SOFTWARE COLLEGE OF NORTHEASTERN UNIVERSITY



3. Embedded Design Lifecycle

Review

- History and Future
- Characteristics of Embedded Systems
- Software issues
- Hardware issues

History and Future

- 经过发展,到80年代初微处理器及微控制器各自已发展为一个庞大的家族,以 Intel公司x86为主流的应用于个人计算机PC的微处理器格局已形成。
- 为了区别于原有使用在PC的通用计算机,把嵌入到对象体系中、实现对象体系 智能化控制的微控制器的计算机,称作嵌入式计算机.
- 因此,嵌入式计算机是诞生于微处理器发展时代;
- 这也标志着计算机进入了通用计算机与嵌入式计算机两大分支、并行发展时代, 从而导致20世纪末,计算机应用的高速发展并由此引发了计算机分类方式的变化。

Characteristics of Embedded Systems

- ➤ Power limitations and constraints (功耗低、体积小、具有专用性)
- Must operate in extreme environmental conditions
- > Far fewer resources than a general purpose computer

Software issues

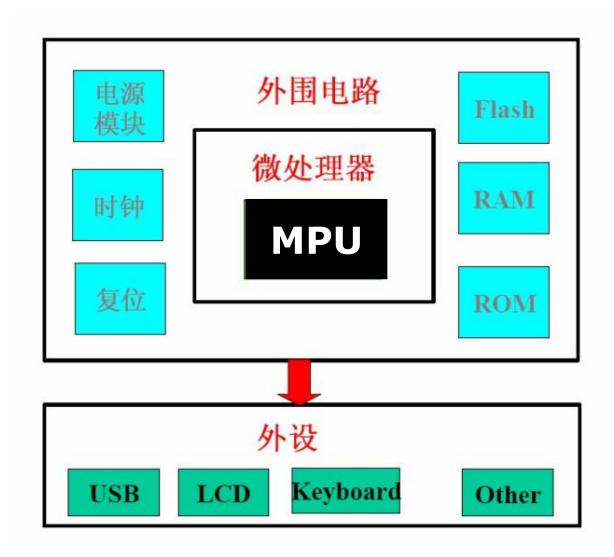
- C语言
- C++
- JAVA
- Objective-c
- ■汇编语言
- 其他

Specialized development tools

Need for a Real-Time Operating System(RTOS)

Hardware issues

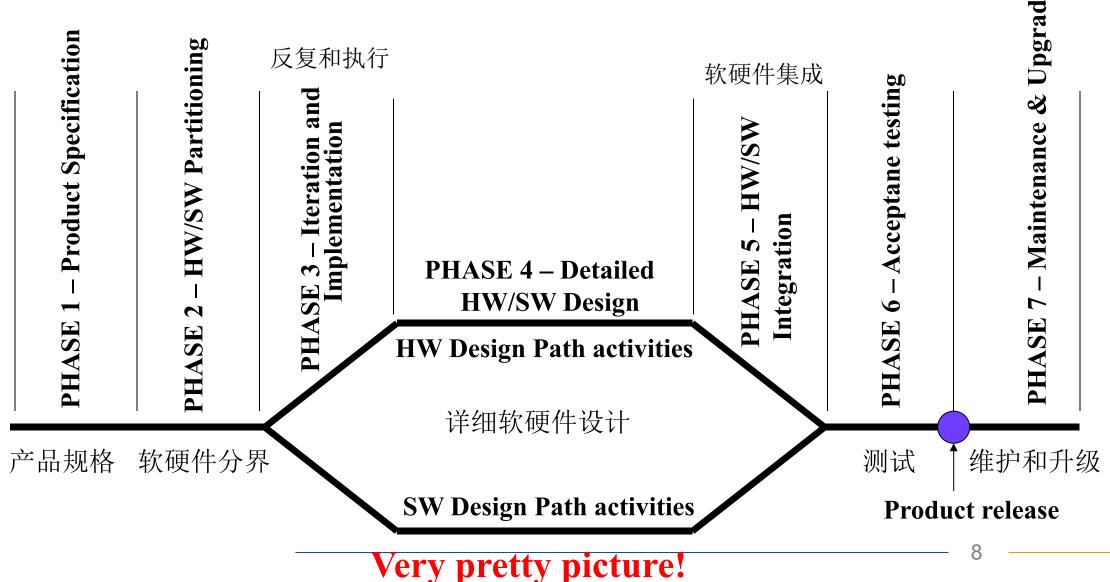
- > Typical Hardware
 - □嵌入式处理器
 - □外围电路
 - ■接口
 - □外设
- Hardware methods
 - □示波器
 - □ 数据分析仪



