

同济大学计算机系
计算机系统实验报告



实验二题目：pmon 引导文件编译和改造

学号：2152118

姓名：史君宝

指导教师：秦国峰

日期：2024 年 5 月 12 日

一、实验环境

操作系统：Windows11

软件系统：VMware Workstation 10.0.3

虚拟机：Ubuntu 20.04LTS

二、实验目的

本实验是后续实验的一个前置实验，通过对后续实验进行环境的配置与测试，以方便进一步的计算机系统结构实验的学些。

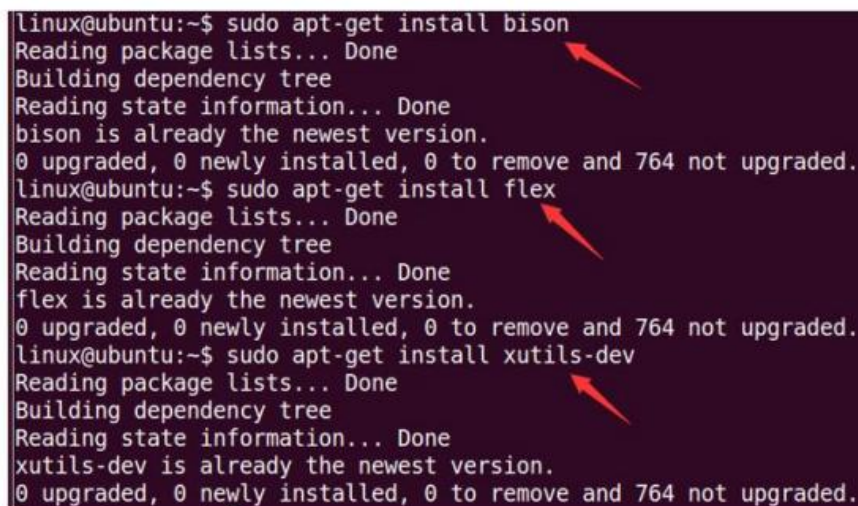
实验的主要内容如下：

- (1) 配置串口调试助手，并进行调试与通信。
- (2) 配置 tftp 服务。
- (3) 搭建 pmon 编译环境，进行 pmon 的源代码编译。
- (4) 生成 pmon 二进制文件 gzrom-dtb.bin。

三、实验过程及内容

- (1) 配置串口调试助手，并进行调试与通信。

我们按照实验教程，首先使用 minicom 来进行配置串口调试助手，在使用前我们需要对 minicom 进行如下配置：



```
linux@ubuntu:~$ sudo apt-get install bison
Reading package lists... Done
Building dependency tree
Reading state information... Done
bison is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 764 not upgraded.
linux@ubuntu:~$ sudo apt-get install flex
Reading package lists... Done
Building dependency tree
Reading state information... Done
flex is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 764 not upgraded.
linux@ubuntu:~$ sudo apt-get install xutils-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
xutils-dev is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 764 not upgraded.
```

