

# 浙江大学

## 本科实验报告

课程名称： 计算机网络基础

实验名称： 使用模拟软件组建互联网络

姓 名： 蒋仕彪

学 院： 计算机学院

系： 计算机系

专 业： 计算机科学与技术

学 号： 3170102587

指导教师： 董玮

2020 年 4 月 9 日

# 浙江大学实验报告

实验名称： 使用模拟软件组建互连网络 实验类型： 设计实验

同组学生： 无 实验地点： 家中

## 一、 实验目的：

- 学习掌握 GNS3 模拟软件的使用法
- 学习掌握交换机、路由器的配置方法
- 学习掌握 VLAN 的工作原理，以及如何配置 VLAN
- 学习掌握 IP 路由的工作原理，以及如何设置静态路由表

## 二、 实验内容

- GNS3 是一款具有图形化界面可以运行在多平台（包括 Windows, Linux, and MacOS 等）的网络虚拟软件。
- 分别采用以下方式组建网络，测试连通性，产生模拟数据包，观察网络数据包流向
  - ✓ 使用 HUB、无线 AP 和 PC 机搭建局域网，
  - ✓ 使用单个交换机和 PC 机搭建局域网并配置 VLAN，观察网络数据包流向
  - ✓ 使用多个交换机和建局域网并配置 VLAN 中继，观察网络数据包流向
  - ✓ 使用多个路由器连接多个局 PC 机搭域网，并配置静态路由

## 三、 主要仪器设备

- 联网的 PC 机
- GNS3 模拟软件

## 四、 操作方法与实验步骤

- 安装 GNS3 模拟软件

### Part 1. 组网

- 使用 1 个 HUB 和 5 个 PC 机搭建第 1 个局域网，并使用子网地址 10.1.0.0/8
- 使用 1 个 HUB 和 3 个 PC 机搭建第 2 个局域网，并使用子网地址 10.2.0.0/8
- 使用 1 个交换机和 3 个 PC 机搭建第 3 个局域网，并使用子网地址 10.3.0.0/8
- 使用 1 个交换机和 3 个 PC 机搭建第 4 个局域网，并使用子网地址 10.4.0.0/8
- 使用第 5 个交换机，将 4 个局域网连接起来
- 使用 Ping 命令查看各个网络之间的连通性
- 修改第 1、2 局域网的子网掩码为 16 位，再次查看各个网络之间的连通性
- 修改第 3、4 局域网的子网掩码为 16 位，再次查看各个网络之间的连通性

## Part 2. VLAN

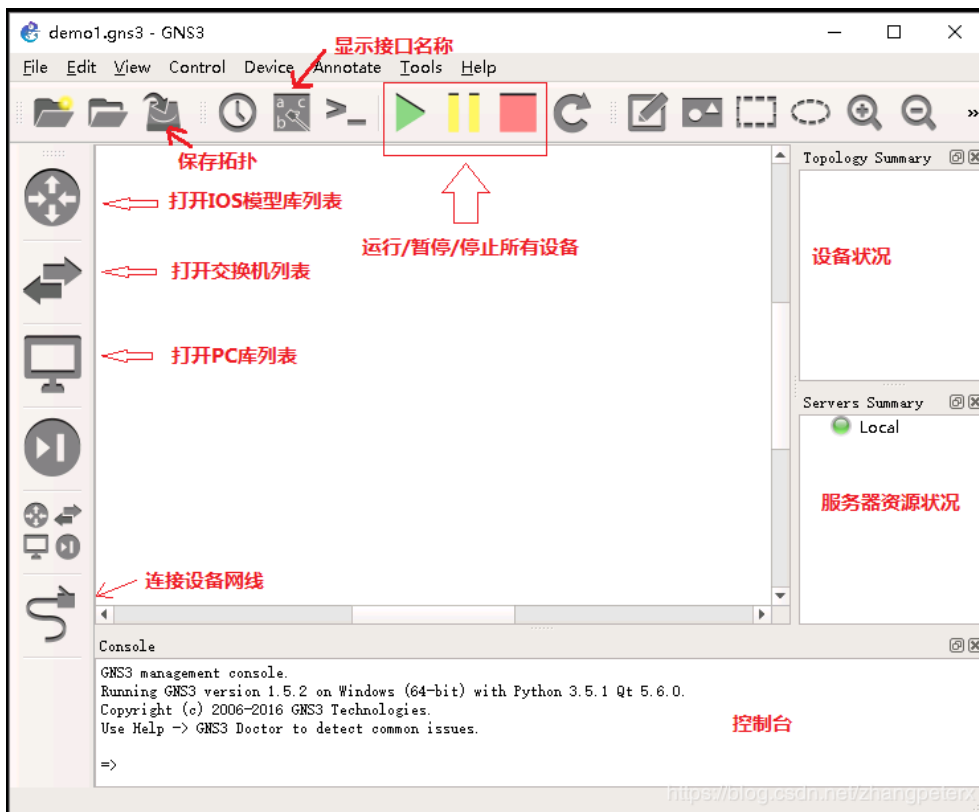
- 将 HUB 换成交换机，并在 4 个局域网交换机上划分出 2 个 VLAN，让 PC 机属于不同 VLAN
- 使用 Ping 命令查看各个网络的联通性
- 修改 4 个局域网的子网地址，给不同组的 VLAN 分配不同的子网地址，再次查看各个网络之间的联通性
- 在第 5 个交换机（互联交换机）上设置 VLAN，使不同局域网内某个 VLAN 组的 PC 之间能够互通
- 在第 5 个交换机（互联交换机）上启用 VLAN Trunk，使不同局域网内相同 VLAN 组的 PC 之间都能够互通

## Part 3. 路由

- 将第 5 个交换机删除，每个局域网分别设立一个路由器(使用 c3745 路由器镜像)
- 给各个路由器创建 2 个子接口，并分配合适的 IP 地址，使得同一局域网内，不同 VLAN 的 PC 之间能够互通
- 使用第 5 台路由器分别连接 4 个局域网的路由器
- 启用动态路由协议 RIP，使得不同子网的 PC 之间能够互通
- 关闭动态路由协议 RIP，给各个路由器设置正确的静态路由，使得不同子网的 PC 之间能够互通

## 五、 实验数据记录和处理

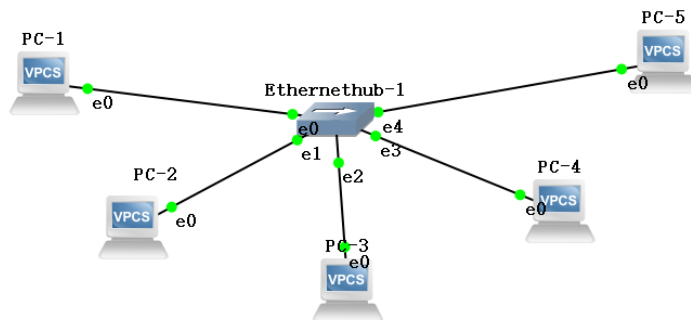
- 运行 GNS3 模拟软件，界面上由哪些部分组成，分别有什么作用？



## -----Part 1. 组网-----

以下在控制台设置每个 PC 的 ip（以及后续其他配置）时需要注意及时 save，否则停止节点（stop nodes）或退出 GNS3 后 ip 配置便会失效。

- 使用 1 个 HUB 和 5 个 PC 机搭建第 1 个局域网，并使用子网地址 10.1.0.0/8 给每个 PC 配置 IP 地址。使用另 1 个 HUB 和 3 个 PC 机搭建第 2 个局域网，并使用子网地址 10.2.0.0/8 给每个 PC 配置 IP 地址。在每个局域网中的其中一台 PC 上使用 Ping 命令 ping 同一局域网内的另外一台 PC，查看主机之间的连通性。



```
PC-1> ip 10.1.0.0/8
Checking for duplicate address...
PC1 : 10.1.0.0 255.0.0.0
PC-1>
```

```
PC-6> ip 10.2.0.0/8
Checking for duplicate address...
PC1 : 10.2.0.0 255.0.0.0
PC-6>
```

```
PC-2> ping 10.1.0.0
84 bytes from 10.1.0.0 icmp_seq=1 ttl=64 time=0.104 ms
84 bytes from 10.1.0.0 icmp_seq=2 ttl=64 time=0.193 ms
84 bytes from 10.1.0.0 icmp_seq=3 ttl=64 time=0.148 ms
84 bytes from 10.1.0.0 icmp_seq=4 ttl=64 time=0.137 ms
84 bytes from 10.1.0.0 icmp_seq=5 ttl=64 time=0.158 ms
```

在局域网 A 和 B 分别配置地址为 10.1.0.0 和 10.2.0.0 且掩码是 8 位的 ip。内部 PC 之间的 ping 通信（我截图的是 PC2 给 PC1）能成功。

以下使用的交换机为二层交换机，如 GNS3 中的 Ethernet switch。

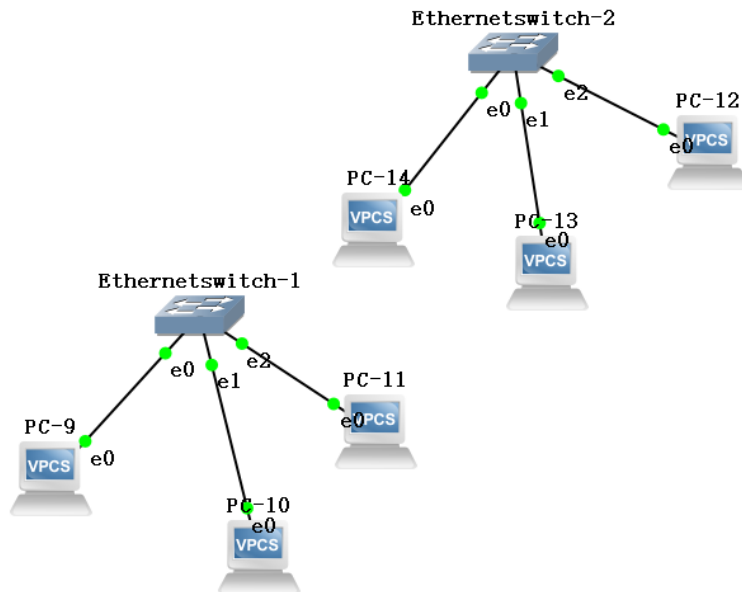
- 使用 1 个交换机和 3 个 PC 机搭建第 3 个局域网，并使用子网地址 10.3.0.0/8 给每个 PC 配置 IP 地址。使用另 1 个交换机和 3 个 PC 机搭建第 4 个局域网，并使用子网地址 10.4.0.0/8 给每个 PC 配置 IP 地址。在每个局域网中的其中一台 PC 上使用 Ping 命令 ping 同一局域网内的另外一台 PC，查看主机之间的联通性。

```
PC-9> ip 10.3.0.0/8
Checking for duplicate address...
PC1 : 10.3.0.0 255.0.0.0

PC-9> save
Saving startup configuration to startup.vpc
. done
```

```
PC-13> ip 10.4.0.1 8
Checking for duplicate address...
PC1 : 10.4.0.1 255.0.0.0

PC-13> save
Saving startup configuration to startup.vpc
. done
```



