

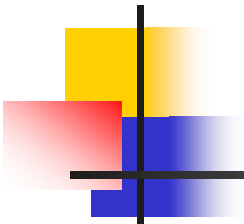
嵌入式Linux学习七步曲

Sailor_forever(扬帆)

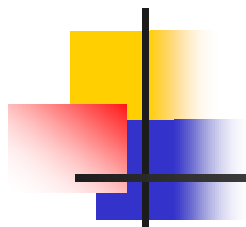
自由传播 版权所有 翻版必究



八一卦-我是who

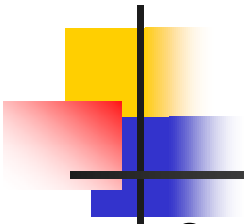
- 
-
- n 目前就职于通信行业某外企研发中心
 - n 参与校园招聘和社会招聘的技术面试工作
 - n 5年嵌入式软件开发经验，擅长嵌入式Linux开发；
 - n 接触的软硬件平台包括ARM，DSP，PowerPC，uC/OS-II，Linux，VxWorks及OSE

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

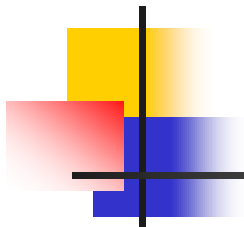
n 1年左右

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



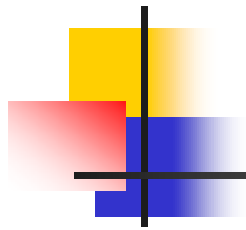
Volunteer Task

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

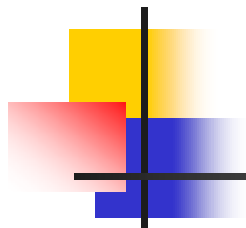
《嵌入式 Linux 学习七步曲》
BUPT BES 系列交流会 Volunteer Task



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

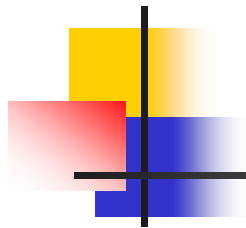
1

Linux主机开发环境



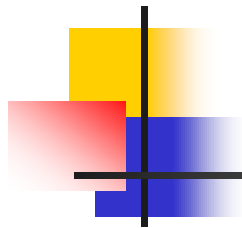
主要内容

- 
- 1 **Linux的安装**
 - 2 **Linux的基本命令及使用**
 - 3 **Linux与Windows的资源共享**
 - 4 **GCC开发工具**
 - 5 **Makefile编写**
 - 6 **简单应用程序**
 - 7 **GDB调试**
 - 8 **主机端的模块编程**



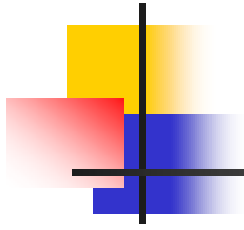
Linux的安装

- n 虚拟机下Linux的安装（优先）
 - n 商业的Vmware及开源的Virtual Box
 - n 安装后的系统是Windows下的一个文件，大小可自动变化，但只升不降，安装完后可拷贝，其他人免安装
 - n 无需独立分区
 - n 安全可靠
 - n 仍然可以使用Windows下的所有工具，切换方便
- n Windows下从硬盘安装Linux（可选）
 - n 无需安装盘，方便
 - n 需独立分区，可靠性差
- n 光盘直接安装（可选）
 - n 需安装盘，需要独立分区



Vmware虚拟机下Linux的安装

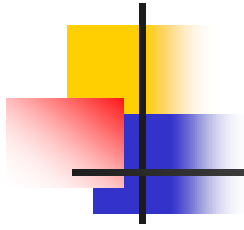
- n 先安装Vmware虚拟机
- n 获得安装光盘或者ISO镜像
 - n Fedora、Ubuntu、SuSe、Debian，个人推荐ubuntu，在软件安装管理方面较方便
- n 创建一个虚拟机，安装Linux
 - n 安装包要尽量选全了，默认方式不是每个包都装的，一定要安装上**TFTP、NFS、FTP、Samba、SSH、Telenet**及内核源码树
- n 设置虚拟机的网络连接方式NAT或者Bridge
- n 添加硬盘、串口
- n 安装Vmware tools



Volunteer task

- n Vmware虚拟机下Linux的安装
 - n 安装文件的下载地址，包括虚拟机及Linux
 - n 主要安装步骤的截图
 - n 网络配置及Vmware tools的配置
- n Windows下从硬盘安装Linux
- n 光盘直接安装

Self Learning, summary, share!