Elements of Embedded Design

- Design lifecycle
- Component selection process and an overview of available technologies
- Partitioning of hardware and software subsystems

Embedded Design Lifecycle

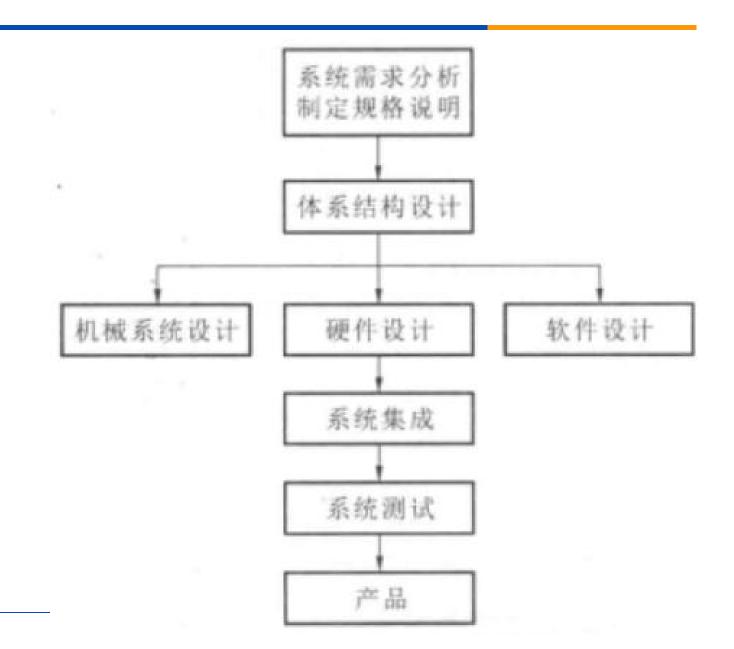


嵌入式系统开发流程

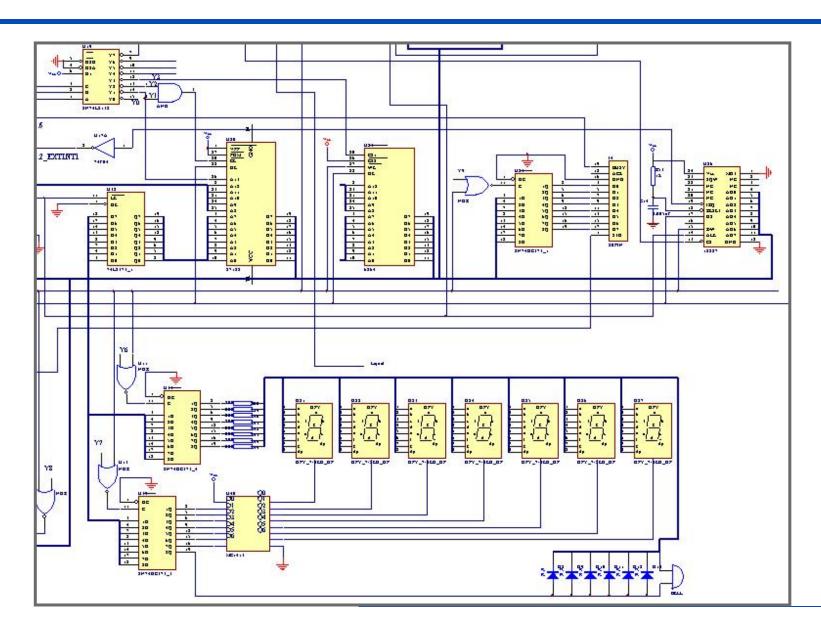
