

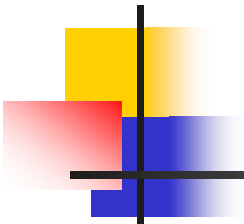
嵌入式Linux学习七步曲

Sailor_forever(扬帆)

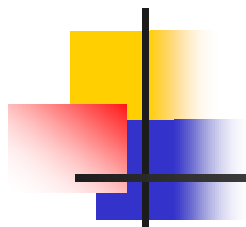
自由传播 版权所有 翻版必究



八一卦-我是who

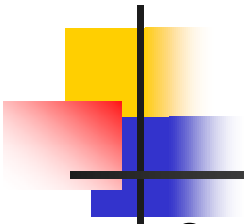
- 
-
- n 目前就职于通信行业某外企研发中心
 - n 参与校园招聘和社会招聘的技术面试工作
 - n 5年嵌入式软件开发经验，擅长嵌入式Linux开发；
 - n 接触的软硬件平台包括ARM，DSP，PowerPC，uC/OS-II，Linux，VxWorks及OSE

八一卦-我是who



- n 嵌入式Linux七步曲 学习群 交流讨论 资源共享
- n 群号 107900817
- n 7steps2linux@gmail.com
- n http://blog.csdn.net/sailor_8318

嵌入式水平小调查



n 0—3个月

n 3—6 个月

n 1年左右

n 2年以上

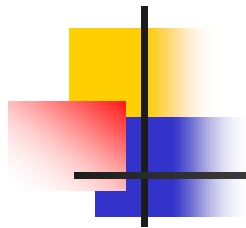
n 多少人参加过系列交流会？

嵌入式Linux学习七步曲

- 
- 1 Linux主机开发环境
 - 2 嵌入式Linux交叉开发环境
 - 3 Linux系统bootloader移植
 - 4 Linux的内核移植
 - 5 Linux的内核及驱动编程
 - 6 文件系统制作
 - 7 Linux的高级应用编程

Grammy for Linux





STAR Volunteer

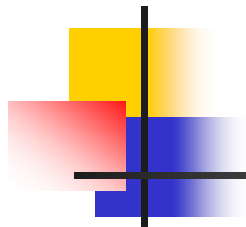
n Linux主机开发环境

- n 于艳丽，豆豆，Mount、hgfs、nfs
- n 王钢，布鲁诺，Samba
- n 黄康莹，lightWind，GDB调试
- n 王晓宇，阿飞，Linux安装



STAR Volunteer





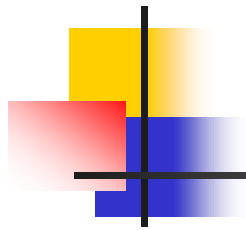
Volunteer Task

n 宗旨

- n 鼓励大家实际的参与嵌入式Linux的开发
- n 自己解决动手解决问题
- n 总结记录、分享
- n 形成知识库
- n 采用统一的模板，争取成为系列交流会的特色项目
- n 扩大BUPT BES的影响力，创造品牌

n 运作

- n 下次交流会之前完成上次的总结文档
- n Share给大家，提建议意见
- n 每次交流会颁奖鼓励
- n 最终将评出 STAR Volunteer



Volunteer Task

n Logo

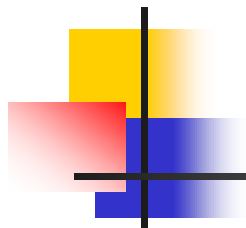
《嵌入式 Linux 学习七步曲》
BUPT BES 系列交流会 Volunteer Task



Key To Success

- n Google、Baidu
- n 理论 + 实践（开发板）
- n 勤于思考，善于总结
- n 多上相关技术论坛，他山之石可以攻玉
- n 良好的文档撰写习惯
- n Passion！