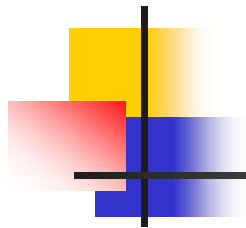
主要内容

- 
- 1 Linux的安装
 - 2 **Linux的基本命令及使用**
 - 3 Linux与Windows的资源共享
 - 4 GCC开发工具
 - 5 Makefile编写
 - 6 简单应用程序
 - 7 GDB调试
 - 8 主机端的模块编程



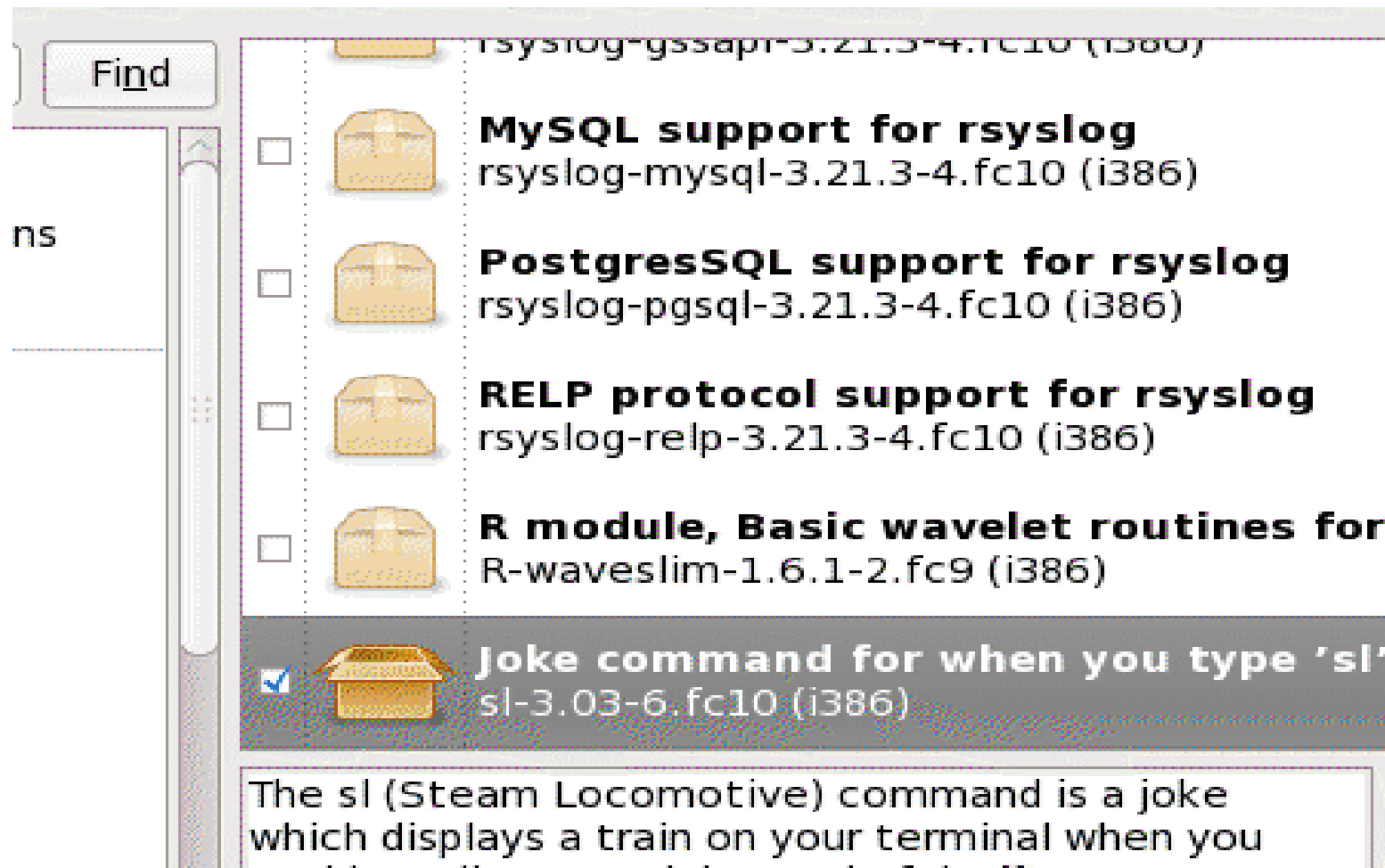
Windows 与Linux的口水战

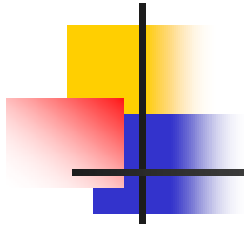
- n 图形界面 Vs 命令行
- n 费用
- n 开源，可重配
- n 支持的软件及软件安装
- n 开发环境及工具
- n 权限，组，域
- n 网络性能



Linux的魅力

- n Sl(Steam Locomotive), Joke command for when you type 'sl' instead of 'ls' on Fedora





程序员为什么喜欢Linux

n unzip, strip, touch, finger, grep, mount, fsck, more, yes, fsck, fsck, fsck, umount, sleep

（说明：unzip：拉开拉链；strip：脱掉衣服；touch：抚摸；finger：手指；grep：摸

索；

。

。。 sleep：睡觉）



Linux的基本命令及使用

n 终端

- n <ctrl>+<alt>+<shift>+<F1>
- n <ctrl>+<alt>+<F1>
- n <alt>+<F7>

n 命令的权限

- n /etc/profile所有用户的权限，用户登录时生效
- n ~/.bash_profile 当前用户登录时生效，仅一次
- n ~/.bashrc当前用户每次登陆虚拟终端时的配置文件
- n **PATH** 设置了可以使用的命令的路径

n Help

- n man cmd
- n info cmd
- n cmd -help

n 查看当前软硬件配置

- n uname -a



Linux的基本命令及使用

n 工作目录、用户主目录与路径

- n pwd, 查看当前工作目录
- n 绝对路径, 从根开始的目录, cd /tmp
- n 相对路径, 以当前工作目录为参考点, cd ../tmp, “..”代表上一级目录; cd ./named, 进入当前目录下的named, “.”代表当前目录
- n 用户主目录, 登陆shell后默认的当前工作目录, cd可以返回主目录, cd ~/named, 可以进入当前用户主目录下的named目录

n 文件的属性

