



哈爾濱工業大學 (深圳)
HARBIN INSTITUTE OF TECHNOLOGY

实验报告

开课学期: 2022 春季
课程名称: 计算机网络
实验名称: 协议栈之 Eth、ARP 协议实现
学生班级: 1 班
学生学号: 190110115
学生姓名: 谢俊安
评阅教师:
报告成绩:

实验与创新实践教育中心制

2022 年 3 月

一、实验详细设计

1. Eth 协议详细设计

a) ethernet_out()

- i. 判断数据包的长度，如果不足 46 则显式填充 0
 1. 根据帧格式，46 往后是负载。
 2. 用 buf_add_padding 填充 0，直到够 46 为止。
- ii. 调用 buf_add_header()函数添加以太网包头
- iii. 填写目的 MAC 地址
 1. 利用 memcpy 函数
- iv. 填写源 MAC 地址（本机地址）
 1. 利用 memcpy 函数
 2. net_if_mac 是本机地址
- v. 填写协议类型 protocol，注意大小端的转换。
- vi. 调用驱动层封装好的 driver_send()发送函数，将添加了以太网包头的的数据帧发送到驱动层。

b) ethernet_in()

- i. 判断数据长度，如果数据长度小于以太网头部长度（14），则认为数据包不完整，丢弃不处理。
- ii. 用本函数传入的数据包，初始化以太网头部。
- iii. 调用 buf_remove_header()函数移除以太网包头
 1. 注意第 2 步第 3 步的顺序，不可逆转，否则包头的的数据会丢失，造成数据错乱。
- iv. 对以太网包头的协议进行大小端的转换，再作为向上层传递的协议号。
- v. 调用 net_in()函数向上层传递数据包。

2. ARP 协议详细设计

阅读并理解实验原理

a) arp_req()

- i. 对网卡缓存区 txbuf 进行初始化；
- ii. 利用 arp_init_pkt 初始化填写 ARP 报头；
- iii. ARP 操作类型 opcode16 为 ARP_REQUEST，注意大小端转换；
- iv. 填写 target_ip；
- v. 目标 MAC 地址填写为广播地址（全 F）；
- vi. 利用 ethernet_out 将 ARP 报文以 ARP 协议的类型发送出去；

b) arp_out()

调用 map_get()函数，根据 IP 地址来查找 ARP 表(arp_table)：

1. 如果找到对应 MAC 地址，则将数据包直接发送给以太网层；
2. 如果没有找到对应的 MAC 地址：
 - a) 如果 arp_buf 已经有包了，则不作为；
 - b) 如果 arp_buf 没有包
 - i. 调用 map_set()函数将来自 IP 层的数据包缓存到 arp_buf
 - ii. 调用 arp_req()函数，发一个请求目标 IP 地址对应的 MAC 地址的 ARP request 报文。

c) arp_in()

- i. 判断数据长度，如果数据长度小于 ARP 报头长度（8 字节），则认为数据包不完整，丢弃不处理。
- ii. 做报头检查，查看报文是否完整。
 1. 与 arp_init_pkt 初始化的内容做对比，相等则完整
 2. 特别地，操作类型 opcode16 应该是 ARP_REQUEST 或者 ARP_REPLY
- iii. 调用 map_set()函数，利用（传入数据包 buf 中 arp 报头部分携带的发送包协议地址 sender_ip，函数的传入参数源 mac 地址 src_mac）更新 ARP 表项
- iv. 调用 map_get()函数查看该接收报文的 IP 地址(sender_ip)是否有对应的 arp_buf 缓存
 1. 如果有，则证明是这个待发送的数据包是：上一次调用 arp_out()发送来自 IP 层的数据包的时候，由于没找到对应的 MAC 地址而发送的 ARP request 报文的应答报文。
 - a) 利用 ethernet_out()将该待发送数据包发送给以太网层
 - i. sender_ip 的包对应的 MAC 地址就是 sender_mac
 - ii. 因为是 IP 协议才会出现此情况，所以传入的协议类型也是 NET_PROTOCOL_IP
 - b) 调用 map_delete()函数将这个缓存的数据包（arp buffer）删除掉
 2. 如果没有：
 - a) 判断接收到的报文是否为 ARP_REQUEST 请求报文（操作类型）
 - b) 并且判断该请求报文的 target_ip 是否为本机的 IP（net_if_ip）
 - c) 如果都是，则认为是请求本主机 MAC 地址的 ARP 请求报文。
 - i. 调用 arp_resp()函数回应一个响应报文
 - ii. 该响应报文的目标 IP/MAC 地址应该是发送方的 IP/MAC 地址，即 sender_ip，因为该报文的功能是“回应”。

d) arp_resp()

- i. 调用 buf_init()来初始化 txbuf
- ii. 利用 arp_init_pkt 初始化填写 ARP 报头
- iii. 填写 arp 报文的 opcode16、target_ip、target_mac。
- iv. ethernet_out()函数将填充好的 ARP 报文发送出去
 1. 此时对应的应该是 ARP 协议，所以是协议类型填写的是 NET_PROTOCOL_ARP

二、实验结果截图及分析

1. Eth 协议实验结果及分析

```
1/1 Test #1: eth_in ..... Passed    0.03 sec
```

```
The following tests passed:
```

```
    eth in
```

```
100% tests passed, 0 tests failed out of 1
```

```
Total Test time (real) =   0.04 sec
```

```
1/1 Test #2: eth_out ..... Passed    0.04 sec
```

```
The following tests passed:
```

```
    eth out
```

```
100% tests passed, 0 tests failed out of 1
```

```
Total Test time (real) =   0.05 sec
```

所有结果和 demo 一致，测试通过。

2. ARP 协议实验结果及分析

```
1/1 Test #3: arp_test ..... Passed    0.06 sec
```

```
The following tests passed:
```

```
    arp_test
```

```
100% tests passed, 0 tests failed out of 1
```

```
Total Test time (real) =   0.51 sec
```

所有结果和 demo 一致，测试通过。