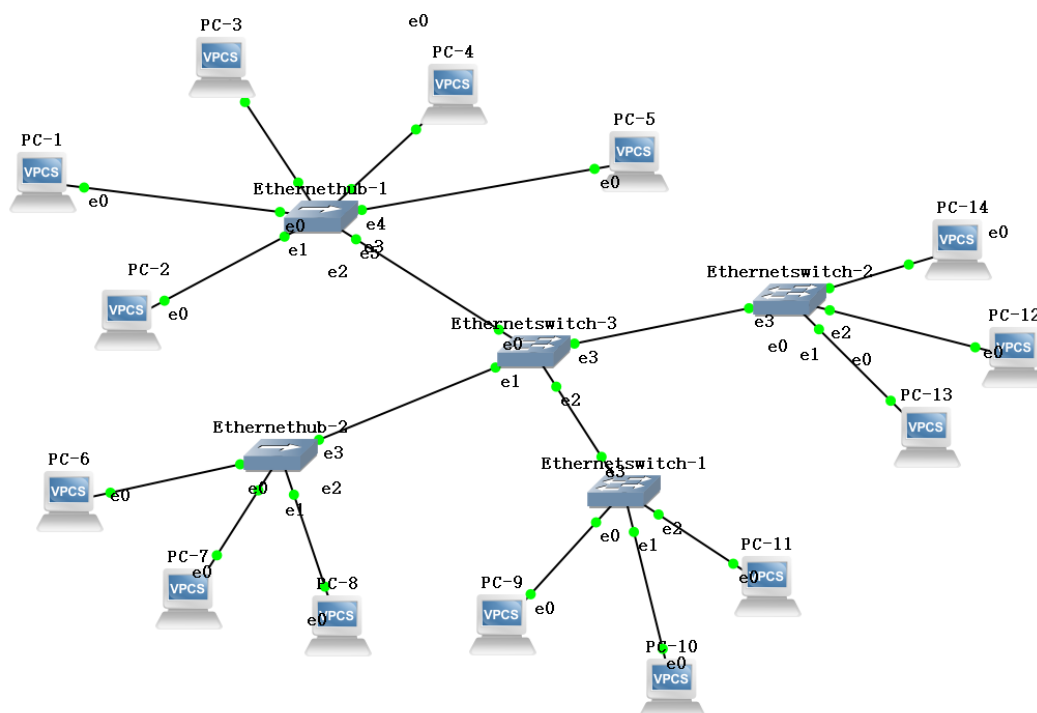
```
PC-9> ping 10.3.0.1
84 bytes from 10.3.0.1 icmp_seq=1 ttl=64 time=0.148 ms
84 bytes from 10.3.0.1 icmp_seq=2 ttl=64 time=0.098 ms
84 bytes from 10.3.0.1 icmp_seq=3 ttl=64 time=0.092 ms
84 bytes from 10.3.0.1 icmp_seq=4 ttl=64 time=0.100 ms
84 bytes from 10.3.0.1 icmp_seq=5 ttl=64 time=0.136 ms

PC-9> ping 10.4.0.0
host (10.4.0.0) not reachable
```

尝试在第三个局域网内部 ping（PC-10 给 PC-9）成功。

尝试从第三个局域网 ping 到第四个局域网（PC-10 给 PC-12）失败。

- 使用第 5 个交换机，将 4 个局域网连接起来。使用 Ping 命令检查各个局域网 PC 之间的联通性（每个局域网选取 2 台 PC 做代表）。是否都能 Ping 通？如果不通，请检查原因。



```
PC-9> ping 10.1.0.4
84 bytes from 10.1.0.4 icmp_seq=1 ttl=64 time=0.158 ms
84 bytes from 10.1.0.4 icmp_seq=2 ttl=64 time=0.206 ms
84 bytes from 10.1.0.4 icmp_seq=3 ttl=64 time=0.198 ms
84 bytes from 10.1.0.4 icmp_seq=4 ttl=64 time=0.199 ms
84 bytes from 10.1.0.4 icmp_seq=5 ttl=64 time=0.299 ms

PC-9> ping 10.2.0.2
84 bytes from 10.2.0.2 icmp_seq=1 ttl=64 time=0.118 ms
84 bytes from 10.2.0.2 icmp_seq=2 ttl=64 time=1.011 ms
84 bytes from 10.2.0.2 icmp_seq=3 ttl=64 time=0.453 ms
84 bytes from 10.2.0.2 icmp_seq=4 ttl=64 time=0.175 ms
84 bytes from 10.2.0.2 icmp_seq=5 ttl=64 time=0.913 ms

PC-9> ping 10.3.0.1
84 bytes from 10.3.0.1 icmp_seq=1 ttl=64 time=0.061 ms
84 bytes from 10.3.0.1 icmp_seq=2 ttl=64 time=0.082 ms
84 bytes from 10.3.0.1 icmp_seq=3 ttl=64 time=0.094 ms
84 bytes from 10.3.0.1 icmp_seq=4 ttl=64 time=0.088 ms
84 bytes from 10.3.0.1 icmp_seq=5 ttl=64 time=0.098 ms

PC-9> ping 10.4.0.2
84 bytes from 10.4.0.2 icmp_seq=1 ttl=64 time=0.092 ms
84 bytes from 10.4.0.2 icmp_seq=2 ttl=64 time=0.111 ms
84 bytes from 10.4.0.2 icmp_seq=3 ttl=64 time=0.106 ms
84 bytes from 10.4.0.2 icmp_seq=4 ttl=64 time=0.143 ms
84 bytes from 10.4.0.2 icmp_seq=5 ttl=64 time=0.116 ms
```

```
PC-1> ping 10.1.0.2
84 bytes from 10.1.0.2 icmp_seq=1 ttl=64 time=0.096 ms
84 bytes from 10.1.0.2 icmp_seq=2 ttl=64 time=0.165 ms
84 bytes from 10.1.0.2 icmp_seq=3 ttl=64 time=0.146 ms
84 bytes from 10.1.0.2 icmp_seq=4 ttl=64 time=0.151 ms
84 bytes from 10.1.0.2 icmp_seq=5 ttl=64 time=0.143 ms

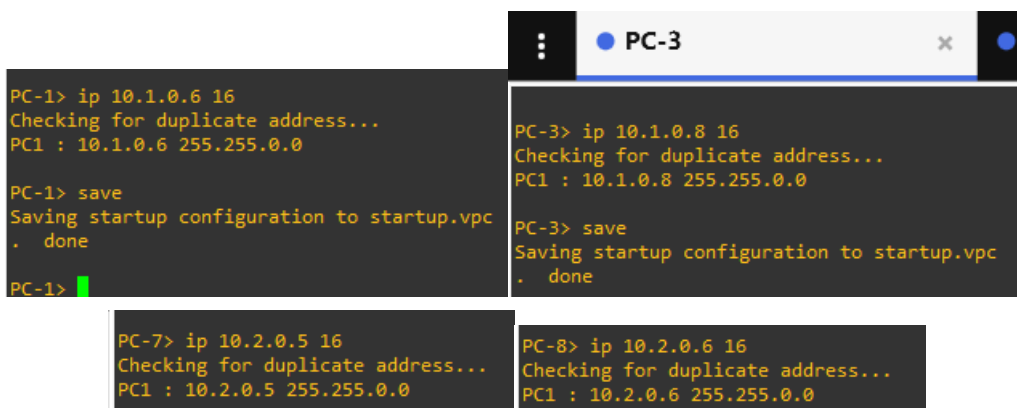
PC-1> ping 10.2.0.1
84 bytes from 10.2.0.1 icmp_seq=1 ttl=64 time=0.257 ms
84 bytes from 10.2.0.1 icmp_seq=2 ttl=64 time=0.214 ms
84 bytes from 10.2.0.1 icmp_seq=3 ttl=64 time=0.216 ms
84 bytes from 10.2.0.1 icmp_seq=4 ttl=64 time=0.204 ms
84 bytes from 10.2.0.1 icmp_seq=5 ttl=64 time=0.217 ms

PC-1> ping 10.3.0.2
84 bytes from 10.3.0.2 icmp_seq=1 ttl=64 time=0.112 ms
84 bytes from 10.3.0.2 icmp_seq=2 ttl=64 time=0.170 ms
84 bytes from 10.3.0.2 icmp_seq=3 ttl=64 time=0.172 ms
84 bytes from 10.3.0.2 icmp_seq=4 ttl=64 time=0.199 ms
84 bytes from 10.3.0.2 icmp_seq=5 ttl=64 time=0.190 ms

PC-1> ping 10.4.0.1
84 bytes from 10.4.0.1 icmp_seq=1 ttl=64 time=0.129 ms
84 bytes from 10.4.0.1 icmp_seq=2 ttl=64 time=0.169 ms
84 bytes from 10.4.0.1 icmp_seq=3 ttl=64 time=0.216 ms
84 bytes from 10.4.0.1 icmp_seq=4 ttl=64 time=0.181 ms
84 bytes from 10.4.0.1 icmp_seq=5 ttl=64 time=0.223 ms
```

以局域网 3 的 PC-9 和局域网 1 的 PC-1 分别举例，往四个局域网的 PC 机器分别 ping，均能够 ping 成功。最开始 ping 的时候曾失败了一次，发现是 ip 地址设错了。