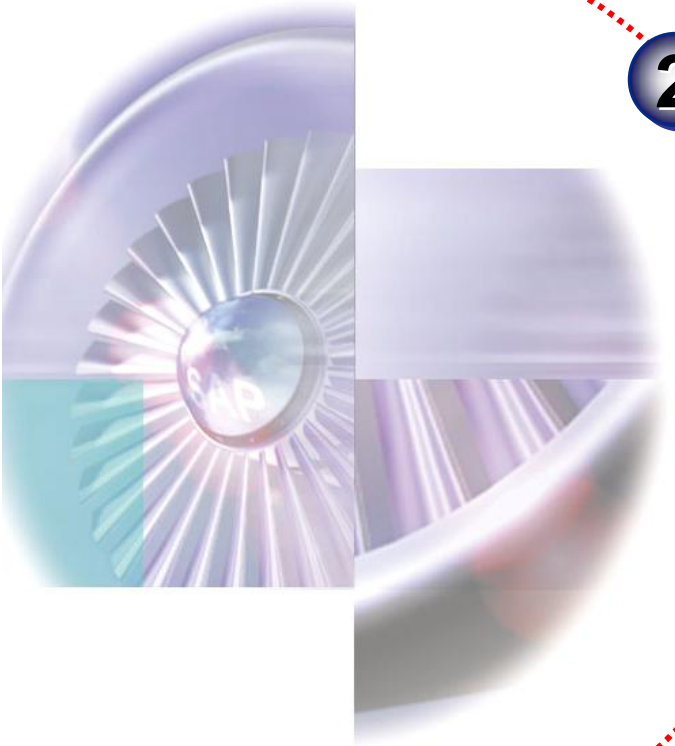


CHAPTER

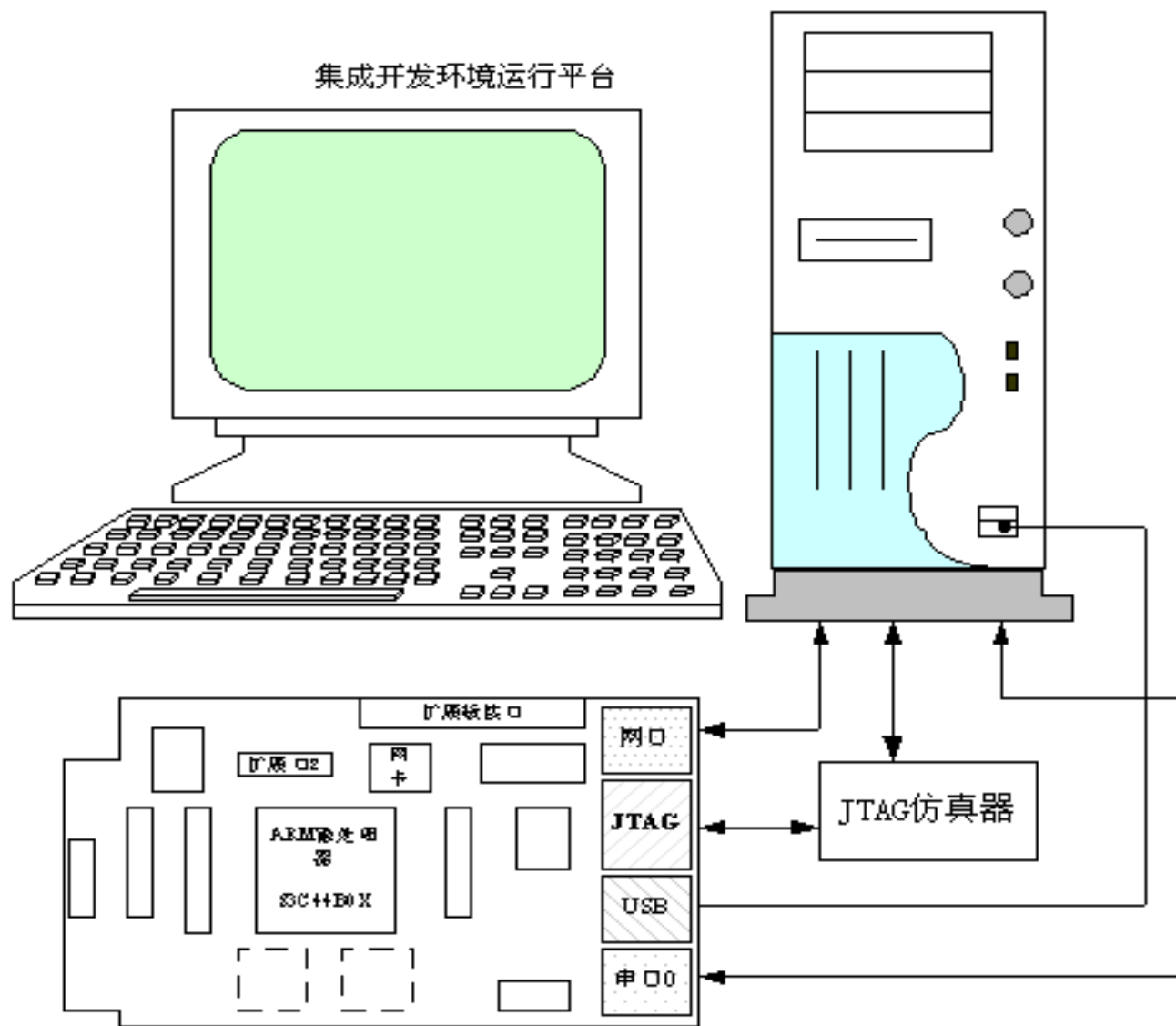
2 嵌入式Linux 交叉开发环境

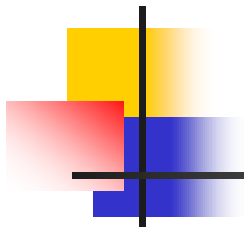


主要内容

- 
- 1 交叉开发环境介绍
 - 2 交叉编译器**cross-tool**
 - 3 配置主机开发环境
 - 4 建立交叉开发环境
 - 5 交叉调试应用程序
 - 6 **ARM**上的简单模块编程

