时钟频率，简单地说也就是 CPU 运算时的工作频率。

外频就是系统总线的工作频率。

倍频则是指 CPU 外频与主频相差的倍数。三者的关系：主频＝外频 × 倍频。

(2) 缓存

L1 Cache(一级缓存)

L2 Cache(二级缓存)

L3 Cache(三级缓存)

2.1.2 嵌入式微处理器的流水线技术

1、微处理器的流水线技术

- 通常微处理器在处理一条指令要经过三个步骤：取指（从存储器装载一条指令）、译码（识别将要被执行的指令）、执行（处理指令并将结果写回寄存器）。
- 流水线技术通过多个功能部件并行工作来缩短程序执行时间，提高微处理器的运行效率和吞吐率。

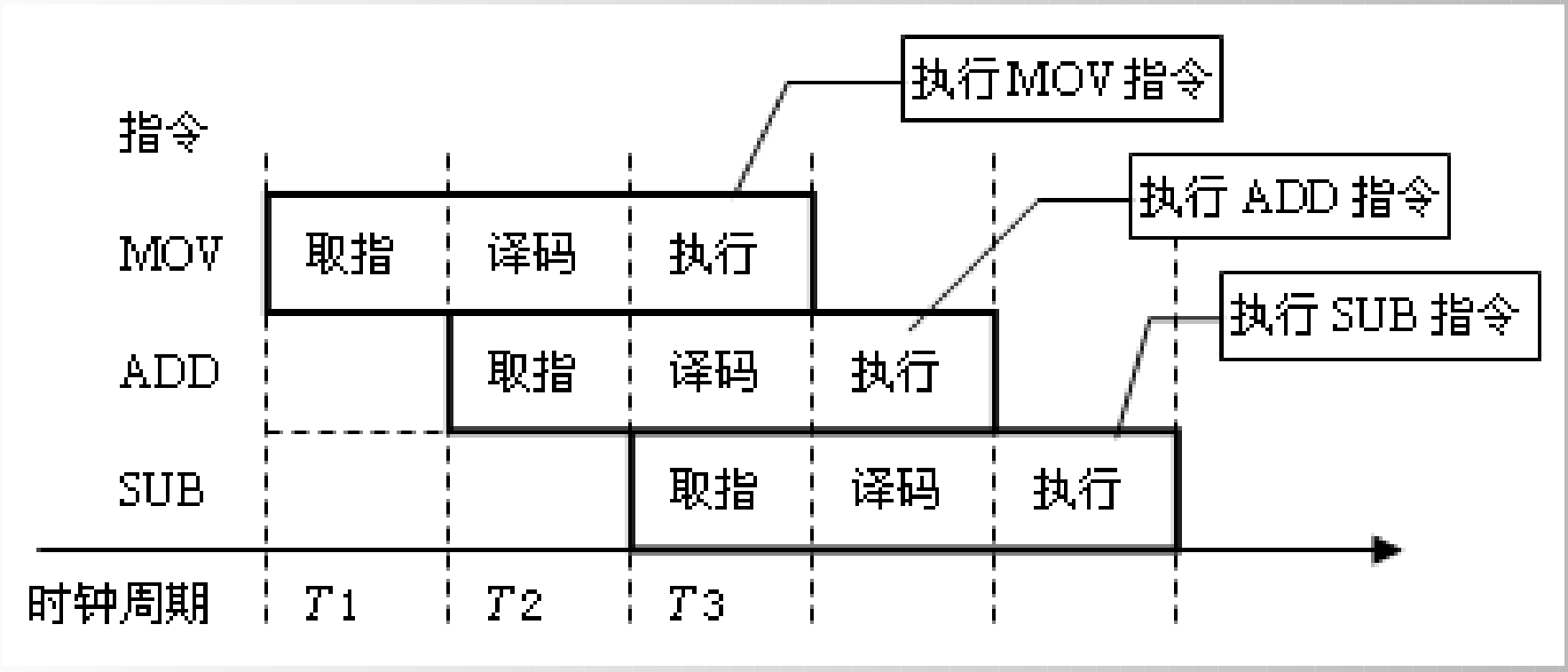
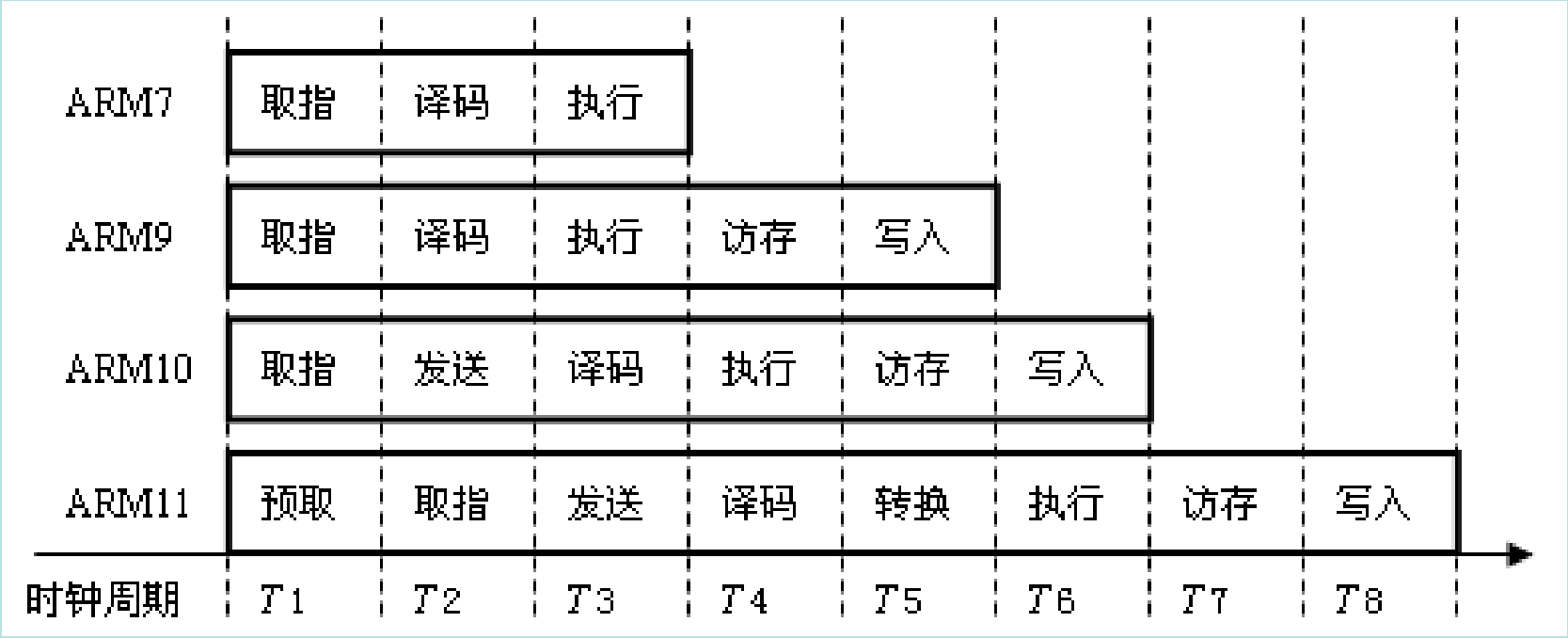