- n 读写权限, rwx, 421, chmod (-R) xxx file/dir
- n 属主, chown (-R) owner:group file/dir
- n ls -al 查看当前目录下所有文件的属性
- n User, group, other



Linux的基本命令及使用

n 文件目录的创建删除及移动

- n mkdir, rmdir
- n cp -r循环拷贝, -dfr循环拷贝并保持属性不变
- n rm -rf强制循环删除
- n mv
- n touch filename, 创建文件

n 文件的压缩及解压缩

- n .tar, tar -cvf创建, -xvf解压缩
- n .gz, gzip压缩, gunzip解压缩
- n .tar.gz, tar -cvfz创建, -xvfz解压缩
- n .tar.bz2, tar -cvfj创建, -xvfj解压缩

n 查看文件

- n cat
- n more,tail,less,head

n 链接

- n Ln -s



Linux的基本命令及使用

n 磁盘管理

- n `df -h`, 显示当前挂载的文件系统及磁盘空间的使用情况
- n `du -h`, 查看文件的使用空间
- n `fdisk -l`, 列出当前的分区类型及编号

n 网络配置及诊断

- n `Ifconfig -a`列出当前的网络参数
- n `Ifconfig ethx xxxxx`, 可以用来配置网络接口的IP地址、掩码、网关、物理地址等, 但配置完的参数不能保存
- n `Ifconfig ethx up/down`
- n `service network restart`, `/etc/init.d/network restart`
- n 特定的工具或者修改配置文件可以保存网络参数
- n `Ping`, `traceroute`, `netstat -r`



Linux的基本命令及使用

n 关机重启

n Halt, shutdown -h now, poweroff, reboot

n 软件安装及卸载

n rpm -qa | grep xxx 查询某软件是否安装

n rpm -ivh xx.rpm 安装

n rpm -r xx 卸载

n <http://rpmfind.net/>

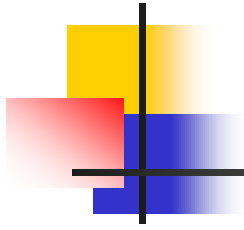
n <http://rpm.pbone.net/>

n 命令查找

n whereis, locate, which, find, grep

n 进程管理

n Ps, top, kill

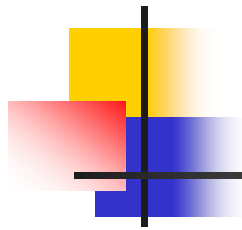


Q & A ?