- 把第 1、2 局域网中所有 PC 机的子网掩码从 8 位改成 16 位，再次用 Ping 检查各个 PC 之间的联通性（每个局域网选取 2 台 PC 做代表）。哪些通？哪些不通？原因是什么？



```

PC-1> ip 10.1.0.6 16
Checking for duplicate address...
PC1 : 10.1.0.6 255.255.0.0

PC-1> save
Saving startup configuration to startup.vpc
. done

PC-1>

PC-3> ip 10.1.0.8 16
Checking for duplicate address...
PC1 : 10.1.0.8 255.255.0.0

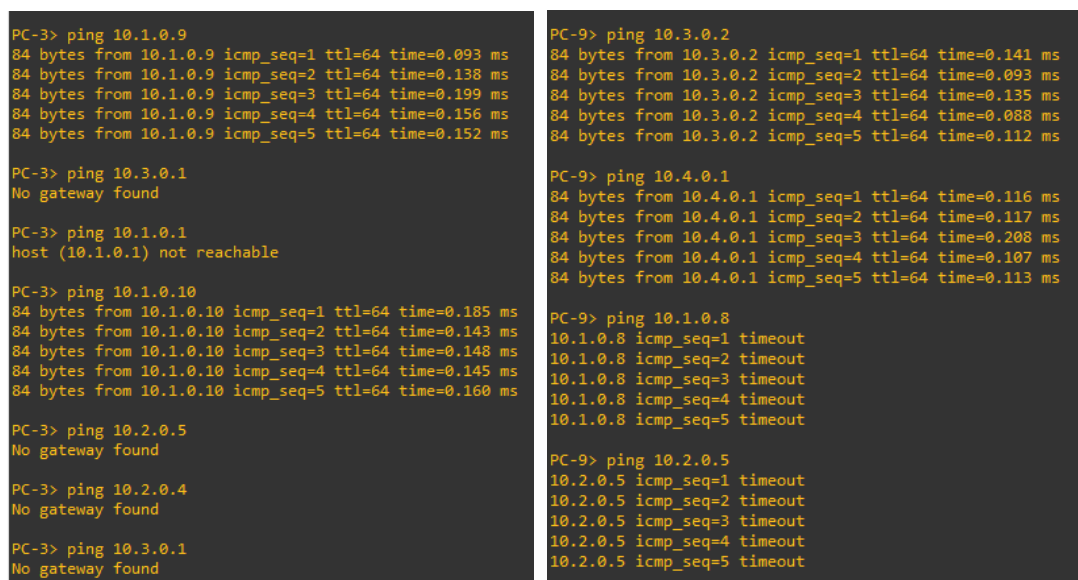
PC-3> save
Saving startup configuration to startup.vpc
. done

PC-7> ip 10.2.0.5 16
Checking for duplicate address...
PC1 : 10.2.0.5 255.255.0.0

PC-8> ip 10.2.0.6 16
Checking for duplicate address...
PC1 : 10.2.0.6 255.255.0.0

```

为了确保 ip 地址修改无误，我不光把掩码改成了 16 位，之前的 ip 地址也重新编号（在原来的基础上继续增大，即原来 PC1~PC5 是 1~5 标，现在是 6~10 标）。



```

PC-3> ping 10.1.0.9
84 bytes from 10.1.0.9 icmp_seq=1 ttl=64 time=0.093 ms
84 bytes from 10.1.0.9 icmp_seq=2 ttl=64 time=0.138 ms
84 bytes from 10.1.0.9 icmp_seq=3 ttl=64 time=0.199 ms
84 bytes from 10.1.0.9 icmp_seq=4 ttl=64 time=0.156 ms
84 bytes from 10.1.0.9 icmp_seq=5 ttl=64 time=0.152 ms

PC-3> ping 10.3.0.1
No gateway found

PC-3> ping 10.1.0.1
host (10.1.0.1) not reachable

PC-3> ping 10.1.0.10
84 bytes from 10.1.0.10 icmp_seq=1 ttl=64 time=0.185 ms
84 bytes from 10.1.0.10 icmp_seq=2 ttl=64 time=0.143 ms
84 bytes from 10.1.0.10 icmp_seq=3 ttl=64 time=0.148 ms
84 bytes from 10.1.0.10 icmp_seq=4 ttl=64 time=0.145 ms
84 bytes from 10.1.0.10 icmp_seq=5 ttl=64 time=0.160 ms

PC-3> ping 10.2.0.5
No gateway found

PC-3> ping 10.2.0.4
No gateway found

PC-3> ping 10.3.0.1
No gateway found

PC-9> ping 10.3.0.2
84 bytes from 10.3.0.2 icmp_seq=1 ttl=64 time=0.141 ms
84 bytes from 10.3.0.2 icmp_seq=2 ttl=64 time=0.093 ms
84 bytes from 10.3.0.2 icmp_seq=3 ttl=64 time=0.135 ms
84 bytes from 10.3.0.2 icmp_seq=4 ttl=64 time=0.088 ms
84 bytes from 10.3.0.2 icmp_seq=5 ttl=64 time=0.112 ms

PC-9> ping 10.4.0.1
84 bytes from 10.4.0.1 icmp_seq=1 ttl=64 time=0.116 ms
84 bytes from 10.4.0.1 icmp_seq=2 ttl=64 time=0.117 ms
84 bytes from 10.4.0.1 icmp_seq=3 ttl=64 time=0.208 ms
84 bytes from 10.4.0.1 icmp_seq=4 ttl=64 time=0.107 ms
84 bytes from 10.4.0.1 icmp_seq=5 ttl=64 time=0.113 ms

PC-9> ping 10.1.0.8
10.1.0.8 icmp_seq=1 timeout
10.1.0.8 icmp_seq=2 timeout
10.1.0.8 icmp_seq=3 timeout
10.1.0.8 icmp_seq=4 timeout
10.1.0.8 icmp_seq=5 timeout

PC-9> ping 10.2.0.5
10.2.0.5 icmp_seq=1 timeout
10.2.0.5 icmp_seq=2 timeout
10.2.0.5 icmp_seq=3 timeout
10.2.0.5 icmp_seq=4 timeout
10.2.0.5 icmp_seq=5 timeout

```

将局域网 1,2 的掩码改成 16 位后，对于局域网 1,2 里面的 PC 来说（截图以 PC-3 为例），可以 ping 到局域网内部（局域网 1）的 PC，但是 ping 不到别的局域网（局域网 2,3,4）。

对于局域网 3,4 里面的 PC 来说（截图以 PC-9）为例，可以 ping 到局域网内部和另外一个掩码位数没改的局域网（局域网 3,4），但是 ping 不到掩码位数修改的地方（局域网 1,2）。

对于局域网 1,2 里面的 PC 来说（截图以 PC-3 为例），它认为子网是后 16 位，所以默认局域网 3 和 4 也是它的子网。但是他在子网里根本找不到这些，就是 No gateway found。

对于局域网 3,4 里面的 PC 来说，首先他们肯定能访问到局域网 3,4。但是对于局域网 1,2 来说由于掩码位数变了，他们虽然能定位但是就 ping 不通了。

- 把第 3、4 局域网的子网掩码从 8 位改成 16 位，再次用 Ping 检查各个 PC 之间的  
联通性（每个局域网选取 2 台 PC 做代表）。哪些通？哪些不通？

```
PC-9> ip 10.3.0.4 16
Checking for duplicate address...
PC1 : 10.3.0.4 255.255.0.0
```

```
PC-11> ip 10.3.0.6 16
Checking for duplicate address...
PC1 : 10.3.0.6 255.255.0.0
```

```
PC-14> ip 10.4.0.6 16
Checking for duplicate address...
PC1 : 10.4.0.6 255.255.0.0
```

```
PC-13> ip 10.4.0.5 16
Checking for duplicate address...
PC1 : 10.4.0.5 255.255.0.0
```

为了确保 ip 地址修改无误，我不光把掩码改成了 16 位，之前的 ip 地址也重新编号（在原来的基础上继续增大，即原来 PC9~PC11 是 1~3 标，现在是 4~6 标）。

```
PC-3> ping 10.1.0.9
84 bytes from 10.1.0.9 icmp_seq=1 ttl=64 time=0.104 ms
84 bytes from 10.1.0.9 icmp_seq=2 ttl=64 time=0.138 ms
84 bytes from 10.1.0.9 icmp_seq=3 ttl=64 time=0.157 ms
84 bytes from 10.1.0.9 icmp_seq=4 ttl=64 time=0.149 ms
84 bytes from 10.1.0.9 icmp_seq=5 ttl=64 time=0.141 ms
```

```
PC-3> ping 10.2.0.3
No gateway found
```

```
PC-3> ping 10.2.0.5
No gateway found
```

```
PC-3> ping 10.3.0.5
No gateway found
```

```
PC-3> ping 10.4.0.5
No gateway found
```

```
PC-9> ping 10.1.0.6
No gateway found
```

```
PC-9> ping 10.2.0.4
No gateway found
```

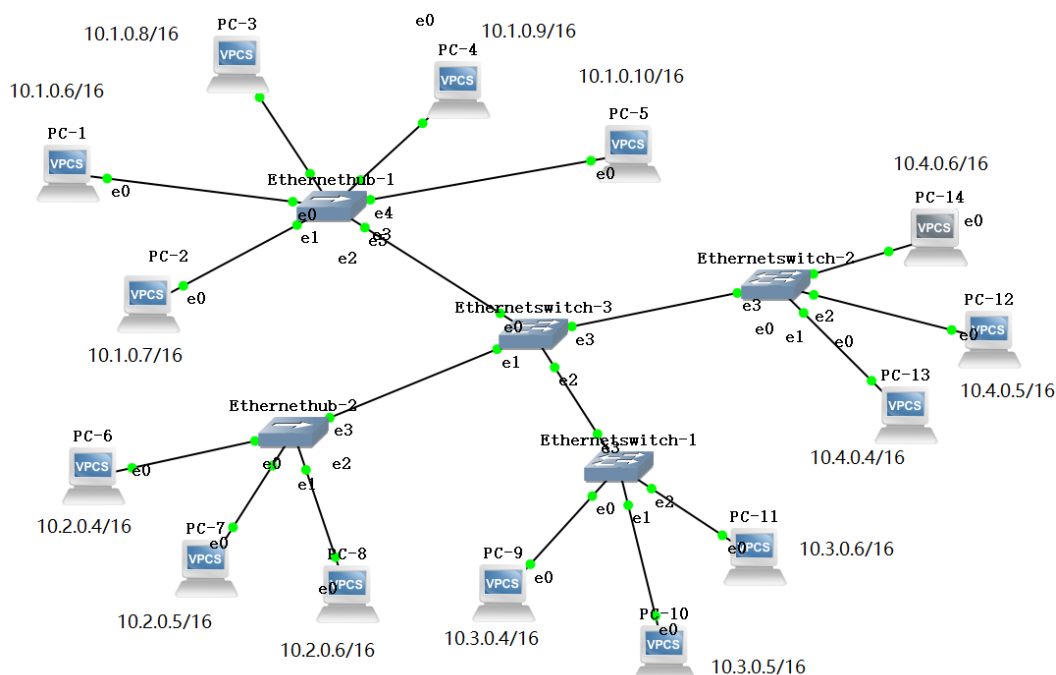
```
PC-9> ping 10.3.0.5
84 bytes from 10.3.0.5 icmp_seq=1 ttl=64 time=0.079 ms
84 bytes from 10.3.0.5 icmp_seq=2 ttl=64 time=0.100 ms
84 bytes from 10.3.0.5 icmp_seq=3 ttl=64 time=0.088 ms
84 bytes from 10.3.0.5 icmp_seq=4 ttl=64 time=0.085 ms
84 bytes from 10.3.0.5 icmp_seq=5 ttl=64 time=0.084 ms
```

```
PC-9> ping 10.4.0.6
No gateway found
```

```
PC-9>
```

