

## 任务二：开发板 BIOS 服务调用

### 1. 介绍

本任务需要实现一个开发板 BIOS 服务调用

### 2. Start code

#### 2.1 文件介绍

- `bootblock.s`: 程序开始运行的程序，需要通过调用 `pmon` 提供的字符输出方式输出一个字符串
- `createimage`: 生成的内核镜像的 Linux 工具，用来将 `bootblock` 生成一个 `image` 文件
- `Makefile`: 编译配置文件
- `ld.script`: 链接器脚本文件

#### 2.2 运行

`make` 命令编译文件

`make clean` 对编译产生的文件进行清除

`sudo dd if=image of=/dev/sdb` 将产生的 `image` 写进 SD 卡中( `/dev/sdb` 为插入的 SD 卡盘符，根据需要修改 )

在 `minicom` 中执行 `loadboot` 运行程序

可以采用反汇编进行调试，调试命令为 `mipsel-linux-objdump -d`

### 3. 任务要求

在 `PMON` 环境下输出一个字符串

#### 3.1 开发要求

在 `bootblock.s` 文件中实现代码输出一个字符串

#### 3.2 注意事项

`PMON` 下的字符输出为串口输出，地址为 `0xbfe48000`，输出字符串可以通过不断向该地址写字符实现。由于串口输出比较慢，输出字符可能会被立即覆盖，需要加入延迟（可以采用空循环方式）。

`Loadboot` 命令的功能是将 `sd` 卡中读取 `image` 到 `0xa0800000` 地址处，然后从 `0xa0800030` 处开始执行，所以需要适当在 `bootblock.s` 中加入 `nop` 指令使得程序从要运行的第一条指令开始运行。如 `start code` 中加入了 12 条 `nop` 指令，每条 `nop` 指令为 4 个字节，共增加了 48 字节的空间，即 `0x30`。

[illegible]