

# 计算机网络实验实验报告

学院（系）名称：信息学院

姓名	高星杰	学号	2021307220712	专业	计算机科学与技术
班级	21 级 2 班	实验项目			
课程名称		计算机网络实验		课程代码	3103009336
实验时间		2023 年 12 月 21 日		实验地点	逸夫楼 C207
考核内容	目的和原理 40	内容及过程分析 30	实验方案设计 10	实验结果（结论正确性以及分析合理性） 20	成绩
各项得分					
考核标准	<input type="radio"/> 原理明确 <input type="radio"/> 原理较明确 <input type="radio"/> 原理不明确	<input type="radio"/> 分析清晰 <input type="radio"/> 分析较清晰 <input type="radio"/> 分析不清晰	<input type="radio"/> 设计可行 <input type="radio"/> 设计基本可行 <input type="radio"/> 设计不可行	<input type="radio"/> 结论正确，分析合理 <input type="radio"/> 结论正确，分析不充分 <input type="radio"/> 结论不正确，分析不合理	教师签字：

## 1. 实验目的：

- (1) 理解 OSPFv3 协议的工作原理
- (2) 能够独立搭建实现动态路由的网络
- (3) 了解基于 OSPFv3 动态路由的原理

## 2. 实验原理：

根据历史的轨迹，任何新技术的出现都是为了解决某种旧技术的痛点问题的。经过上次实验我们学会了通过手动的配置静态路由，让不同 VLAN 之间可以进行定向的通讯，但是又出现了问题，网络规模较大或网络拓扑经常发生改变时，静态路由有很大的缺陷，静态路由需要人工配置路由表，当网络拓扑结构发生变化时，需要手动调整路由表，这样会增加网络管理员的工作量。而动态路由正好解决了这个问题，动态路由可以自动计算新的路由信息，适应网络拓扑结构的变化，减轻了网络管理员的工作负担。

那么是什么让动态路由可以自适应的变化呢？

### (1) 什么是动态路由？

想象你是一位邮递员，你需要把邮件从一个地方送到另一个地方。在这个过程中，你选择的路线就像是网络中的路由。

静态路由就像是事先规划好的固定路线。无论天气如何，道路是否拥堵，你都会按照这条路线去送邮件。这很简单，因为你不需要每次都重新规划路线，但如果路上出现了意外（比如道路封闭），你可能就无法及时调整路线，导致送邮件耽误。

动态路由则像是你边开车边使用导航系统。导航系统会根据当前的交通状况（比如哪些道路堵车，哪些道路畅通）来实时指引你最快的路线。这意味着如果路上出现了意外，导航系统可以立即告诉你

一个新的路线，帮助你更快地送达邮件。但是，这也意味着你需要一直关注导航系统的指引，并且可能经常需要调整你的路线。

总的来说，静态路由就像是预先规划好的固定路线，简单但在条件变化时不够灵活；而动态路由就像是实时更新的导航系统，更加灵活和适应变化，但需要更多的注意和调整。对于网络来说，选择哪一种路由方式取决于网络的大小、复杂度以及需求的灵活性。小而简单的网络可能更适合静态路由，而大型或经常变化的网络则更适合动态路由。

(1) 路由器之间适时地交换路由信息；

(2) 路由器根据某种路由算法（不同的动态路由协议算法不同）把收集到的路由信息加工成路由表，供路由器在转发 IP 报文时查阅。

## (2) 为什么动态路由可以自适应的变换？

我们了解了什么是动态路由后，我们难免会对其自适应改变的原理产生问题。动态路由能够自适应变化的主要原因在于它使用了一套协议和算法来不断地监测和响应网络条件的变化。这些协议和算法使得动态路由系统能够在以下几个方面自适应地调整：