对于局域网 1,2,3,4 里面的所有 PC 来说（截图以 PC-3 和 PC-9 为例），可以 ping 到局域网内部的 PC，但是 ping 不到别的局域网。

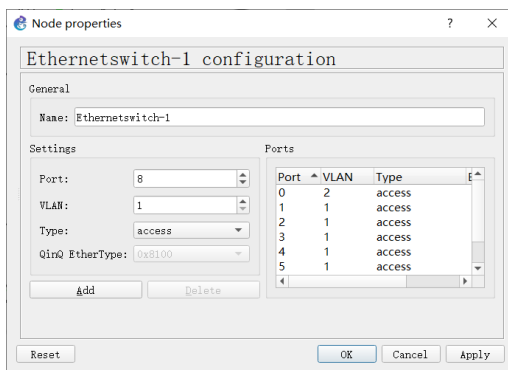
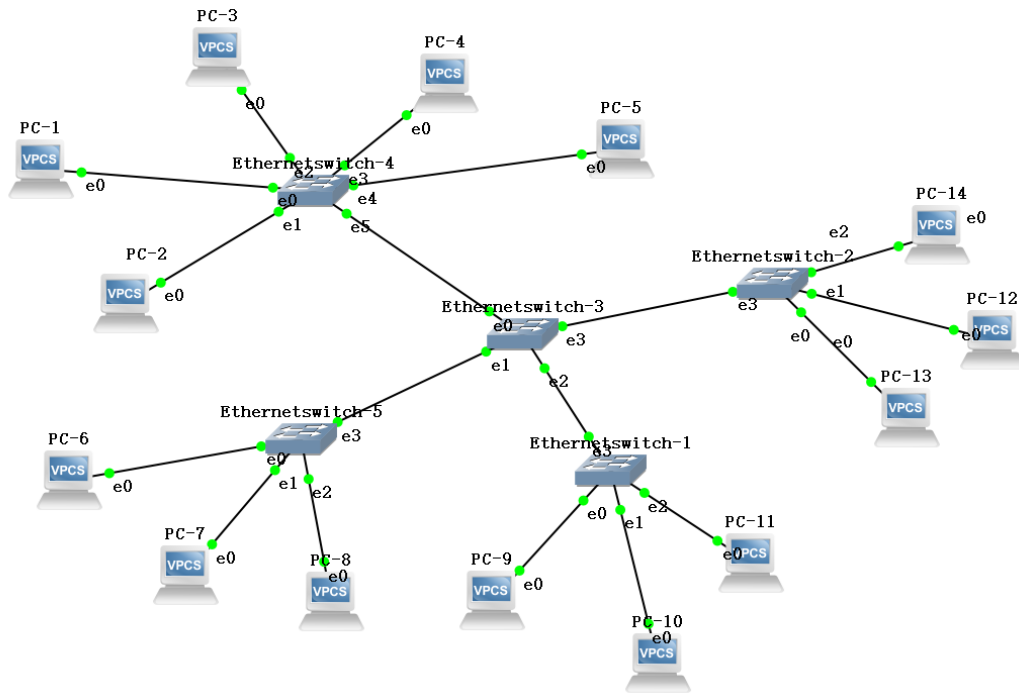
- 请把本部分的最后完整拓扑图记录在此。并标注每台 PC 的 IP 地址、子网掩码。（将实验配置保存为 part1 目录，随实验报告一起打包上交）





## -----Part 2. VLAN-----

- 将 HUB 都换成交换机。在第 1-4 局域网交换机上都新增 1 个 VLAN 2（请参考指南“十四、二层交换机”进行配置并截图），让每个局域网中都有一部分 PC 属于 VLAN 2（默认所有的 PC 都属于 VLAN 1）。使用 Ping 命令检查各个 PC 之间的联通性。哪些通？哪些不通？不通的 PC 之间的数据包，什么时候显示是可达的，什么时候显示是超时的？



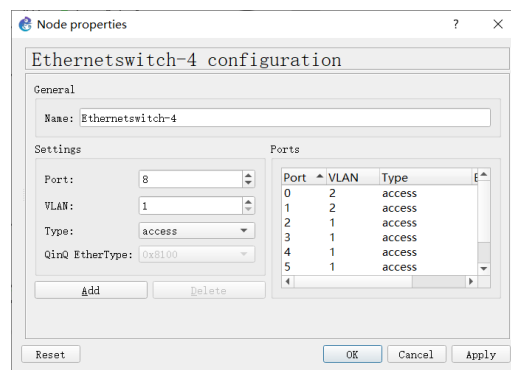
```
PC-1> ping 10.1.0.7
84 bytes from 10.1.0.7 icmp_seq=1 ttl=64 time=0.085 ms
84 bytes from 10.1.0.7 icmp_seq=2 ttl=64 time=0.089 ms
84 bytes from 10.1.0.7 icmp_seq=3 ttl=64 time=0.094 ms
84 bytes from 10.1.0.7 icmp_seq=4 ttl=64 time=0.090 ms
84 bytes from 10.1.0.7 icmp_seq=5 ttl=64 time=0.088 ms

PC-1> ping 10.1.0.8
host (10.1.0.8) not reachable

PC-1> ping 10.2.0.4
No gateway found

PC-1> ping 10.3.0.4
No gateway found

PC-1> ping 10.4.0.4
No gateway found
```



```
PC-4> ping 10.1.0.6
host (10.1.0.6) not reachable

PC-4> ping 10.1.0.8
84 bytes from 10.1.0.8 icmp_seq=1 ttl=64 time=0.059 ms
84 bytes from 10.1.0.8 icmp_seq=2 ttl=64 time=0.086 ms
84 bytes from 10.1.0.8 icmp_seq=3 ttl=64 time=0.084 ms
84 bytes from 10.1.0.8 icmp_seq=4 ttl=64 time=0.130 ms
84 bytes from 10.1.0.8 icmp_seq=5 ttl=64 time=0.089 ms

PC-4> ping 10.2.0.4
host (10.2.0.4) not reachable

PC-4> ping 10.2.0.5
host (10.2.0.5) not reachable
```

首先按要求将 hub 全都转成 switch，并将局域网 1 的 switch 端口 0,1（对应 PC-1,2）和局域网 2,3,4 的 switch 端口 0（对应 PC-6,9,12）调成 VLAN2。

对于连接在 VLAN2 端口的 PC（以 PC-1 为例），它可以 ping 到本地也是 VLAN2 的 PC（PC-2），但是 ping 不到本地 VLAN1 的 PC 且显示“不可达”；它也 ping 不到其他局域网的 PC 且显示“No gateway found”。

对于连接在 VLAN1 端口的 PC（以 PC-4 为例），它可以 ping 到本地也是 VLAN1 的 PC（PC-3,5），但是 ping 不到本地 VLAN2 的 PC 且显示“不可达”；它也 ping 不到其他局域网的 PC 且显示“不可达”。

- 在第 1-4 局域网上，把属于 VLAN 1 的 PC 的 IP 地址都改成 10.1.0.0/16 子网内的地址，把属于 VLAN 2 的 PC 的 IP 地址都改成 10.2.0.0/16 子网内的地址。使用 Ping 命令检查同属于 VLAN 1，但在不同局域网的 PC 之间的连通性（应该通）。使用 Ping 命令检查同属于 VLAN 2，但在不同局域网的 PC 之间的连通性（应该不通）。

```
PC-3> ip 10.1.0.3 16
Checking for duplicate address...
PC1 : 10.1.0.3 255.255.0.0

PC-3> save
Saving startup configuration to startup.vpc
. done
```

```
PC-6> ip 10.2.0.6 16
Checking for duplicate address...
PC1 : 10.2.0.6 255.255.0.0

PC-6> save
Saving startup configuration to startup.vpc
. done
```

我发现本实验要频繁修改 ip，于是打算重新设定 ip 的分配方式，以防操作的时候搞混。每次分配子网 ip 的时候，最后一格（8bit）的 ip 我都设为该 PC 的编号。由上一个实验步骤，已知在 VLAN1 的有 PC-3,4,5,7,8,10,11,13,14，在 VLAN2 的有 PC-1,2,6,9,12，所以比如 PC-3 的编号为 10.1.0.3，PC-6 的编号为 10.2.0.6。

```
PC-4> ping 10.1.0.7
84 bytes from 10.1.0.7 icmp_seq=1 ttl=64 time=0.365 ms
84 bytes from 10.1.0.7 icmp_seq=2 ttl=64 time=0.130 ms
84 bytes from 10.1.0.7 icmp_seq=3 ttl=64 time=0.111 ms
84 bytes from 10.1.0.7 icmp_seq=4 ttl=64 time=0.156 ms
84 bytes from 10.1.0.7 icmp_seq=5 ttl=64 time=0.121 ms

PC-4> ping 10.1.0.6
host (10.1.0.6) not reachable
```