我们对上述的内容进行一定的解释：

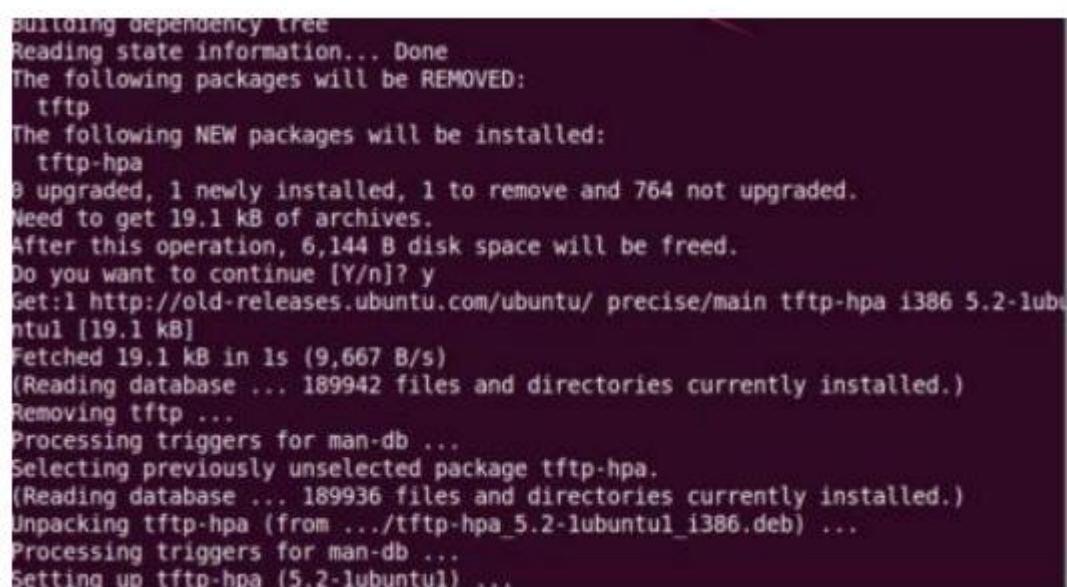
如上图所示，在上面的拷贝的镜像中，我们需要对串口设备和硬件流进行重

新设置。重新设置的具体内容为：首先我们设置串口设备为 `ttyUSB0`，并让波特率保持默认的 115200，并将硬件流改为关闭，软件流也保持默认关闭。

完成了上述的操作之后，我们可以选择“Save setup as df1”。将上述修改好的配置文件设置为默认配置文件，即完成了配置串口调试助手。

(2) 配置 tftp 服务。

在配置 tftp 服务的步骤中，我们首先需要安装 tftp 和 tftpd-hpa 服务。在安装完成之后，我们需要在 Vim 界面中对 tftpd-hpa 服务的配置文件进行进一步的修改。修改的重点内容是对服务端的最大权限 TFTP_OPTIONS 权限进行修改。之后我们需要创建新的服务端文件夹并将其设置为最高权限。最后我们需要重启 tftp 服务使其生效即可。



```
building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  tftp
The following NEW packages will be installed:
  tftp-hpa
0 upgraded, 1 newly installed, 1 to remove and 764 not upgraded.
Need to get 19.1 kB of archives.
After this operation, 6,144 B disk space will be freed.
Do you want to continue [Y/n]? y
Get:1 http://old-releases.ubuntu.com/ubuntu/ precise/main tftp-hpa i386 5.2-lubuntul [19.1 kB]
Fetched 19.1 kB in 1s (9,667 B/s)
(Reading database ... 189942 files and directories currently installed.)
Removing tftp ...
Processing triggers for man-db ...
Selecting previously unselected package tftp-hpa.
(Reading database ... 189936 files and directories currently installed.)
Unpacking tftp-hpa (from .../tftp-hpa_5.2-lubuntul_i386.deb) ...
Processing triggers for man-db ...
Setting up tftp-hpa (5.2-lubuntul) ...
```

(3) 测试 tftp 服务。

之后我们需要对之前配置的 tftp 服务进行测试，首先我们在 tftp 的服务目录下创建新的文件 `test.txt`，并在终端上输入执行 `tftp 192.168.1.249` 的命令。我们可以看到进入了 tftp 服务中，然后输入 `get test.txt`，发现可以正常查看文件，这说明测试完成，tftp 服务配置成功。

(4) 搭建 pmon 编译环境，准备进行 pmon 的源代码编译。

之后我们需要搭建 pmon 的编译环境，并准备进行 pmon 的源代码的编译。首先我们需要安装编译 pmon 所必备的相关工具，比如 bison、flex、xutils-dev 等等。安装好相关的工具之后，我们需要将所给的 pmon 源码和编译 pmon 的编译工具链拷贝到环境中。之后我们切换环境，切换到 pmoncfg 文件夹下，通过 ls 命令对 pmoncfg 文件夹中的内容进行检查，检查编译 pmon 所必备的相关工具之后将 和编译 pmon 的编译工具链拷贝到环境中，切换 bison、flex、xutils-dev 是否已经成功安装。安装完成之后，说明编译环境已经完成，可以准备进行编译工作。

```
linux@ubuntu:~$ sudo apt-get install bison
Reading package lists... Done
Building dependency tree
Reading state information... Done
bison is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 764 not upgraded.
linux@ubuntu:~$ sudo apt-get install flex
Reading package lists... Done
Building dependency tree
Reading state information... Done
flex is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 764 not upgraded.
linux@ubuntu:~$ sudo apt-get install xutils-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
xutils-dev is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 764 not upgraded.
```

```
linux@ubuntu:~/loongson/ls2k/pmon-loongson3/tools/pmoncfg$ ls
common.h  gram.o  main.o      mkioconf.o  pmoncfg      sem.h
config.h  gram.y  Makefile    mkmakefile.c  pmoncfg.8    sem.o
files.c   hash.c  mkheaders.c  mkmakefile.o  scan.l       util.c
files.o   hash.o  mkheaders.o  pack.c        scan.o       util.o
gram.h    main.c  mkioconf.c  pack.o        sem.c
```

(5) 进行 pmon 的源代码编译，生成 pmon 二进制文件 gzrom-dtb.bin。

我们首先需要进入 zloader.ls2k 目录，并使用 Vim 工具编写一个名为 cmd.sh 的编译脚本。在脚本中要确保在 DEBUG=-g 指令之前有空格，这非常重要性，能够避免意外错误导致编译失败。编写完 cmd.sh 的编译脚本后，我们需要执行该脚本用来编译 PMON 工具。尽管编译过程比较冗长，但在此省略了详细步骤。编译完成后，我们需要检查了文件夹中的文件，确认所需的二进制 gzrom-dtb.bin 文件已经成功生成，则表明编译成功。如果编译过程中出现错误，可以将 dtc 文件复制到 zloader.ls2k 目录下再次尝试。

```
linux@ubuntu:~/loongson/ls2k/pmon-loongson3/zloader$ vim cmd.sh
linux@ubuntu:~/loongson/ls2k/pmon-loongson3/zloader$ cat cmd.sh
#!/bin/bash

export PATH=/opt/gcc-4.4-gnu/bin/:$PATH
make cfg all tgt=rom CROSS_COMPILE=mipsel-linux- DEBUG=-g
make dtb
```

四、实验小结

在这次实验中，我探索了嵌入式系统中的 PMON，这是一个具有 BIOS 和 bootloader 功能的开源软件，专为基于龙芯架构的系统而设计。PMON 的改进提供了许多基础服务，如 BIOS 启动配置、内核加载、程序调试、内存寄存器显示、设置以及内存反汇编等功能。此外，PMON 还能够有效地加载操作系统或内核，并为内核提供良好的运行环境，对于龙芯系统的初始化和内核执行至关重要。

通过深入了解 PMON 的配置和编译，以及其他嵌入式系统开发工具，我在嵌入式系统领域的知识和技能得到了显著提升。这次实验不仅提升了我的技术水平，还增强了我解决复杂问题的能力。每次成功解决问题都让我信心倍增，激励我在未来的学习和工作中继续挑战更复杂的技术难题。

这次实验让我更深入地了解了嵌入式系统的运作原理，同时也让我意识到持续学习和探索对于不断提升自己的能力至关重要。我期待在未来的学习和工作中继续应用所学，探索更多领域的技术挑战，不断完善自己的技能和知识储备。这次实验为我打开了嵌入式系统领域的大门，让我对这个领域充满了兴趣和热情。