图 2.2　微处理器的三级流水线技术

微处理器在同一时间周期并行执行若干条指令的取指、译码、执行操作，其运行效率是逐条执行指令的 3 倍。

2、嵌入式微处理器的流水线

嵌入式微处理器 ARM7 采用三级流水线，而 ARM9 则采用五级流水线技术，而 ARM11 则更是使用了 8 级流水线。



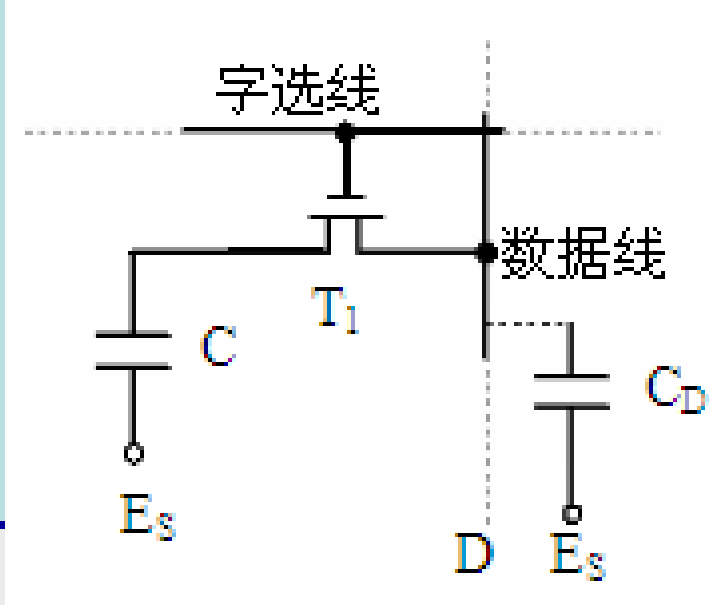
2.1.3 寄存器与存储器

1、寄存器

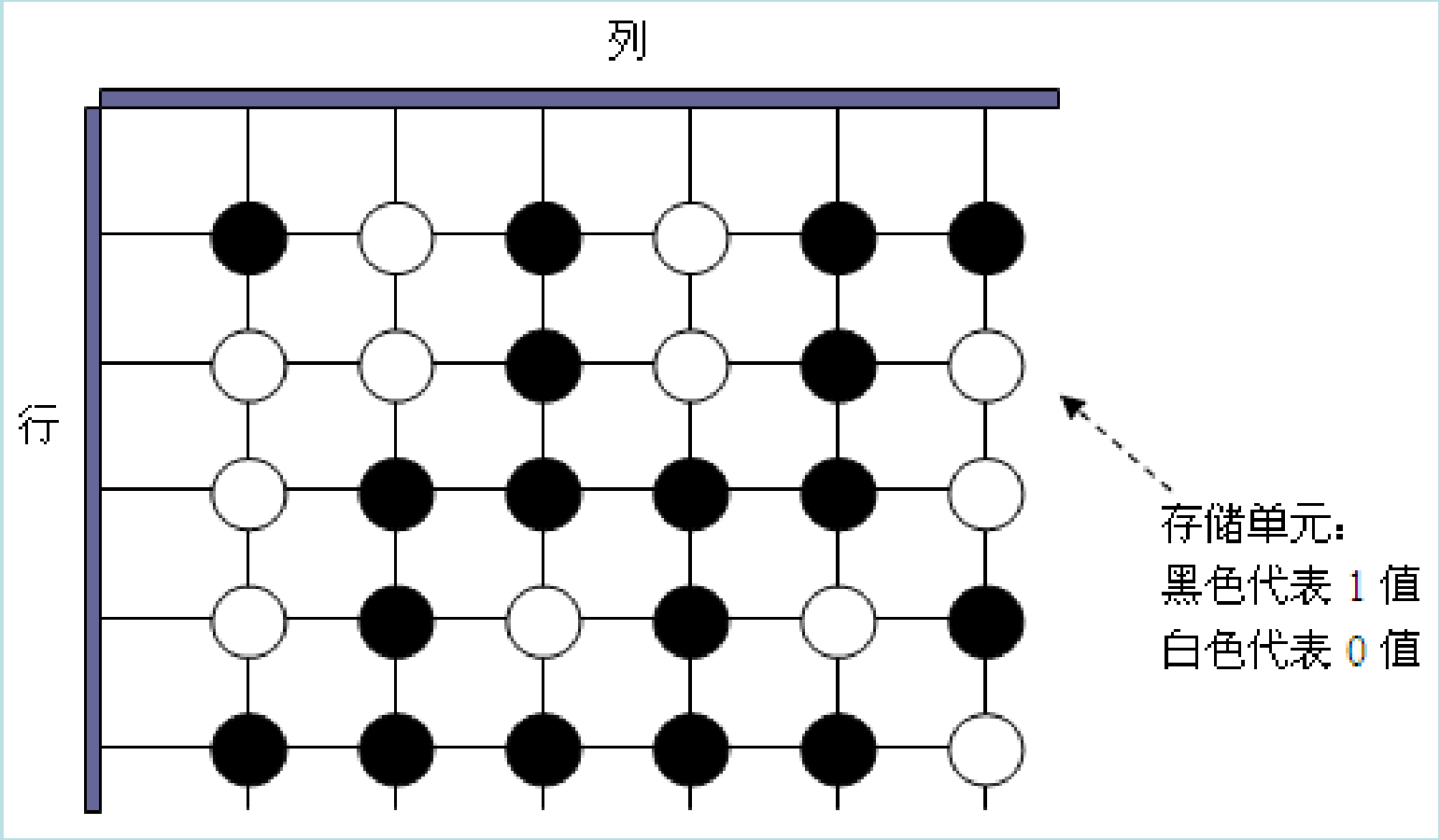
寄存器（register）是CPU的组成部分，是CPU内部用来存放数据的一些小型存储区域，用于暂时存放参与运算的数据和运算结果。

2、随机存取存储器（RAM）

嵌入式系统的存储器中，最为常见的一种是动态随机存取存储器（DRAM），在DRAM中晶体管和电容器合在一起就构成一个存储单元，代表一个数据位元。电容器保存一位二进制信息位——0或1（电容器有无电荷表示数据1或0）。



将很多 DRAM 基本存储单元连接到同一个列线（位线）和同一个行线（字线）组成一个矩阵结构，位线和字线相交，就形成了存储单元的地址



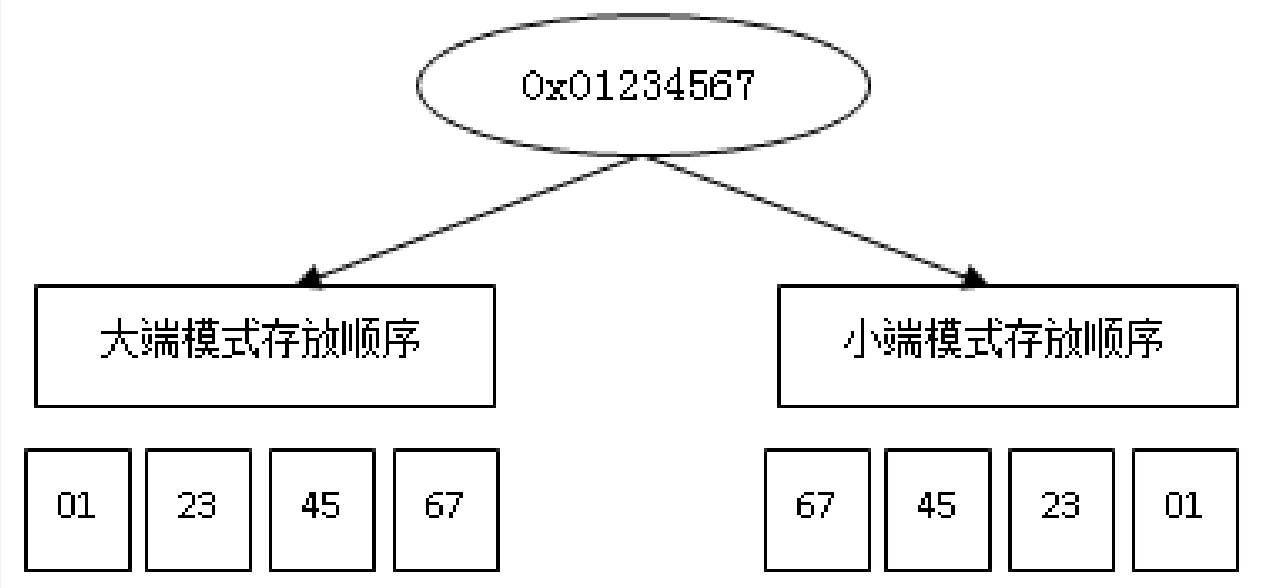
3. 内存中数据存放的大小端模式

在嵌入式系统中，存储是以字节为单位的，每个地址单元都对应着一个字节，一个字节为 8 位。对于位数大于 8 位的处理器，例如 16 位或者 32 位的处理器，由于寄存器宽度大于一个字节，如何安排多个字节的存储，这就有了大端存储模式和小端存储模式。

大端模式：数据的高字节保存在内存的低地址中，而数据的低字节保存在内存的高地址中。

小端模式：数据的高字节保存在内存的高地址中，而数据的低字节保存在内存的低地址中。

- 例如，一个 32 位宽的数的十六进制表示为 0x01234567（地址从低位开始存放），如果是小端模式，则存储方式为： 0x67 0x45 0x23 0x01，如果是大端模式，则存储方式为： 0x01 0x23 0x45 0x67 。如图所示。



2.1.4 总线

- 总线（BUS）是接口电路与CPU或者接口电路与I/O外部设备之间连接的主要形式，是各功能部件之间传送信息的公共通路。
- 采用一组公共的信号线作为嵌入式系统各部件之间的通信线，这组公共信号线就称为总线。