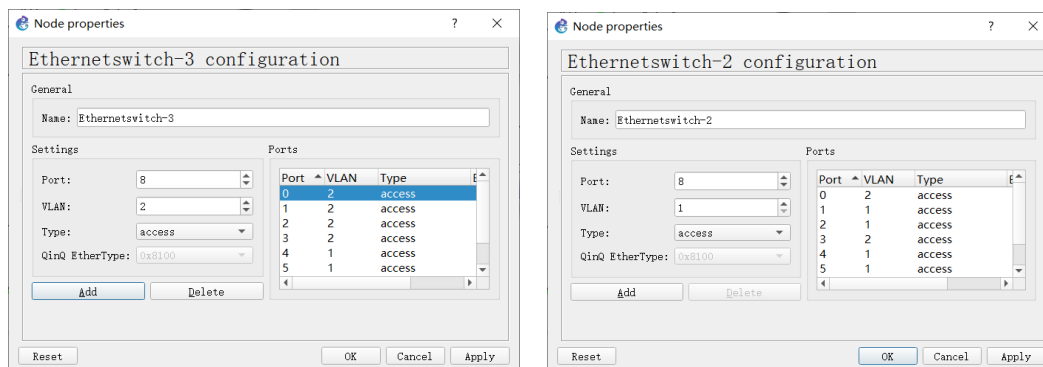
```
PC-1> ping 10.2.0.2
84 bytes from 10.2.0.2 icmp_seq=1 ttl=64 time=0.239 ms
84 bytes from 10.2.0.2 icmp_seq=2 ttl=64 time=0.117 ms
84 bytes from 10.2.0.2 icmp_seq=3 ttl=64 time=0.169 ms
84 bytes from 10.2.0.2 icmp_seq=4 ttl=64 time=0.089 ms
84 bytes from 10.2.0.2 icmp_seq=5 ttl=64 time=0.092 ms

PC-1> ping 10.2.0.6
host (10.2.0.6) not reachable

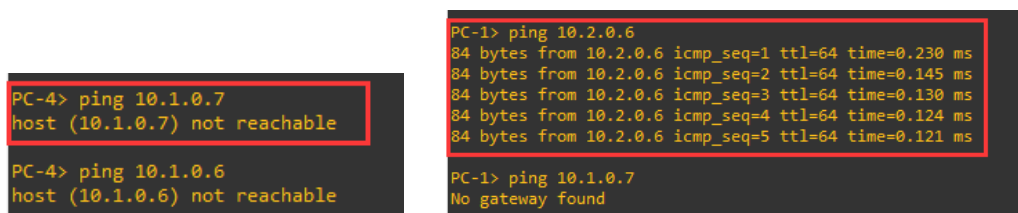
PC-1> ping 10.1.0.7
No gateway found
```

首先我取 VLAN1 的 PC-4，它往不同局域网（PC-7）同一个 VLAN ping 可以通。然后再取 VLAN2 的 PC-1，它往不同局域网的同一个 VLAN ping（PC-6）的确不能通（当然在同一个局域网同一个 VLAN 上依然是可以通的）。

- 在第 5 个交换机（互联交换机）上新增 VLAN 2，将该交换机上连接 4 个局域网的端口（包括 4 个交换机的端口）都修改为属于 VLAN 2。使用 Ping 命令检查同属于 VLAN 2，但在不同局域网的 PC 之间的联通性（应该通）。此时，再次使用 Ping 命令检查同属于 VLAN 1，但在不同局域网的 PC 之间的联通性（应该不通）。

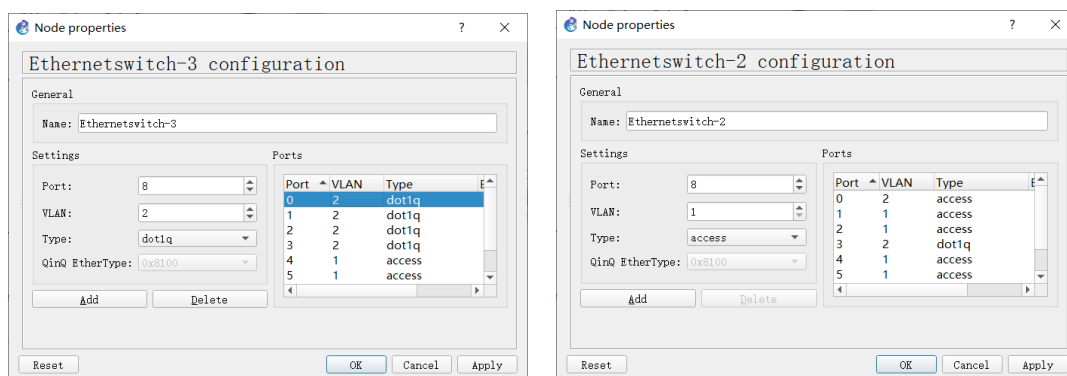


把中心交换机（switch-3）的 0,1,2,3 四个端口的 VLAN 都设置成了 2，再把其他交换机（switch-1,2,4,5）连接到中心交换机的端口也设置成 2。



重复上一个步骤的命令。首先我取 VLAN1 的 PC-4，它往不同局域网同一个 VLAN 的机器（PC-7）ping 现在通不了了。然后再取 VLAN2 的 PC-1，它往不同局域网的同一个 VLAN 的机器（PC-6）ping 变成通了。

- 在第 5 个交换机（互联交换机）上将连接 4 个局域网的端口都修改为 VLAN Trunk 模式（注意同时需要修改对应的 4 个交换机的端口）。使用 Ping 命令检查同属于 VLAN 1，但在不同局域网的 PC 之间的联通性。使用 Ping 命令检查同属于 VLAN 2，但在不同局域网的 PC 之间的联通性。



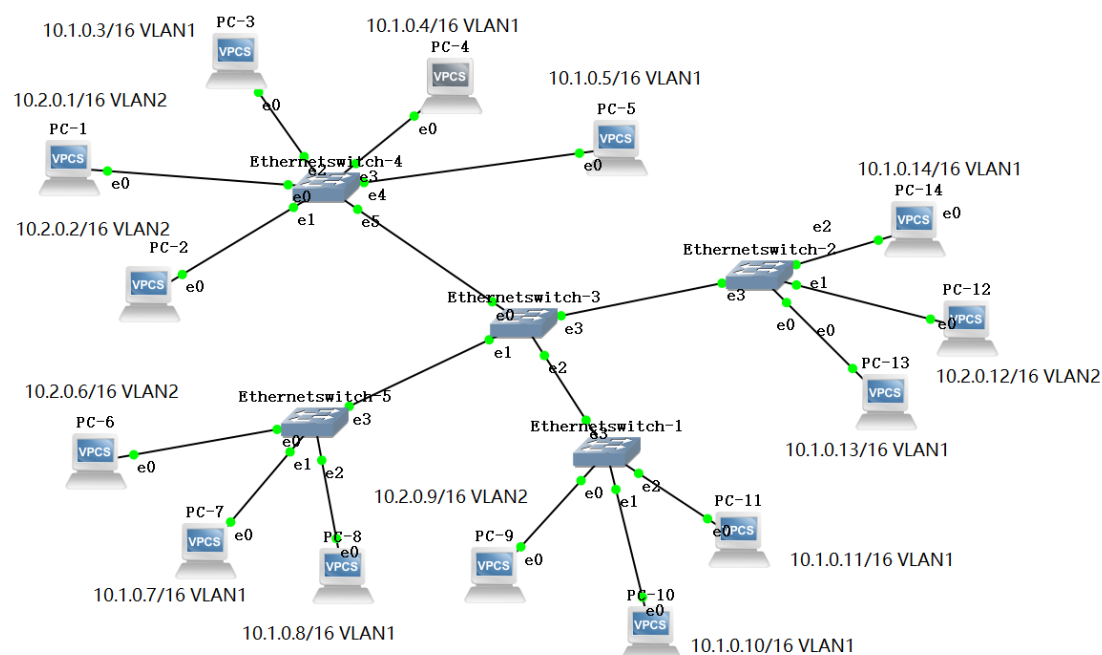
把中心交换机（switch-3）的 0,1,2,3 四个端口的都设置成 trunk（dot1q），再把其他交换机（switch-1,2,4,5）连接到中心交换机的端口也相应的修改。

```
PC-4> ping 10.1.0.7
84 bytes from 10.1.0.7 icmp_seq=1 ttl=64 time=0.140 ms
84 bytes from 10.1.0.7 icmp_seq=2 ttl=64 time=0.121 ms
84 bytes from 10.1.0.7 icmp_seq=3 ttl=64 time=0.126 ms
84 bytes from 10.1.0.7 icmp_seq=4 ttl=64 time=0.122 ms
84 bytes from 10.1.0.7 icmp_seq=5 ttl=64 time=0.139 ms
```

```
PC-1> ping 10.2.0.6
84 bytes from 10.2.0.6 icmp_seq=1 ttl=64 time=0.255 ms
84 bytes from 10.2.0.6 icmp_seq=2 ttl=64 time=0.175 ms
84 bytes from 10.2.0.6 icmp_seq=3 ttl=64 time=0.117 ms
84 bytes from 10.2.0.6 icmp_seq=4 ttl=64 time=0.133 ms
84 bytes from 10.2.0.6 icmp_seq=5 ttl=64 time=0.120 ms
```

说明改成 trunk 模式后，所有机器均能和别的局域网的相同 VLAN 端口的机器 ping 通。

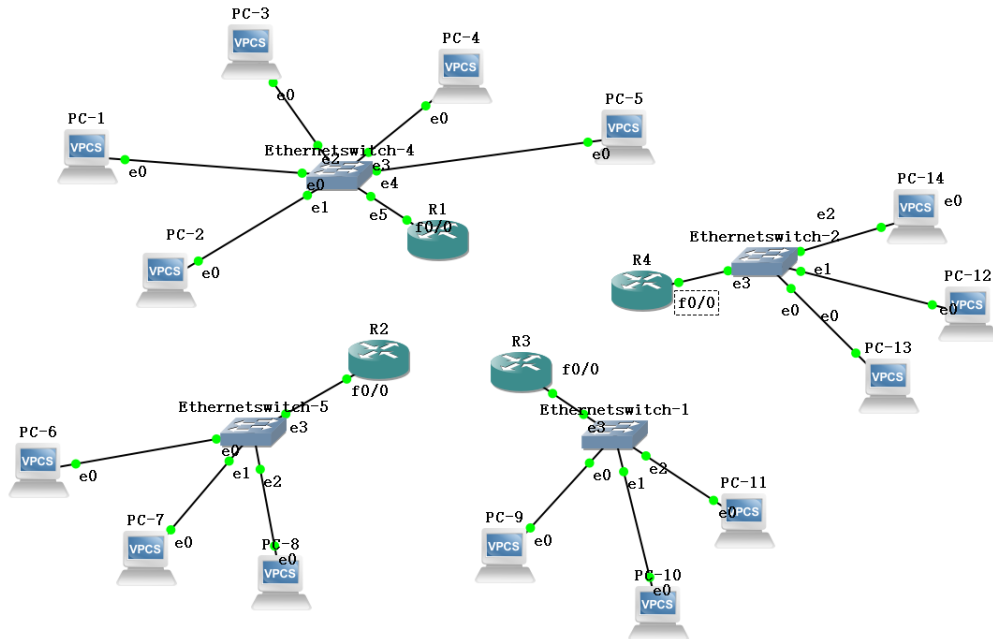
- 请把本部分的最后完整拓扑图记录在此。并标注每台 PC 的 IP 地址、子网掩码和属于的 VLAN。（将实验配置为 part2 目录，随实验报告一起打包上交）



