

嵌入式系统原理与应用

西安邮电大学计算机学院

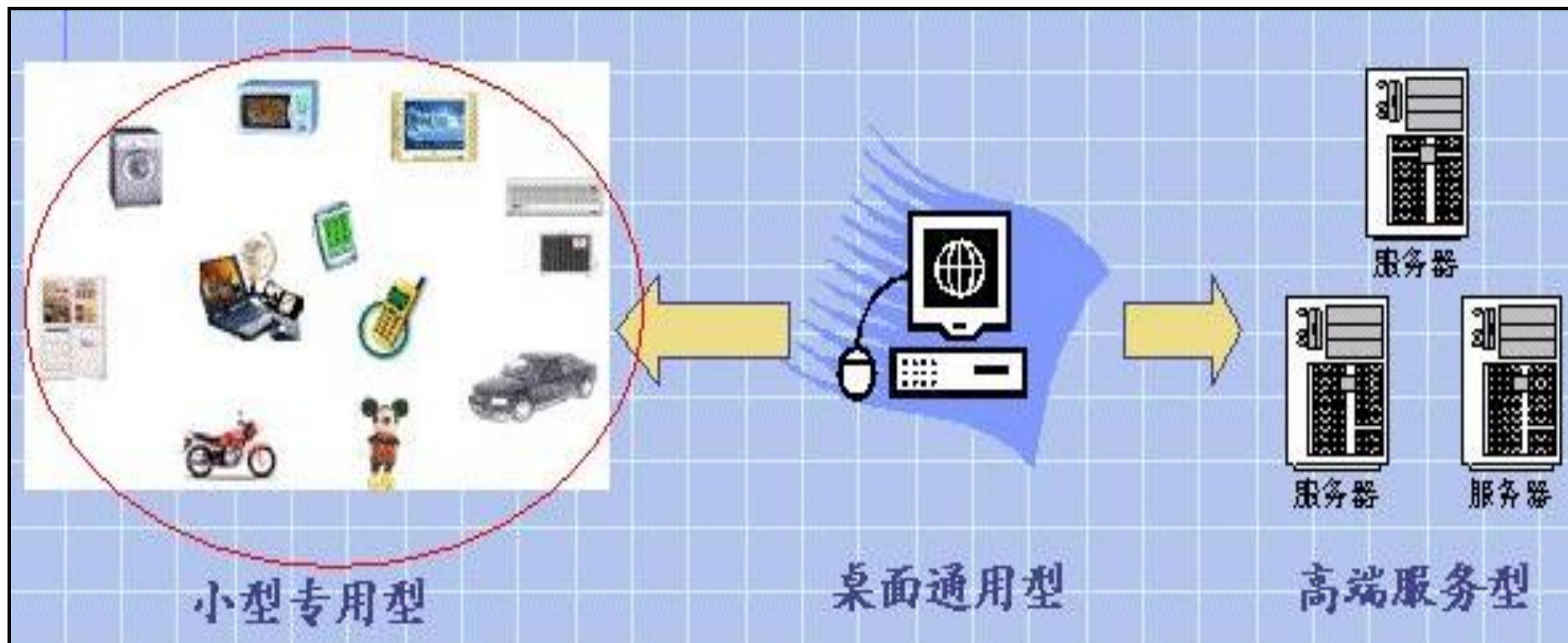
贺炎



西安邮电大学
XI'AN UNIVERSITY OF POSTS & TELECOMMUNICATIONS

第一章 嵌入式系统概述

引子：计算机系统的三大领域



专家预言：大机时代→PC时代→后PC时代（嵌入式时代）



第一章 嵌入式系统概述

- ◆ 基本概念--- 概念
- ◆ 发展历史--- 过去
- ◆ 应用领域--- 现在
- ◆ 未来趋势--- 将来
- ◆ 系统开发--- 方法



1 基本概念

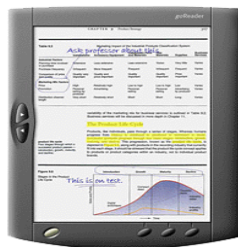
- 1.1 嵌入式系统的定义
- 1.2 嵌入式系统的特点
- 1.3 嵌入式系统的组成
- 1.4 嵌入式系统的分类



第一章 嵌入式系统概述——嵌入式系统的定义

1.1 嵌入式系统的定义

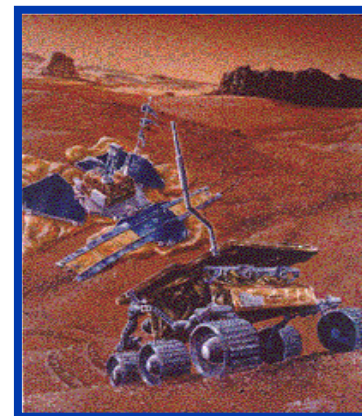
- 从广义的角度：嵌入式系统是一切非PC和大型机的计算机系统



goReader Internet eBook



Nixvue Digital Album
Digital Photo Album



Tektronix TDS7000
Digital Oscilloscopes



Samsung AnyWeb
Internet Screen Phone



1.1 嵌入式系统的定义

- 从应用的角度：

devices used to control, monitor or assist the operation of equipment, machinery or plants.

嵌入式系统是控制、监视或辅助设备、机器和车间运行的装置。

IEEE（国际电气和电子工程师协会）定义

1.1 嵌入式系统的定义

- 从系统的角度：

嵌入式系统是设计完成复杂功能的硬件和软件，并使其紧密耦合在一起的计算机系统，是更大系统的一个完整的子系统。



1.1 嵌入式系统的定义

- 从技术的角度：

嵌入式系统是一个以应用为中心、以计算机技术为基础，并融合微电子技术、通信技术和自动控制技术，而且软硬件可裁剪，适用于应用系统对功能、可靠性、成本、体积、功耗和应用环境有特殊要求的专用计算机系统。



第一章 嵌入式系统概述——嵌入式系统与桌面通用系统比较

相同： 硬件和软件的集合

不同：

- ◆形态：“嵌入”于不同设备 Vs 基本雷同的标准形态独立存在
- ◆价值：“嵌入”的不同设备 Vs 通用指标(计算能力、存储能力)
- ◆功耗：几mW—几W Vs 几百W
- ◆功能：专用单一 Vs 通用复杂
- ◆资源：够用就行 Vs 大而全
- ◆实时：系统最坏情况下的行为 Vs 系统平均响应时间和用户方便
- ◆可靠：恶劣环境无人值守长时间 Vs 一般环境
- ◆开发：交叉开发在线仿真固化存储 Vs 编程环境与运行环境一致
- ◆生命：8—10年 Vs 18个月
- ◆竞争：百家争鸣 Vs 巨头垄断



1.2 嵌入式系统的特点

➤ 面向特定应用（专用性）

- ARM: 手机
- MIPS: 机顶盒
- PowerPC: 网络设备
- SH: 工业控制
- MSP430: 超低功耗要求

➤ 量体裁衣（可裁剪性）

- 硬件
- 软件



1.2 嵌入式系统的特点

➤ 实时性

实时性是指系统能够及时（限定时间内）处理外部事件。大多数实时系统都是嵌入式系统，而嵌入式系统多数也有实时性的要求，如微波炉加热过程中炉门一旦被打开必须在数十毫秒内停止加热。

➤ 可靠性

很多嵌入式系统必须持续工作甚至在极端环境下正常运行。因此，大多数嵌入式系统都具有可靠性机制，例如硬件的看门狗定时器、软件的内存保护和重启机制等，以保证嵌入式系统在出现问题时能够重新启动，保障系统的健壮性。



1.2 嵌入式系统的特点

➤ 较长的生命周期

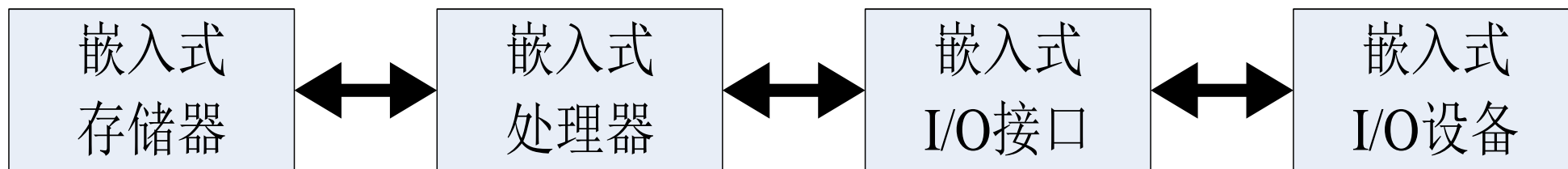
- 与具体嵌入的产品同步：导入期→成长期→成熟期→衰退期

➤ 不易垄断

- 将计算机技术、电子技术与各行业的具体应用相结合的产物
- 资金密集、技术密集、高度分散、不断创新的知识集成系统

1.3 嵌入式系统的组成（硬件）

➤ 硬件



- 嵌入式处理器（核心）
- 嵌入式存储器
- 嵌入式I/O设备
- 嵌入式I/O接口

1.3 嵌入式系统的组成（嵌入式处理器的分类）

- 嵌入式微控制器

（MCU, Micro-programmed Control Unit）

（MCU, Micro-Controller Unit）

- 嵌入式微处理器

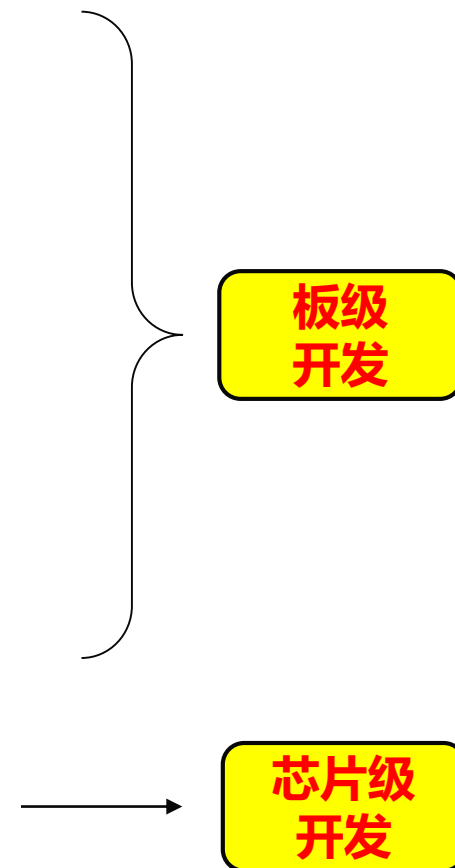
（MPU, Micro-Processor Unit）

- 嵌入式数字信号处理器

（DSP, Digital Signal Processor）

- 嵌入式片上系统

（SOC, System On Chip）



第一章 嵌入式系统概述——嵌入式系统的组成

1.3 嵌入式系统的组成（嵌入式处理器的分类）

■ 嵌入式微控制器（MCU）

单片机

➤ 定义：

内部集成中央处理单元、ROM、RAM、总线接口和各种外设

➤ 特点：

将整个计算机系统集成到一块芯片上
体积小、功耗小、成本低、可靠性强

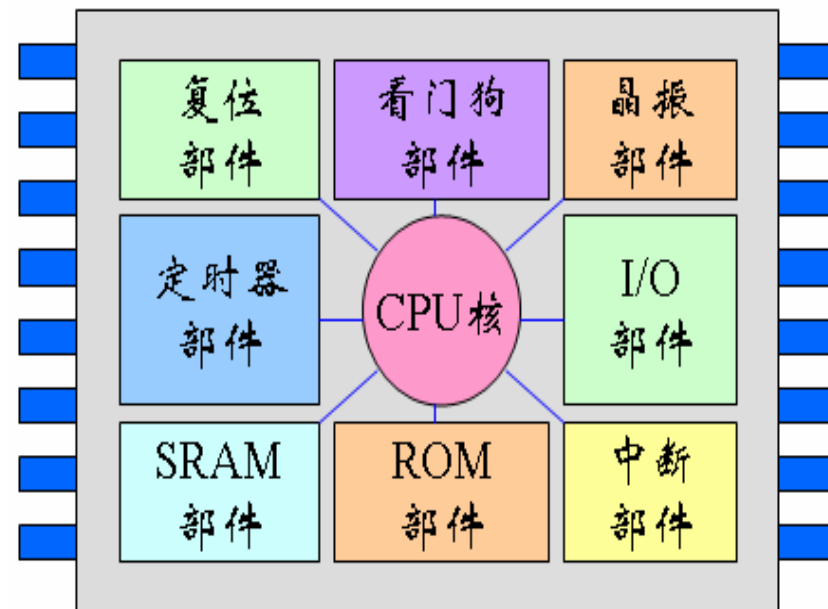
➤ 典型：（[哈佛结构](#)）

ATMEL公司的AT89(51系列)和AVR90

INTEL公司的MCS51

Microchip公司的PIC

TI公司的MSP430



1.3 嵌入式系统的组成（嵌入式处理器的分类）

■ 嵌入式微处理器（MPU）★ 类似于PC机里的CPU

➤ 定义：

内部仅包含单纯的中央处理单元

➤ 特点：（与PC机里的CPU的区别）

减：保留和应用紧密部分裁去冗余部分

增：将通用计算机中许多由板卡完成的I/O电路集成到芯片内部

➤ 典型：

ARM（[冯诺伊曼结构](#)[ARM7]、[哈佛结构](#)[ARM9/10/11]）

MIPS（冯诺伊曼结构）



1.3 嵌入式系统的组成（嵌入式处理器的分类）

■ 嵌入式数字信号处理器（DSP）

➤ 定义：

具有专门的硬件乘法器

➤ 特点：

适合对离散数字信号进行快速处理和运算

应用用于视频、音频处理

➤ 典型：（[哈佛结构](#)）

TI公司的TMS320C2000系列(控制)和C5000系列(通信)