软硬件

协同设计

- ➤需求分析
- > 概要设计
- > 详细设计
- >绘制PCB
- ▶ 制板、安装调试硬件
- > 移植操作系统
- > 软件设计
- > 编码、测试

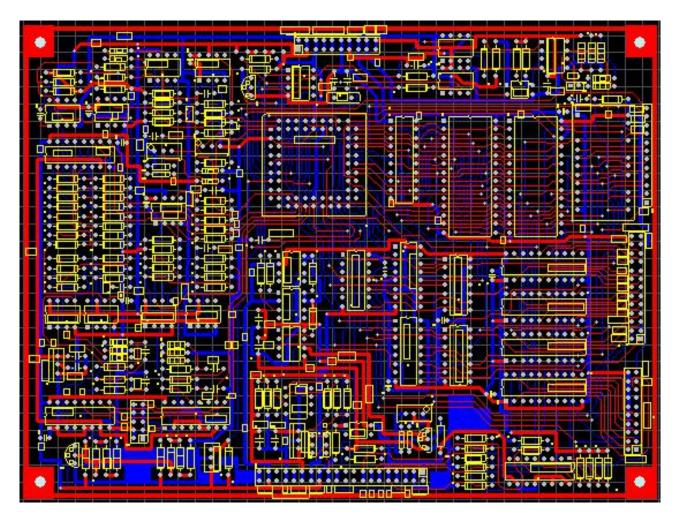


PCB图



PCB图

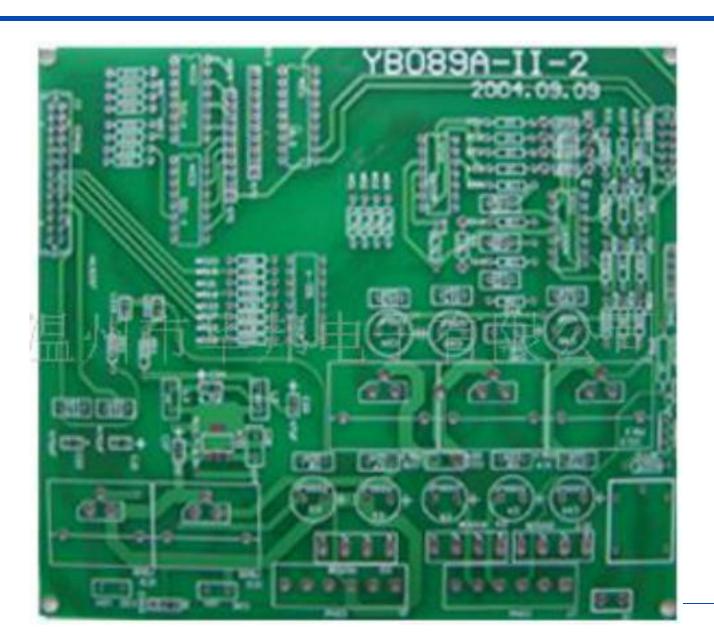
- ➤ 印刷电路板 (PCB板)
 - □ (阻燃环氧树脂复合) 基板
 - □电源线、信号线
 - □焊盘、过孔
 - □覆铜、丝印层