主要内容

- 
- 1 Linux的安装
 - 2 Linux的基本命令及使用
 - 3 **Linux与Windows的资源共享**
 - 4 **GCC开发工具**
 - 5 **Makefile编写**
 - 6 简单应用程序
 - 7 **GDB调试**
 - 8 主机端的模块编程



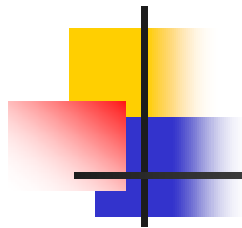
Linux与Windows的资源共享

n Samba

- n Windows与Linux的网络共享协议，适用于虚拟机和本机以及远程server与本机的共享
- n 对于同一个共享文件，针对不同的用户可设置不同的权限，可设置密码保护
- n 共享文件在Linux上，适合团队开发
- n 即时通步

n mount

- n 仅适用于fat32文件系统的文件
- n 需要独立的Windows分区
- n 共享文件在Windows上，适合于个人开发
- n 适用于虚拟机和本机以及独立的Linux主机与本机上硬盘的共享



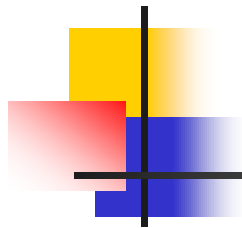
Linux与windows的资源共享

n 共享文件夹hgfs

- n 仅适用于虚拟机和本机，需安装Vmvare tools
- n 共享文件在Windows上，适合于个人开发
- n 虚拟机异常损坏时，共享文件不会丢失
- n 无权限控制和密码保护
- n 即时通步

n NFS

- n Linux主机之间的共享
- n 开发板和开发主机之间的共享



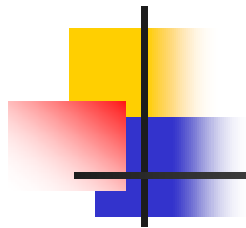
如何操作共享的Linux服务器

n 特点

- n 服务器可以是虚拟机，也可以是独立的Linux服务器
- n 在服务器上建立多个账户，多人共享，节省资源
- n 可以有独立的个人目录，安全可靠

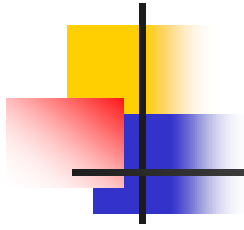
n 如何登录

- n 文本模式，ssh协议及telnet，多用户无干涉(windows下面的远程登录 多用户之间就会有干扰)，登录工具如putty，SecureCRT
- n 图形界面方式，vncserver及vncviewer，多用户无干涉



开发工具的组合模式

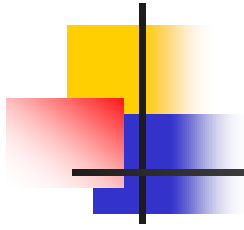
- n Samba/mount/hgfs + source insight
- n Vim/vi + Linux本机



Volunteer task

- n Samba
- n mount
- n Hgfs
- n 要求
 - n 明确Linux版本及相应的软件版本
 - n 详细步骤
 - n 图形界面的主要步骤要截图

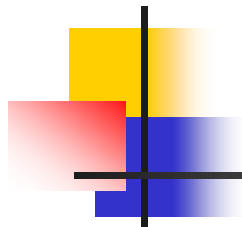
Self Learning, summary, share!