通信协议

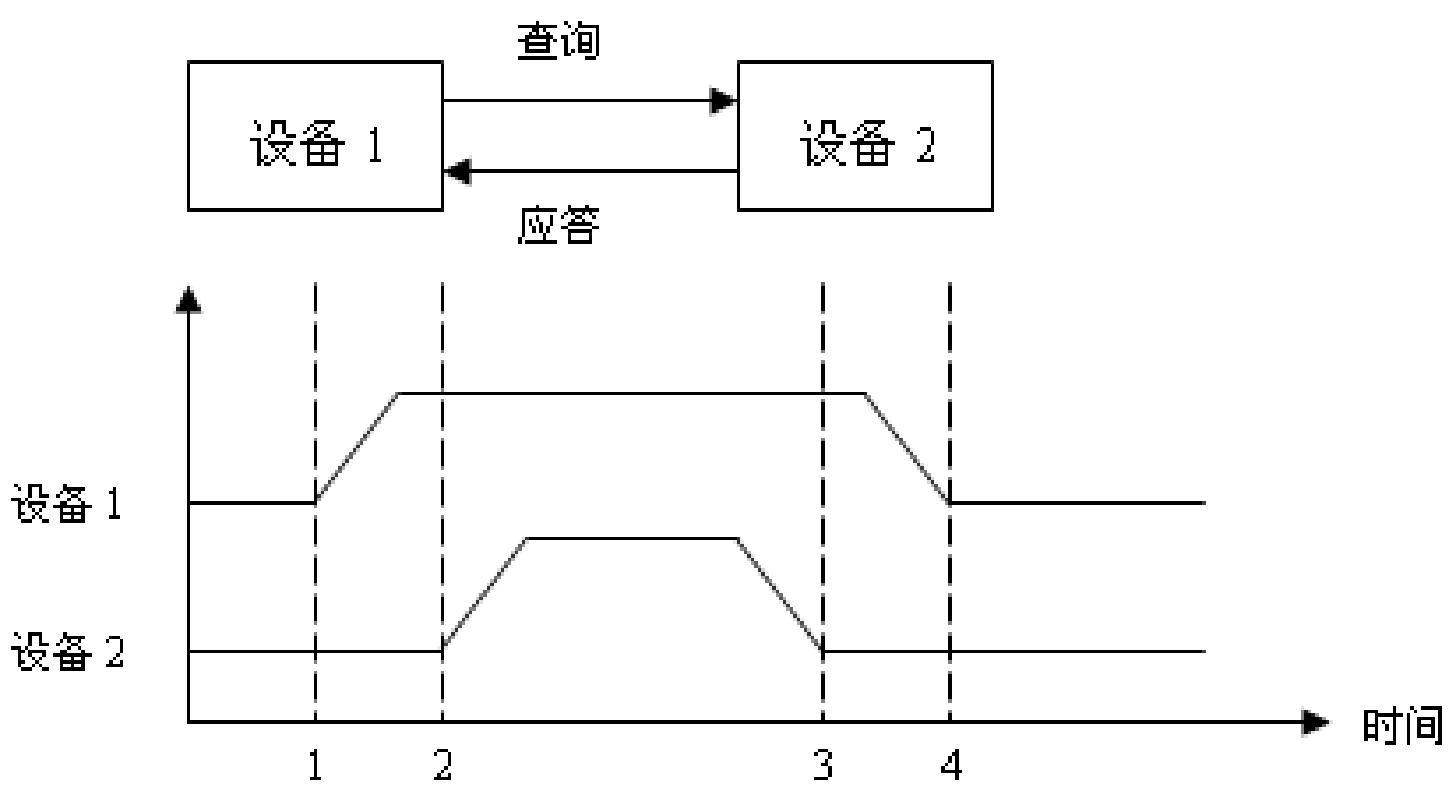
- 通信协议是指通信双方的一种约定。约定包括对数据格式、同步方式、传送速度、传送步骤等问题做出统一规定，通信双方必须共同遵守。
- （1）总线时序协议
- （2）异步时序协议的握手协议
- （3）总线仲裁方式
- （4）总线标准

（1）总线时序协议

- 同步时序：
- 总线上所有事件共用同一时钟脉冲进行操作过程的控制，所有事件都在时钟周期的开始发生。
- 异步时序：
- 操作由源或目的模块发出的特定信号确定。双方相互提供联络信号。

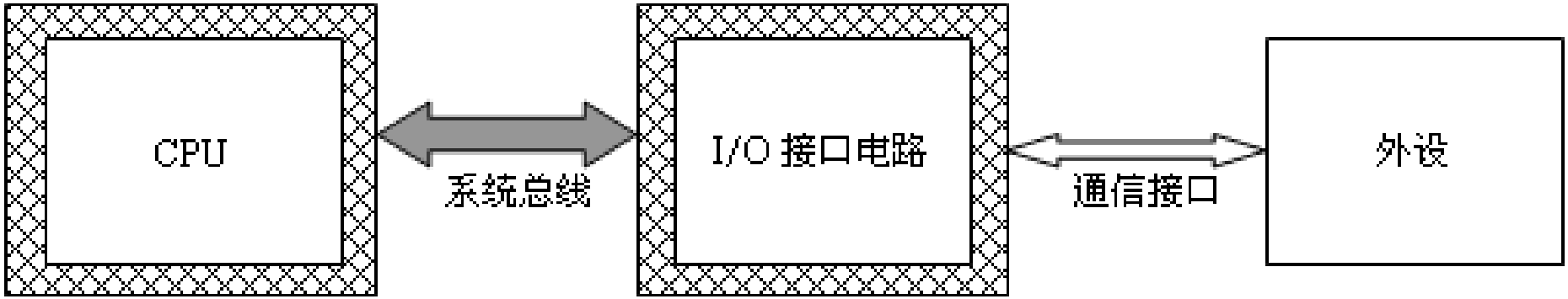
（2）异步时序协议的握手协议

- 握手协议是总线异步时序的基本构件。
- 握手协议数据传送过程的 4 个周期：



2.1.5 I/O 端口

- I/O 端口又称为 I/O 接口，它是微处理器对外控制和信息交换的必经之路，是 CPU 与外部设备连接的桥梁，它在 CPU 与外部设备之间起信息转换和匹配的作用。 I/O 端口有串行和并行之分，串行 I/O 端口一次只能传送一位二进制数信息，而并行 I/O 端口一次能传送一组二进制数信息。



- 图 2.8　　I/O 接口电路的位置

- CPU 对外设 I/O 端口物理地址的编址方式有两种：
- 一种是 I/O 映射方式（ I/O － mapped ）；
- 另一种是内存映射方式（ Memory － mapped ）。
- 具体采用哪一种则取决于 CPU 的体系结构。

2.1.6 中断

- 中断方式是指，当外部设备准备与 CPU 进行数据传输时，外部设备首先向 CPU 发出中断请求， CPU 接收到中断请求并在一定条件下，暂时停止原来的程序并执行中断服务处理程序，执行完毕以后再返回原来的程序继续执行。

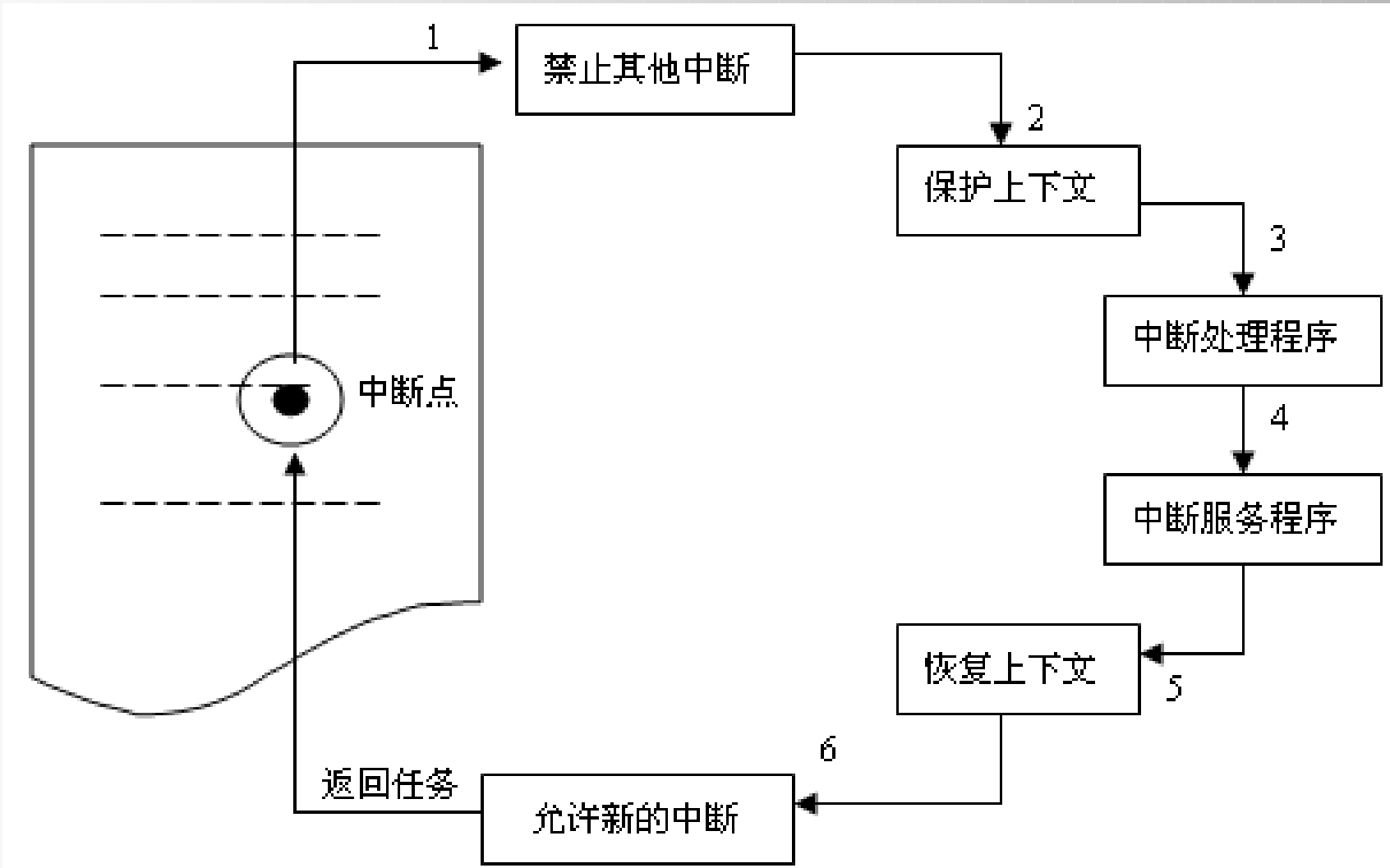
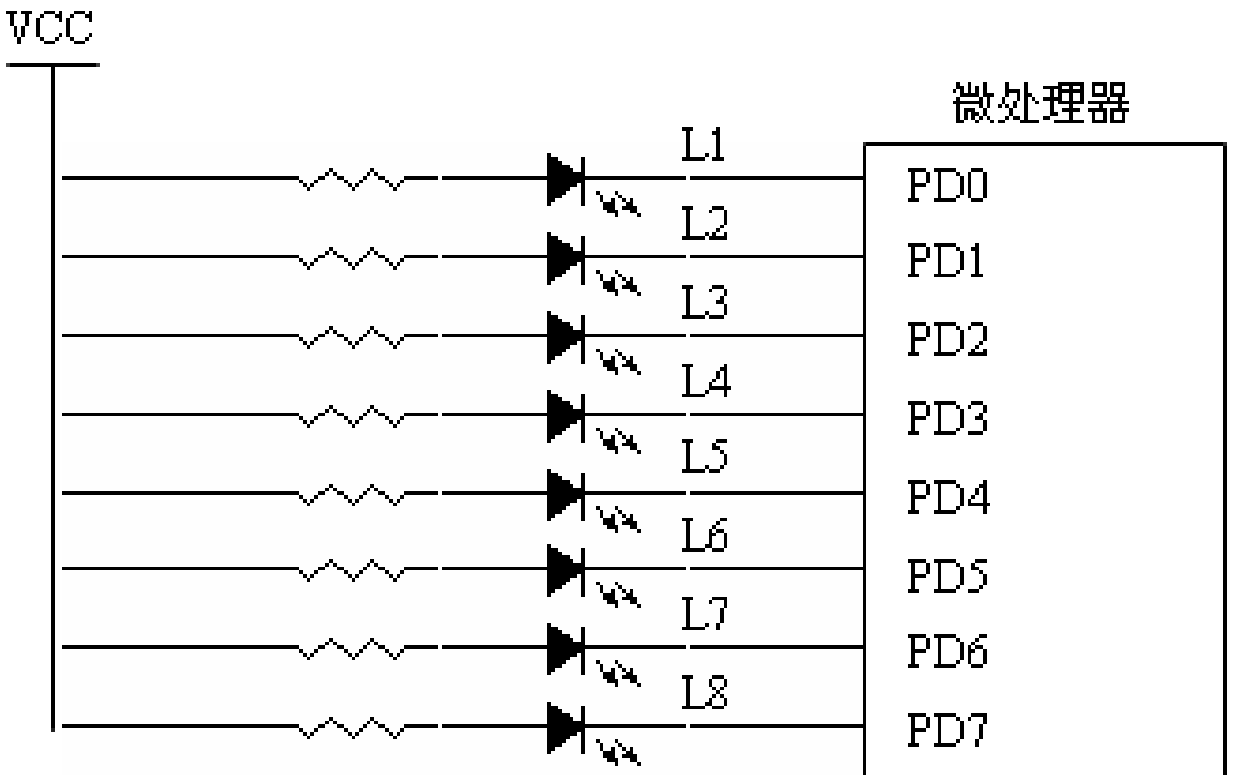