制板

- > 2层板
- > 4层板
- > 6层板

> 检测



焊接硬件

> 先小后大的原则





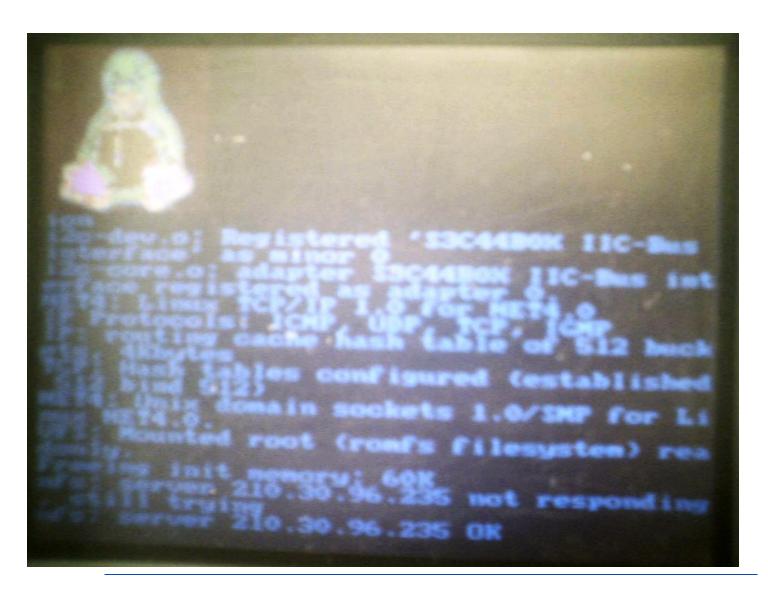
调试硬件

▶ 先调试显示部分,再调试其他部分



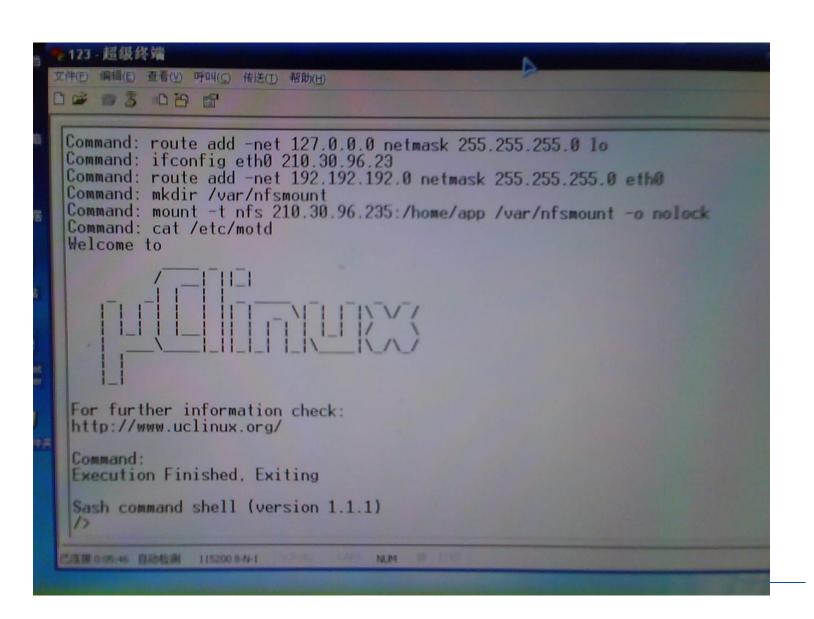
移植操作系统

> uClinux



移植操作系统 (续)

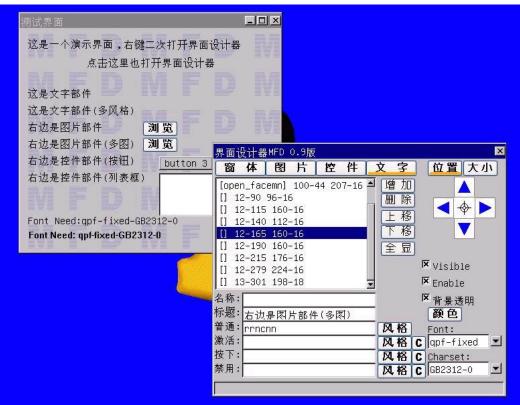
> uClinux



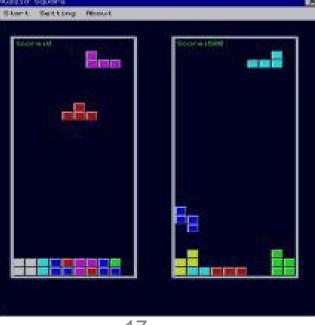
图形界面

> miniGUI









17

图形界面 (续)

