

《嵌入式系统》实验报告 7

学号： 21013134 姓名： 徐昊博 班级： 计 213 日期： 2024.6.14

成绩： _____ 指导教师： 罗飞

| | |
|--|-------|
| 实验名称： Openwrt 编译/嵌入式系统的烧写实验 | 实验地点： |
| 实验仪器： Linux 实验环境 | |
| <p>一、实验目的：</p> <ul style="list-style-type: none">1、了解 Openwrt 源代码的目录结构以及各目录的相关内容2、了解 Openwrt 各配置选项内容和作用3、掌握 Openwrt 配置文件 feeds 的作用4、掌握 Openwrt 的编译过程5、掌握 Openwrt 的烧录过程6、掌握嵌入式 Linux 镜像的烧写过程 | |
| <p>二、实验内容：</p> <p>一>镜像编译</p> <p>Openwrt 用到了 feeds，其作用是什么？如何组织？</p> <p>在 OpenWrt 中，feeds 是用于管理和集成额外软件包的机制。它们的主要作用包括：</p> <p>扩展包管理：允许用户访问和管理 OpenWrt 默认包之外的额外包，增加系统的功能和灵活性。</p> <p>简化包更新：通过更新 feeds，可以轻松地获取包的最新版本，确保系统和应用程序保持最新和安全。</p> <p>模块化开发：开发者可以独立于核心 OpenWrt 源代码进行包的开发和维护，这样可以更方便地进行更新和维护。</p> <p>Feeds 的组织</p> <p>Feeds 的组织通过一个配置文件来实现，该文件称为 feeds.conf。这个文件位于 OpenWrt 源代码的根目录中。以下是具体的组织步骤：</p> <p>配置 feeds.conf 文件：</p> <p>feeds.conf 文件中列出不同 feeds 的来源 URL。</p> <p>每个 URL 指向一个包含软件包的仓库，这些仓库可以是官方的，也可以是社区维护的。</p> <p>例如，官方的 packages 仓库和 luci 仓库。</p> <p>更新 feeds：</p> <p>更新 feeds 列表是为了确保系统可以获取到最新的包信息。</p> <p>更新过程中，系统会从 feeds.conf 文件中指定的 URL 获取最新的包信息。</p> <p>安装 feeds：</p> <p>在更新了 feeds 列表后，需要安装这些包到系统中，使其可用于构建和使用。</p> <p>安装过程会将指定的包添加到 OpenWrt 构建系统中，使其在配置和构建时可以选择。</p> <p>使用 feeds：</p> <p>在配置 OpenWrt 系统时，例如使用 make menuconfig 命令，用户可以选择并启用来自不同 feeds 的包。</p> <p>用户可以浏览和选择可用的包，并将其包括在固件构建中。</p> <p>这一步允许用户根据自己的需求定制化他们的固件，包括网络、系统工具、应用程序等各类包。</p> <p>通过这种组织方式，feeds 提供了一种灵活而高效的包管理方法，使得 OpenWrt 用户和开发者能够轻松扩展和更新系统功能。</p> <p>这样，不仅可以丰富 OpenWrt 的功能，还能方便地保持系统更新和安全。</p> <p>所用目标板为 mt7688，如何进行配置的？（通过配置来验证目标板是 mt7688）</p> | |

(1) 下载并准备 OpenWrt 源代码

首先，从 OpenWrt 官方仓库中获取最新的源代码。确保开发环境中已经安装了必要的构建工具和依赖项。

(2) 更新和安装 Feeds

在源代码根目录中，更新并安装 feeds，以获取最新的包和工具。

(3) 运行配置菜单

运行 `make menuconfig` 命令打开 OpenWrt 的配置界面。在这个界面中，可以选择目标板、包和其他配置选项。

(4) 配置目标板为 MT7688

在配置菜单中按以下步骤选择 MT7688 目标板：

选择 Target System:

进入 Target System 选项，选择 MediaTek Ralink MIPS。这个选项对应 MT7688 的架构。

选择 Subtarget:

在 Subtarget 选项中，选择 MT76x8 based boards。

选择 Target Profile:

进入 Target Profile 选项，选择与 MT7688 设备对应的配置。例如，如果设备是 LinkIt Smart 7688，选择相应的配置文件。

(5) 保存并退出

完成配置后，选择保存并退出配置菜单。配置会被保存到 `.config` 文件中。

(6) 验证配置

查看 `.config` 文件或运行以下命令来验证配置是否正确：

```
grep CONFIG_TARGET .config | grep "=y"
```

输出中应包含以下条目：

```
CONFIG_TARGET_ramips=y
```

```
CONFIG_TARGET_ramips_mt76x8=y
```

```
CONFIG_TARGET_ramips_mt76x8_DEVICE_mt7688=y
```

这些条目表明目标系统和子目标都已正确配置为 MT7688。

7. 编译固件

配置完成并验证后，运行以下命令开始编译固件：

```
make
```

编译完成后，固件映像将生成在 `bin/targets/ramips/mt76x8/` 目录下。

8. 烧录固件

使用适当的工具和方法将生成的固件烧录到 MT7688 设备上。例如，可以通过设备的 web 界面、TFTP 或串行控制台进行烧录。

特别地，通过 `make menuconfig`，说明以下配置别代表了什么，且其选项下的选项主要都有些什么作用？

Target System: 定义了目标设备的体系结构和平台类型。在这个选项中，你可以选择适当的硬件架构，例如 x86、ARM、MIPS 等。这个选项决定了构建系统的底层支持，包括编译器和工具链的配置。选择正确的目标系统是确保固件能够正确运行在目标设备上的第一步。