图 2.9　中断处理的各个阶段

2.1.7 数据编码

- 设用微处理器控制一串彩灯（发光二极管）的亮灭。如图所示。



- 我们设不发光的口线（高电平）为 1，发光的口线（低电平）为 0 。
- 当彩灯 L1 发光时， PD0 口线为低电平，而其余口线均为高电平。

则可以表示为以下对应值：

PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
1	1	1	1	1	1	1	0

用二进制数表示为： 11111110 。

用十六进制编码 ，其值为： FEH 。

- 再如，要彩灯 L8 发光，其余均不发光，则有

则可以表示为以下对应值：

PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
0	1	1	1	1	1	1	1

- 用二进制数表示为： 01111111 。
- 十六进制编码为： 7FH 。

- 若希望两边亮，中间暗，则：

PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
0	1	1	1	1	1	1	0

- 十六进制编码为： 7EH 。

2.2 嵌入式系统硬件平台

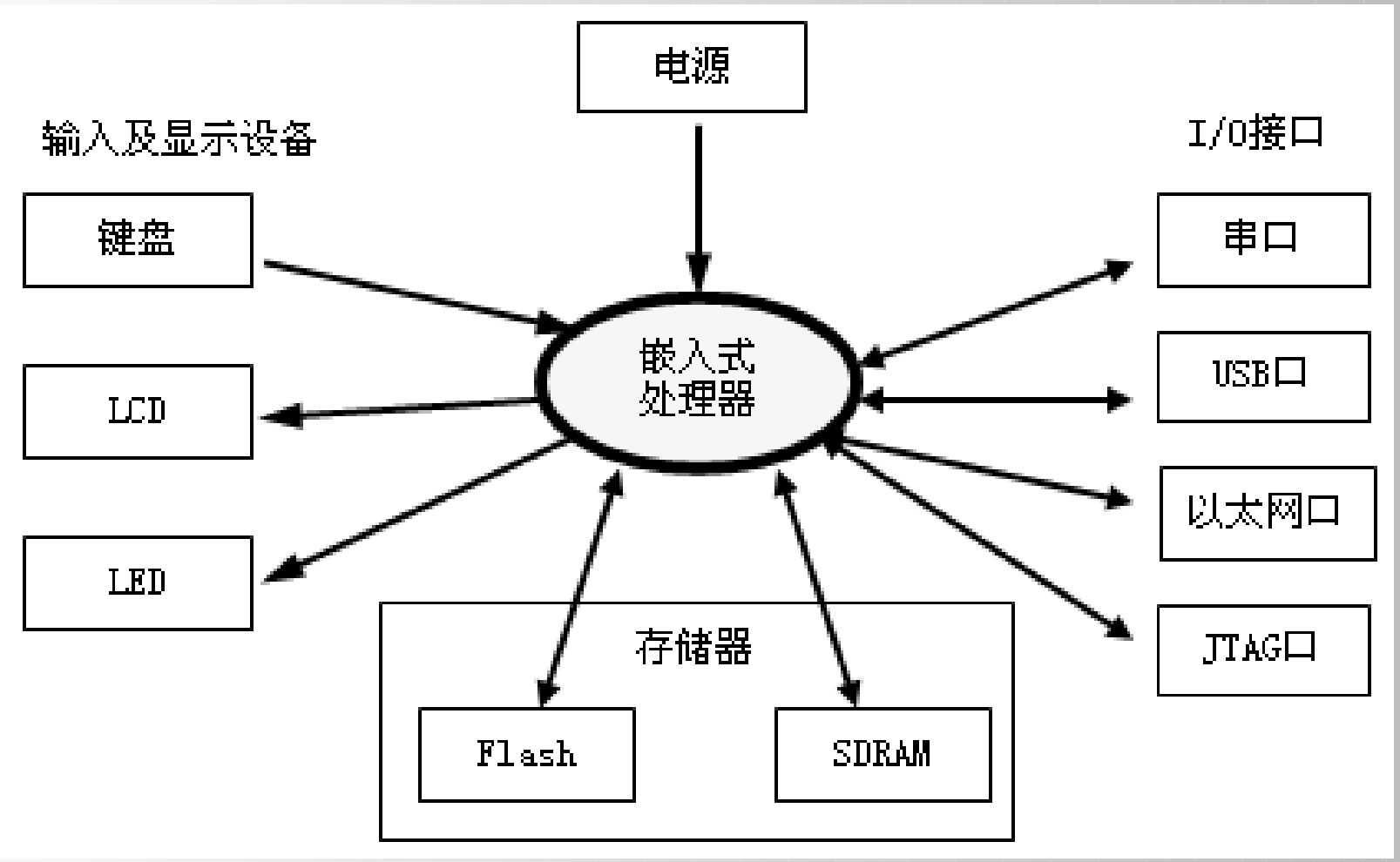


图 2.12　嵌入式系统硬件结构

1、嵌入式处理器

- 嵌入式处理器通常包括几个部分：处理器内核、地址总线、数据总线、控制总线、片上 I/O 接口电路及辅助电路（如时钟、复位电路等）。
- 嵌入式处理器可以分为 3 类：