➤ MiniGUI



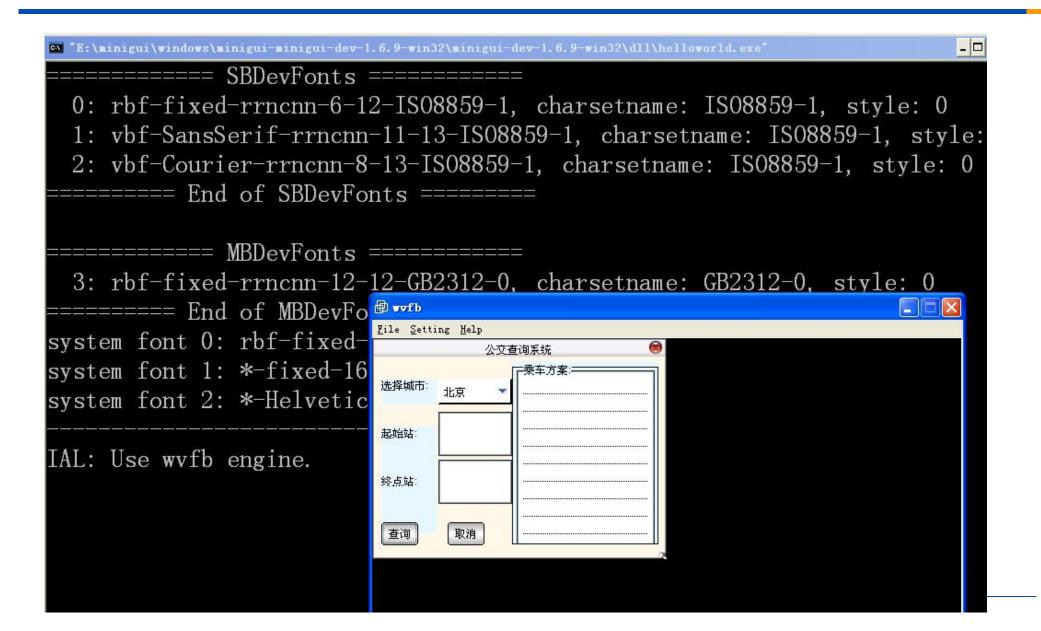
图形界面 (续)

> miniGUI





编码、测试



- > CPU
 - ☐ Intel x86
 - □ PowerPC(Mac) G3,G4,G5
 - ☐ SPARC, Alpha
 - □ ARM
 - MIPS
 -

> CPU DATA Wide

- 8 Bit controllers (still!)
- □ 16 Bit controllers (mainly)
- □ 32 Bit controllers (popular)
- □ 64 Bit controllers (high performance)

- Choices for processor architecture
 - □ CISC (Complex Instruction Set Computer)
 - Many instructions which can perform involved operations: compact code
 - Can be many clock cycles per instruction
 - Large silicon area > Higher cost per die

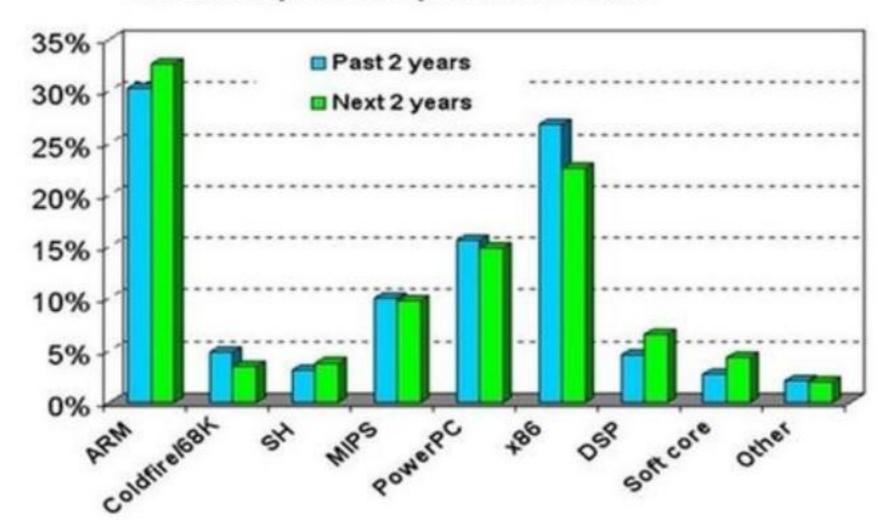
- > Choices for processor architecture
 - □ RISC (Reduced Instruction Set Computer)
 - More modern architecture
 - One instruction executed per clock cycle > Very fast