网络拓扑的变化：动态路由协议能够检测到网络中的变化，如新增或移除的路由器、链路的更改等。当网络拓扑发生变化时，路由器使用这些协议来自动更新它们的路由信息。

链路状态的变化：动态路由还能检测链路的状态变化，比如链路故障、延迟的增加或减少等。当这些状态发生变化时，动态路由算法会重新计算最佳路径。

流量状况的变化：某些动态路由协议能够根据网络流量的变化调整路由。例如，当一条路径变得过于拥塞时，路由器可能会选择一个较少拥塞的备用路径。

算法的优化：动态路由协议如 OSPF（开放最短路径优先）和 BGP（边界网关协议）使用复杂的算法来确保数据以最有效的方式传输。这些算法不断更新和优化路由决策。

总的来说，动态路由之所以能够自适应地变化，是因为它依赖于实时的网络信息和高级的算法来不断优化路由决策，从而确保数据在网络中高效、可靠地传输。

动态路由算法共同的原理是：①路由器之间适时地交换路由信息；②路由器根据某种路由算法（不同的动态路由协议算法不同）把收集到的路由信息加工成路由表，供路由器在转发 IP 报文时查阅。

## (3) 动态路由算法类型

通过上一个问题我们知道了，动态路由同通过高级的算法和协议时间检测网络信息，不断改变路由，从而能够实现自适应，但是动态路由有哪些协议呢？



## (4) 什么是 OSPF 协议？

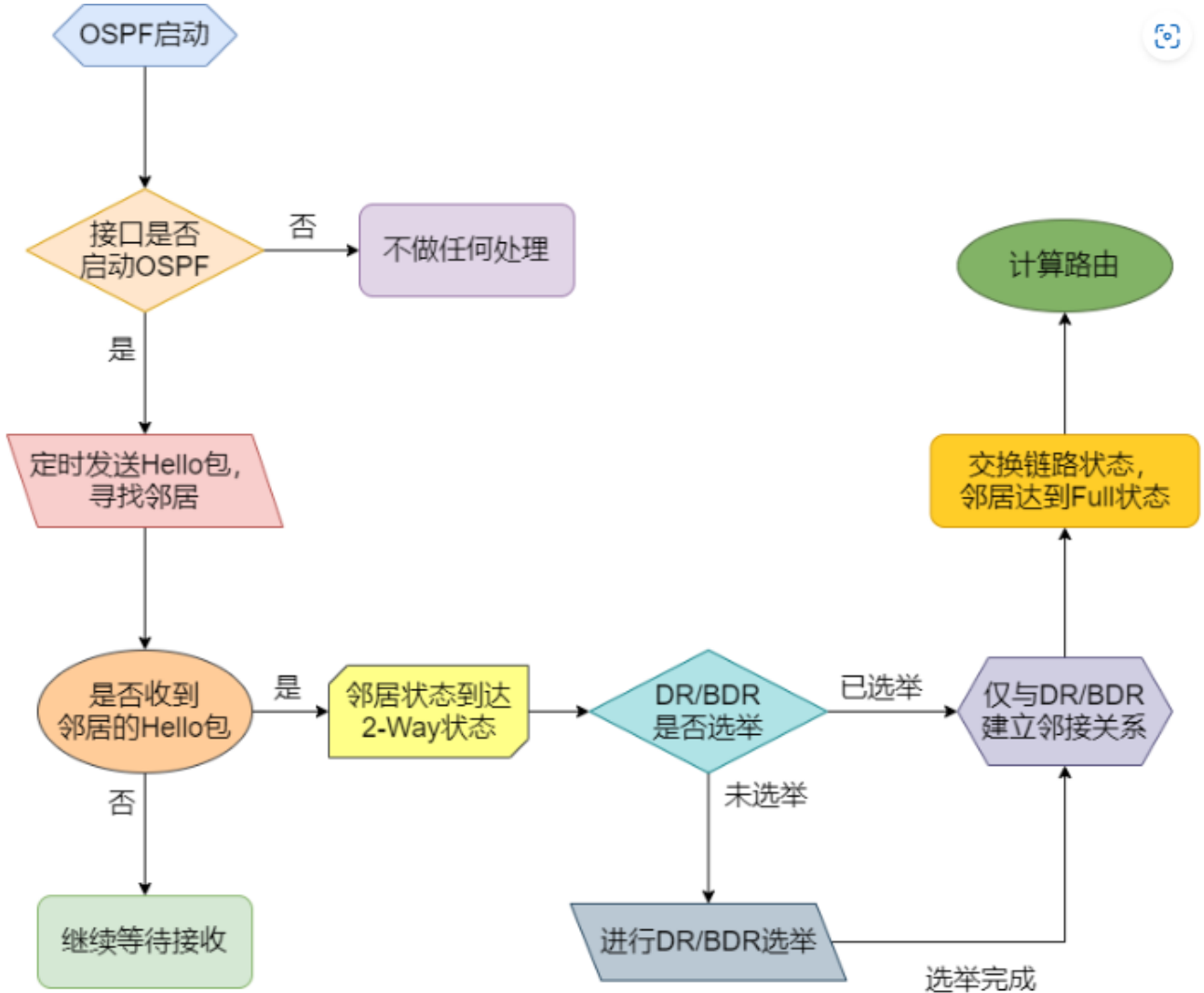
OSPF，是 Open Shortest Path First 的缩写，译为开放最短路径优先。OSPF 是基于链路状态（Link State）的自治系统内部路由协议，用来替代 RIP 协议