交叉开发环境介绍





交叉开发环境介绍

n 为何生出了这个怪胎呢？

- n 嵌入式系统资源受限
- n 板子运行之前不能用于开发

n 解决方案

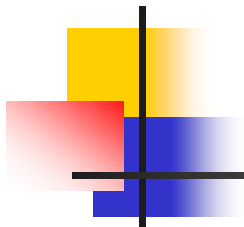
- n 在PC上编辑编译程序，生成二进制代码，下载至目标平台运行

n 何谓Cross Development

- n 交叉开发环境是指编译、链接和调试嵌入式应用软件的环境

n 组成

- n 宿主机Host，即开发机
- n 目标机Target，及开发板



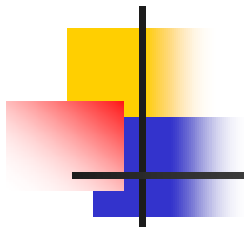
交叉开发环境介绍

n 通信工具

- n 串口，程序下载(速率较慢，适用于网口不能用的时候)、打印终端
- n 网口，程序下载及更新(速率较快)，信息反馈
- n JTAG，仿真器调试，程序烧录(裸板)
- n USB，启动、存储(大部分嵌入式设备不支持USB启动且板子上没有USB接口)

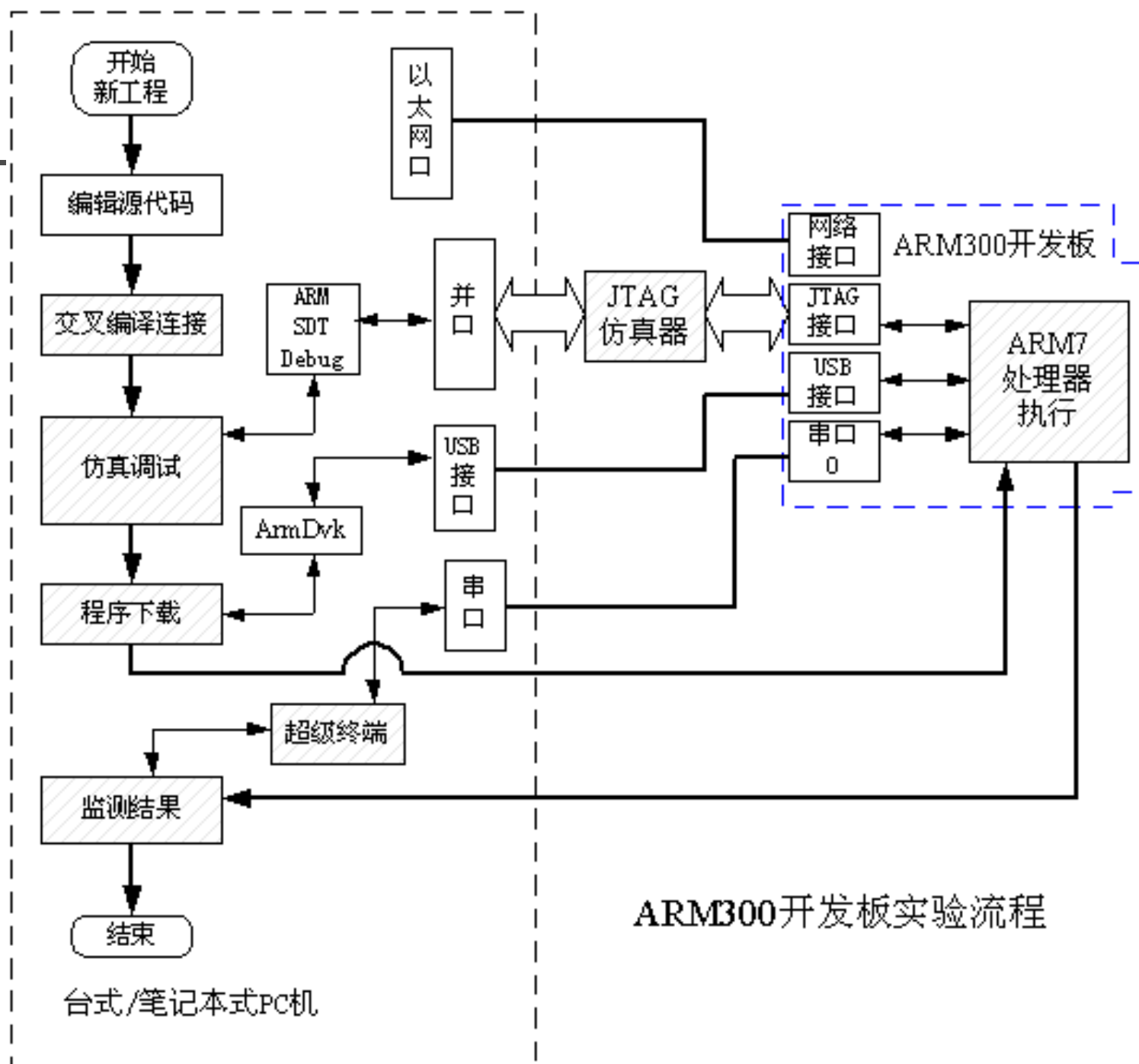
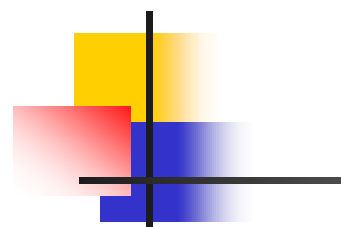
n 通信协议