- ➤ Choices for processor architecture
 - □DSP (Digital Signal Processor)
 - Specialized type of uP
 - Designed for real time mathematical manipulation of data streams
 - Includes instructions designed for multiplication and accumulation

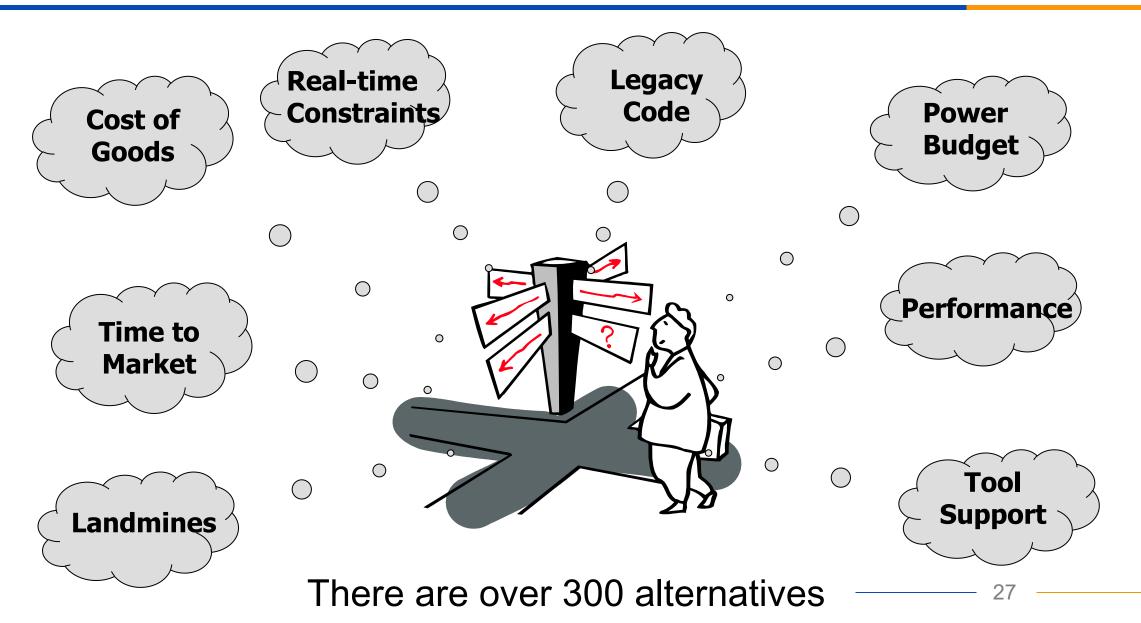
Choice can be one of the three or a combination