Motorola公司的DSP56000系列



1.3 嵌入式系统的组成（嵌入式处理器的分类）

■ 嵌入式片上系统（SOC）

➤ 定义：

将一个或多个CPU单元以及功能部件集成在一块芯片上

TI(3G): OMAP = TMS320C54X + ARM925 + 多功能周边电路

➤ 特点：

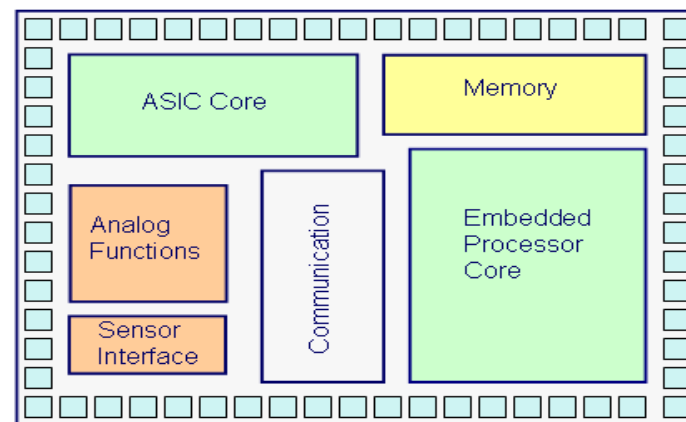
电路板更简洁、体积和功耗更小

可靠性更高

➤ 典型：

通用系列：Motorola公司的M-Core

专用系列：OMAP(Open Multimedia Application Platform)



1.3 嵌入式系统的组成（主流的嵌入式处理器）

■ ARM处理器

- 32位RISC内核，除ARM7采用**冯诺依曼架构**外，其他均**哈佛架构**
- 在智能手机、平板电脑等领域应用尤为广泛

■ MIPS处理器

- 32/64位RISC内核
- 在机顶盒领域应用尤为广泛

■ PowerPC处理器

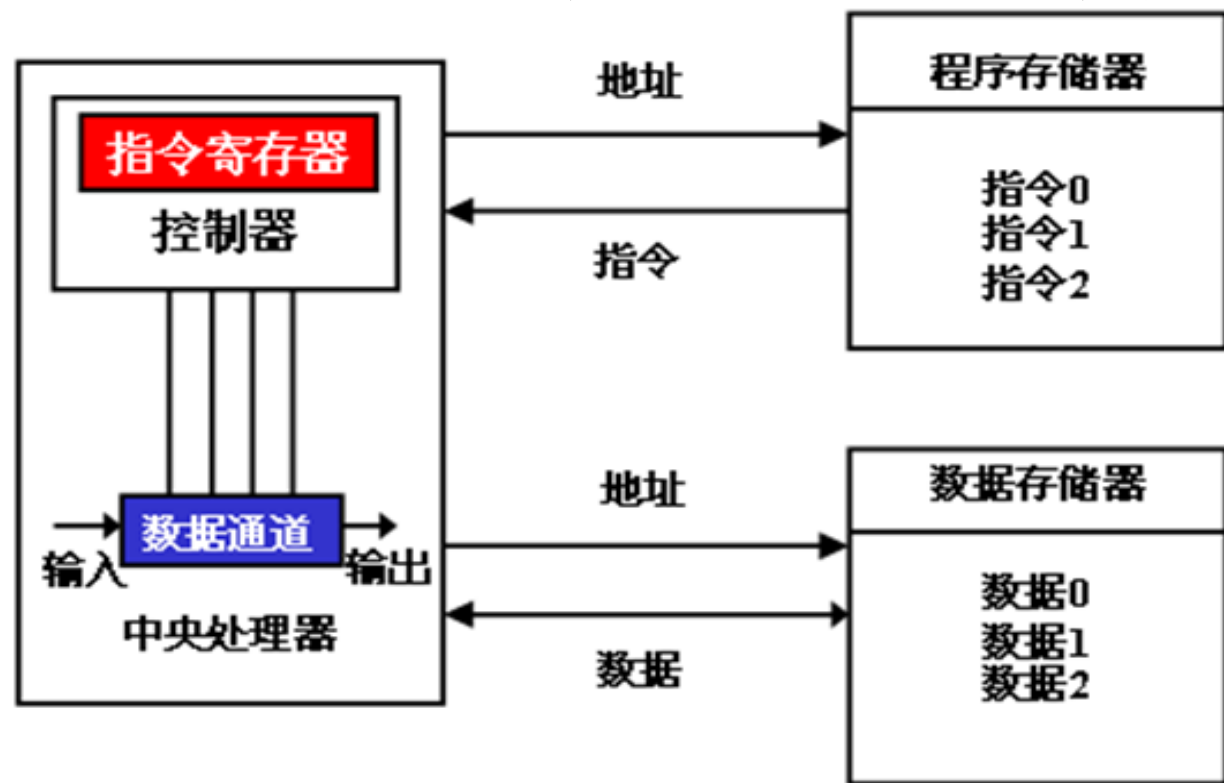
- 64位RISC内核
- 在网络通信、办公自动化、多媒体娱乐和消费电子等领域应用广泛



第一章 嵌入式系统概述——嵌入式系统的组成

哈佛体系结构

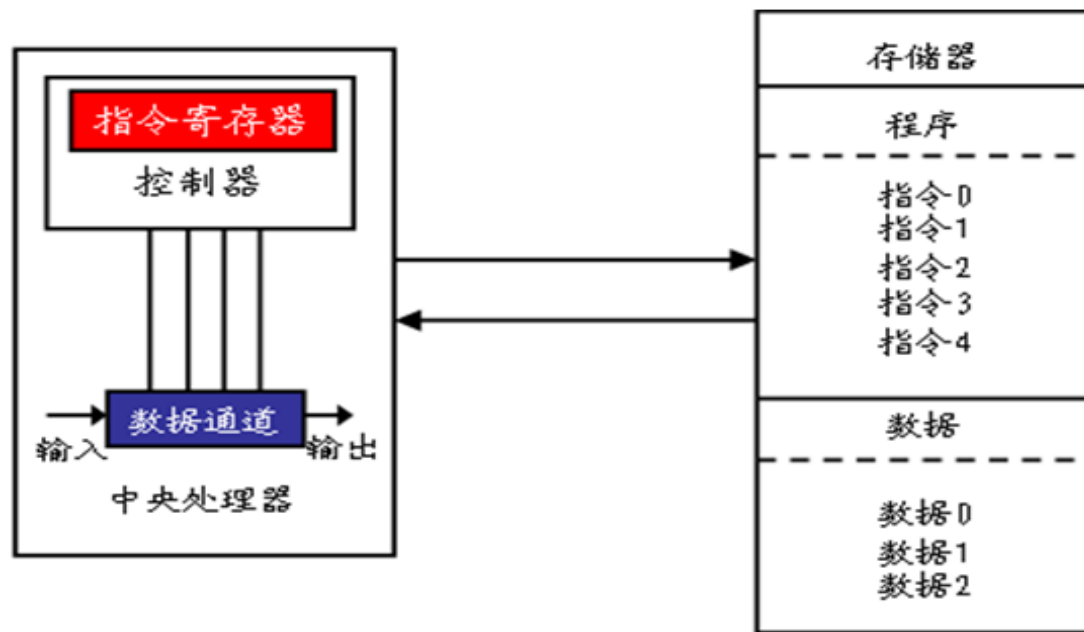
指令与数据分开存储在不同的存储空间，使得指令读取和数据访问可以并行处理，显著提高了系统性能，但需要较多数量的总线。大多数嵌入式处理器都采用哈佛结构。



第一章 嵌入式系统概述——嵌入式系统的组成

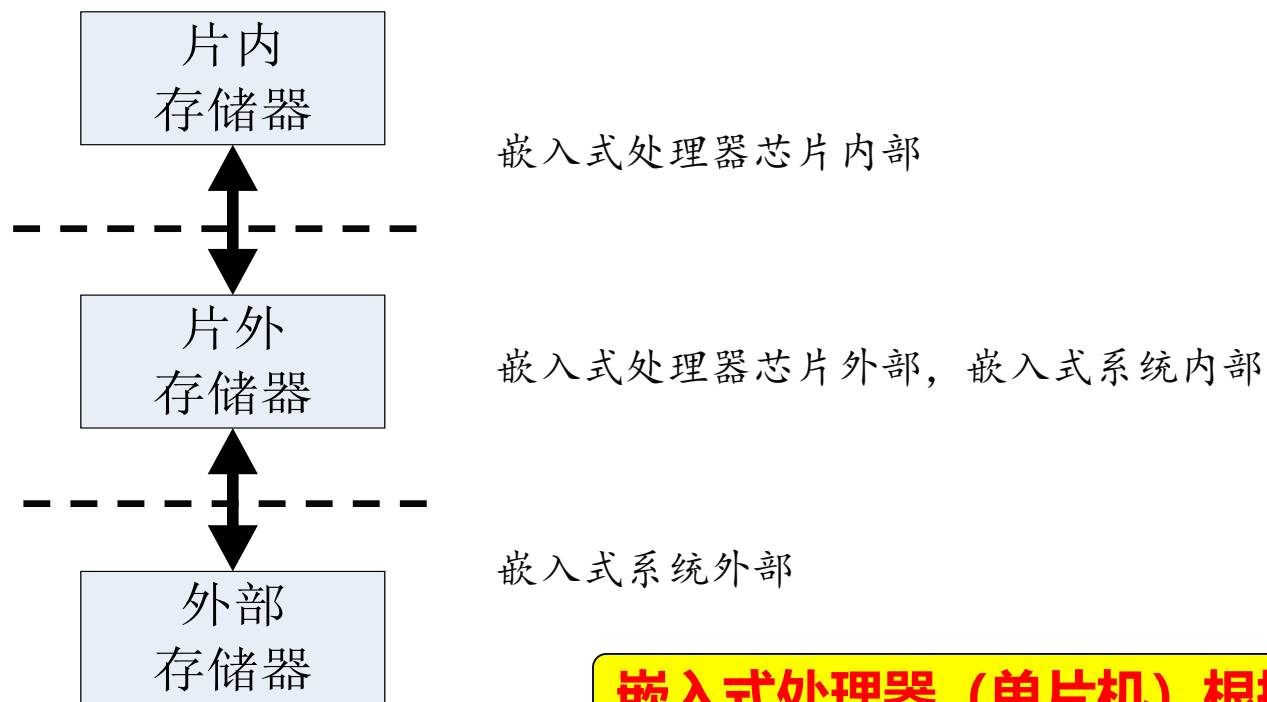
冯诺依曼体系结构

指令和数据不加以区分，都通过数据总线进行传输。因此，指令读取和数据访问不能同时进行，数据吞吐量低，但总线数量相对较少且管理统一。大多数通用计算机的处理器（如Intel X86）和嵌入式系统中的ARM7处理器均采用冯诺依曼结构。



1.3 嵌入式系统的组成（硬件）

■ 硬件——嵌入式存储器（层次结构）



嵌入式处理器（单片机）根据实际情况需要扩展外部存储器

1.3 嵌入式系统的组成（硬件）

■ 硬件

➤ 嵌入式存储器（主要类型）

易失性存储器
非易失性存储器

用来存放程序（固化的应用软件）和数据（其他信息）

NAND FLASH
大容量数据存储

基本存储单元
触发器

基本存储单元
MOS管栅极电容

- ROM: 生成时固定，不可再次编程，只读，存放启动代码
- FLASH: 可读可写，速度慢，软件控制读写，存放断电保存数据
- SRAM: 速度快无需动态刷新体积大容量小价格高，Cache
- DRAM: 体积小容量大价格低速度慢需控制器动态刷新，常用
- SDRAM: 时钟频率与处理器总线同步提高传输效率，最常用

第一章 嵌入式系统概述——嵌入式系统的组成

1.3 嵌入式系统的组成（硬件）

■ 硬件——嵌入式外围设备

一般包括：

- 数字键盘
- LCD
- LED
- 通讯卡
- 传感器
- 开关



诺基亚（Nokia 5110）
5110屏驱动（STM32）

一般不包括：

- 键盘
- 显示器
- 硬盘
- 光驱
- 喇叭
- 软驱
- 鼠标

1.3 嵌入式系统的组成（硬件）

■ 硬件——嵌入式外围设备

■ 用于人机交互的设备：

——LED、蜂鸣器、七段LED、LCD、...

——按键、4*4键盘、拨盘、摇杆、触摸屏、...

■ 用于机机交互的设备：

——传感器（Sensor）：温度传感器、压力传感器、...

——伺服执行机构（Actuator）：继电器和电机、...

1.3 嵌入式系统的组成（硬件）

■ 硬件——嵌入式I/O接口

- UART/USART
- SPI
- I²C
- USB
- CAN
- 以太网
- 红外
- WIFI
- ZigBee



第一章 嵌入式系统概述——嵌入式系统的组成

1.3 嵌入式系统的组成（硬件小结）

■ 硬件

➤ 嵌入式处理器

嵌入式微控制器（ARM）及其最小系统

➤ 嵌入式存储器（SDRAM + NAND FLASH）

片内存储器 — 片外存储器 — 外部存储器

➤ 嵌入式外围设备和接口电路

设备：Switch, LED, LCD, 7-SEG-LED ...

接口：UART/USART, USB, ETH, ...