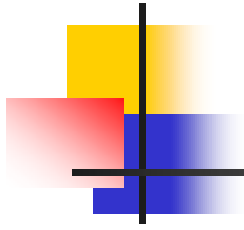
- n UART
- n GDB
- n JTAG
- n USB

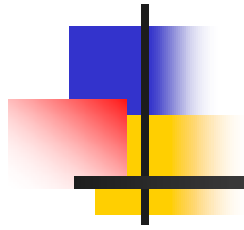


交叉开发环境介绍

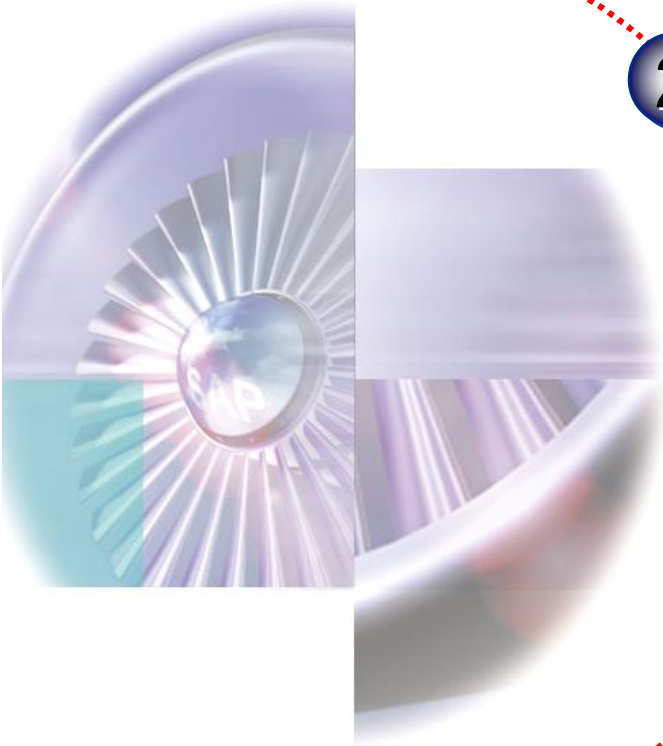
- n 典型环境及工具
 - n ARM ADS或者SDT
 - n MS E-VC或者Windows Mobile
 - n Windriver的Tornado, 用于VxWorks
 - n TI的CCS, 用于DSP开发
 - n Freescale 的Metrowerks CodeWarrior, 用于PowerPC
- n GCC用于Linux开发

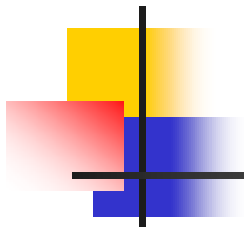






主要内容

- 
- 1 交叉开发环境介绍
 - 2 交叉编译器**cross-tool**
 - 3 配置主机开发环境
 - 4 建立交叉开发环境
 - 5 交叉调试应用程序
 - 6 **ARM**上的简单模块编程



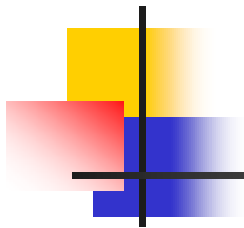
交叉编译器cross-tool

n ARM cross tool

- n 适用于ARM平台
- n 2.95.3, 2.4内核
- n 3.3.2, 2.6内核
- n 可自己通过相关工具编译生成, 也可以直接网上下载
- n <http://www.kegel.com/crosstool/>
- n <http://handhelds.org/download/projects/toolchain/>

n ELDK

- n Embedded Linux Development Kit
- n 适用于ARM、PowerPC等多种平台
- n <http://ftp.denx.de/pub/eldk/>



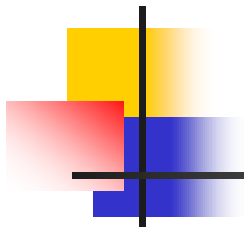
交叉编译器cross-tool

n 版本匹配问题

- n **kernel**、软件包及编译器三种的版本需要**match**
- n 编译器的版本并非越高越好
- n 根据各个包**Release**的时间可以初步判断对应关系
- n 参照软件包的相关说明

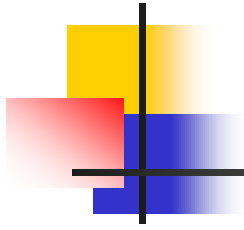
n 相关工具

- n arm-linux-gcc, 编译器
- n arm-linux-ld, 链接器
- n arm-linux-objcopy, 反汇编相关
- n arm-linux-strip, 剥离符号信息
- n arm-linux-objdump, 反汇编
- n arm-linux-ar, 打包成lib