  嵌入式微处理器、

  嵌入式微控制器、

  嵌入式 DSP （ Digital Signal Processor，数字信号处理器），

2、嵌入式系统中的存储设备

（1） RAM、 SRAM、 DRAM

- RAM 即是我们通常所说的内存。 RAM 又可分为 SRAM （静态存储器）和 DRAM （动态存储器）。

（2） Flash

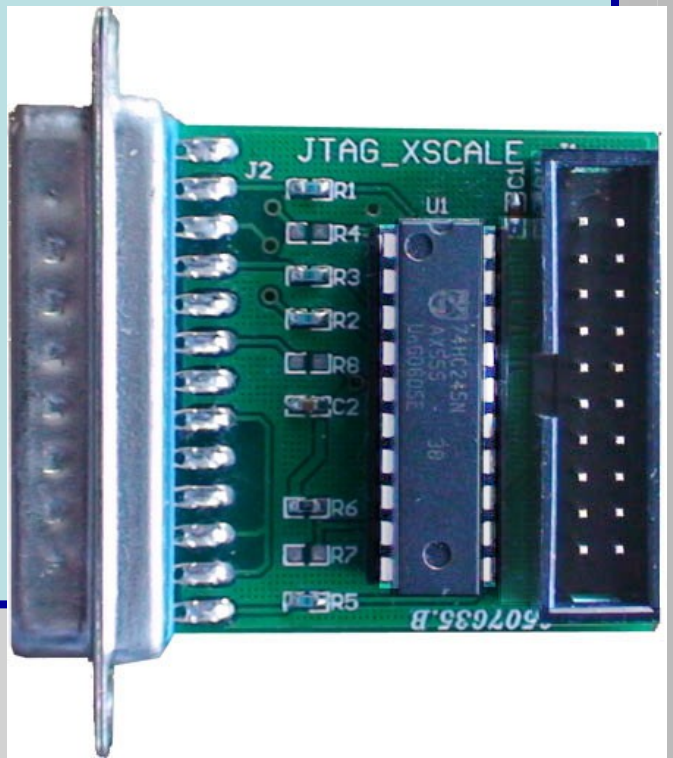
- Flash 是一种非易失闪存，它具有和 ROM 一样掉电后数据不会丢失的特性。 Flash 是目前嵌入式系统中广泛采用的主流存储器，它的主要特点是按整体 / 扇区擦除和按字节编程，具有低功耗、高密度、小体积等优点。

Flash 分为 NOR Flash 、 NAND Flash 两种。

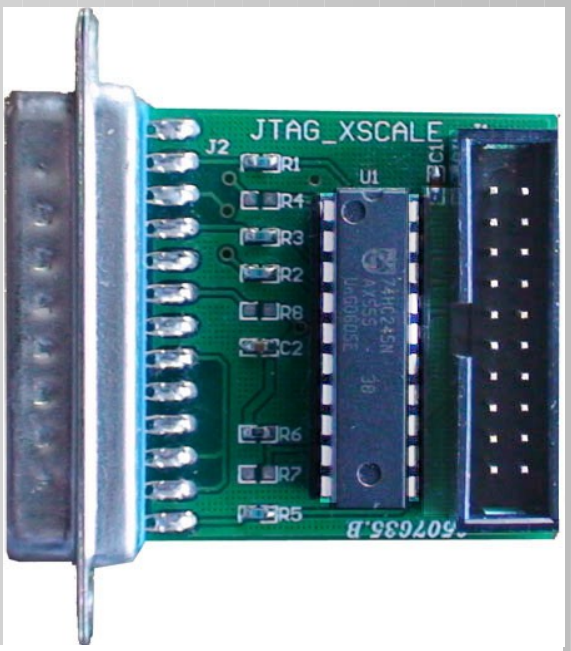
- NOR Flash 的特点是在芯片内执行，可以直接读取芯片内储存的数据，因而速度比较快。应用程序直接在 Flash 内运行，不必把代码读到系统 RAM 中运行。
- NAND Flash 不能直接在 Flash 内运行应用程序，需要将数据复制到 RAM 中运行。
- NAND Flash 的特点是容量大。

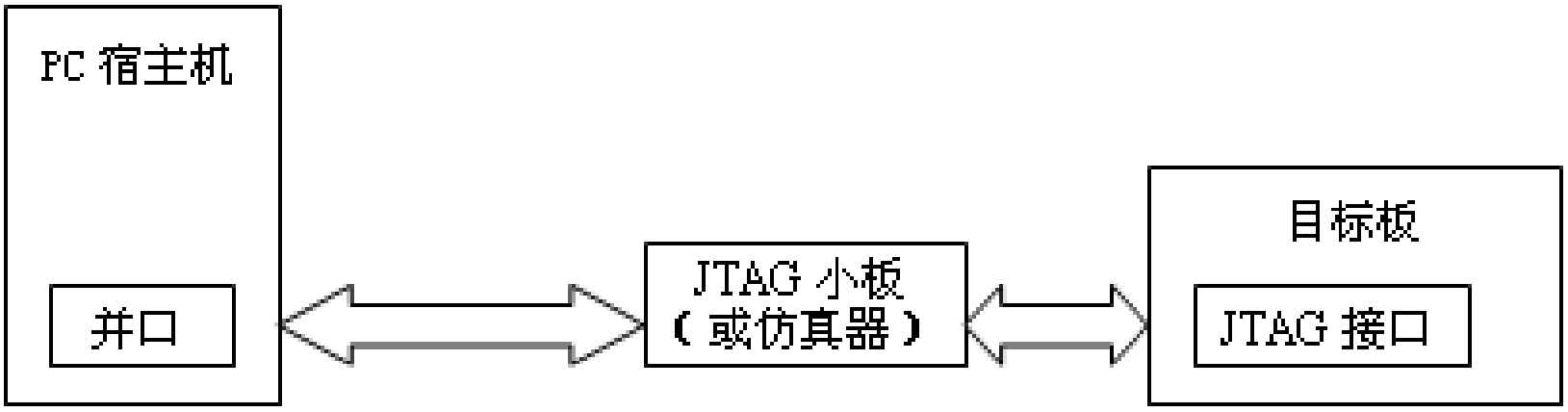
3、JTAG 接口

- JTAG（Joint Test Action Group，联合测试行动小组）是一种国际标准测试协议（IEEE 1149.1 兼容），主要用于芯片内部测试。



- 我们经常用简易 JTAG 接口直接烧写嵌入式系统 Flash 存储器。这种烧写方式是通过一根并口电缆和一块信号转换集成电路板以建立 PC 机与开发板之间的通信





2.3 ARM 微处理器体系

2.3.1
ARM 公司及 ARM 体系结构

1、ARM 公司简介

- ARM （Advanced RISC Machines），既可以认为是一个公司的名字，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字。
- ARM 公司是专门从事基于 RISC 技术芯片设计开发的公司，作为知识产权供应商，本身不直接从事芯片生产，靠转让设计许可，由合作公司生产各具特色的芯。

2、ARM 微处理器体系

- — ARM7 系列
- — ARM9 系列
- — ARM10 系列
- — Cortex-M 系列
- — Cortex-R 系列
- — Cortex-A 系列