1.3 嵌入式系统的组成

■ 软件

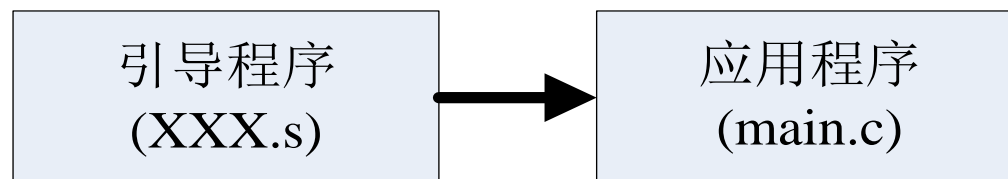
- ◆ 无操作系统下的嵌入式软件（功能简单）
- ◆ 带操作系统的嵌入式软件（功能较为复杂）



1.3 嵌入式系统的组成

■ 软件（无操作系统下的嵌入式软件）

➤ 引导程序



一般由汇编语言编写，在嵌入式系统上电后运行，完成自检、存储映射、时钟系统和外设接口配置等硬件初始化操作。

➤ 应用程序

一般由C语言编写，直接架构在硬件之上，在引导程序之后运行，负责实现嵌入式系统的主要功能。

1.3 嵌入式系统的组成

- 软件（无操作系统下的嵌入式软件：应用程序结构）

- 循环轮询系统

```
//main.c
```

```
void main(){
```

```
    Initialize();
```

```
    while(1){
```

```
        ...
```

```
    }
```

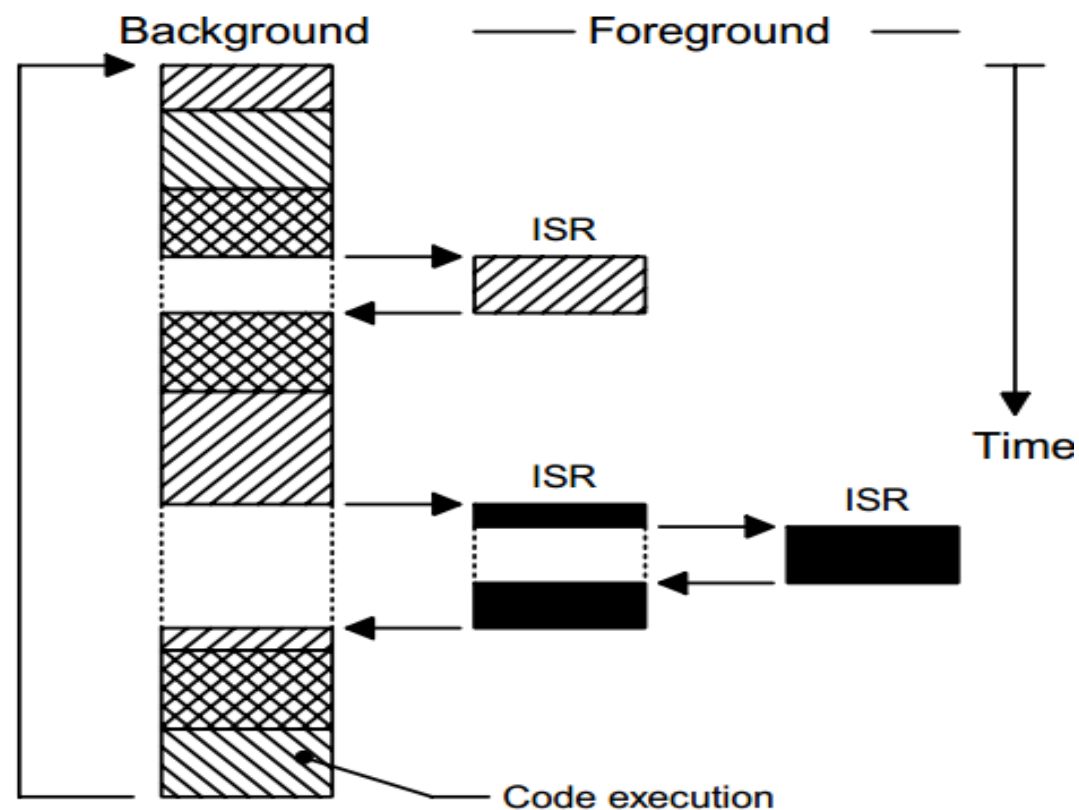
```
}
```



1.3 嵌入式系统的组成

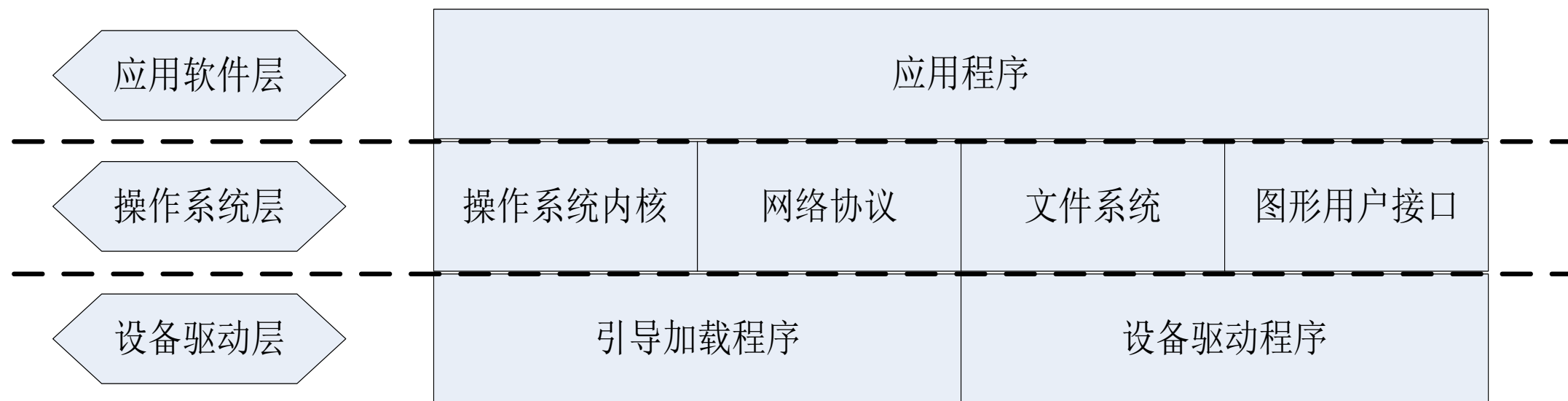
■ 软件（无操作系统下的嵌入式软件：应用程序结构）

➤ 前后台系统



1.3 嵌入式系统的组成

■ 软件（有操作系统下的嵌入式软件）



1.3 嵌入式系统的组成

- 软件（有操作系统下的嵌入式软件）：设备驱动层

- 引导程序（Bootloader）

- 设备驱动程序（Device Driver）

为上层软件提供透明的设备操作接口

1.3 嵌入式系统的组成

- 软件（有操作系统下的嵌入式软件）：操作系统层

- ▶ 嵌入式操作系统

- μ C/OS-II、TinyOS、Vxworks

- 嵌入式Linux

- Android、IOS



1.3 嵌入式系统的组成

- 软件（有操作系统下的嵌入式软件）： 应用软件层

- 嵌入式应用软件/嵌入式应用程序

可基于或不基于嵌入式操作系统，实现用户应用功能



1.4 嵌入式系统的分类

■ 按硬件（嵌入式处理器的字长）划分

➤ 嵌入式处理器

—— 4 位：Intel 4004、...

—— 逐渐淘汰

—— 8 位：Intel MCS-51、...

—— 16位：Intel MCS-96、...

} 大量应用

—— 32位：ARM7、ARM9、...

—— 高速发展

—— 64位：RENESAS SH5、...

—— 技术成熟



1.4 嵌入式系统的分类

■ 按软件（有无嵌入式操作系统）划分

- 无操作系统控制的嵌入式系统
- 小型操作系统控制的嵌入式系统
- 大型操作系统控制的嵌入式系统



1.4 嵌入式系统的分类

■ 按实时性划分

➤ 非实时嵌入式系统

不具备实时性要求的嵌入式系统（温湿度计）

➤ 软实时嵌入式系统

外部事件在截止时间到达时偶尔未得到及时处理并不会带来致命失误的嵌入式系统（DVD播放器、机顶盒）

➤ 硬实时嵌入式系统

外部事件在截止时间到达时偶尔未得到及时处理，会带来致命失误的嵌入式系统（导弹控制系统、雷达导航系统）

问题：
生活中哪些地方需要使用实时系统？



2 发展历史

- 2.1 计算机的发展历史
- 2.2 嵌入式系统硬件的发展历史
- 2.3 嵌入式系统软件的发展历史



计算机发展历史

自1946年世界上第一台电子计算机问世以来，计算机的发展已经历了四代：

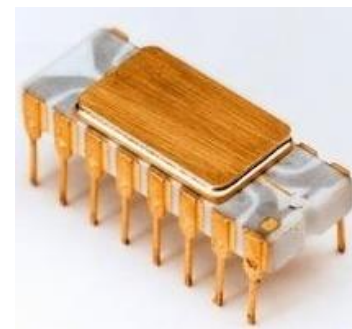
- ◆ 第一代：电子管计算机
- ◆ 第二代：晶体管计算机
- ◆ 第三代：集成电路计算机
- ◆ 第四代：大规模、超大规模集成电路计算机