Subtarget: 用于进一步细分目标系统，定义特定的设备变体。对于某些硬件平台，可能存在多个变体或版本，在此可以选择具体的型号或板卡版本。这个选项决定了系统的特定驱动程序和硬件支持，以确保固件能够充分利用目标设备的硬件特性。

Target Profile: 用于选择具体的设备配置文件。通过选择一个预定义的设备配置文件，可以快速配置针对特定设备的设置。这个配置文件包含特定设备的初始化脚本、设备树文件 (DTB) 和其他硬件相关设置，使得配置过程更加简便和准确。

Target Images: 用于配置固件映像的生成和格式。在这个选项中，你可以配置生成的固件映像的格式，例

如 ext4、squashfs 等。同时，可以设置是否生成带有内核、根文件系统的组合镜像，或包含附加工具如 initramfs 镜像。这些设置决定了固件的最终格式和内容。

Kernel Modules: 用于选择要包含在内核中的模块。通过这个选项，可以配置内核支持的额外功能和驱动程序，例如网络驱动、文件系统驱动、USB 驱动等。选择需要的内核模块，以便在设备启动时自动加载，确保设备具备所需的功能和硬件支持。

Languages: 用于选择和配置系统支持的编程语言和相关运行时。在这个选项中，你可以选择要包含在系统中的编程语言运行时，如 Python、Perl、Lua 等。同时，可以配置语言相关的库和工具，以支持应用程序开发和运行，确保系统能够满足特定开发需求。

Luci: 用于配置 OpenWrt 的 Web 管理界面。通过这个选项，可以选择和安装 LuCI 相关的组件和应用程序，配置 Web 界面的主题、翻译和附加功能插件。确保设备可以通过 Web 浏览器进行管理和配置，为用户提供直观、方便的管理界面。

Network: 用于配置网络相关的软件包和服务。在这个选项中，可以选择和配置各种网络协议支持，如 DHCP、DNS、VPN、WiFi 等。同时，可以安装和配置网络工具和服务，如防火墙、流量监控、网络共享等，确保设备具备全面的网络功能。

Sound: 用于配置音频支持和相关驱动。通过这个选项，可以选择音频驱动程序和相关的内核模块，安装和配置音频应用程序和工具如 alsa-utils，确保设备能够支持音频输入和输出功能，满足音频处理需求。

二>嵌入式 Linux 镜像烧写（可选）

结合嵌入式硬件实验（虚拟仿真），说明：

1. 如何烧写 Bootloader?

连接编程器：将编程器连接到目标板的 Flash 存储器接口。

配置编程软件：使用编程软件加载 Bootloader 文件。

烧写 Bootloader：

使用编程软件将 Bootloader 文件写入 Flash 存储器。

2. 如何烧写内核镜像?

（1）加载内核镜像文件

假设内核镜像文件名为 **zImage**，并且存放在当前目录中。

（2）烧写内核镜像

在 Telnet 会话中输入以下命令：

复位并停止目标板：

reset halt

检测 Flash 存储器：

对于 NAND Flash：

nand probe 0

擦除 Flash 存储器：

对于 NAND Flash：

nand erase 0 0 0

烧写内核镜像文件：

对于 NAND Flash：

nand write 0 zImage 0x80000

重启目标板：

reset run

对于 NOR Flash 存储器，可以使用如下命令：

检测 NOR Flash：

flash probe 0

擦除 NOR Flash：

```
flash erase_sector 0 0 last
```

烧写内核镜像文件：

```
flash write_image erase zImage 0x80000
```

重启目标板：

```
reset run
```

三>OpenWrt 烧录镜像 （可选）

编译好的 OpenWrt 镜像烧录有几种方式？如何配置？

方式一：SCP 方式烧写

SCP 拷贝镜像文件：

将编译好的镜像文件通过 SCP 传输到目标板的 /tmp/ 目录下。

SSH 登录目标板：

使用 SSH 登录到目标板，确保能够通过命令行访问目标设备。

使用 sysupgrade 命令更新目标板：

在目标板上使用 sysupgrade 命令更新固件。此命令将会用新的固件文件替换当前系统并重启设备。

方式二：Luci 方式烧写

连接设备：

将网线插入开发板的 LAN 口与电脑的网口，或连接到同一局域网下的路由器网口。

访问 Luci 界面：

在浏览器中输入开发板的默认 IP 地址访问 Luci 界面，使用默认的用户名和密码登录。

刷写固件：

在 Luci 界面的系统菜单中，选择备份与升级选项。

选择需要刷写的固件文件进行升级。该过程会自动完成固件替换和设备重启。

方式三：串口方式烧写

连接开发板与电脑：

使用配套的数据线将开发板与电脑连接，并在电脑端安装必要的串口驱动。

安装并配置 SecureCRT：

安装 SecureCRT 软件，打开并进行连接设置，将串口端口号和波特率配置为与开发板一致的设置。

启动开发板并进入 U-Boot：

给开发板上电，当 U-Boot 打印信息并开始计时时，通过键盘选择进入 U-Boot 的 TFTP 模式。

配置 TFTP：

在电脑端配置 TFTP 服务器，选择固件所在目录，并确保防火墙设置允许 TFTP 传输。

传输固件：

在 SecureCRT 中输入固件文件名，通过 TFTP 传输固件到开发板。

确保电脑和开发板在同一网段，并关闭防火墙以允许文件传输。

登录并查看启动状态：

烧写完成并重启开发板后，将开发板与 PC 网口或路由器网口连接。

根据编译时设置的 IP 地址访问开发板，确认成功启动。

访问 IP 地址查看 Luci 管理界面，设置 root 密码后通过 SSH 进行访问。