3、哈佛总线体系结构

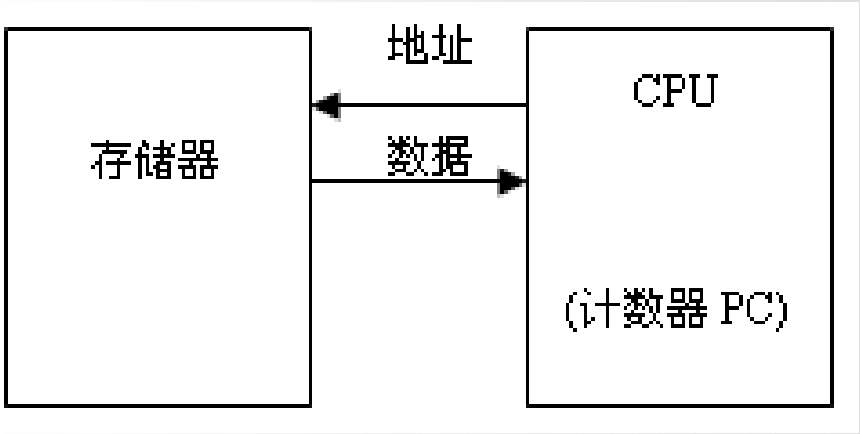


图 2.15 冯·诺依曼结构

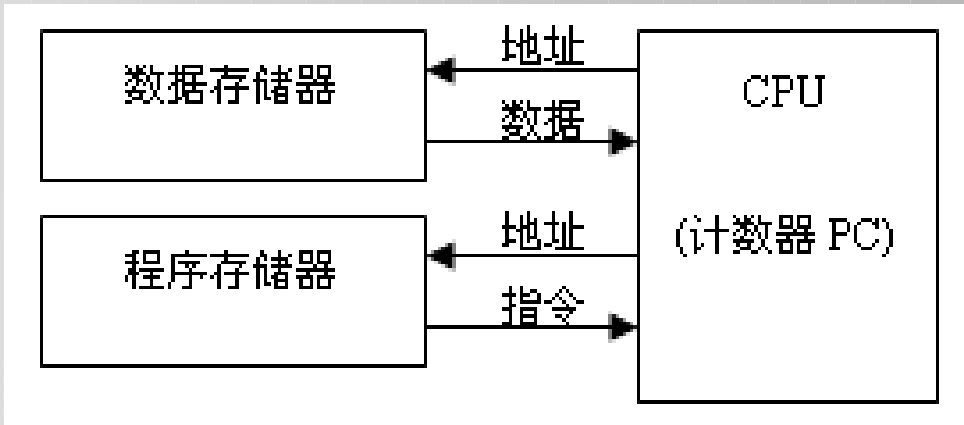


图 2.16 哈佛结构

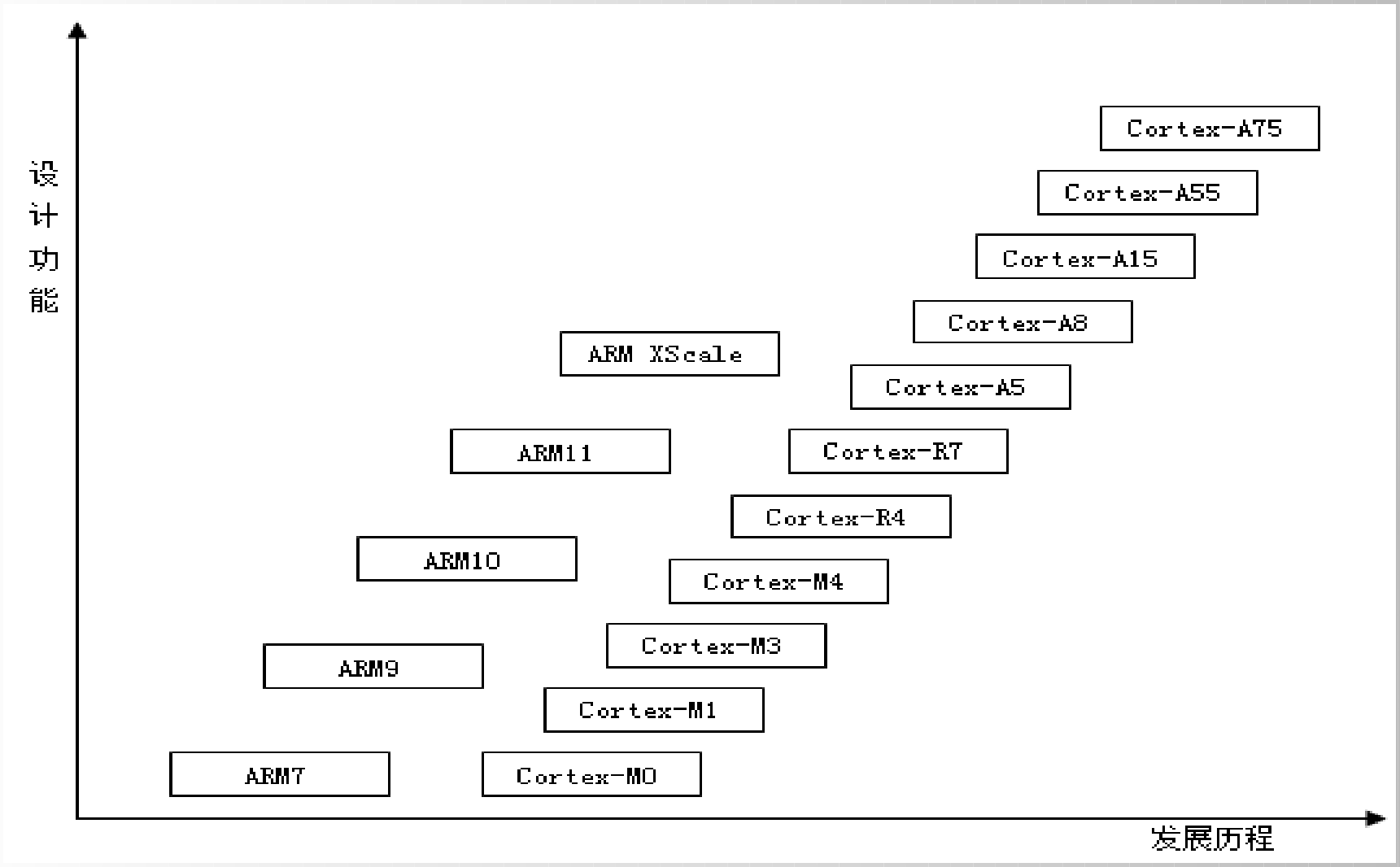
4、ARM 微处理器的特点

- （1）体积小、低功耗、低成本、高性能；
- （2）支持 Thumb(16 位)/ARM(32 位) 双指令集，能很好的兼容 8/16 位器件；
- （3）大量使用寄存器，指令执行速度更快；
- （4）大多数数据操作都在寄存器中完成；
- （5）寻址方式灵活简单，执行效率高；
- （6）指令长度固定。

2.3.2 ARM 系列微处理器简介

- 1、 ARM7 系列微处理器
- 2、 ARM9 系列微处理器
- 3、 Xscale 系列微处理器
- 4、 Cortex 系列微处理器

ARM Cortex 系列微处理器是 ARM 公司推出的第二代微处理器，它的发展历程如图所示。



2.4 微处理器的结构

2.4.1 RISC 体系结构和 ARM 设计思想

1、RISC 体系结构

- 在 CISC 指令集的各种指令中，其使用频率却相差悬殊，大约有 20 %的指令被反复使用，占整个程序代码的 80 %。而余下的 80 %的指令却不经常使用，在程序设计中只占 20 %。
- RISC 结构优先选取使用频率最高的简单指令，避免复杂指令；将指令长度固定，指令格式和寻址方式种类减少；以控制逻辑为主。

2、ARM 设计思想

- 1、ARM 微处理器被设计成较小的核，降低功耗，延长电源的使用时间。
- 2、存储量有限，这就要求嵌入式系统需要使用高密度代码。
- 3、嵌入式系统对成本敏感，一般选用速度不高，成本较低的存储器，以降低系统成本。
- 4、ARM 内核不是一个纯粹的 RISC 体系结构，这是为使它能够更好地适应其嵌入式的应用领域。
- 对嵌入式系统的应用项目来说，系统的关键并不单纯在于微处理器的速度，而在于系统性能、功耗和成本。