与距离矢量协议不同，链路状态协议使用最短路径优先算法（Shortest Path First，SPF）计算和选择

路由。这类路由协议关系网络链路或接口的状态，比如 up、down、IP 地址、掩码、带宽、利用率和时延等。每台路由器将已知的链路状态向其它路由器通告，让网络上每台路由器对网络结构有相同的了解。然后，路由器以此为依据，使用 SPF 算法计算和选择路由。

OSPF 协议使用组播发送协议包，节约资源，又减少对其它网络设备的干扰。

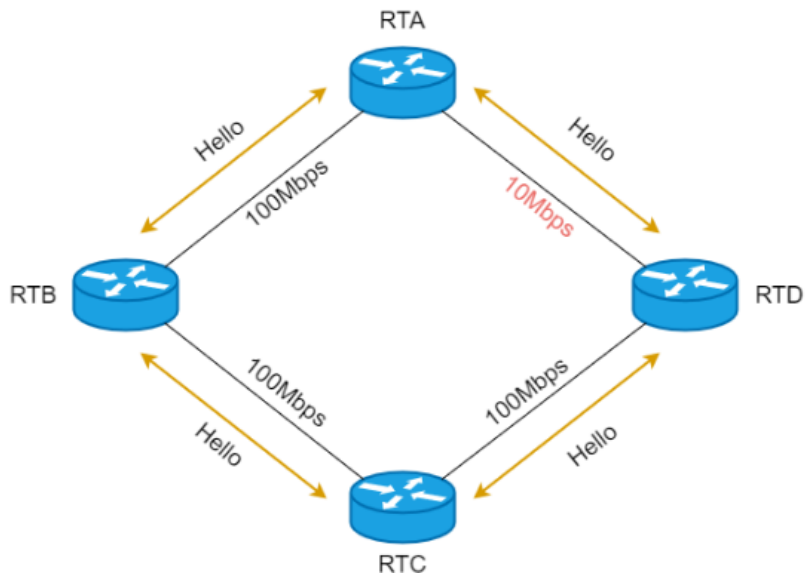
### (5) OSPF 是如何工作的？



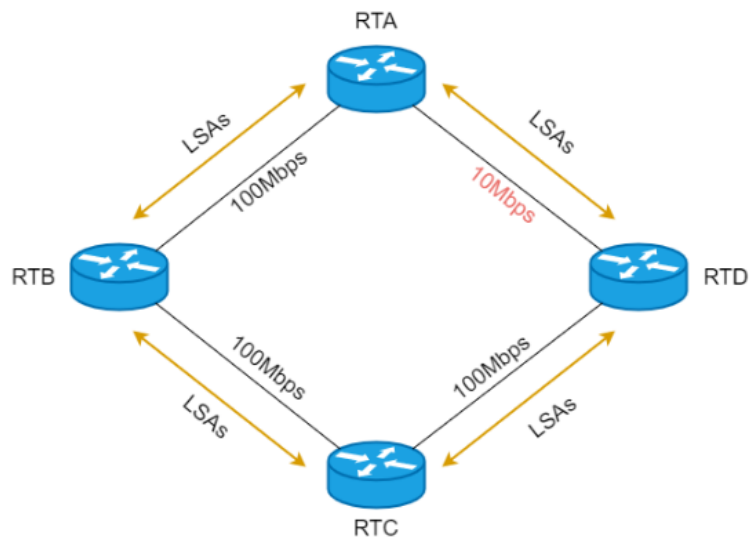
OSPF 将协议包封装在 IP 包中，协议号 89。由于 IP 协议是无连接的，OSPF 定义了一些机制保证协议包安全可靠的传输。