电子管



晶体管

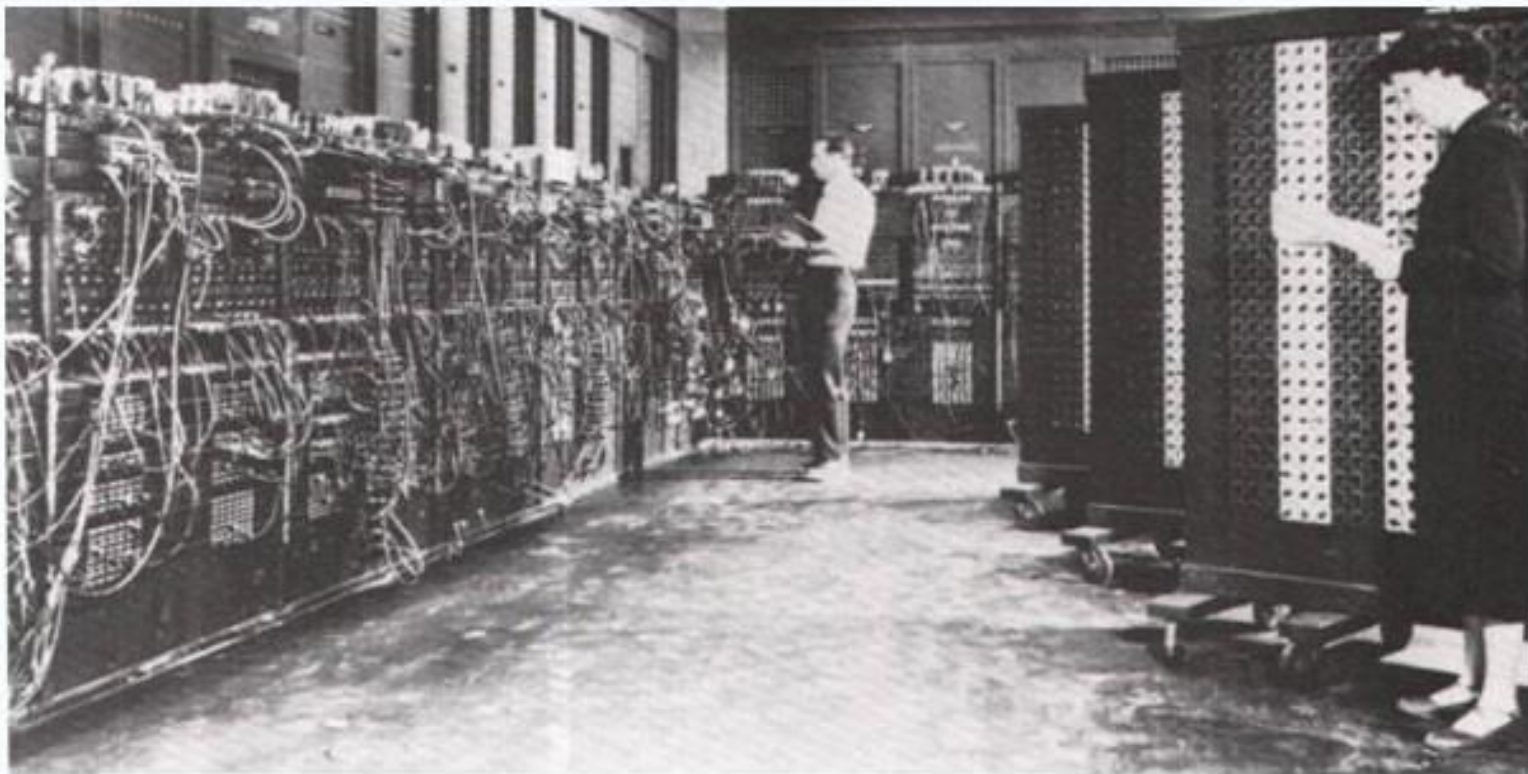


集成电路芯片



超大规模集成电路芯片

计算机发展历史



第一台电子计算机ENIAC的历史照片

17468个电子三极管、
7200个电子二极管、
70000个电阻, 10000个电
容器、1500个继电器、
6000多个开关、重达30吨。

时间：1946年2月14日

研制者：美国军方+宾西

法伊亚大学

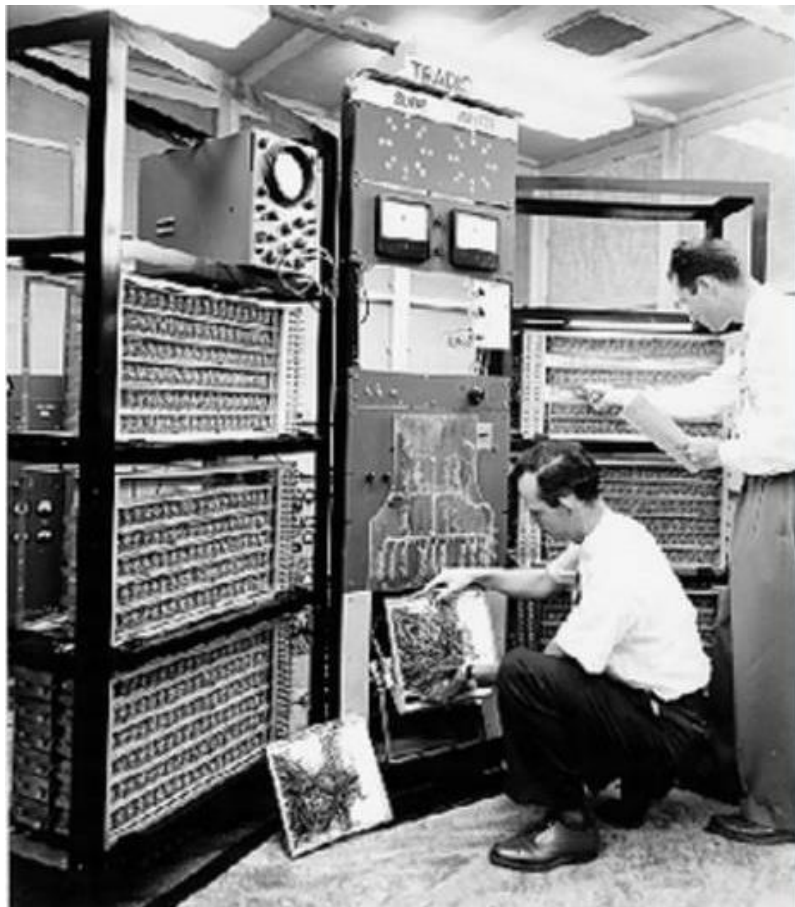
用途：美国军方用于炮弹
弹道的计算

<http://www.icpcw.com/Information/Industry/News/3232/323257.htm>



西安邮电大学
XI'AN UNIVERSITY OF POSTS & TELECOMMUNICATIONS

计算机发展历史



首台晶体管计算机“催迪克”

有800个晶体管。而计算机的体积，也由ENIAC的房间般大小，缩小为衣橱般体积。

时间：1954年

研制者：美国军方+贝尔实验室

用途：波音B-52亚音速远程战略轰炸机（即B-52轰炸机），1955年，美国阿塔拉斯洲际导弹。



计算机发展历史



IBM 360成为首款使用集成电路的计算机

时间：1964年

研制者：IBM

用途：美国的银行在线交易系统、登月计划



世界上第一款商用处理器：Intel4004

时间：1971年

研制者：Intel

用途：用于日本人商场的收款机



西安邮电大学
XI'AN UNIVERSITY OF POSTS & TELECOMMUNICATIONS

微型计算机发展历史

微型计算机是计算机发展到第四代（上世纪70年代后）出现的！

- ◆ 第一代：4位或抵挡8位微处理器（1971-1973）
- ◆ 第二代：中高档8位微处理器（1974-1977）
- ◆ 第三代：16位微处理器（1978-1984）
- ◆ 第四代：32位微处理器（1985-2004）
- ◆ 第五代：64位微处理器（2005---）

摩尔定律：

集成电路上可容纳的元器件的数目，约每隔18-24个月便会增加一倍，性能提升一倍。

——英特尔公司创始人之一戈登●摩尔（Gordon Moore）1965年提出



计算机发展历史



APPLE II (1977)
中央处理器: MOStek 6502
(摩斯太克公司)



IBM PC/XT (1981)
中央处理器: Intel 8088



IBM PC/AT (1984)
中央处理器: Intel 80286



康柏LTE 286笔记本
(1989)

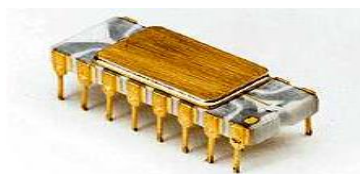
早期的一些由不同厂家推出的个人计算机



■ 2 发展历史

➤ 2.2 发展 —— 硬件（嵌入式处理器）

- 开创历史的Intel 4004（1971年）
- 永远经典的8051（1978年）
- 独步当时的TMS320C10（1982年）--德州仪器TI的DSP芯片
- 流行当代的ARM（1996年）



● 12 mm²

● 4 bit architecture

● 2,250 transistors

● Clock: 108 kHz



■ 2 发展历史

➤ 2.3 发展 —— 软件（嵌入式操作系统）

■ 无操作系统时代（20世纪70年代）

- 初期，受硬件条件限制追求时空效率采用汇编语言直接控制系统
- 后来，逐步采用高级语言和汇编语言相结合开发嵌入式应用软件

■ 简单嵌入式操作系统时代（20世纪80年代）

- 出现了控制系统负载和监视应用程序运行的简单嵌入式操作系统
- 简单而经典的嵌入式操作系统uCos内核大小只有几K

■ 嵌入式实时操作系统时代（20世纪90年代至今）

- 嵌入式操作系统的实时性得到很大提高
- 嵌入式操作系统的功能日益完善使嵌入式应用软件开发更加简单
 - 文件系统、网络协议栈、图形用户接口、多任务、设备管理、应用程序接口、...

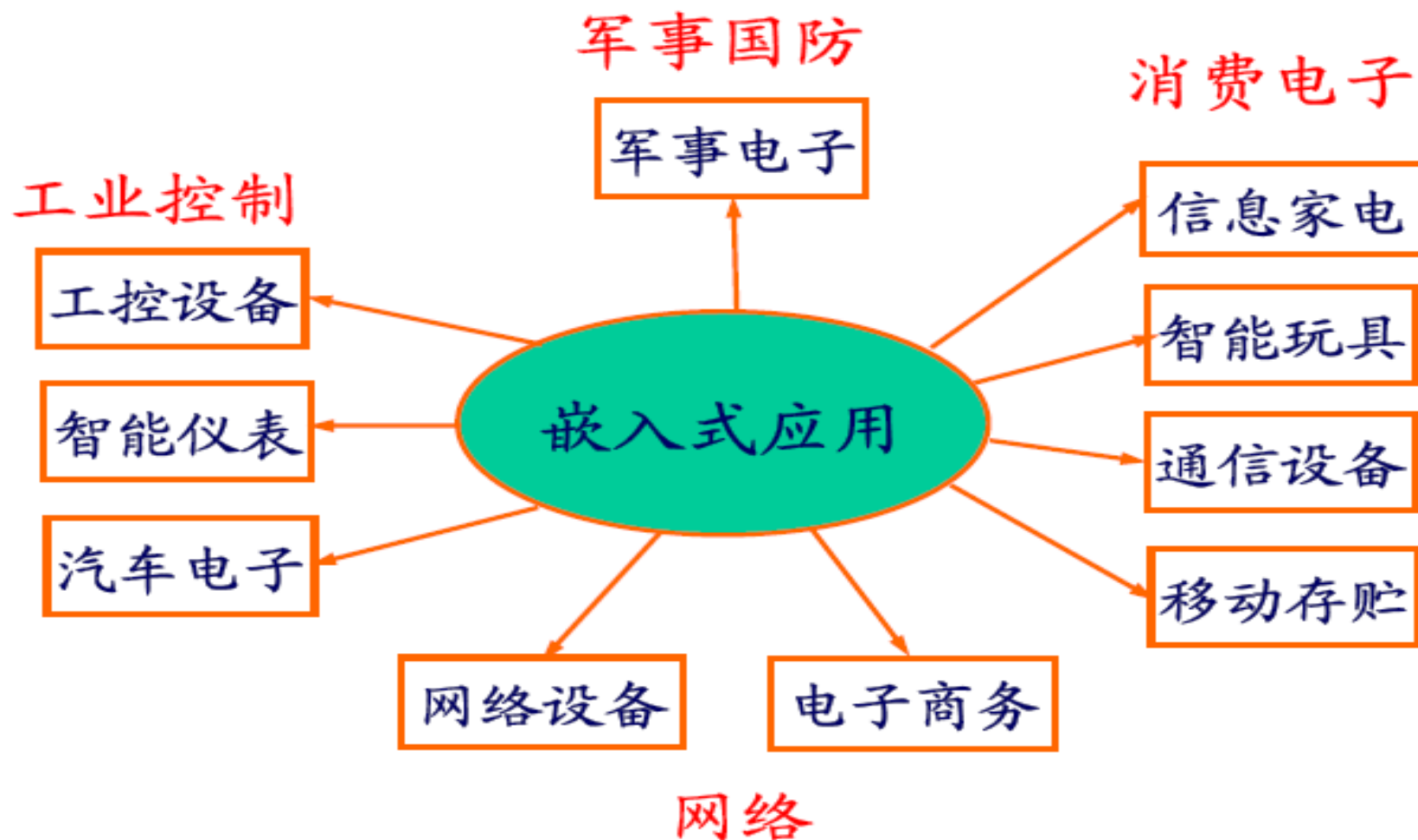


3 应用领域

- 3.1 军事电子
- 3.2 工控设备
- 3.3 智能仪表
- 3.4 汽车电子
- 3.5 网络设备
- 3.6 信息家电
- 3.7 其他消费电子



3 应用领域



3.1 军事电子

■ 军事侦察



微型飞行器
“黑寡妇”



阿富汗参加反恐作战的“赫耳墨斯”价值4万美元，可携带2架摄像机，发挥了很好作用。

3.1 军事电子

■ 科学探测



基于VxWorks的“火星探路者”