CPU Market

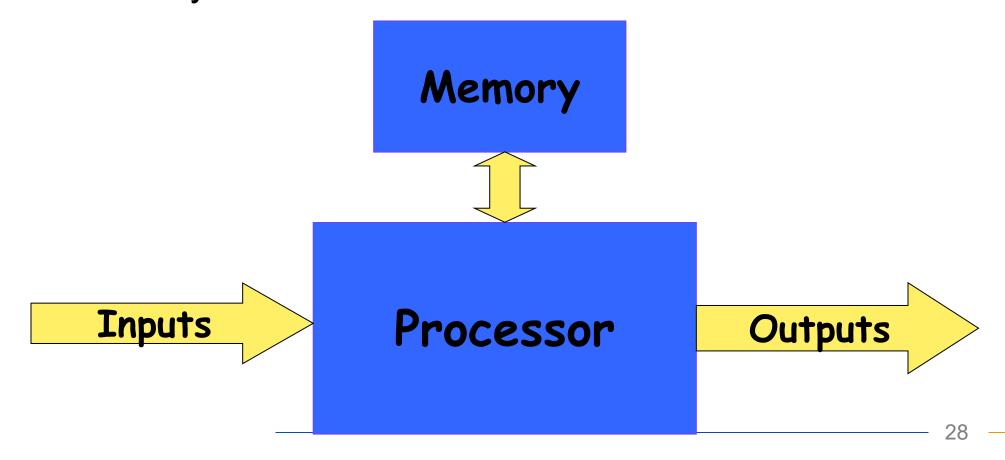
Embedded processor preference trends



How to choose a processor?



Besides CPU and software, what else is common among embedded systems?



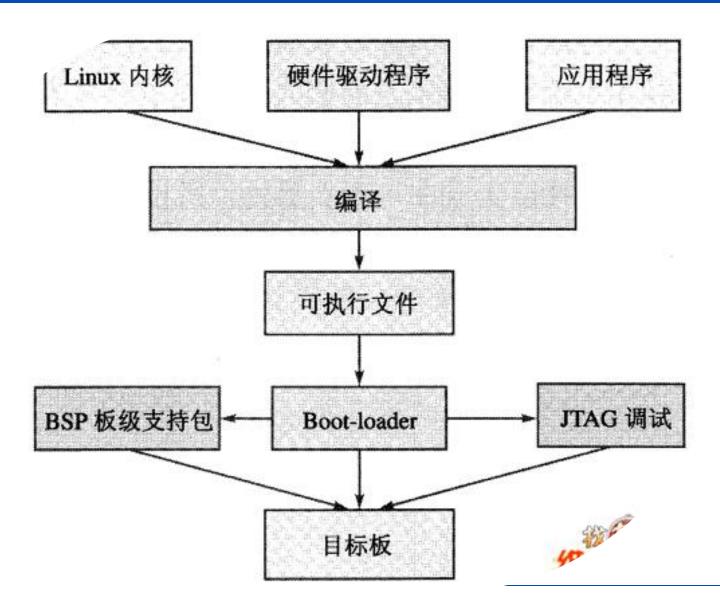
- Besides CPU and software, what else is common among embedded systems?
 - ☐ Memory storage: ROM, RAM, ...
 - □ Input: knobs, buttons, probes, sensors, communication signals, ...
 - ☐ Output: human-readable display, microwave radiation, communication signals, changes to physical world

- ➤ Adequacy of I/O and processor capabilities
 - □ Digital and analog input/output facilities.
 - ☐ Timers, PWM channels, encoder interfaces
 - □ Communication channels: I2C, RS232, parallel, CAN, ethernet
 - □ Interrupt sources and configurability

- ➤ Performance (性能)
 - ☐ Processor core speed
 - □ I/O performance (i.e. speed, accuracy)
 - □ Dedicated hardware for communications and timing operations
 - □ Requires a collection of benchmarking (基准) standard and tools

- > RTOS and tool chain availability
 - □ RTOS compatibility, device drivers and debugging tools
 - □ Tool chain: compilers, debugging facilities and performance evaluation tools

Programming Embedded Systems



Programming Embedded Systems

- Need to have knowledge of hardware/electronics
- Need to operate in constrained environment
 - No or simple Operating System
 - □ Low memory, slow processor
- ➤ Infinite Loop
 - ☐ E.g. anti lock brakes
- Reliability

Programming Small Embedded Systems

No hard disk, where does the code go?