主要内容

- 
- 1 Linux的安装
 - 2 Linux的基本命令及使用
 - 3 Linux与Windows的资源共享
 - 4 **GCC**开发工具
 - 5 **Makefile**编写
 - 6 简单应用程序
 - 7 **GDB**调试
 - 8 主机端的模块编程



GCC开发工具

- n 无完善的图形化开发界面

- n gcc

- n -v 查看编译器版本

- n -c 只编译不链接

- n -o 链接，适用于单个c文件，无链接脚本

- n -g 编译时生成调试信息

- n -gdwarf-2 附带宏调试信息

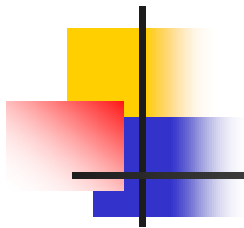
- n -fomit-frame-pointer

- n -o1/2/3 编译优化级别

- n -I 指定头文件目录

- n ld

- n 链接多个目标文件，可以使用链接脚本，需要makefile



GCC开发工具

n strip

- n 剥离符号信息，减小目标或者lib大小

n objdump

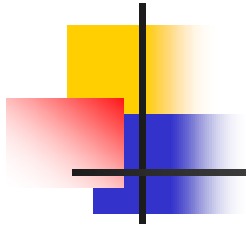
- n 反汇编代码，获取各个符号段的信息

n gdb

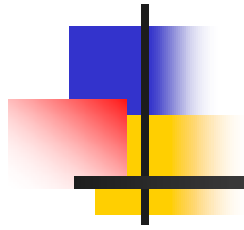
- n 命令行方式调试-g参数编译的elf可执行文件

n ddd

- n 图形界面方式

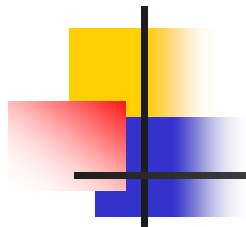


Q & A ?



主要内容

- 
- 1 Linux的安装
 - 2 Linux的基本命令及使用
 - 3 Linux与Windows的资源共享
 - 4 GCC开发工具
 - 5 **Makefile编写**
 - 6 简单应用程序
 - 7 GDB调试
 - 8 主机端的模块编程



Makefile-基础

n 功能特性

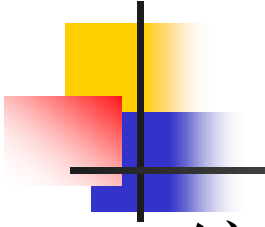
- n IDE集成开发环境下只需工程无需makefile
- n 用于层次化的管理工程代码，指定编译链接规则
- n 支持通配符、变量、**shell**命令，可配置，比工程具有更大的灵活性
- n 实现自动化编译
- n 会不会makefile是一个人是否具备开发大型项目的能力体现