### -----Part 3. 路由-----

以下使用的路由器是前面导入的 c3745 路由器。注意在路由器的 Console 中输入配置命令前需要先输入 **conf t** 进入配置模式。

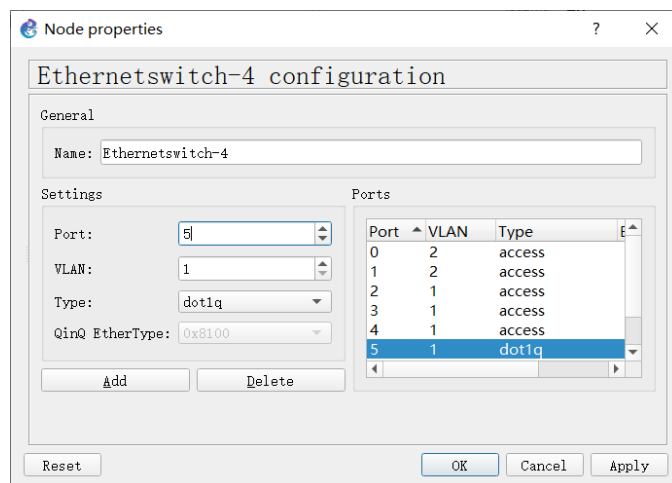
- 将第 5 个交换机删除，为每个局域网增加一个路由器，并用网线将本局域网的路由器与交换机连接起来（记录下拓扑图）。



- 下面的 2 个步骤在第 1 个局域网中进行
  - 步骤 1: 在路由器上与交换机连接的物理端口上创建接口（命令格式: **interface 物理接口.接口**, 如 **interface e0/0**），再创建 2 个逻辑子接口（命令格式: **interface 物理接口.子接口**, 如 **interface e0/0.1**）；然后让 2 个子接口分别属于 VLAN 1 和 VLAN 2（命令: **encapsulation dot1q VLAN 编号**），并给 2 个子接口的 IP 地址分别配置为 10.1.0.0/16 和 10.2.0.0/16 子网内的地址，最后激活端口（命令: **no shutdown**）。（本步骤截取实际使用的配置命令）

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#int f0/0
R1(config-if)#no shut
R1(config-if)#int f0/0.1
R1(config-subif)#encapsulation dot1q 1
R1(config-subif)#ip addr 10.1.0.20 255.255.0.0
R1(config-subif)#no shut
R1(config-subif)#exit
R1(config)#int f0/0.2
R1(config-subif)#encapsulation dot1q 2
R1(config-subif)#ip addr 10.2.0.20 255.255.0.0
R1(config-subif)#no shut
R1(config-subif)#exit
R1(config)#exit
```

- 步骤 2: 给 2 个 VLAN 内的 PC 机配置默认路由器/网关（gateway）地址，分别设置为路由器上所属 VLAN 的子接口的 IP 地址。注意要将交换机上与路由器连接的端口设置为 VLAN Trunk 模式，且将 VLAN 号设置为 1。使用 Ping 检查属于不同 VLAN 的 PC 之间的联通性（应该通）。



```

show ip interface brief

```

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	unassigned	YES	unset	up	up
FastEthernet0/0.1	10.1.0.20	YES	manual	up	up
FastEthernet0/0.2	10.2.0.20	YES	manual	up	up
FastEthernet0/1	unassigned	YES	unset	administratively down	down
FastEthernet1/0	unassigned	YES	unset	up	down
FastEthernet1/1	unassigned	YES	unset	up	down
FastEthernet1/2	unassigned	YES	unset	up	down
FastEthernet1/3	unassigned	YES	unset	up	down
FastEthernet1/4	unassigned	YES	unset	up	down
FastEthernet1/5	unassigned	YES	unset	up	down
FastEthernet1/6	unassigned	YES	unset	up	down

```

PC-1> ping 10.1.0.3
84 bytes from 10.1.0.3 icmp_seq=1 ttl=63 time=23.073 ms
84 bytes from 10.1.0.3 icmp_seq=2 ttl=63 time=12.206 ms
84 bytes from 10.1.0.3 icmp_seq=3 ttl=63 time=12.918 ms
84 bytes from 10.1.0.3 icmp_seq=4 ttl=63 time=12.199 ms
84 bytes from 10.1.0.3 icmp_seq=5 ttl=63 time=13.499 ms

```

```

PC-3> ping 10.2.0.1
84 bytes from 10.2.0.1 icmp_seq=1 ttl=63 time=19.390 ms
84 bytes from 10.2.0.1 icmp_seq=2 ttl=63 time=19.916 ms
84 bytes from 10.2.0.1 icmp_seq=3 ttl=63 time=20.996 ms
84 bytes from 10.2.0.1 icmp_seq=4 ttl=63 time=16.019 ms
84 bytes from 10.2.0.1 icmp_seq=5 ttl=63 time=16.045 ms

```

发现位于 VLAN2 的 PC-1 可以 ping 通 VLAN1 的 PC-3，PC-3 也可以 ping 通 PC-1。

- 修改第 2 个局域网中 PC 的 IP 地址，把 10.1.0.0/16、10.2.0.0/16 子网内的 IP 地址分别改成 12.1.0.0/16、12.2.0.0/16 子网内的 IP 地址。然后按照第 1 个局域网的 2 个步骤，给路由器分配 IP 地址，给 PC 配置默认路由器/网关地址。用 Ping 检查不同 VLAN 的 PC 之间联通性。

```

PC-6> ip 12.2.0.6 255.255.0.0 12.2.0.20
Checking for duplicate address...
PC1 : 12.2.0.6 255.255.0.0 gateway 12.2.0.20

PC-6> save
Saving startup configuration to startup.vpc
. done

```

```

PC-7> ip 12.1.0.7 255.255.0.0 12.1.0.20
Checking for duplicate address...
PC1 : 12.1.0.7 255.255.0.0 gateway 12.1.0.20

PC-7> save
Saving startup configuration to startup.vpc
. done

```

首先对局域网里的 PC 设置 IP 和默认网关地址。

```

R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#int f0/0
R2(config-if)#no shut
R2(config-if)#int f0/0.1
R2(config-subif)#encapsulation dot1q 1
R2(config-subif)#ip addr 12.1.0.20 255.255.0.0
R2(config-subif)#no shut
R2(config-subif)#exit
R2(config)#int f0/0.2
R2(config-subif)#encapsulation dot1q 2
R2(config-subif)#ip addr 12.2.0.20 255.255.0.0
R2(config-subif)#no shut
R2(config-subif)#exit

```

然后对 R2 进行相应的配置。

```

PC-6> ping 12.1.0.7
84 bytes from 12.1.0.7 icmp_seq=1 ttl=63 time=17.026 ms
84 bytes from 12.1.0.7 icmp_seq=2 ttl=63 time=12.085 ms
84 bytes from 12.1.0.7 icmp_seq=3 ttl=63 time=16.054 ms
84 bytes from 12.1.0.7 icmp_seq=4 ttl=63 time=16.063 ms
84 bytes from 12.1.0.7 icmp_seq=5 ttl=63 time=13.995 ms

```

```

PC-7> ping 12.2.0.6
84 bytes from 12.2.0.6 icmp_seq=1 ttl=63 time=16.359 ms
84 bytes from 12.2.0.6 icmp_seq=2 ttl=63 time=18.919 ms
84 bytes from 12.2.0.6 icmp_seq=3 ttl=63 time=18.910 ms
84 bytes from 12.2.0.6 icmp_seq=4 ttl=63 time=18.413 ms
84 bytes from 12.2.0.6 icmp_seq=5 ttl=63 time=18.596 ms

```

发现位于 VLAN2 的 PC-6 可以 ping 通 VLAN1 的 PC-7，PC-7 也可以 ping 通 PC-6。

- 修改第 3 个局域网中 PC 的 IP 地址，把 10.1.0.0/16、10.2.0.0/16 子网内的 IP 地址分别改成 13.1.0.0/16、13.2.0.0/16 子网内的 IP 地址。然后按照第 1 个局域网的 2 个步骤，给路由器分配 IP 地址，给 PC 配置默认路由器/网关地址。用 Ping 检查不同 VLAN 的 PC 之间联通性。

```

PC1 : 10.2.0.9 255.255.0.0

PC-9> ip 13.2.0.9 255.255.0.0 13.2.0.20
Checking for duplicate address...
PC1 : 13.2.0.9 255.255.0.0 gateway 13.2.0.20

PC-9> save
Saving startup configuration to startup.vpc
. done

```

```

PC1 : 10.1.0.10 255.255.0.0

PC-10> ip 13.1.0.10 255.255.0.0 13.1.0.20
Checking for duplicate address...
PC1 : 13.1.0.10 255.255.0.0 gateway 13.1.0.20

PC-10> save
Saving startup configuration to startup.vpc
. done

```

首先对局域网里的 PC 设置 IP 和默认网关地址。

```

R3(config)#int f0/0
R3(config-if)#no shut
R3(config-if)#int f0/0.1
R3(config-subif)#encapsulation dot1q 1
R3(config-subif)#ip addr 13.1.0.20 255.255.0.0
R3(config-subif)#no shut
R3(config-subif)#exit
R3(config)#int f0/0.2
R3(config-subif)#encapsulation dot1q 2
R3(config-subif)#ip addr 13.2.0.20 255.255.0.0
R3(config-subif)#no shut

```

然后对 R3 进行相应的配置。

```

PC-9> ping 13.1.0.11
84 bytes from 13.1.0.11 icmp_seq=1 ttl=63 time=13.886 ms
84 bytes from 13.1.0.11 icmp_seq=2 ttl=63 time=16.999 ms
84 bytes from 13.1.0.11 icmp_seq=3 ttl=63 time=19.434 ms
84 bytes from 13.1.0.11 icmp_seq=4 ttl=63 time=18.423 ms
84 bytes from 13.1.0.11 icmp_seq=5 ttl=63 time=19.900 ms

```