OSPF 协议有四个主要过程：

- ① **OSPF 协议启动后**，先寻找网络中的邻居（Neighbor），也就是通过 Hello 报文确认可以双向通信。

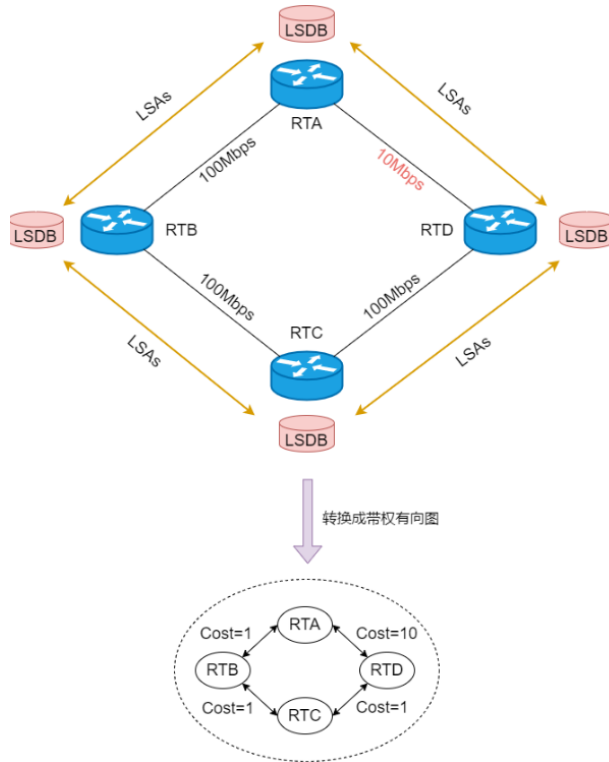


② **建立邻居关系**  
 一部分路由器形成邻居关系后，就开始进行建立邻接关系（Adjacency）。建立了邻居关系的路由器才能互相传递链路状态信息。

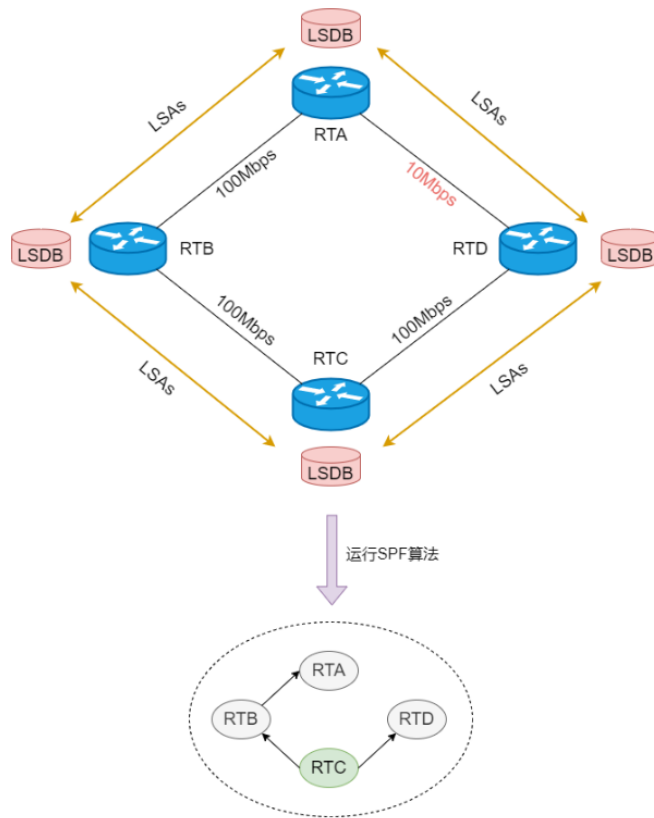


③ **链路状态信息同步**

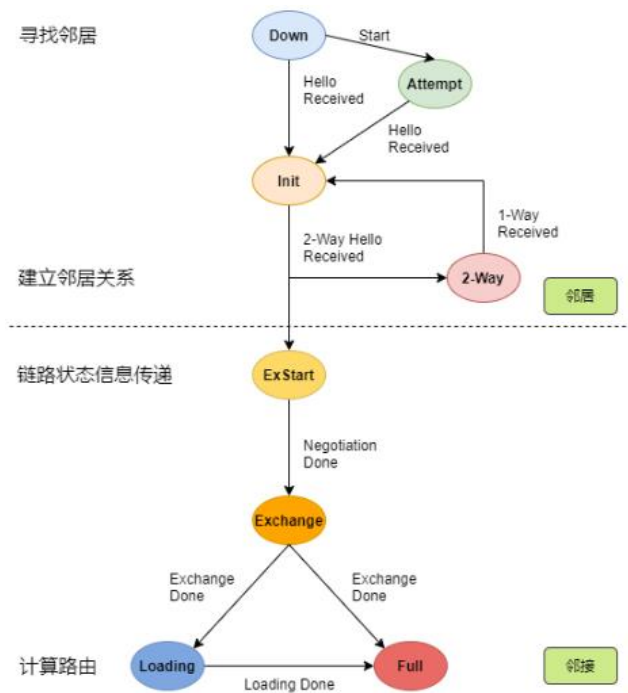
建立邻接关系的 OSPF 路由器在网络中交互 LSA（链路状态通告），最后形成包含网络完整链路状态信息的 LSDB（链路状态数据库）。



LSDB 同步完成后，OSPF 区域内的每个路由器对网络结构有相同的认识，邻居路由器之间形成完全的邻接关系。然后，每台路由器根据 LSDB 的信息使用 SPF（最短路径优先）算法独立计算出路由。



将 OSPF 的四个过程展开来讲，就是一个个邻居状态的切换，不同的邻居状态有不同的行为。



## (6) 什么是 Router-ID ?

Router-ID 用于标识 OSPF 路由器，是一个 32 位的数值，跟 IPv4 地址格式一样。连续的 OSPF 路由器组成的网络叫做 OSPF 域，域内 Router-ID 必须唯一，也就是在同一个域内不允许出现两台相同 Router-ID 的路由器。Router-ID 可以手动设置，也可以自动生成。