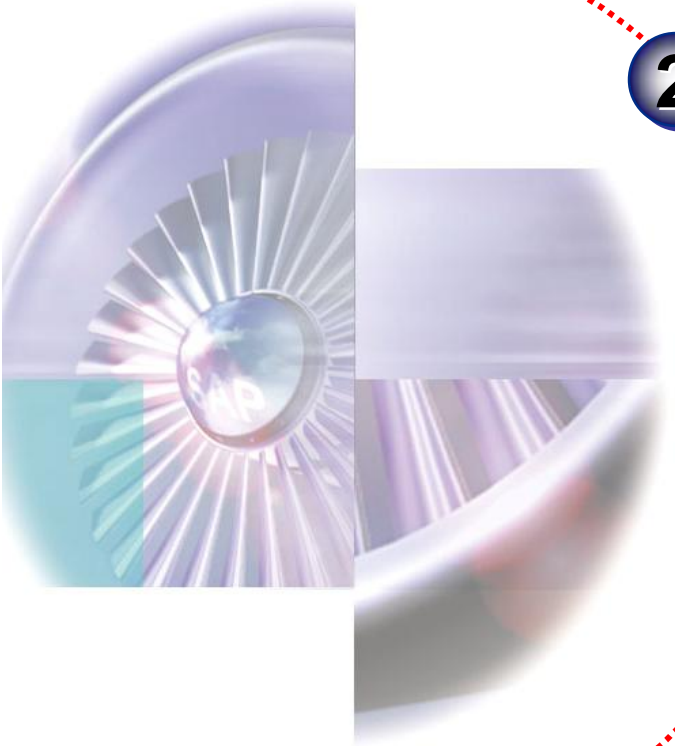
交叉编译器cross-tool

- n 如何设置生效
 - n 命令行里输入 `export`
`PATH=$PATH:/home/sea/linux_dev/cross_tool/usr/local/arm/3.4.1/bin`
 - n 也可在`~.bashrc`或`~.profile`或`/etc/profile`中设置此搜索路径，否则仅仅在当前终端下有效
- n 如何交叉编译
 - n `Make CC=arm-linux-gcc`
 - n `export CROSS_COMPILE=arm-linux-`
 - n `Make CROSS_COMPILE=arm-linux-`





主要内容

- 
- 1 交叉开发环境介绍
 - 2 交叉编译器**cross-tool**
 - 3 **配置主机开发环境**
 - 4 建立交叉开发环境
 - 5 交叉调试应用程序
 - 6 **ARM上的简单模块编程**



配置主机开发环境

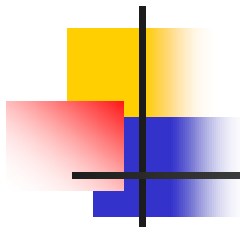
- n 串口
- n TFTP
- n NFS
- n Telnet
- n SSH
- n FTP

配置主机开发环境

n 串口

- n 用于打印调试，输出终端
- n Windows下Hyperterminal
DNW、SecureCRT、Putty
- n Linux下面的Minicom、
Kermit
- n **8N1**，无流控

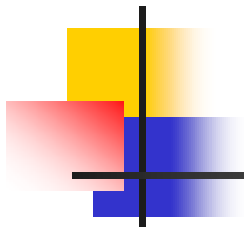




配置主机开发环境

n TFTP

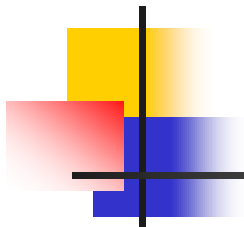
- n Trivial File Transfer Protocol, 中文名叫简单文件传输协议
- n 与FTP不同的是, 它使用的是**UDP**的**69**端口, 因此它可以穿越许多防火墙
- n 不过它也有缺点, 比如传送不可靠、没有密码验证等。只能从文件服务器上获得或写入文件, 不能列出目录。
- n 非常适合传送小型文件
- n 传输模式, **ASC11** 和 **Binary**(更安全可靠)
- n 512字节块, ACK确认
- n 主要用于**U-boot**下程序印象的下载更新



配置主机开发环境

n TFTP

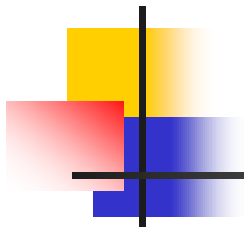
- n Windows下面有tftpd.exe，因为协议实现上的细微差异，大文件传输时会出现**socket recv error**
- n Linux上面配置tftp server性能较好，和U-boot match
- n 典型问题**Transfer timed out**，防火墙设置问题，打开**UDP 69**
- n `iptables -A OUTPUT -p UDP --dport 69 -j ACCEPT`
- n `iptables -A INPUT -p UDP --dport 69 -j ACCEPT`
- n `iptables -A OUTPUT -p UDP --dport 69 -j REJECT`
- n `iptables -A INPUT -p UDP --dport 69 -j REJECT`