3.2 工控设备

■ 工业控制



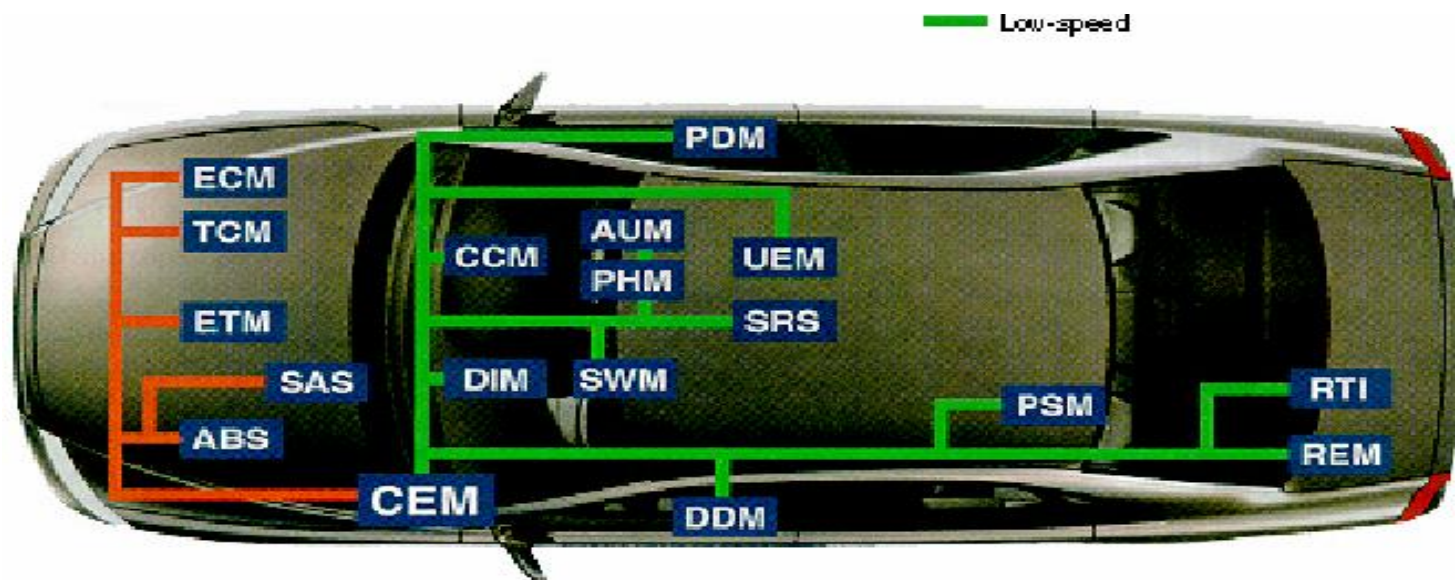
3.3 智能仪表

■ 手持抄表仪



3.4 汽车电子

- 含有18个嵌入式控制模块的汽车电子产品



VOLVO S80汽车的CAN总线网络

3.5 网络设备

■ 路由器/交换机/防火墙



宽带路由器



快速以太网交换机



千兆网关防火墙



VPN产品

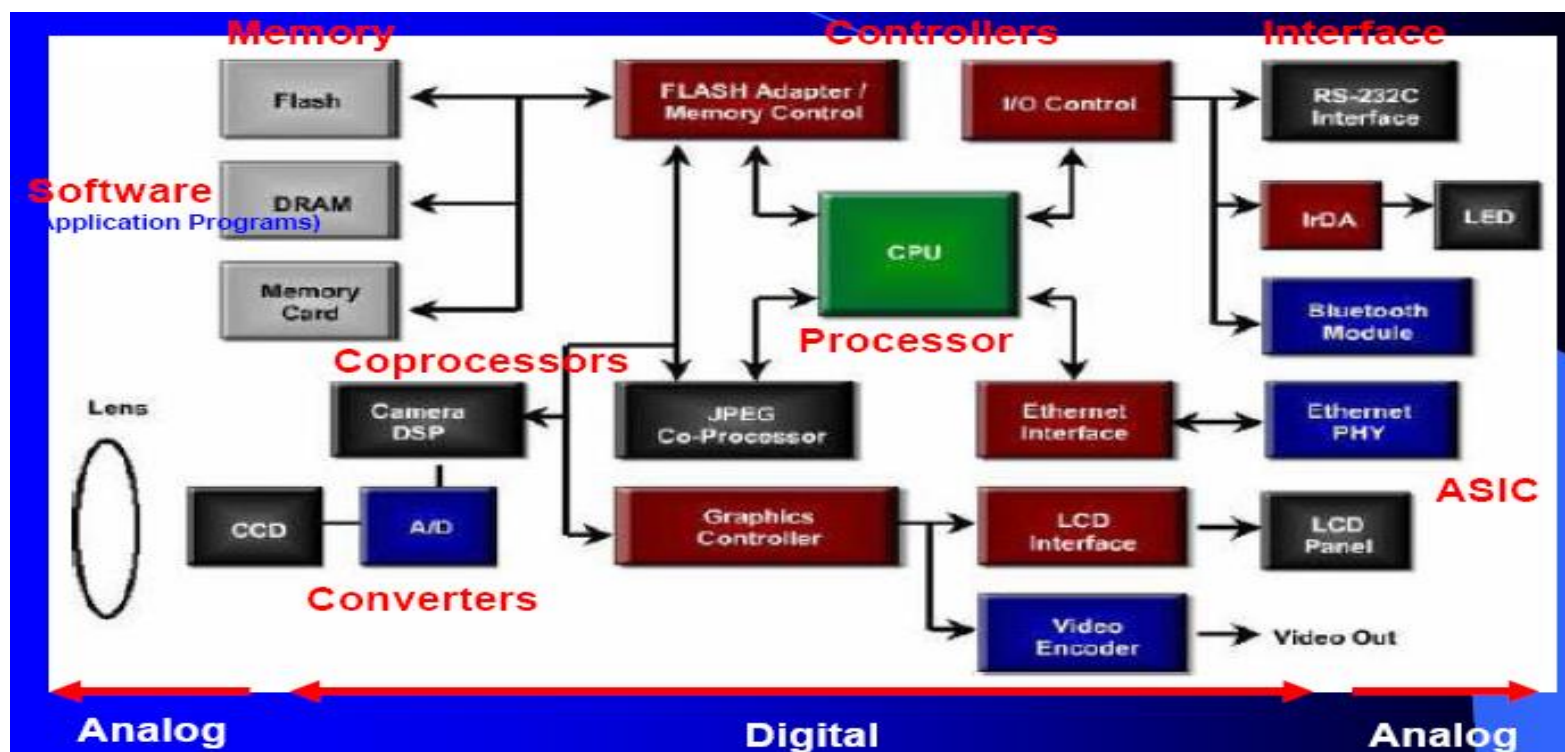
3.6 信息家电

■ 现代化家庭



3.7 其他消费电子

■ 数码相机



4 未来趋势

- 硬件的发展趋势
- 软件的发展趋势
- 系统的发展趋势

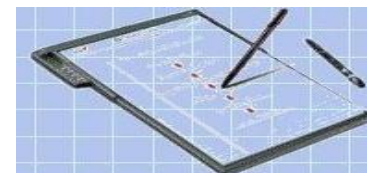
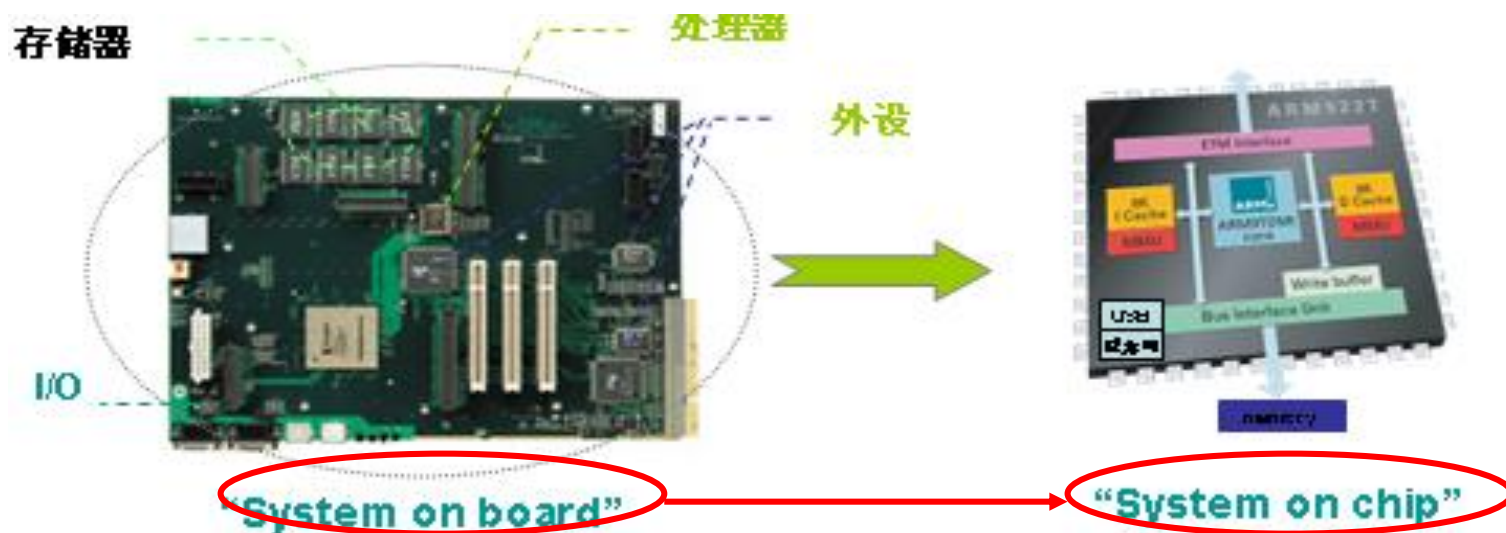


第一章 嵌入式系统概述——未来趋势

4 未来趋势

- 硬件的发展趋势：更小
——嵌入式处理器小型化

- 32位嵌入式微处理器为核心的SoC成为主流



数字笔



中国版
月球车

4 未来趋势

- 软件的发展趋势：更“聪明”
——嵌入式应用软件智能化



机器狗

- 系统的发展趋势：更“紧密”
——嵌入式通信接口网络化



GPS跑鞋

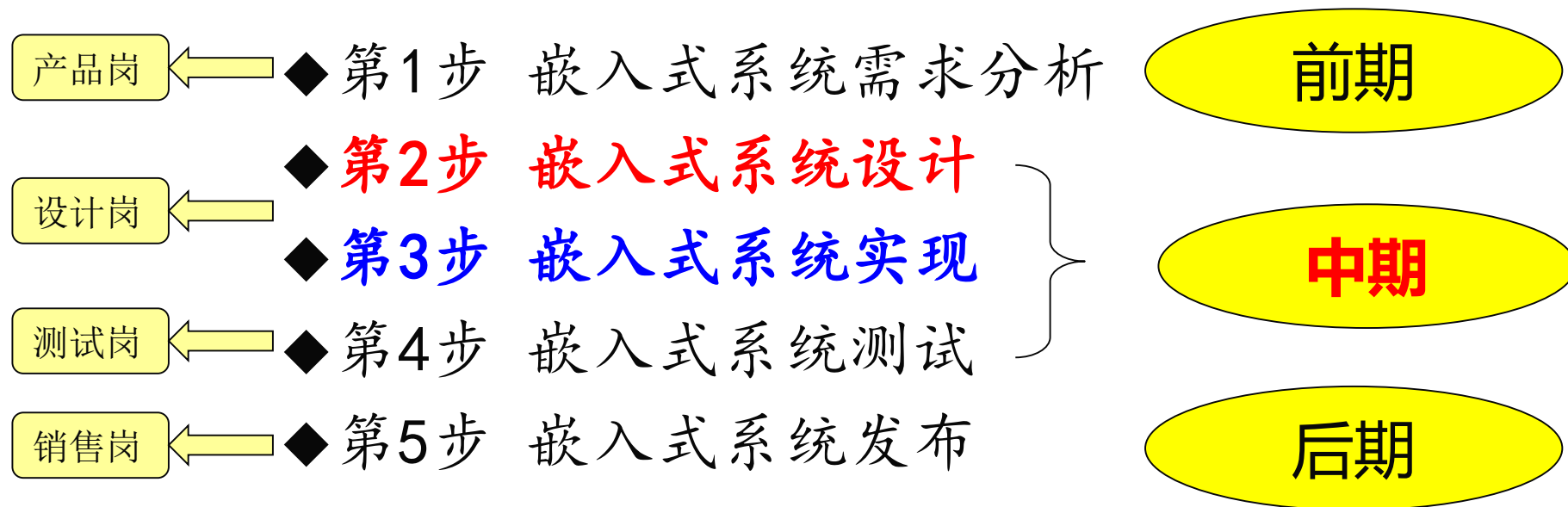
- 嵌入式Internet

5 嵌入式系统开发

- 5.1 嵌入式系统开发过程
- 5.2 嵌入式系统交叉开发环境
- 5.3 嵌入式系统开发工具

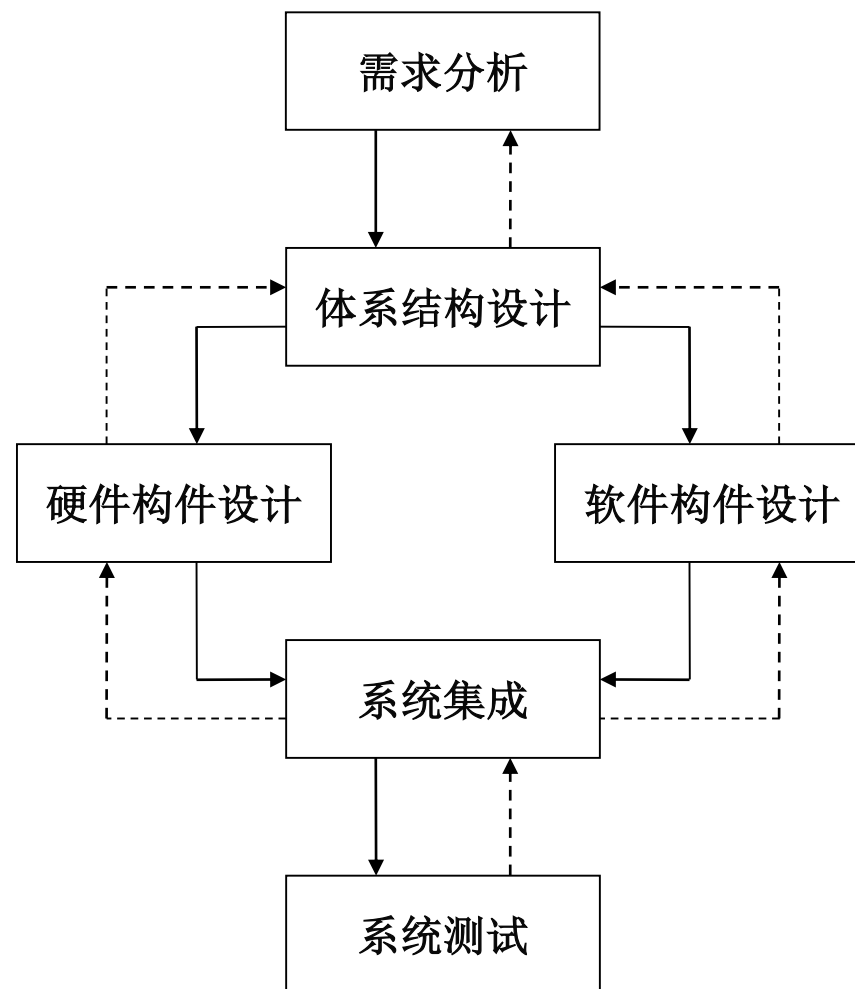


嵌入式系统开发过程



5.1 嵌入式系统开发过程

嵌入式系统的开发过程可以分为需求分析、体系结构设计、软件设计、硬件设计、系统集成和系统优化与测试6个阶段。



5.1 嵌入式系统开发过程

➤ 需求分析

- **需求：**包括功能和非功能两部分，非功能信息包括性能、功耗、成本、系统尺寸和重量等。
- **确认需求：**建立一个模型，使用已存数据来模拟功能，确认需求的合理性。
- **建立需求分析表格：**
- **设计规格说明书，按照需求撰写。**

5.1 嵌入式系统开发过程

➤ 体系结构设计

- 体系结构设计是通过对规格说明中的内容作进一步的分解，建立硬件和软件架构，并确立各个模块、各个子系统之间的关系，定义各子系统和各功能模块之间的接口，规定设计约束，进而给出每个功能模块的功能描述。

5.1 嵌入式系统开发过程

➤ 软/硬件构件设计

- 按照体系结构设计要求，分别对硬件和软件进行设计，在建立的开发环境中，可能提供了大量的硬件构件和软件构件，如CPU、存储器、接口电路、操作系统、各种驱动程序、数据库等标准构件，但还需要设计专用的硬件构件和软件构件，同时可能要做大量定制编程。

5.1 嵌入式系统开发过程

➤ 系统集成

- 软/硬件构件建立以后就要把它们集成在一起以得到一个可以运行的系统，在系统集成中通常可以发现一些构件设计时的错误，所以按阶段搭建系统并且在每个阶段结束后进行测试，就有助于定位这些错误。