n 组成

- n Target，生成的目标对象
- n Dependency，各个对象的层次及依赖关系
- n Phony，伪目标，辅助功能

n rule

- n 参见《跟我一起写makefile》



Makefile-编译链接

n 编译

- n 将源文件转换为中间代码
- n 条件是语法正确，变量及函数声明正确，函数及变量未声明则warning
- n .o, .obj

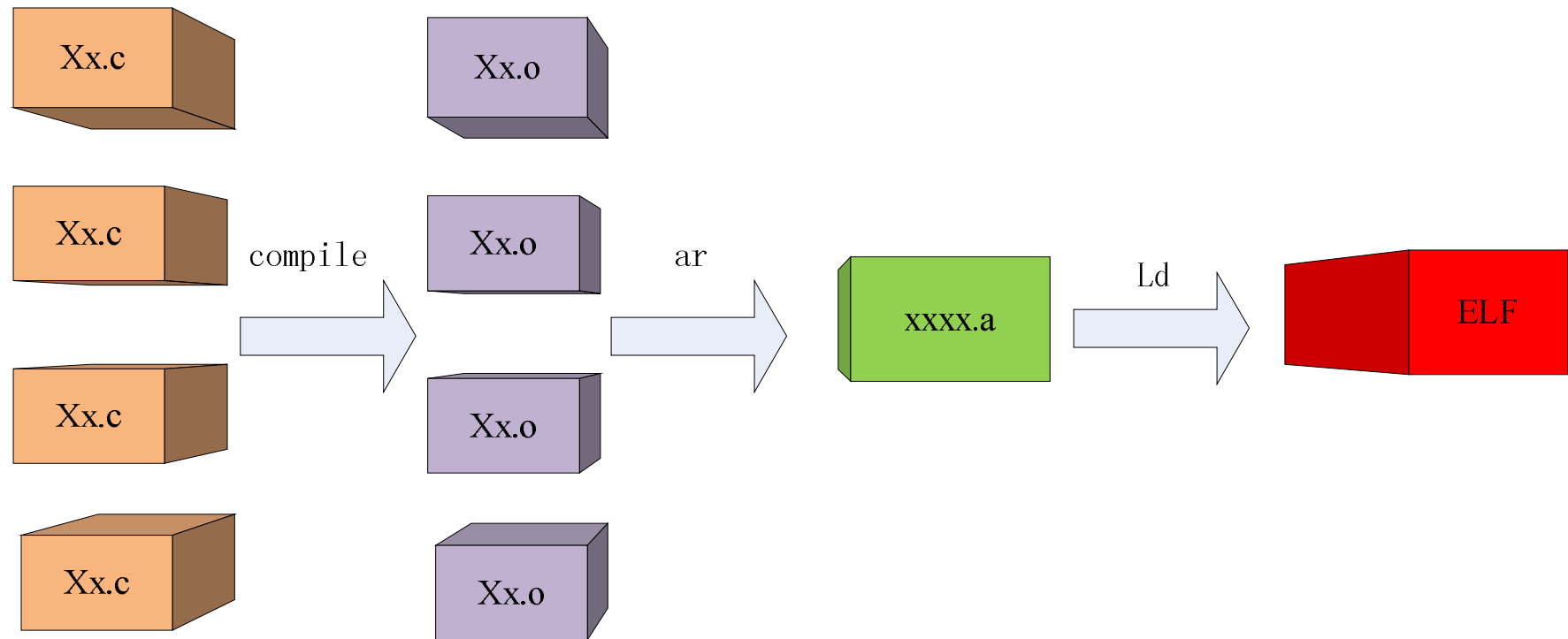
n 链接

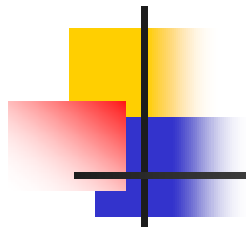
- n 链接函数及变量，寻找其实现及定义
- n 链接时需要明确知道中间目标文件
- n 将中间代码及库组合在一起，形成可执行文件

n 库

- n 避免过多的显示指定目标文件，将相关目标文件层层打包
- n .a, .lib

Makefile-编译链接

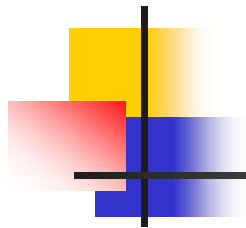




Makefile-规则

n 核心规则

- n 如果这个工程没有编译过，那么我们的所有C文件都要编译并被链接。
- n 如果这个工程的某几个C文件被修改，那么我们只编译被修改的C文件，并链接目标程序。
- n 如果这个工程的头文件被改变了，那么我们需要编译引用了这几个头文件的C文件，并链接目标程序。
- n **Make**命令自动根据文件修改情况决定编译的内容



Makefile-规则

n 基本语法

n target ... : prerequisites ...

n command

n ...

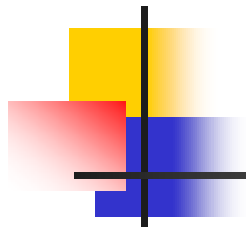
n ...

n 三个基本要素

n target也就是一个目标文件

n prerequisites就是，要生成那个target所需要的文件或是目标

n command也就是make需要执行的命令



Makefile-规则

n Make的动作

- n 读入所有的Makefile。
- n 读入被include的其它Makefile。
- n 初始化文件中的变量。
- n 推导隐晦规则，并分析所有规则。
- n 为所有的目标文件创建依赖关系链。
- n 根据依赖关系，决定哪些目标要重新生成。
- n 执行生成命令。