2.4.2 ARM Cortex 微处理器结构的最小系统设计

1、什么是最小系统

- 嵌入式微处理器芯片自己是不能独立工作的，需要一些必要的外围元器件给它提供基本的工作条件。
- 一个 ARM 最小系统一般包括：

（1） ARM 微处理器芯片，

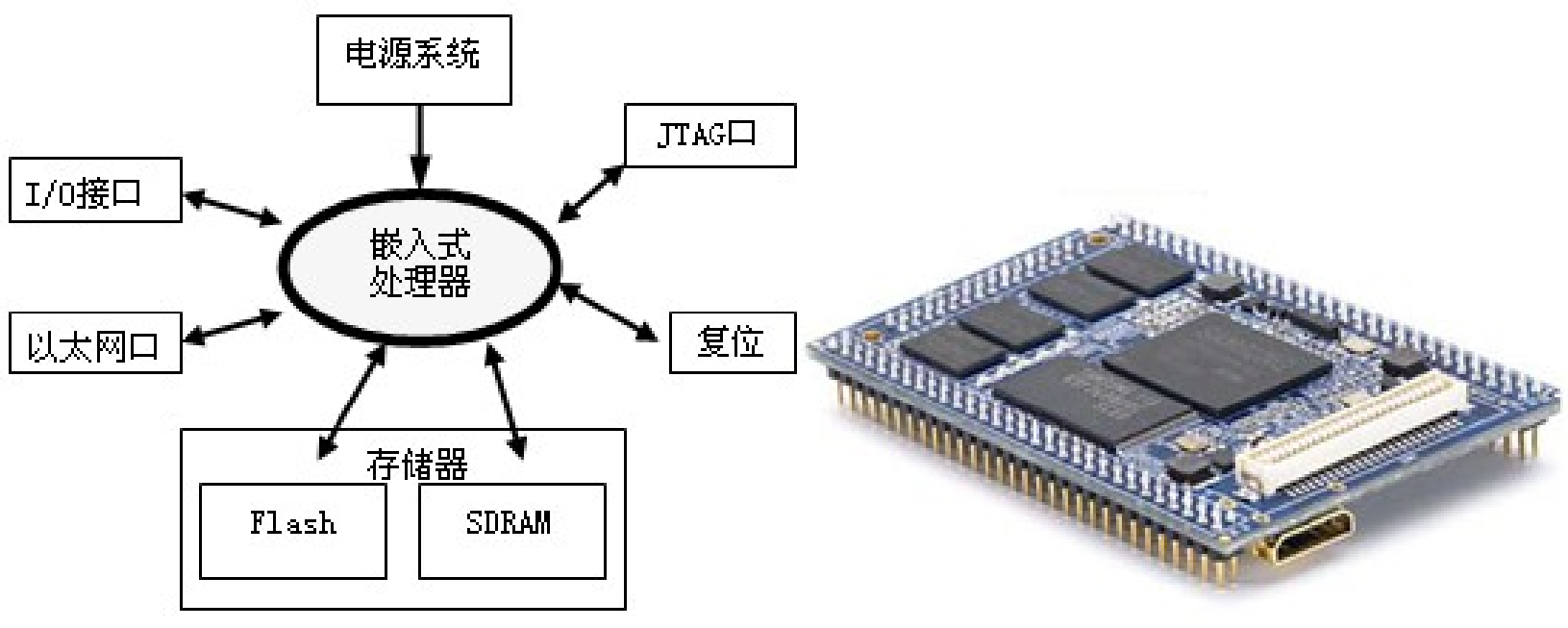
（2） 电源电路、复位电路，晶振电路，

（3） 存储器（ FLASH 和 SDRAM ），

（4） UART （ RS232 及以太网）接口电路。

（5） JTAG 调试接口。

2、 Cortex A8 微处理器



2.4.3 Cortex 微处理器结构

Cortex A8 微处理器 Snmsung S5PV210 核心板系统包括：

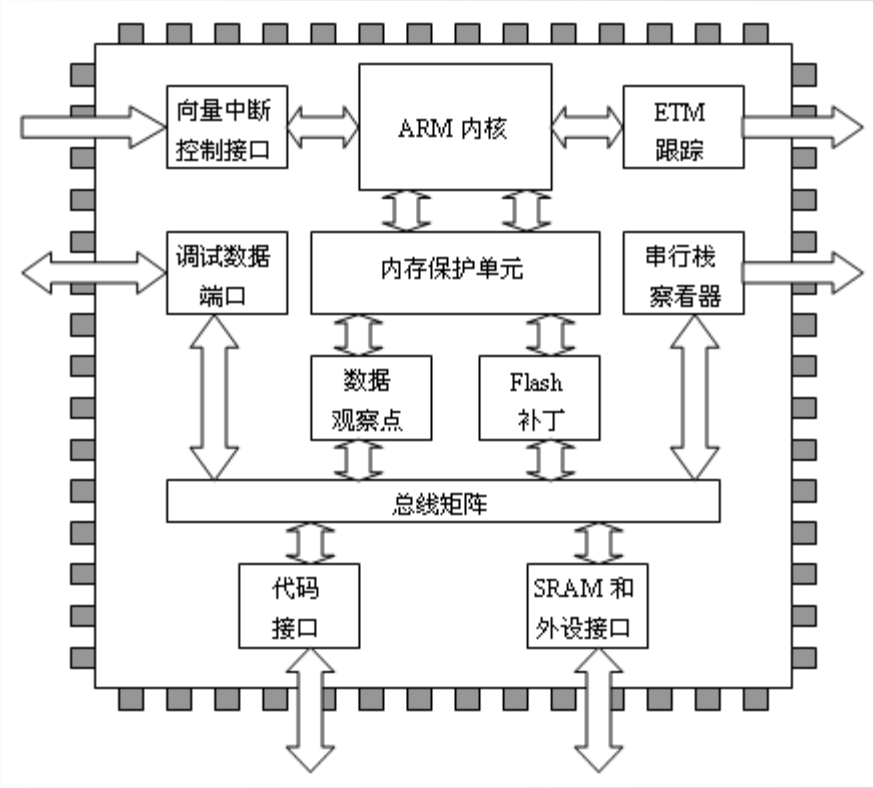
CPU： Snmsung S5PV210 基于 Cortex-A8，运行主频 1GHz；

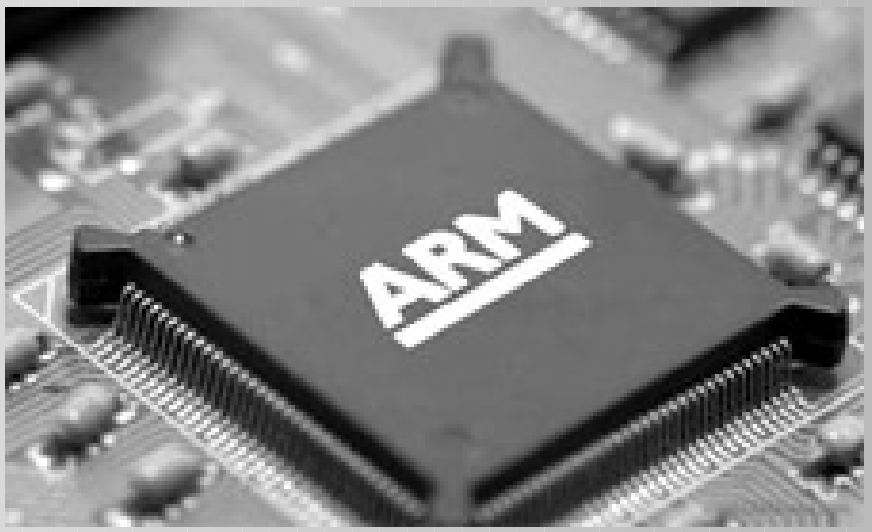
DDR2 RAM : 512MB 工作在 200MHz 外频上；

FLASH :512MB SLC NAND FALSH；

Ethernet CON： 100MB 网络控制器

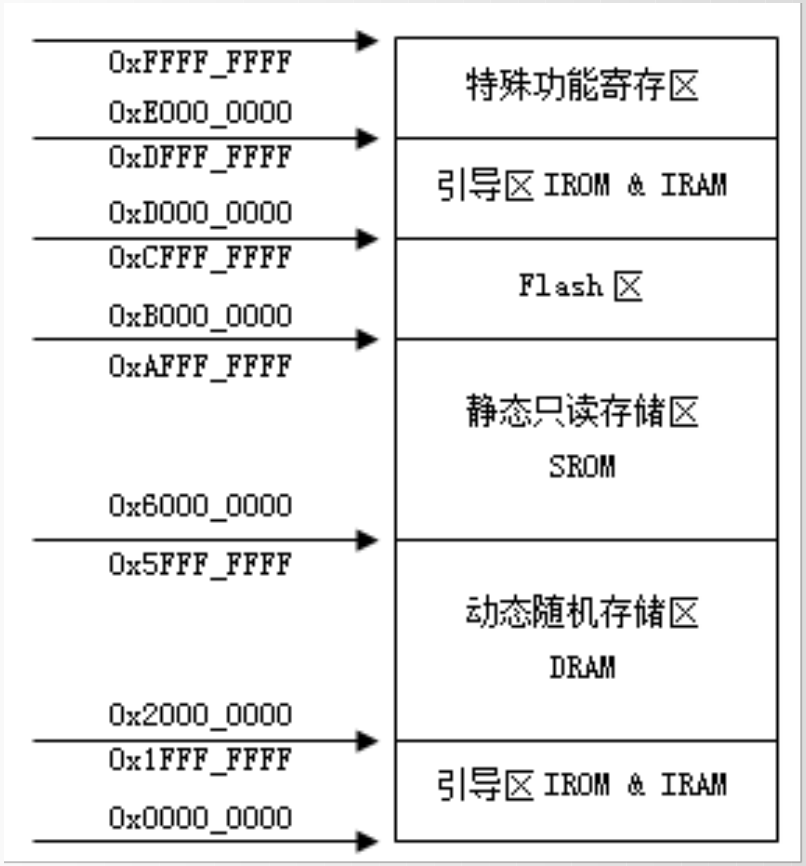
2.4.3 Cortex 微处理器结构





2.4.4 Cortex A8 的存储地址空间

- S5PV210 存储器地址映射如图 2.20 所示。



从图 2.20 中可以看到， S5PV210 的引导区分为两个部分，分别是：
0x0000_0000 ～ 0x1FFFF_FFFF 和
0xD000_0000 ～ 0xDFFFF_FFFF
的地址空间。
系统上电后，从引导区开始执行 BootLoader 引导程序。

2.4.5 Cortex A8 的 GPIO 端口

- 通用 I/O 接口（General Purpose IO， GPIO）是嵌入式系统中一种非常重要的 I/O 接口。它具有使用灵活、可配置性好、硬件代价小等优点，在嵌入式系统中广泛应用。

- 每个 GPIO 端口通常至少有两个寄存器：
  - 一个为“ IO 端口控制寄存器”，
  - 另一个为“ IO 端口数据寄存器”。

1. S5PV210 微处理器的 GPIO 端口分组

- 表 2.2 Cortex A8 的 GPIO 端口

端口分组	端 口 引 脚 数
GPA0	8 个 输入 / 输出引脚 - 2xUART 带控制流。
GPA1	4 个 输入 / 输出引脚 - 2xUART 不带控制流或 1xUART 带控制流。
GPB	8 个 输入 / 输出引脚 - 2x SPI 总个接口。
GPC0	5 个 输入 / 输出引脚 - I²S 总个接口， PCM 接口， AC97 接口。
GPC1	5 个 输入 / 输出引脚
GPD0	4 个 输入 / 输出引脚 - I²C 总个接口， PWM 接口，扩展 DMA 接口， SPDIF 接口。
GPD1	6 个 输入 / 输出引脚
GPE0,1	13 个 输入 / 输出引脚 - 摄像头接口， SD/MMC 接口。
GPF0,1,2,3	30 个 输入 / 输出引脚 - LCD 接口。
GPG0,1,2,3	28 个 输入 / 输出引脚 - 3xMMC channel， SPI， I²S， PCM， SPDIF 各种接口。
GPH0,1,2,3	32 个 输入 / 输出引脚 - 摄像头通道接口，键盘，最大支持 32 位可中断接口。
GPI	低功率 I²S 、 PCM 接口。
GPJ0,1,2,3,4	35 个 输入 / 输出引脚 - Modem IF， HIS， ATA 接口。
MP0_1,2,3	20 个 输入 / 输出内存端口引脚。
MP0_4,5,6,7	32 个 输入 / 输出内存端口引脚。
MP1_0 ～ 8	71 个 DRAM1 端口引脚。
MP2_0 ～ 8	71 个 DRAM2 端口引脚。
ETC0, ETC1, ETC2, ETC4	28 个 输入 / 输出 ETC 端口及 JTAG 端口。