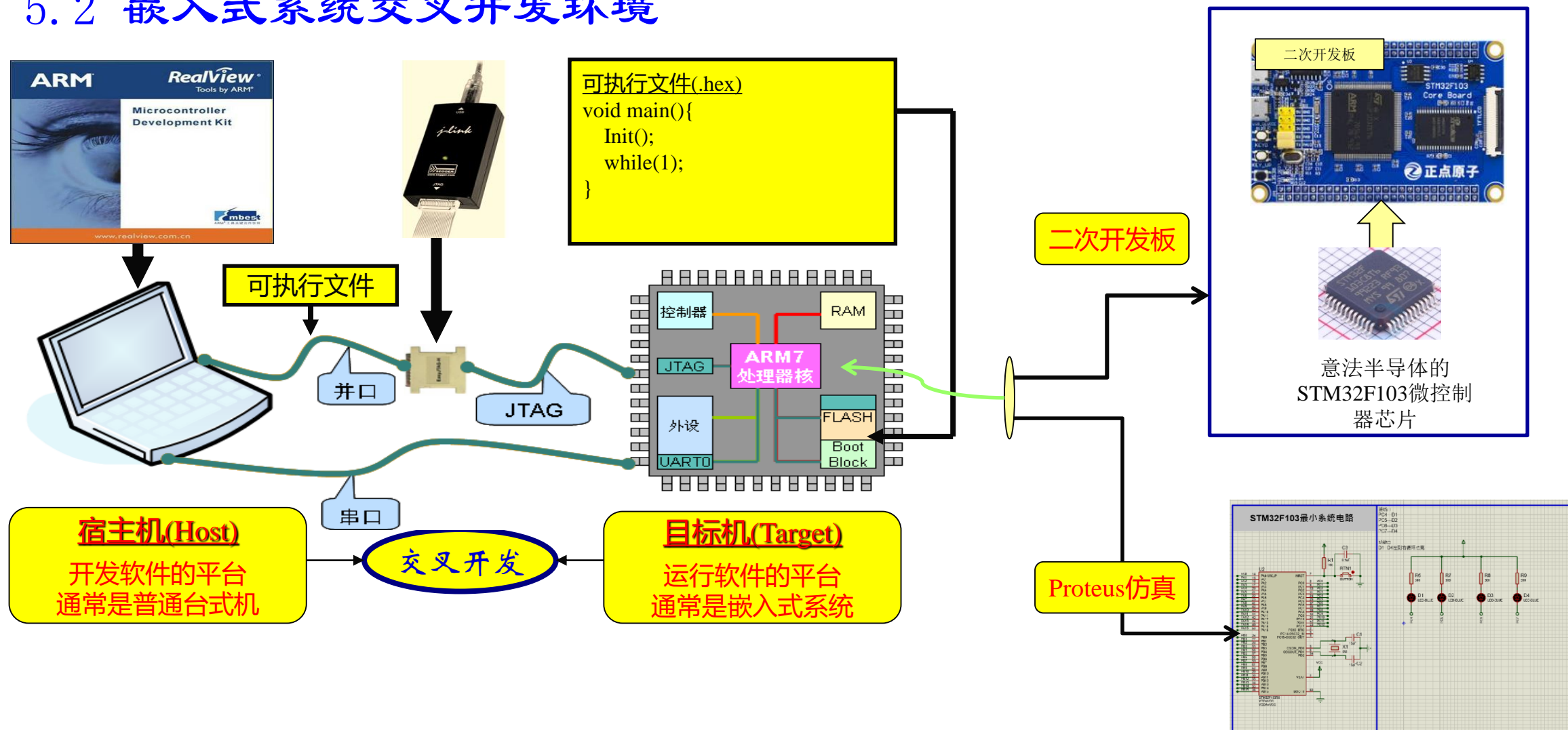
5.1 嵌入式系统开发过程

➤ 系统测试

- 在系统集成基础上，进行各种组装测试和确认测试，通过与系统的需求相比较，发现与用户需求不符或矛盾的地方，进行修改，从而提出更加完善的方案。

第1章 嵌入式系统概述—嵌入式系统开发环境

5.2 嵌入式系统交叉开发环境



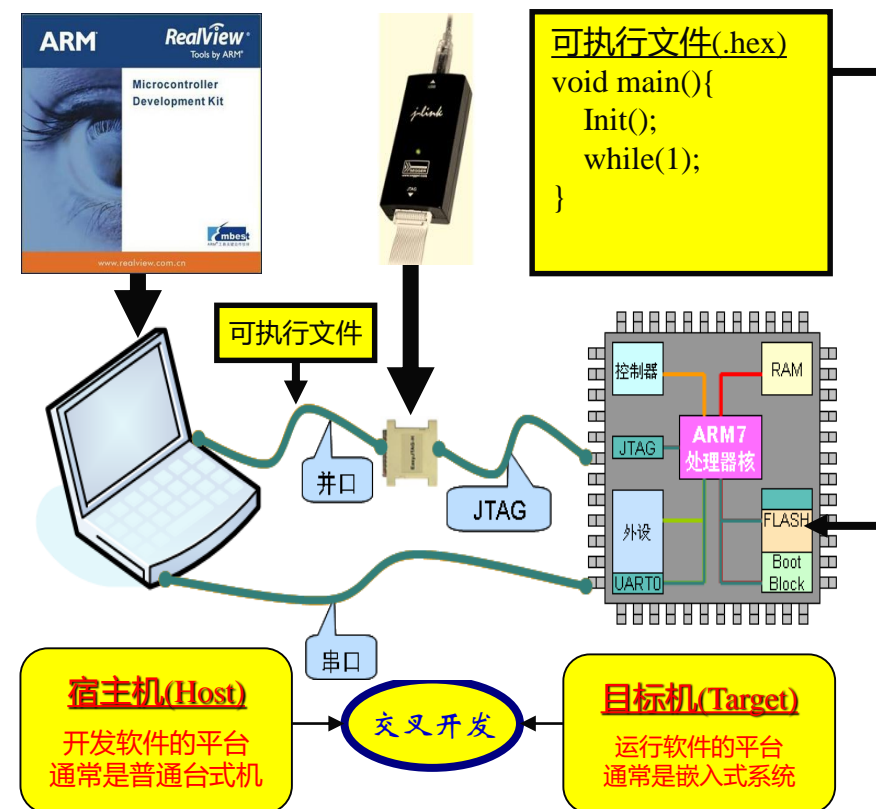
第1章 嵌入式系统概述—嵌入式系统开发环境

宿主机

- 用于开发嵌入式系统的PC机
- 嵌入式应用软件开发环境
- 具备丰富的软硬件资源
- 为嵌入式应用软件开发提供全过程支持

目标机

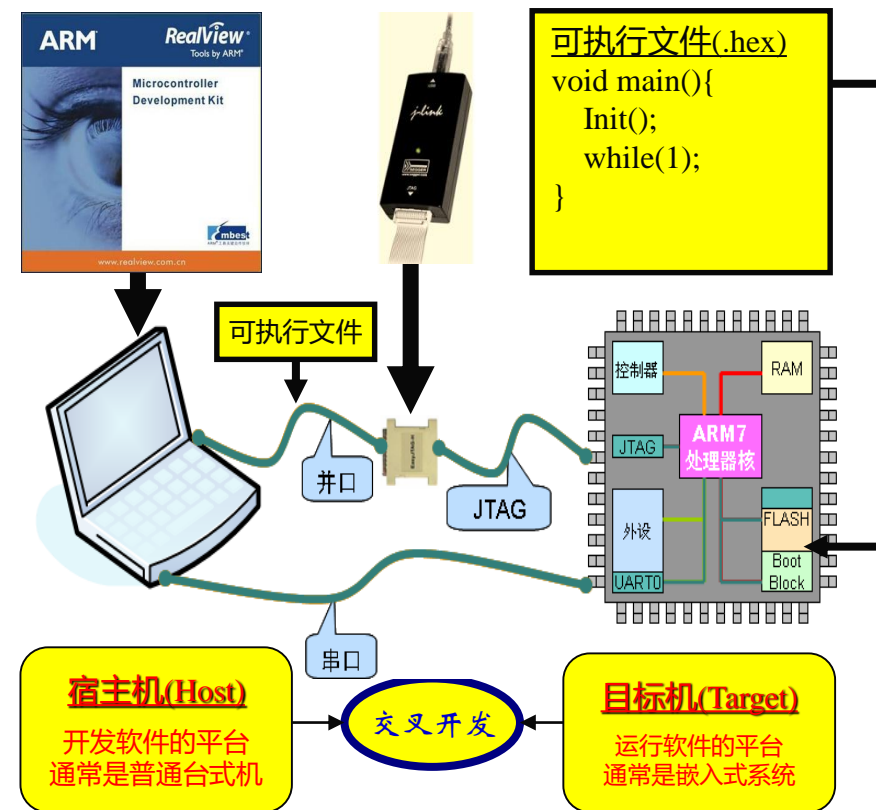
- 嵌入式软件的运行环境
- 软硬件是为特定应用定制
- 在开发过程中，目标机需接收和执行宿主机发出的各种命令如设置断点、读内存、写内存等，将结果返回给宿主机，配合宿主机各方面的工作



第1章 嵌入式系统概述—嵌入式系统开发环境

嵌入式系统的调试

- 调试器是运行于宿主机上的应用程序，被调试程序是运行在目标机上的应用程序，两个程序间需要实时通信。
- 嵌入式系统调试时，调试器通过仿真器与目标机相连。仿真器处理宿主机与目标机之间的通信，这个通信口可以是串口、并口、网口及USB口，仿真器通过JTAG口或其他接口与目标机相连。



第1章 嵌入式系统概述—嵌入式系统开发环境

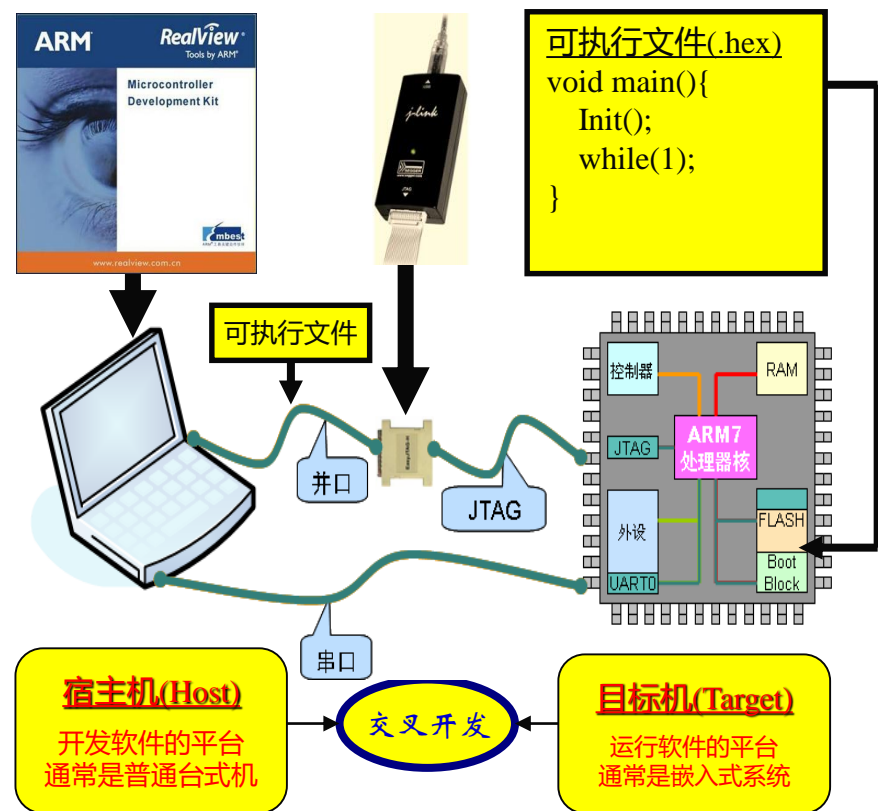
宿主机与目标机的连接

物理连接：宿主机与目标机通过物理线路连接，是逻辑连接的基础

- ◆ 串口
- ◆ 以太网口
- ◆ 在线调试接口（如JTAG等）

逻辑连接：宿主机与目标机间按某种通信协议建立起来的通信连接；前已逐步形成了一些通信协议的标准

J-Link是一个USB到JTAG适配器，支持基于ARM7、ARM9、Cortex-M3处理器的设备，JTAG速度高达12MHz。它使用标准的20针或14针(可选)JTAG连接器与所有评估板和目标硬件接口。



第1章 嵌入式系统概述—嵌入式系统开发环境

- JTAG是一种通用的测试和调试接口标准。它最初由美国电子工业协会的Joint Test Action Group组织制定，旨在提供一种标准化的方式来测试集成电路（ICs）和PCB（Printed Circuit Board）。
- JTAG协议在定义时，由于当时的计算机(PC机)普遍带有并口，因而在连接计算机端是定义使用的并口。
- 因此出现将并口转换为USB与主机相连的下载调试器

J-link(德国SEGGER公司)