配置主机开发环境

n NFS

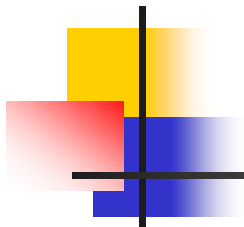
- n Network File System，用于Linux机器之间资源共享
- n 嵌入式系统中主要用于**NFS**网络根文件系统启动或者网络加载调试应用程序
- n 包括nfs-utilis及portmap等套件
- n 相关诊断工具包括exportfs、showmount -a/-e、rpcinfo，df
- n 需客户端和server端同时提供nfs支持
- n server端文件本身的权限及共享的权限
- n Port mapper failure，portmap未启用
- n **access denied by server**，防火墙或者host deny



配置主机开发环境

n NFS

- n not responding, timed out, 添加-o nolock参数即可
- n warning: mount version older than kernel
- n iptables -A INPUT -p TCP --dport 111 -j ACCEPT
- n iptables -A INPUT -p UDP --dport 111 -j ACCEPT
- n iptables -A INPUT -p UDP --dport 111 -j REJECT
- n iptables -A INPUT -p TCP --dport 111 -j REJECT



配置主机开发环境

n ssh

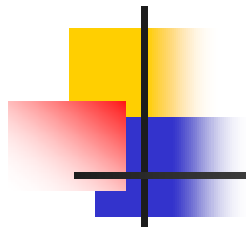
n Secure shell, 远程登录, 调试, 安全拷贝

n telnet

n Non secure shell, 远程登录, 调试

n ftp

n 主要用于将文件从目标板上拷贝到主机上



Volunteer task

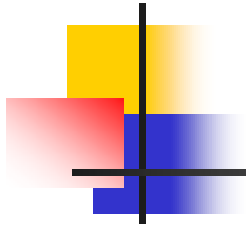
n Tftp server

- n 编译安装步骤
- n 典型问题分析
- n 主机端验证实例
- n 目标板上测试实例

n Nfs server及client搭建

- n 编译安装步骤
- n 典型问题分析
- n 主机端验证实例
- n 目标板上测试实例

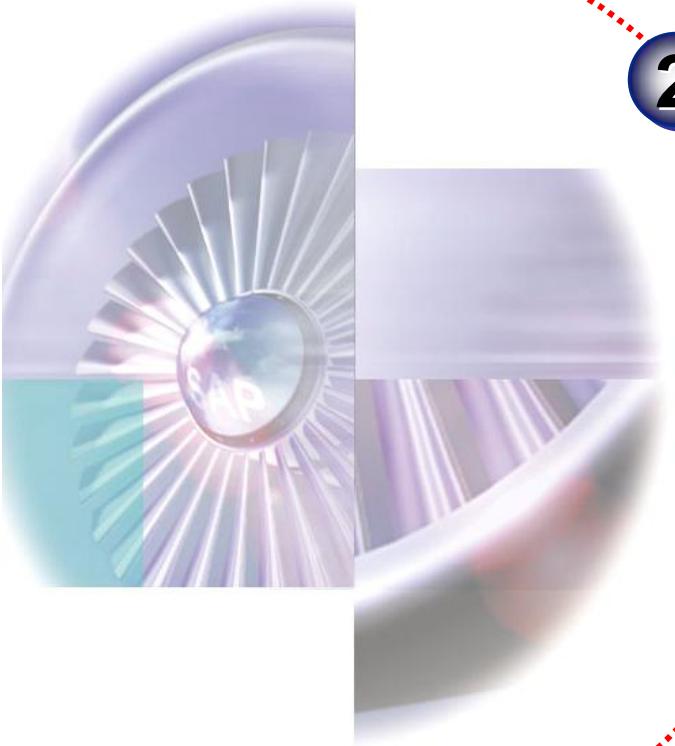
Self Learning, summary, share!



Q & A ?



主要内容

- 
- 1 交叉开发环境介绍
 - 2 交叉编译器**cross-tool**
 - 3 配置主机开发环境
 - 4 **建立交叉开发环境**
 - 5 交叉调试应用程序
 - 6 **ARM上的简单模块编程**

建立交叉开发环境

n 配置目标板上的NFS客户端

n 需要内核支持NFS，打开相应的编译选项

```
Linux Kernel v2.6.20 Configuration

Network File Systems

Arrow keys navigate the menu. <Enter> selects submenus --->. Highlighted letters are
hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press
<Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ] excluded
<M> module < > module capable

<*> NFS file system support
[*] Provide NFSv3 client support
[ ] Provide client support for the NFSv3 ACL protocol extension
[*] Provide NFSv4 client support (EXPERIMENTAL)
[ ] Allow direct I/O on NFS files
<*> NFS server support
[ ] Provide NFSv3 server support
[*] Provide NFS server over TCP support
[*] Root file system on NFS
--- Secure RPC: Kerberos V mechanism (EXPERIMENTAL)
< > Secure RPC: SPKM3 mechanism (EXPERIMENTAL)
< > SMB file system support (to mount Windows shares etc.)
< > CIFS support (advanced network filesystem for Samba, Window and other CIFS)
< > NCP file system support (to mount NetWare volumes)
< > Coda file system support (advanced network fs)
< > Andrew File System support (AFS) (EXPERIMENTAL)
< > Plan 9 Resource Sharing Support (9P2000) (Experimental)

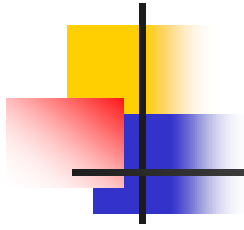
<Select> < Exit > < Help >
```