```

PC-11> ping 13.2.0.9
84 bytes from 13.2.0.9 icmp_seq=1 ttl=63 time=20.341 ms
84 bytes from 13.2.0.9 icmp_seq=2 ttl=63 time=19.957 ms
84 bytes from 13.2.0.9 icmp_seq=3 ttl=63 time=19.543 ms
84 bytes from 13.2.0.9 icmp_seq=4 ttl=63 time=21.983 ms
84 bytes from 13.2.0.9 icmp_seq=5 ttl=63 time=14.081 ms

```

发现位于 VLAN2 的 PC-9 可以 ping 通 VLAN1 的 PC-13，PC-13 也可以 ping 通 PC-9。







```

R5#config t
Enter configuration commands, one per line. End with CNTL/Z.
R5(config)#int f0/0
R5(config-if)#no shut
R5(config-if)#ip addr 192.168.1.87 255.255.255.0
R5(config-if)#exit
R5(config)#int f0/1
R5(config-if)#no shut
R5(config-if)#ip addr 192.168.2.87 255.255.255.0
R5(config-if)#exit
R5(config)#int f1/0
R5(config-if)#no shut
R5(config-if)#ip addr 192.168.3.87 255.255.255.0
R5(config-if)#exit
R5(config)#int f2/0
R5(config-if)#no shut
R5(config-if)#ip addr 192.168.4.87 255.255.255.0
R5(config-if)#exit
R5(config)#
R5(config)#exit

```

我的学号末尾是 87，我就以 87 标最后一位。

- 启用各路由器上的动态路由协议 RIP(命令: `router rip`), 版本设置成 2(命令: `version 2`, 感兴趣的同学可以研究不同协议版本的影响), 将本路由器上的各端口所在子网加入到路由信息交换 (命令: `network 子网`, 如 `network 10.1.0.0`)。等待一段时间后, 使用 Ping 命令测试下各 PC 之间的联通性。截图显示各路由器上的路由表信息 (命令: `show ip route`, 标记哪些路由是直连的, 哪些是通过路由协议动态获取的)。

```

R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#network 10.1.0.0
R1(config-router)#network 10.2.0.0
R1(config-router)#network 192.168.1.0
R1(config-router)#exit
R1(config)#exit
R1#
*Mar 1 00:30:56.031: %SYS-5-CONFIG_I: Configured from console by console

R5#config t
Enter configuration commands, one per line. End with CNTL/Z.
R5(config)#router rip
R5(config-router)#version 2
R5(config-router)#network 192.168.1.0
R5(config-router)#network 192.168.2.0
R5(config-router)#network 192.168.3.0
R5(config-router)#network 192.168.4.0
R5(config-router)#exit
R5(config)#exit
R5#

```

需要注意的是, 不仅要配置 R5 朝向 R1~R4 的端口地址, 激活 R1~R5 的 network, **还要配置 R1~R4 朝向 R5 的端口地址**。这一步在实验步骤里没有明说, 但是如果没有配置就会 ping 不通。以下是我配置 R1~R4 朝向 R5 的端口地址的举例。

```

R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#int f0/1
R1(config-if)#ip addr 192.168.1.20 255.255.255.0
R1(config-if)#no shut
R1(config-if)#exit
R1(config)#

```

```

PC-1> ping 12.1.0.8
12.1.0.8 icmp_seq=1 timeout
12.1.0.8 icmp_seq=2 timeout
84 bytes from 12.1.0.8 icmp_seq=3 ttl=61 time=53.698 ms
84 bytes from 12.1.0.8 icmp_seq=4 ttl=61 time=60.638 ms
84 bytes from 12.1.0.8 icmp_seq=5 ttl=61 time=61.623 ms

PC-1> ping 10.1.0.3
10.1.0.3 icmp_seq=1 timeout
10.1.0.3 icmp_seq=2 timeout
84 bytes from 10.1.0.3 icmp_seq=3 ttl=63 time=15.465 ms
84 bytes from 10.1.0.3 icmp_seq=4 ttl=63 time=18.421 ms
84 bytes from 10.1.0.3 icmp_seq=5 ttl=63 time=18.967 ms

PC-1> ping 13.1.0.10
13.1.0.10 icmp_seq=1 timeout
13.1.0.10 icmp_seq=2 timeout
84 bytes from 13.1.0.10 icmp_seq=3 ttl=61 time=47.686 ms
84 bytes from 13.1.0.10 icmp_seq=4 ttl=61 time=50.670 ms
84 bytes from 13.1.0.10 icmp_seq=5 ttl=61 time=60.129 ms

PC-1> ping 14.2.0.12
14.2.0.12 icmp_seq=1 timeout
14.2.0.12 icmp_seq=2 timeout
84 bytes from 14.2.0.12 icmp_seq=3 ttl=61 time=42.570 ms
84 bytes from 14.2.0.12 icmp_seq=4 ttl=61 time=52.144 ms
84 bytes from 14.2.0.12 icmp_seq=5 ttl=61 time=50.217 ms

```

启动了 RIP 协议后所有 PC 之间都能 ping 通。容易发现，局域网内部的速度比较快（15ms~20ms），跨局域网的速度比较慢（40ms~60ms）。路由器信息表见下：

```

R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    192.168.4.0/24 [120/1] via 192.168.1.87, 00:00:19, FastEthernet0/1
10.0.0.0/16 is subnetted, 2 subnets
C    10.2.0.0 is directly connected, FastEthernet0/0.2
C    10.1.0.0 is directly connected, FastEthernet0/0.1
R    12.0.0.0/8 [120/2] via 192.168.1.87, 00:00:19, FastEthernet0/1
C    192.168.1.0/24 is directly connected, FastEthernet0/1
R    13.0.0.0/8 [120/2] via 192.168.1.87, 00:00:19, FastEthernet0/1
R    192.168.2.0/24 [120/1] via 192.168.1.87, 00:00:19, FastEthernet0/1
R    14.0.0.0/8 [120/2] via 192.168.1.87, 00:00:20, FastEthernet0/1
R    192.168.3.0/24 [120/1] via 192.168.1.87, 00:00:20, FastEthernet0/1

```

```

R5#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.4.0/24 is directly connected, FastEthernet2/0
R    10.0.0.0/8 [120/1] via 192.168.1.20, 00:00:18, FastEthernet0/0
R    12.0.0.0/8 [120/1] via 192.168.2.20, 00:00:24, FastEthernet0/1
C    192.168.1.0/24 is directly connected, FastEthernet0/0
R    13.0.0.0/8 [120/1] via 192.168.3.20, 00:00:06, FastEthernet1/0
C    192.168.2.0/24 is directly connected, FastEthernet0/1
R    14.0.0.0/8 [120/1] via 192.168.4.20, 00:00:25, FastEthernet2/0
C    192.168.3.0/24 is directly connected, FastEthernet1/0

```

标有 C 的是直连的，标有 R 的是根据协议自动生成的。

- 关闭各路由器上的动态路由协议 RIP（命令：no router rip）。等待一段时间后，显示各路由器上的路由表信息（动态获取的路由信息是否消失了？）。再次使用 Ping 命令测试下各 PC 之间的联通性。哪些通？哪些不通？不通的数据包在哪个环节不再转发了（是不是在该路由器上缺少对应的路由信息？）。

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#no router rip
R1(config)#exit

10.0.0.0/16 is subnetted, 2 subnets
C    10.2.0.0 is directly connected, FastEthernet0/0.2
C    10.1.0.0 is directly connected, FastEthernet0/0.1
C    192.168.1.0/24 is directly connected, FastEthernet0/1
```

No router rip 后，动态获取的路由信息消失了。

```
PC-1> ping 10.1.0.3
10.1.0.3 icmp_seq=1 timeout
10.1.0.3 icmp_seq=2 timeout
84 bytes from 10.1.0.3 icmp_seq=3 ttl=63 time=17.069 ms
84 bytes from 10.1.0.3 icmp_seq=4 ttl=63 time=15.048 ms
84 bytes from 10.1.0.3 icmp_seq=5 ttl=63 time=25.033 ms

PC-1> ping 12.1.0.8
*10.2.0.20 icmp_seq=1 ttl=255 time=19.960 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=2 ttl=255 time=9.009 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=3 ttl=255 time=8.533 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=4 ttl=255 time=9.011 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=5 ttl=255 time=8.493 ms (ICMP type:3, code:1, Destination host unreachable)

PC-1> ping 13.1.0.10
*10.2.0.20 icmp_seq=1 ttl=255 time=8.005 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=2 ttl=255 time=9.524 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=3 ttl=255 time=8.006 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=4 ttl=255 time=8.497 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=5 ttl=255 time=8.469 ms (ICMP type:3, code:1, Destination host unreachable)

PC-1> ping 14.2.0.12
*10.2.0.20 icmp_seq=1 ttl=255 time=9.098 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=2 ttl=255 time=10.072 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=3 ttl=255 time=16.077 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=4 ttl=255 time=15.088 ms (ICMP type:3, code:1, Destination host unreachable)
*10.2.0.20 icmp_seq=5 ttl=255 time=6.114 ms (ICMP type:3, code:1, Destination host unreachable)
```

同一个局域网内部依然可以 ping 到，但是跨局域网无法 ping 通。

局域网之间的信息没有联通，在 R1-R4 的路由上缺少了信息，不再转发。

- 在相应的路由器上为某些子网添加正确的静态路由（命令：ip route 目标网络 子网掩码 下一跳地址，如 ip route 11.1.0.0 255.255.0.0 192.168.1.100）。使用 Ping 命令测试下各 PC 之间的联通性。不断的添加静态路由，让所有的 PC 之间都能互相 Ping 通。完成后，截图显示 5 个路由器上的路由表。

参照刚才自动生成的 route 进行静态路由器配置。以下以 R1 和 R5 为例：

```
R1#config t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)#ip route 12.0.0.0 255.0.0.0 192.168.1.87
R1(config)#ip route 13.0.0.0 255.0.0.0 192.168.1.87
R1(config)#ip route 14.0.0.0 255.0.0.0 192.168.1.87
R1(config)#exit
```

```

R5#config t
Enter configuration commands, one per line. End with CNTL/Z.
R5(config)#ip route 10.0.0.0 255.0.0.0 192.168.1.20
R5(config)#ip route 12.0.0.0 255.0.0.0 192.168.2.20
R5(config)#ip route 13.0.0.0 255.0.0.0 192.168.3.20
R5(config)#ip route 14.0.0.0 255.0.0.0 192.168.4.20
R5(config)#exit
R5#