U-link（ARM Keil公司）

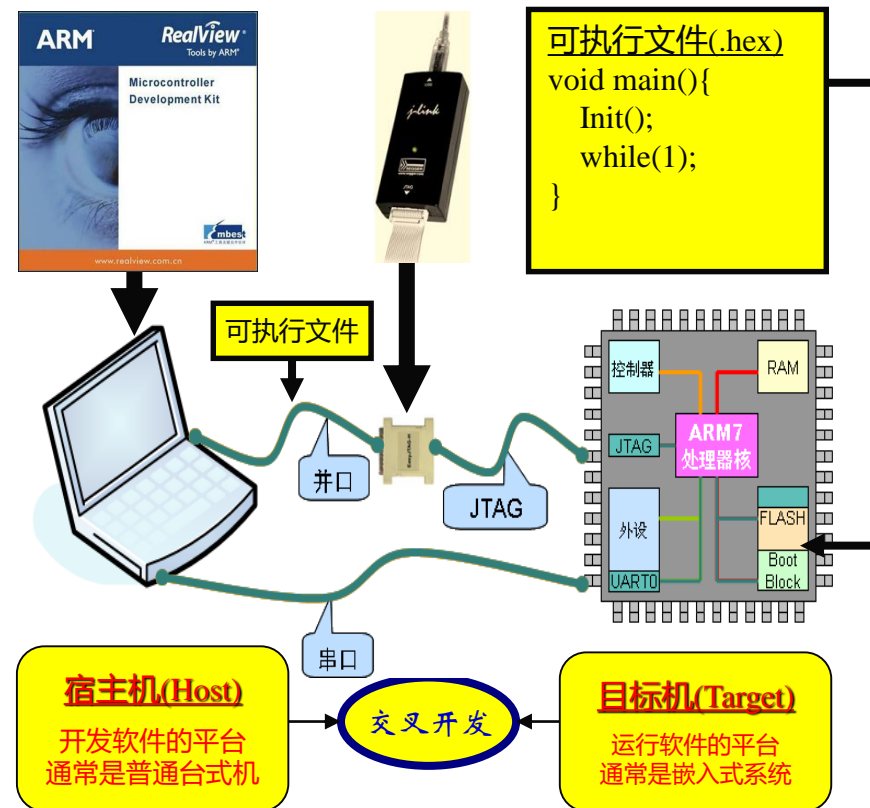
ST-Link（意法半导体公司）



第1章 嵌入式系统概述—嵌入式系统开发环境

嵌入式软件开发过程：

- 首先，在宿主机（host）上建立开发环境，进行应用程序编写和交叉编译；
- 然后，在宿主机和目标机（target）之间建立连接将应用程序下载到目标机上
进行交叉调试；
- 最后，将应用程序固化到目标机中实际运行。



5.3 嵌入式系统开发工具

◆ ARM 嵌入式系统开发工具

---ADS (ARM Developer Suite) --2002年后不再维护

---Keil MDK (Microcontroller development kit)

◆ ARM嵌入式系统仿真工具

---Proteus



第1章 嵌入式系统概述—ADS1.2开发工具

ARM公司的ADS嵌入式开发环境

- ARM ADS 全称为 ARM Developer Suite，是ARM公司为自己的IP核开发的集成开发环境。ADS1.2是 ARM 公司2007年前后推出的新一代 ARM 集成开发工具。它取代了早期的 ADS1.1 和 ADS1.0 。目前ARM公司（2016年已被日本软银收购）不再维护该平台，该平台不支持ARM11后新的IP核。
- ARM ADS1.2集成开发环境软件，包括了四个模块分别是：SIMULATOR；C 编译器；实时调试器；应用函数库。支持的硬件调试器有Multi-ICE以及兼容Multi-ICE的调试工具如FFT-ICE。



第1章 嵌入式系统概述—Keil 开发工具

Keil公司简介

Keil公司是一家业界领先的微控制器（MCU）软件开发工具的独立供应商。Keil公司由两家私人公司联合运营，分别是德国慕尼黑的Keil Elektronik GmbH和美国德克萨斯的Keil Software Inc。Keil公司制造和销售种类广泛的开发工具，包括ANSI C编译器、宏汇编程序、调试器、连接器、库管理器、固件和实时操作系统核心（real-time kernel）。有超过10万名微控制器开发人员在使用这种得到业界认可的解决方案。其Keil C51编译器自1988年引入市场以来成为事实上的行业标准，并支持超过500种8051变种。

Keil公司在2005年被ARM公司收购（收购当时,Keil 公司有23名员工,总资产价值为 460 万美元）。



第1章 嵌入式系统概述—Keil 开发工具

Keil与ARM

Keil公司2005年由ARM公司收购。Keil公司首席执行官Reinhard Keil表示：“作为ARM Connected Community中的一员，Keil和ARM保持着长期的良好关系。通过这次收购，我们将能更好地向高速发展的32位微控制器市场提供完整的解决方案，同时继续在 μ Vision环境下支持我们的8051和C16x编译器。”

而后ARM Keil推出基于 μ Vision界面，用于调试ARM7，ARM9，ARM11，Cortex-M内核的MDK-ARM开发工具，用于为控制领域的开发。



第1章 嵌入式系统概述—Keil 开发工具

界面

Keil μ Vision2: 是美国Keil Software公司出品的51系列兼容单片机C语言软件开发系统，使用接近于传统C语言的语法来开发，与汇编相比，C语言易学易用,而且大大的提高了工作效率和项目开发周期，C51已被完全集成到 μ Vision2的集成开发环境中，这个集成开发环境包含：编译器，汇编器，实时操作系统，项目管理器，调试器。 μ Vision2 IDE可为它们提供单一而灵活的开发环境。

Keil μ Vision3: 2006年1月30日ARM推出全新的针对各种嵌入式处理器的软件开发工具，集成Keil μ Vision3的RealView MDK开发环境。RealView MDK开发工具Keil μ Vision3源自Keil公司。RealView MDK集成了业内领先的技术，包括Keil μ Vision3集成开发环境与RealView编译器。支持ARM7、ARM9和最新的Cortex-M3核处理器，**自动配置启动代码**，集成Flash烧写模块，强大的Simulation设备模拟，性能分析等功能，与ARM之前的工具包ADS等相比，RealView编译器的最新版本可将性能改善超过20%。

Keil μ Vision4: 2009年2月发布Keil μ Vision4，Keil μ Vision4引入灵活的窗口管理系统，使开发人员能够使用多台监视器，并提供了视觉上的表面对窗口位置的完全控制的任何地方。新的用户界面可以更好地利用屏幕空间和更有效地组织多个窗口，提供一个整洁，高效的环境来开发应用程序。新版本支持更多最新的ARM芯片，还添加了一些其他新功能。2011年3月ARM公司发布最新集成开发环境RealView MDK开发工具中集成了最新版本的Keil μ Vision4，其编译器、调试工具实现与ARM器件的最完美匹配。

Keil μ Vision5: 2013年10月，Keil正式发布了Keil μ Vision5 IDE。



第1章 嵌入式系统概述—Keil 开发工具

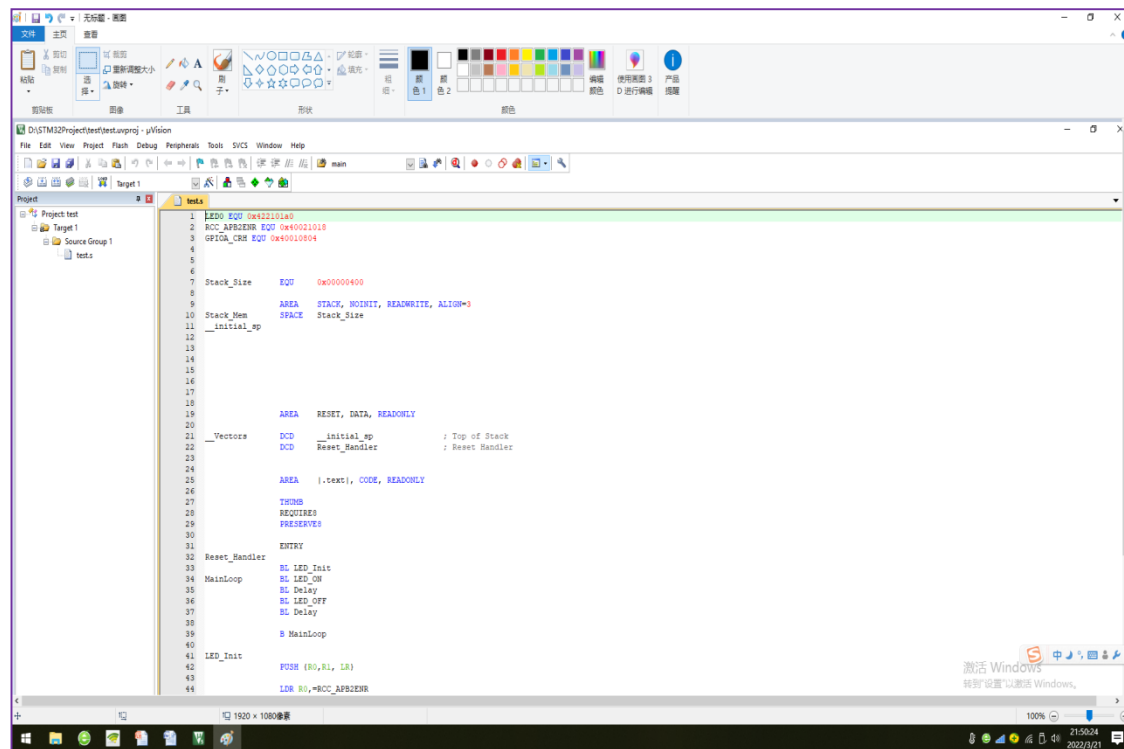
Keil公司推出的微控制器集成开发工具MDK

(Microcontroller development kit)

Keil公司被ARM 收购（2005年）后， Keil先后开发的针对ARM核的 μ Vision系列集成开发工具（ Keil μ Vision3---2006年， Keil μ Vision4---2011年， Keil μ Vision5---2013年） ，目前广泛使用的为Keil μ Vision5。

该版本2013年10月正式推出，是目前针对ARM微控制器，尤其是ARM Cortex-M内核微控制器最佳的一款集成开发工具。 MDK v5向后兼容Keil MDK-ARM μ Vision4，以前的项目同样可以在MDK v5上进行开发， MDK v5同时加强了针对Cortex-M微控制器开发的支持，并且对传统的开发模式和界面进行升级，将分成两个部分，MDK Core和Software Packs。其中，Software Packs可以独立于工具链进行新芯片支持和中间库的升级。

观看视频：开发环境搭建



第1章 嵌入式系统概述—Keil 开发工具

MDK-ARM相比ADS1.2有什么优势？

ADS是ARM的老版本的开发工具，而MDK-ARM是ARM新的开发工具，MDK与ADS相比较，从外观、仿真流程以及内部二进制编译链接工具上都有了不少改进，用法稍有不同。

- ① ADS模拟器只能模拟指令集，而MDK的模拟器能提供指令集、启动代码、外设、中断等整个MCU行为的模拟。
- ② MDK提供性能分析器，而ADS没有。
- ③ MDK提供启动代码生成向导，轻松完成启动生成，而ADS没有此功能。
- ④ ADS不支持CM3，MDK支持。
- ⑤ MDK的项目管理窗口、编译窗口、调试窗口等都在同一个界面，操作方便，上手更易；而ADS的项目管理窗口、编译窗口、调试窗口等需要在不同界面操作，相对不便。
- ⑥ ADS不支持Flash烧写，MDK支持。



第1章 嵌入式系统概述—Keil 开发工具使用

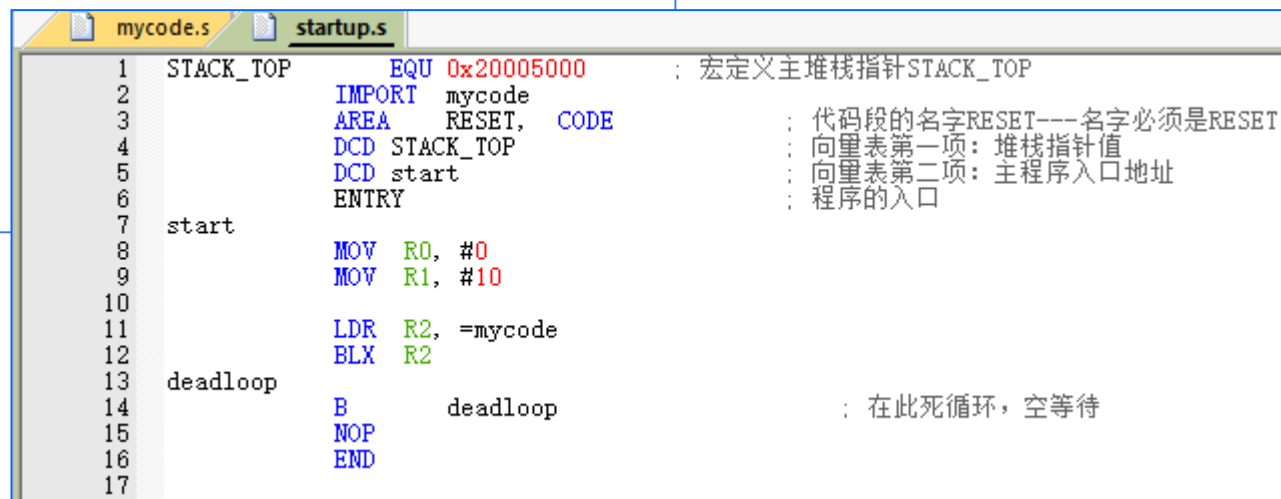
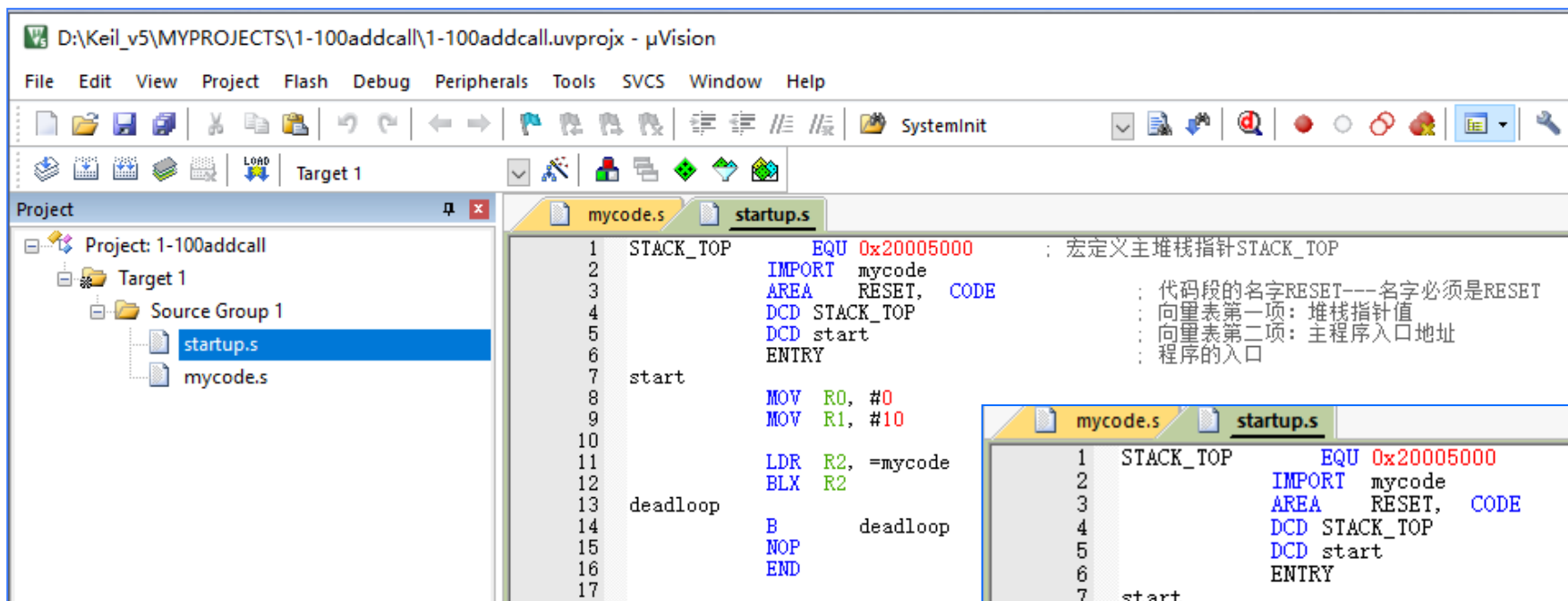
◆纯汇编工程构建（基于ARM的IP核）