Makefile-简单示例

n 项目构成

n Main_test.c subfun.c

n 依赖关系

n Main_test.c需要subfun.h，即调用subfun.c提供的函数接口

n Makefile

n test_makefile : main_test.o subfun.o

n @echo "Now build for test_makefile"

n gcc -o test_makefile main_test.o subfun.o

n main_test.o : main_test.c subfun.h

n gcc -c main_test.c

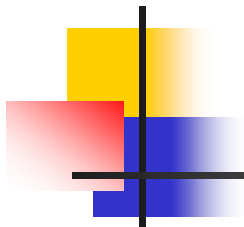
n subfun.o : subfun.c

n gcc -c subfun.c

n .PHONY : clean

n clean :

n -rm -f *.o test_makefile



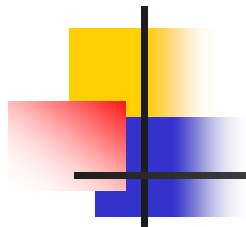
Makefile-改善

n 改善的对象

- n 可执行目标文件
- n 源文件列表
- n 头文件的依赖关系
- n 编译器类型
- n 编译器编译选项
- n 编译器链接选项

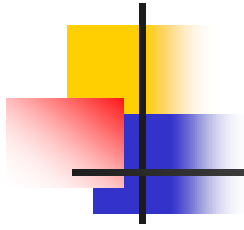
n 可移植性的makefile构成

- n 显示规则, 生成的目标、依赖关系及生成命令
- n 隐晦规则, 自动推导
- n 变量的定义
- n 文件指示, `include`, `#if`
- n 注释, `#`



Makefile-改善

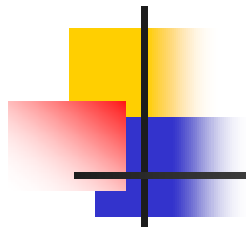
- n 优化的makefile实例
 - n 参见附件中的相关源代码





主要内容

- 
- 1 Linux的安装
 - 2 Linux的基本命令及使用
 - 3 Linux与Windows的文件共享
 - 4 GCC开发工具
 - 5 Makefile编写
 - 6 简单应用程序
 - 7 GDB调试
 - 8 主机端的模块编程



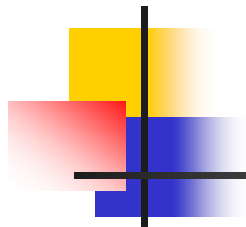
简单应用程序

n 注意事项

- n 头文件
- n 宏定义
- n 主函数
- n 文件及IO操作
- n 返回值判断
- n 错误处理
- n 异常退出，资源释放

n 参考资源

- n C编程的API, 《Linux C 函数参考》
- n Man, help



简单应用程序

n 程序实例

n 参见附件中的相关代码



主要内容

- 
- 1 Linux的安装
 - 2 Linux的基本命令及使用
 - 3 Linux与Windows的文件共享
 - 4 GCC开发工具
 - 5 Makefile编写
 - 6 简单应用程序
 - 7 **GDB调试**
 - 8 主机端的模块编程



GDB调试

- n 命令行调试工具

- 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

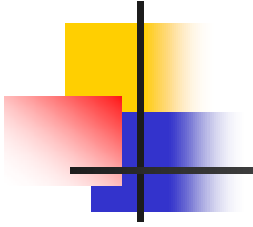
- n Next, step, 单步运行

- n Continue, 断点停止后继续运行

- n Bt, backtrace, 显示调用栈

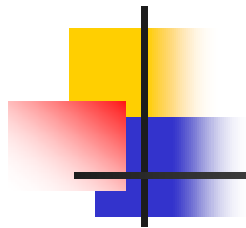
- n Print, 查看变量

- n Display, 动态监视变量



GDB调试-高级进阶

- n show print pretty
- n set print pretty on, 图形化的显示结构体
- n ddd, 图形界面调试
- n 自动生成Core dump文件
- n 崩溃时自动打印调用栈
- n 崩溃时自动调用GDB调试工具



Volunteer task

- n GDB调试命令
- n 要求
 - n 列举gdb调试的主要命令
 - n 缩写
 - n 详细用法
 - n 操作实例

Self Learning, summary, share!



主要内容

- 
- 1 Linux的安装
 - 2 Linux的基本命令及使用
 - 3 Linux与Windows的文件共享
 - 4 GCC开发工具
 - 5 Makefile编写
 - 6 简单应用程序
 - 7 GDB调试
 - 8 主机端的模块编程



主机端的模块编程

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
- n 需要时动态添加删除

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 源码树



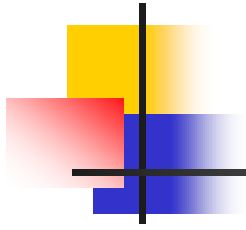
主机端的模块编程

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



Q & A ?