D EPROM

- > No monitor, how can you see it?
- Use a standard computer to develop, and put the program onto EPROM.
- No monitor, how can you debug it?
 - detective work. And advanced tools (e.g. Oscilloscope)

- (1) 目前国内外这方面的人都很稀缺
 - □ 一方面,是因为这一领域入门门槛较高.
 - □ 要懂较底层软件(如操作系统级、驱动程序级软件);
 - □ 对软件专业水平要求较高(嵌入式系统对软件设计的时间和空间效率要求较高);
 - □ 而且必须懂得硬件的工作原理,所以非专业IT人员很难切入这一领域;

- (1) 目前国内外这方面的人都很稀缺
 - □ 另一方面,是因为这一领域较新,目前发展太快,很多软硬件技术出现时间不长或正在出现(如ARM处理器、嵌入式操作系统、MPEG技术、无线通信协议等),掌握这些新技术的人当然很少。嵌入式人才稀缺,身价自然就高,越有经验价格就越高。其实嵌入式人才稀少,根本原因可能是大多数人无条件接触,这需要相应的嵌入式开发板和软件,另外需要有经验的人进行指导开发流程。

- (2) 嵌入式领域人才的工作强度略低,但收入不低
- □ 搞企业应用软件的IT企业,这个用户的系统搞完了,又得去搞下一个用户的,而且每个用户的需求和完成时间都得按客户要求改变,往往疲于奔命,重复劳动。相比而言,搞嵌入式系统的公司,都有自己的产品计划,按自己的节奏行事。所开发的产品通常是通用的,不会因客户的不同而修改。一个产品型号开发完了,往往有较长一段空闲时间,有时间进行充电和休整。

- (3) 产品加密性好
 - □ 从事创业,搞自己的产品,那么嵌入式是一个不错的主意。这可不像应用软件那样容易被盗版。

□ 饭馆用的点菜PDA(WinCE平台,可无线连网和上网),这些PDA的硬件设计一般都是请其它公司给订做(这叫"贴牌":OEM),都是通用的硬件,我们只管设计软件就变成自己的产品了。

从事嵌入式软件开发的缺点是:

- 入门起点较高,所用到的技术往往都有一定难度,若软硬件基础不好,则可能不适于此行。
- ➤ 这方面的企业数量要远少于计算类企业。特别是从事嵌入式的小企业数量较多 (小企业要搞自己的产品创业),知名大公司较少(主要有 Intel、Motorola、TI、 Philip、Samsung、Sony、研华、华为、中兴通信、上广电等制造类企业)。这 些企业的习惯思维方式是到电子、通信等偏硬专业找人。

高校同学学习嵌入式

- ▶ 正规高校软件学院同学若学习嵌入式,显然应偏重于嵌入式软件, 对于搞嵌入式软件的人,最重要的技术显然是(实际上很多公司的招聘广告上就是这样写的):
 - □掌握主流嵌入式微处理器的结构与原理
 - □必须掌握一个嵌入式操作系统
 - □必须熟悉嵌入式软件开发流程并至少做过一个嵌入式软件项目。

电子信息工程专业

- 1、嵌入式实时操作系统(RXT51tiny、uccosII、Linux、android等)、
- 2、ARM指令系统
- 3、主流ARM芯片体系结构
- 4、外围接口电路原理及应用
- 5、嵌入式Linux及驱动程序设计等基本内容
- 6、嵌入式硬件平台
- 7、主流FPGA芯片体系结构以及外围接口电路
- 8、SOC现状及发展前景

软件工程专业

- 1、嵌入式操作系统WINCE、Linux、Android、iOS
- 2、嵌入式数据库及其应用程序设计
- 3、嵌入式软件方面对功耗、成本、实时性、运算量等条件限制下的常用算法及实现
- 4、嵌入式软件和应用
- 5、工业现场自动控制技术
- 6、现场总线及传感器技术
- 7、掌握嵌入式系统外围接口电路。
- 8、嵌入式Linux图形系统程序设计
- 9、FPGA驱动程序设计

Embedded Software Engineering

ENJOY THE COURSE!!!