建立交叉开发环境

n Gdb和Gdb Server

- n arm-linux-gdb在主机上运行，提供交叉调试功能
- n arm-linux-gdbserver在板子上提供服务
- n 版本匹配问题，一定根据内核版本选取合适的Gdb源代码版本
- n 可借鉴主机端的软件环境和GDB的版本的匹配关系
- n gcc -v， gdb -v ， uname -a



建立交叉开发环境

n Gdb

n `./configure --target=arm-linux --
prefix=/home/sea/linux_dev/cross_tool/arm
-gdb`

n `make`

n `make install`

n `export`

`PATH=$PATH:/home/sea/linux_dev/cross_t
ool/usr/local/arm/3.4.1/bin/:/home/sea/linu
x_dev/cross_tool/arm-gdb/bin`



建立交叉开发环境

n Gdb Server

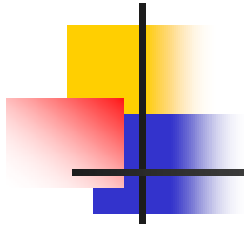
- n `/home/sea/linux_dev/gdb-6.6/gdb/gdbserver` 进入此目录
- n `./configure --target=arm-linux --host=arm-linux --prefix=/home/sea/linux_dev/cross_tool/arm-gdb`
- n `make CC=arm-linux-gcc`
- n `make install`
- n 将在项目及安装目录目录下生成arm-linux-gdbserver
- n 拷贝到板子上即可



建立交叉开发环境

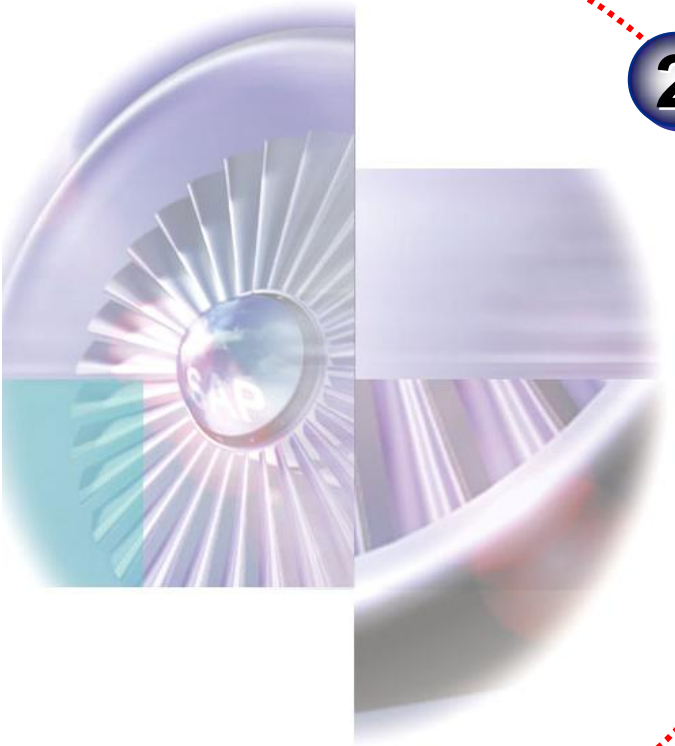
n Gdb Server

- n 编译时注意动态库的问题，板子上资源有限，最好选static编译成静态链CFLAGS += -static
- n arm-linux-strip arm-linux-gdbserver裁剪大小
- n error in loading shared libraries:
libthread_db.so.1
- n Ldd gdbserver查看动态库依赖关系





主要内容

- 
- 1 交叉开发环境介绍
 - 2 交叉编译器**cross-tool**
 - 3 配置主机开发环境
 - 4 建立交叉开发环境
 - 5 交叉调试应用程序
 - 6 **ARM上的简单模块编程**