◆寄存器工程构建（基于ST的微控制器芯片）

◆标准外设库工程构建（基于ST的微控制器芯片）

◆HAL(Hardware Abstraction Layer)库工程构建（芯片）

◆LL(Low-Level)库工程构建（芯片）



见1-100addcall纯汇编工程



第1章 嵌入式系统概述—Proteus仿真工具

文件(F) 系统(Y) 帮助(H)

主页 X

PROTEUS DESIGN SUITE 8.13

入门教程

- Schematic and PCB (Basic)
- Schematic and PCB (Advanced)
- 仿真
- 新特性

帮助中心

- 帮助主页
- 原理图设计
- PCB布版
- 仿真
- 可视化设计
- IoT Builder

关于软件

(C) Labcenter Electronics 1989-2021
中文版由广州风标教育技术股份有限公司提供
发布版本 8.13 SP0 (31525) 具有高级仿真功能
www.labcenter.com

注册给:
Windway PSTL Training

用户编号: 27-21610-705
评估版许可证过期: 18/01/2022

风标教育 (总代理): www.fengbiaoedu.com
中文社区 (普客圈): www.proteusedu.com

开始设计

打开工程 新建工程 新建流程图 打开示例工程

最近打开的工程

> 更多...

最新消息

Proteus 设计套件评估版

软件已经更新到最新。上次检测在 1 天前。 [手动升级检测。](#)

New in Version 8.13

<ul style="list-style-type: none">Zone InspectorLibrary Part Editing	<ul style="list-style-type: none">Design WalkPre Production Check (Updated)	<ul style="list-style-type: none">VSM Studio (Updated)
---	--	--

New in Versions 8.8 to 8.12

<ul style="list-style-type: none">Multi Board SupportDiff Pair Pass ThroughDXF Importing	<ul style="list-style-type: none">Non-Functional PadsPick and Place ConfiguratorDirect Library Part Importer	<ul style="list-style-type: none">SnapEDA Library IntegrationPCB Panelization (Updated)	<ul style="list-style-type: none">Component Placement ReportAuto Complete Routing
--	--	--	--

[more guides](#)

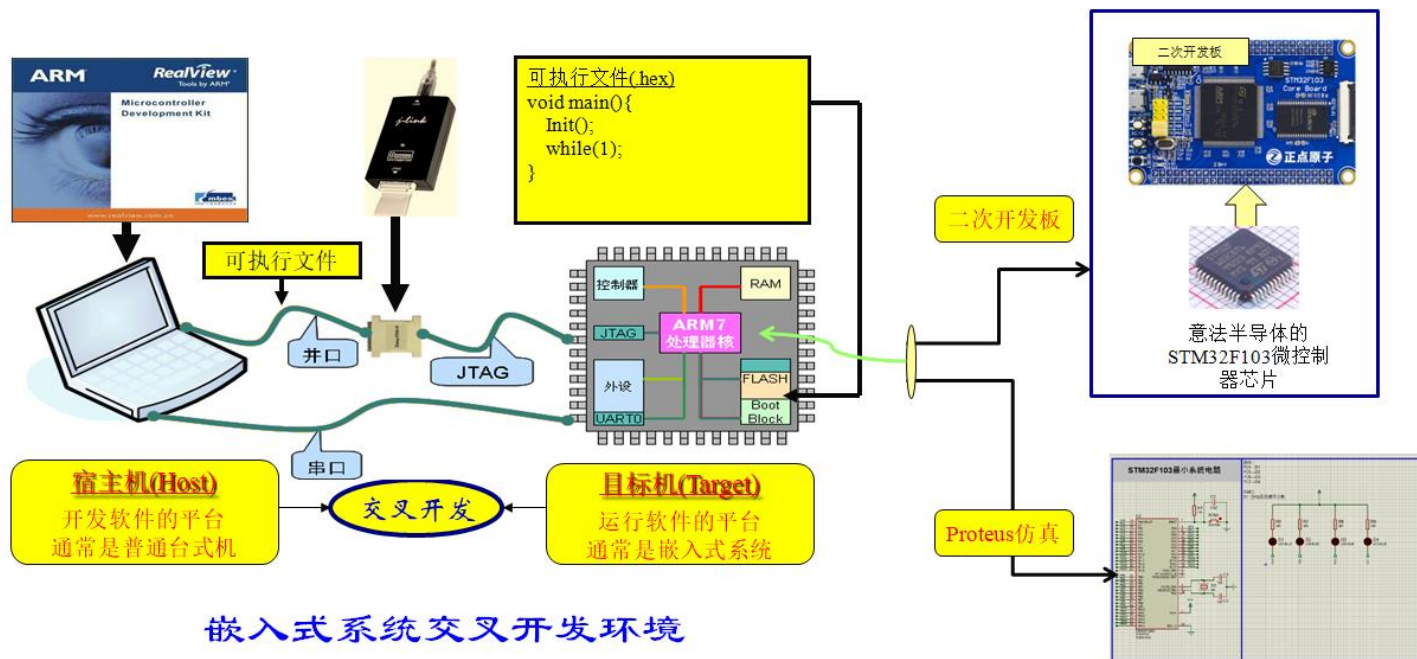
Getting Started Movies

<ul style="list-style-type: none">Installation and FilingApplication Overview	<ul style="list-style-type: none">New Project (PCB)New Project (VSM)	<ul style="list-style-type: none">Import Project (PCB)Import Project (VSM)
--	---	---



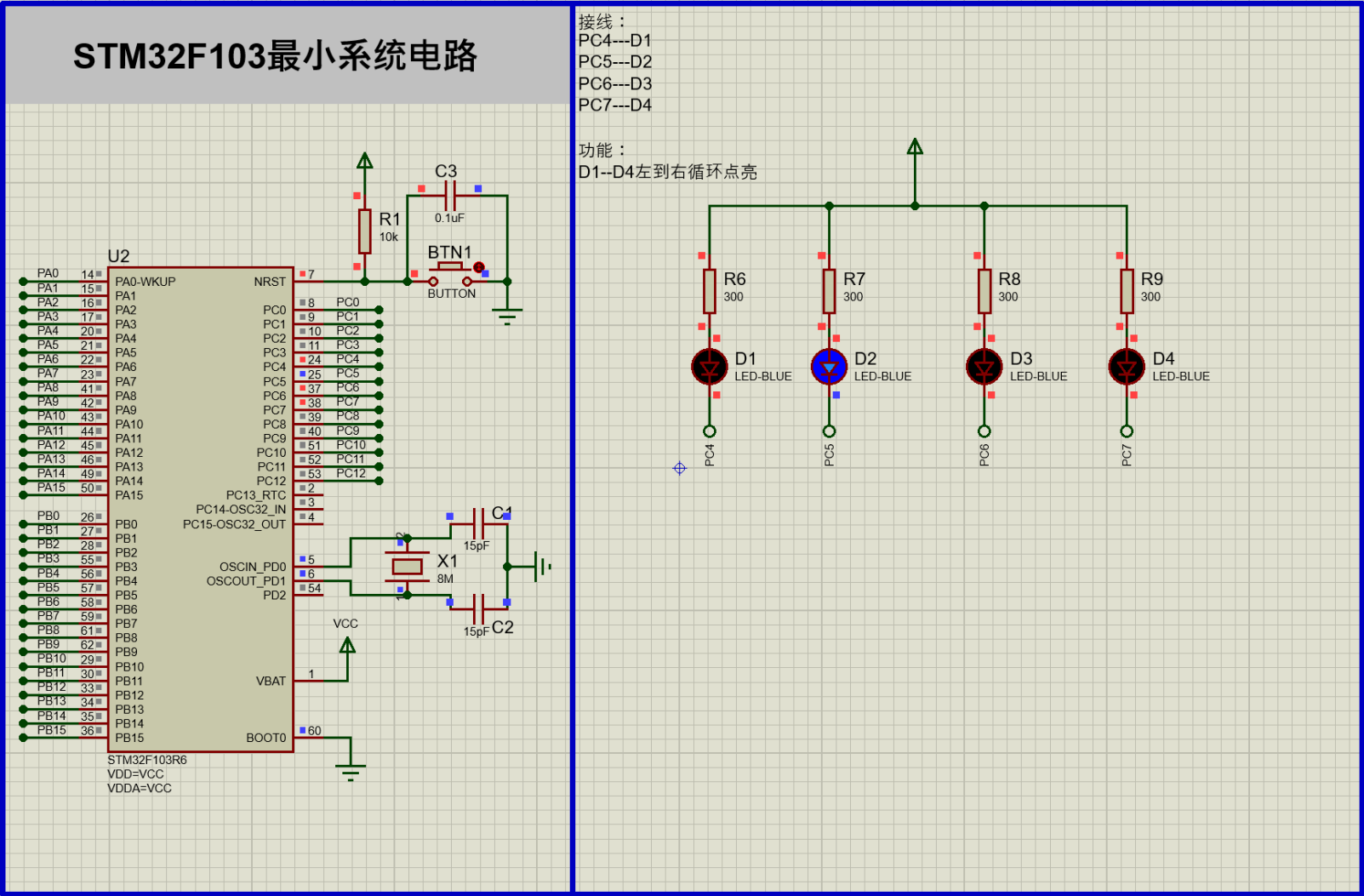
第1章 嵌入式系统概述—嵌入式系统设计与仿真举例

- EXP1_GPIO实验
- EXP2_按键控制实验
- EXP3_外部中断实验
- EXP4_定时器中断实验
- EXP5_串口通讯实验
- EXP6_数码管驱动显示实验
- EXP7_模数转换实验
- EXP8_温度检测实验
- EXP9_DMA单通道ADC采集实验
- EXP10_UCOS-II的LED控制任务



大家可以参考实例进行片内 I/O 的学习

第1章 嵌入式系统概述—STM32微控制器最小系统



使用STM32F103R6微控制器设计的流水灯

见EXP1_GPIO实验



广州市星翼电子科技有限公司STM32F1系列开发板

◆ 学习资料下载网站：

<https://www.yuanzige.com/>

◆ 官方网站：

<http://www.alientek.com>

开源电子网(正点原子官方品牌店)

www.openedv.com



本课程学习建议——献给新手们——STM32学习 建议

- ◆ 步骤一，安装嵌入式开发环境：Keil for ARM (MDK)、J-Link、ISP（如果需要从串口下载的话）。
- ◆ 步骤二，构建基于IP核的纯汇编工程，了解Cortex-M3内核结构和工作原理，掌握基本的汇编语言程序设计方法。
- ◆ 步骤三，在第三方的开发板上运行一个简单的例子。如LED灯的例程HEX文件，下载到STM32开发板中，观察LED灯的闪烁情况。找找感觉，对嵌入式系统的开发有个感性认识。
- ◆ 步骤四，准备几个常用的文档，比如《STM32的用户手册》，《STM32固件库使用手册》等文档。用于平时查阅。
- ◆ 步骤五，开始查看例程的编写，看看例程是如何写的，自己可否修改下例程，达到自己想要的效果呢？秉火STM32开发板的光盘中为大家提供了丰富的例程代码，可以参考。您一定可以修改出更精彩的例程！
- ◆ 步骤六，Ucos-II的移植，是否需要试一下？
- ◆ 恭喜你，至此，你已经可以自如进行独立的开发了。最后一步，给自己一个目标（项目），把它实现出来！



第一章 嵌入式系统概述——本章小结

■ 什么是嵌入式系统 —— 基本概念

➤ 定义³

➤ 特点⁶

➤ 分类⁴

➤ 组成²⁽³⁺³⁾⁺¹

嵌入式系统与桌面系统的十大不同

■ 嵌入式系统的过去 —— 发展进程

■ 嵌入式系统的现在 —— 应用领域

■ 嵌入式系统的将来 —— 未来趋势

➤ 硬件的发展：更小

➤ 软件的发展：更聪明

➤ 系统的发展：更紧密

■ 嵌入式系统开发



本次课程内容就介绍完了，
同学们，再见！