2. Cortex A8 的 常用 GPIO 寄存器

- 在使用 Cortex A8 微处理器时，由于大多数引脚都是可复用的，因此需要对每个引脚进行配置。
- Cortex A8 架构的 S5PV210 微处理器有 4 种 GPIO 寄存器，它们是：
  - 控制寄存器 GPxnCON 、
  - 数据寄存器 GPxnDAT 、
  - 上拉 / 下拉寄存器 GPxnPUD 、
  - 掉电模式上拉 / 下拉寄存器 GPxnPUDPDN 。

（1） GPIO 寄存器地址表

- 表 2.3　GPC0 端口组控制寄存器地址
  (Base Address = 0xE020_0060)

寄存器	地址	描述	初始值
GPC0CON	0xE020_0060	GPC0 端口组控制寄存器	0x00000000
GPC0DAT	0xE020_0064	GPC0 端口组数据寄存器	0x00
GPC0PUD	0xE020_0068	GPC0 端口组上拉 / 下拉寄存器	0x0155

（2）端口控制寄存器 GPxCON （x = A，B，D，E，F，G，H，I，J）

- 每一个 I/O 端口都有一个 CON （端口控制）寄存器，用于控制 GPIO 引脚的功能。该寄存器每 4 位控制一个引脚。
- 当输入 0000 时，引脚设置为输入口，可以从引脚读入外部输入的数据；
- 当输入 0001 时，引脚设置为输出口，向该位写入的数据被发送到对应的引脚上。

表 2.4 GPC0CON 端口控制寄存器定义 (Address = 0xE020_0060)

GPC0CON	位	描述	初始状态
GPC0CON[4]	[19:16]	0000 = 输入 , 0001 = 输出 , 0010 = I²S_1_SDO, 0011 = PCM_1_SOUT, 0100 = AC97SDO, 0101 ～ 1110 = 保留 , 1111 = GPC0_INT[4]	0000
GPC0CON[3]	[15:12]	0000 = 输入 , 0001 = 输出 , 0010 = I²S_1_SDI , 0011 = PCM_1_SIN , 0100 = AC97SDI, 0101 ～ 1110 = Reserved , 1111 = GPC0_INT[3]	0000
GPC0CON[2]	[11:8]	0000 = 输入 , 0001 = 输出 , 0010=I²S_1_LRCK, 0011=PCM_1_FSYNC , 0100=AC97SYNC, 0101 ～ 1110 = Reserved , 1111 = GPC0_INT[2]	0000
GPC0CON[1]	[7:4]	0000 = 输入 , 0001 = 输出 , 0010=I²S_1_CDCLK, 0011=PCM_1_EXTCLK , 0100=AC97RESETn, 0101 ～ 1110 = Reserved , 1111 = GPC0_INT[1]	0000
GPC0CON[0]	[3:0]	0000 = 输入 , 0001 = 输出 , 0010=I²S_1_SCLK, 0011=PCM_1_SCLK , 0100=AC97BITCLK , 0101 ～ 1110 = Reserved , 1111 = GPC0_INT[0]	0000

（3）端口数据寄存器 GPxDAT （x = A，B，D，E，F，G，H，I，J）

- 每一个 I/O 端口都有一个 DAT （数据）寄存器，它是一个读写寄存器。该寄存器每 2 位表示一个数据。
- 当端口被设置为输出端口时，如果向 GPxDAT 的相应位写入数据 1，则该引脚输出高电平，如果向 GPxDAT 的相应位写入数据 0，则该引脚输出低电平；
- 当端口被设置为输入端口时，则可以向 GPxDAT 的相应位读出数据，得到端口电平状态。

表 2.5　GPC0DAT 端口数据寄存器的定义

GPC0DAT	位	描述	初始状态
GPC0DAT[4:0]	[4:0]	决定输入或者输出的电平状态	0x00

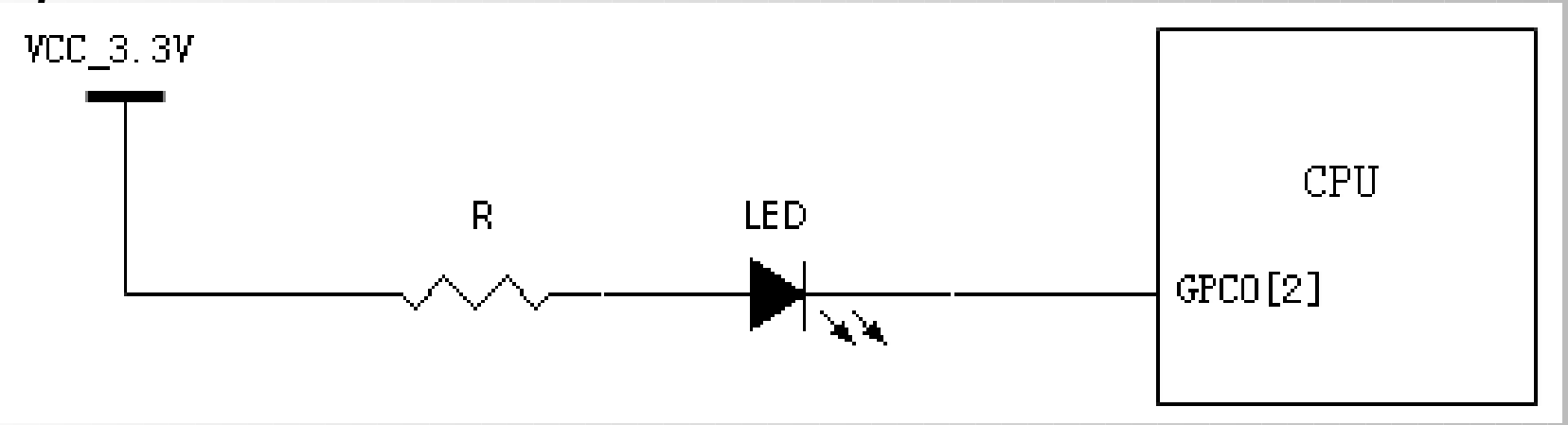
（4）端口上拉 / 下拉寄存器 GPxPUD （x = A，B，D，E，F，G，H，I，J）

•每一个 I/O 端口都有一个 PUD （上拉 / 下拉使能）寄存器，该寄存器控制了每个端口组的上拉 / 下拉电阻的使能 / 禁止。根据对应位的 0/1 组组合，设置对应端口的上拉 / 下拉电阻功能是否使能。

•如果端口的上拉电阻被使能，无论在哪种状态（输入、输出、 DATAn 、 EINTn 等）下，上拉电阻都起作用。

3. GPIO 寄存器功能设置应用示例

【例 2-1 】设在 Cortex A8 微处理器 GPIO 端口的 GPC0[2] 引脚连接一个 LED 发光二极管，如图 2.21 所示。现对该端口的控制寄存器 GPC0CON 和数据寄存器 GPC0DAT 进行设置，使 LED 发光二极管点亮或熄灭。（对于本例，上拉 / 下拉寄存器不需要设置）



- （1）问题分析
- 若要使一个 LED 发光二极管点亮，必须有一个正向电压，即寄存器引脚端必须是低电平。
- 反之，若要使 LED 发光二极管熄灭，则寄存器引脚端必须为高电平。
- 也就是说，寄存器引脚输出低电平时， LED 发光二极管点亮，寄存器引脚输出高电平时， LED 发光二极管熄灭。

（2） GPC0 的端口控制寄存器 GPC0CON 的设置

- 经上述分析，需要把 GPC0[2] 引脚设置为输出模式，也就是 GPC0CON[2] 引脚设为输出模式。按表 2-3 可知， GPC0CON[2] = （0001）2 。
- GPC0CON 的设置如图 2.22 所示。

所以，设置 GPC0CON[2] 为输出模式的值用二进制表示为：

0000 0000 0001 0000 0000

也可以表示为：（1<<12）

即： GPC0CON = （1<<12）

（3） 端口数据寄存器 GPC0DAT 的设置

- GPC0DAT 有 5 位 ([4： 0])，每一位对应一个 GPIO 端口引脚，当该寄存器的某位设置为 1 时，则对应引脚输出高电平，该寄存器的某位设置为 0 时，对应引脚输出低电平。
- 所以，在 GPC0CON[2] 已经设置为输出模式的前提下， GPC0DAT 设置为 0x01 时， GPC0[2] 引脚输出高电平， GPC0DAT 设置为 0x00 时， GPC0[2] 引脚输出低电平。
- 即：
- GPC0DAT = 0x01 时， GPC0[2] 引脚输出高电平， LED 发光二极管熄灭；
- GPC0DAT = 0x00 时， GPC0[2] 引脚输出低电平， LED 发光二极管点亮。

本章小结

- 本章首先简单介绍了嵌入式系统相关的基础知识，这些知识和概念在以后学习嵌入式系统设计时都要用到。之后介绍了嵌入式系统硬件平台的基本组成，并对 ARM 系列微处理器作了简介，还介绍了 Cortex A8 架构 S5PV210 微处理器的 GPIO 寄存器。本章重点要掌握嵌入式系统硬件平台的组成，这是学习和应用嵌入式系统的基础。
- 本章重点要掌握嵌入式系统硬件平台的组成，这是学习和应用嵌入式系统的基础。