交叉调试应用程序

n 试验环境

- n 板子通过nfs映射开发机目录，以便访问主机上的程序及源代码
- n arm-linux-gdbserver运行在开发板上，等待arm-linux-gdb的调试链接
- n arm-linux-gdb运行在开发主机上，连接目标板上的arm-linux-gdbserver
- n arm-linux-gdb必须在项目目录下启动，需要访问源代码

n 测试问题

- n 单步定位segmentation fault



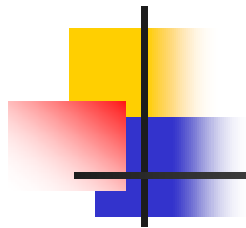
交叉调试应用程序

n 编译选项

- n 可执行目标文件
- n -g,-ggdb
- n 禁止和-O1/2/3、-Os一块使用

n 调试命令

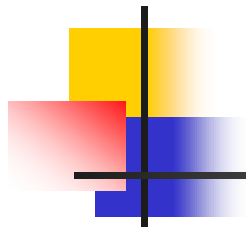
- n File, 载入elf文件
- n Run, 启动运行
- n List, 显示当前程序及函数或变量
- n Break, 设置断点
- n Delete b, 删除断点
- n Next, step, 单步运行
- n Continue, 断点停止后继续运行
- n Bt, backtrace, 显示调用栈
- n Print, 查看变量
- n Display, 动态监视变量



交叉调试应用程序

n 调试实例

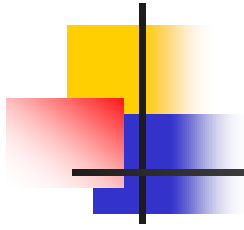
n 参见附件中的相关代码



Volunteer task

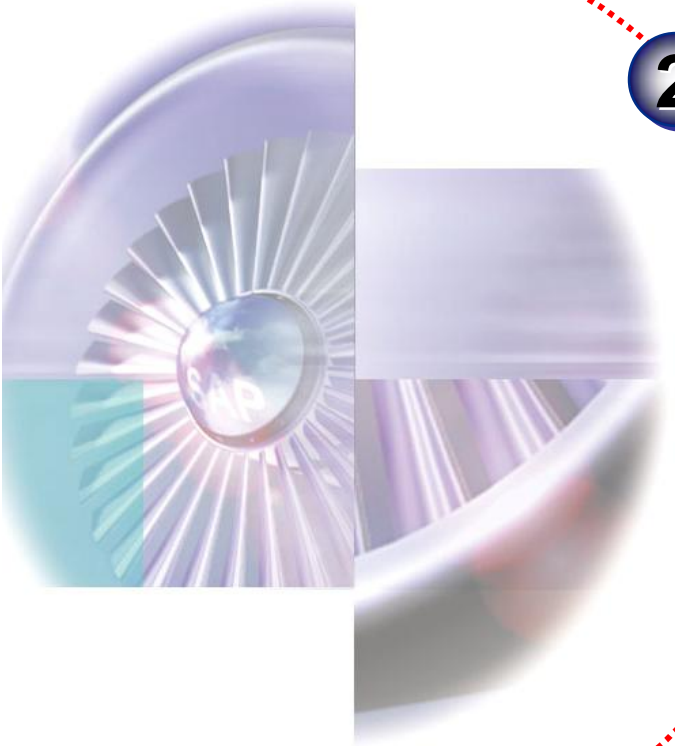
- n GDB和Gdb server交叉调试APP
- n 要求
 - n 编译安装步骤
 - n 典型问题
 - n 调试实例
 - n 用到大部分gdb调试命令

Self Learning, summary, share!





主要内容

- 
- 1 交叉开发环境介绍
 - 2 交叉编译器**cross-tool**
 - 3 配置主机开发环境
 - 4 建立交叉开发环境
 - 5 交叉调试应用程序
 - 6 **ARM上的简单模块编程**



ARM上的简单模块编程

n 何谓模块？

- n 动态改变内核的功能，提供更大的灵活性
- n 不与内核静态链接在一起，避免侵权及开源等问题
- n 需要时动态添加删除
- n 主要用于前期调试驱动模块

n 相关工具

- n insmod，将模块加载到内核中
- n Lsmod，查看当前已经加载的模块
- n rmmod，移除内核模块

n 前提条件

- n Arm-Linux内核源码树必须存在
- n <http://www.kernel.org/pub/linux/kernel/>



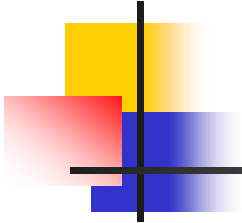
ARM上的简单模块编程

n 模块组成

- n 内核头文件
- n 模块加载及卸载的方法
- n 设备节点
- n File operation数据结构
- n Open, read, write及close当方法

n makefile

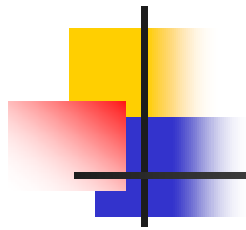
- n Stand alone
- n Part of kernel module



ARM上的简单模块编程

n 调试实例

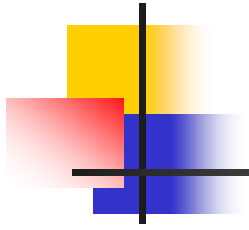
n 参见附件中的相关代码



Volunteer task

- n 在开发板上测试模块
- n 要求
 - n 给出模块源代码
 - n 给出makefile
 - n Nfs加载模块到板子上的调试实例

Self Learning, summary, share!



Q & A ?