```

全部配置完成后，由 PC-1 发起的命令，其他 PC 都能接收到。

```

PC-8> ping 10.2.0.1
10.2.0.1 icmp_seq=1 timeout
10.2.0.1 icmp_seq=2 timeout
84 bytes from 10.2.0.1 icmp_seq=3 ttl=61 time=63.909 ms
84 bytes from 10.2.0.1 icmp_seq=4 ttl=61 time=58.886 ms
84 bytes from 10.2.0.1 icmp_seq=5 ttl=61 time=59.044 ms

PC-8> ping 12.2.0.6
84 bytes from 12.2.0.6 icmp_seq=1 ttl=63 time=23.315 ms
84 bytes from 12.2.0.6 icmp_seq=2 ttl=63 time=16.420 ms
84 bytes from 12.2.0.6 icmp_seq=3 ttl=63 time=16.424 ms
84 bytes from 12.2.0.6 icmp_seq=4 ttl=63 time=17.020 ms
84 bytes from 12.2.0.6 icmp_seq=5 ttl=63 time=18.966 ms

PC-8> ping 13.1.0.10
84 bytes from 13.1.0.10 icmp_seq=1 ttl=61 time=49.172 ms
84 bytes from 13.1.0.10 icmp_seq=2 ttl=61 time=49.656 ms
84 bytes from 13.1.0.10 icmp_seq=3 ttl=61 time=50.662 ms
84 bytes from 13.1.0.10 icmp_seq=4 ttl=61 time=50.187 ms
84 bytes from 13.1.0.10 icmp_seq=5 ttl=61 time=50.150 ms

```

五个路由表 show ip route 后显示如下：

```

10.0.0.0/16 is subnetted, 2 subnets
C    10.2.0.0 is directly connected, FastEthernet0/0.2
C    10.1.0.0 is directly connected, FastEthernet0/0.1
S    12.0.0.0/8 [1/0] via 192.168.1.87
C    192.168.1.0/24 is directly connected, FastEthernet0/1
S    13.0.0.0/8 [1/0] via 192.168.1.87
S    14.0.0.0/8 [1/0] via 192.168.1.87
R1#

```

```

S    10.0.0.0/8 [1/0] via 192.168.2.87
12.0.0.0/16 is subnetted, 2 subnets
C    12.1.0.0 is directly connected, FastEthernet0/0.1
C    12.2.0.0 is directly connected, FastEthernet0/0.2
S    13.0.0.0/8 [1/0] via 192.168.2.87
C    192.168.2.0/24 is directly connected, FastEthernet0/1
S    14.0.0.0/8 [1/0] via 192.168.2.87
R2#

```

```

S    10.0.0.0/8 [1/0] via 192.168.3.87
S    12.0.0.0/8 [1/0] via 192.168.3.87
13.0.0.0/16 is subnetted, 2 subnets
C    13.1.0.0 is directly connected, FastEthernet0/0.1
C    13.2.0.0 is directly connected, FastEthernet0/0.2
S    14.0.0.0/8 [1/0] via 192.168.3.87
C    192.168.3.0/24 is directly connected, FastEthernet0/1

```

```

C    192.168.4.0/24 is directly connected, FastEthernet0/1
S    10.0.0.0/8 [1/0] via 192.168.4.87
S    12.0.0.0/8 [1/0] via 192.168.4.87
S    13.0.0.0/8 [1/0] via 192.168.4.87
14.0.0.0/16 is subnetted, 2 subnets
C    14.2.0.0 is directly connected, FastEthernet0/0.2
C    14.1.0.0 is directly connected, FastEthernet0/0.1

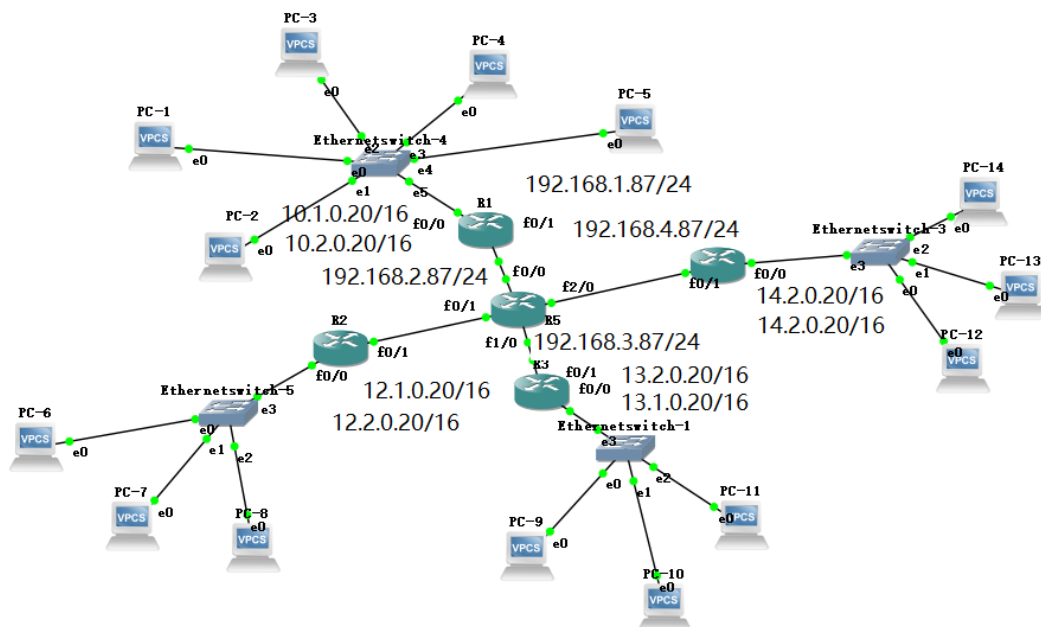
```

```

C    192.168.4.0/24 is directly connected, FastEthernet2/0
S    10.0.0.0/8 [1/0] via 192.168.1.88
      [1/0] via 192.168.1.20
S    12.0.0.0/8 [1/0] via 192.168.2.88
      [1/0] via 192.168.2.20
C    192.168.1.0/24 is directly connected, FastEthernet0/0
S    13.0.0.0/8 [1/0] via 192.168.3.88
      [1/0] via 192.168.3.20
C    192.168.2.0/24 is directly connected, FastEthernet0/1
S    14.0.0.0/8 [1/0] via 192.168.4.88
      [1/0] via 192.168.4.20
C    192.168.3.0/24 is directly connected, FastEthernet1/0
R5#

```

- 请把本部分的最后完整拓扑图记录在此。并标注每台 PC 的 IP 地址、子网掩码和属于的 VLAN，以及每个路由器的端口分配的 IP 地址、子网掩码。（将实验配置保存为 part3 目录，随实验报告一起打包上交）



## 六、 实验结果与分析

根据你观察到的实验数据，分别解答以下问题（看完请删除本句）：

- 在同一个局域网内的 2 台 PC 机，IP 地址分别为 10.0.0.1/8 和 10.1.0.1/8，都属于 VLAN1，一开始可以互相 Ping 通，为什么把子网掩码长度从 8 位变成 16 位，就不通了？

因为改成 16 位后，他们就不属于同一个子网了。

不同网段通信需要路由表，所以他们不能直接通信。

- 仅使用二层交换机的情况下，同一个局域网内，属于不同 VLAN 的 PC 之间为何不能 Ping 通呢？

因为交换机是 Access 模式，根据其规则，不同 VLAN 之间的信息无法传递。

- 交换机的端口设置为 VLAN Trunk 模式后，在通过该端口转发数据包时，交换机会插入什么信息，使得对方交换机能够将数据包转发到正确的 VLAN？

数据包在进入交换机的时候会加入 VID，在进入其他交换机的时候，其他交换机可以通过这个 ID 来确定数据包所属的 VLAN。

- 为了让不同局域网的 PC 之间能够互相 Ping 通，在设置静态路由时，所有路由器之间互联的子网是否一定要全部加入到第 1-4 个局域网路由器的路由表中？

不需要，可以通过其他结构，只要两两之间在借住其他子网的通信下能相互到达就行（比如可以排成  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 1$ ）这种环形结构。

## 七、 讨论、心得

计网的这个实验实在是繁琐，加之做实验之前我对子网、交换机、路由器的知识几乎一无所知，所以整整做了我三天（一个晚上加两个白天）才完美完成这个实验。

在这段时间里我还陆续遇到了很多错误，坚持不懈下终于成功。

第一个问题是配置 GNS3。最开始 GNS3 和虚拟机用的 ova 版本不同一直报错，反反复复折腾了好久才配置完。

第二个问题是我卡得最久的问题，整整卡了两个小时。在 PART3 的开头需要加入路由器并观察同一局域网的不同 VLAN 的 PC 相互通信。我配置了半天，启动后却怎么也 ping 不通。第一个小时的时候我一直在自行 debug，重启 GNS3 重新配置，反复翻看试验资料观察有没有漏掉的点，上网查询一些相关的命令资料，但是最终都没有解决。第二个小时我开始求助，首先询问了上学期上过计网的室友。我们两个对这些配置前前后后研究了好几遍也没发现什么问题，但就是 ping 不通。走投无路的时候我联系了助教，没想到他一针见血地指出问题：我路由器创建两个子接口后，没有把父接口也执行 **no shutdown** 的命令。我把这个加上后终于 ping 通了。我觉得这个错误很大程度上也和实验报告里步骤的不详细有关。本实验报告的步骤里只强调要对子接口执行 **no shutdown**，我觉得如果要想到父接口也要执行还是有点困难的。希望以后的实验能给出更清晰的步骤（当然，直接给出详实的 GNS3 资料让我们学习更好，很多命令我都没在试验资料里找到，自己反复折腾了很久）。

第三个问题是对 R1~R4 每个路由器设置 RIP 协议的时候，我反复检查和配置依旧不能 ping 通。最后发现实验步骤里漏下了“R1~R4 每一个路由器也要对 R5 配置”这个提示。其实如果我早点动脑筋的话，这个错误应该是能早点解决的。我之前对实验报告里的提示太依赖了，无脑地按照它的指示做。这样如果报告里漏下了某个步骤我自然无法成功。

总的来说收获挺大，但是时间花费的也很多。