## (7) 什么是 AREA? 为什么要引入这个概念?

随着网络的不断扩大，越来越多的路由器加入网络大家庭中，由于每个路由器都需要学习整个网络的拓扑结构，也需要将路由变化通过 LSA 泛洪出去，因此，路由条目过于冗杂，链路计算过于困难，存储路由条目的路由表过于消耗内存空间，等等一系列的问题随之而来，为了解决这个问题，于是就提出了区域的概念；

在 OSPF 中，网络被划分为多个区域 (Areas) 来优化路由。每个区域都是一个自治的系统，它包含了网络的一部分。想象一个大城市，这个城市被划分为不同的区域，比如住宅区、商业区和工业区。在 OSPF 中，这些区域就像网络中的不同区域。每个区域管理自己的路由，这就像每个城市区域有自己的街道和路线。

通过这种方式，OSPF 通过将大型网络划分为更小、更易于管理的区域，来提高路由的效率和稳定性。每个区域只需处理自己的路由，而不是整个网络的路由，从而减少了网络的复杂性和负载。

## 3. 实验分析:

### (1) 动态路由如何设计?

首先选择适合网络需求的路由协议。常见的动态路由协议包括 OSPF、BGP、EIGRP、RIP 等。不同协议有不同的特点和用途，因此选择合适的协议至关重要。本次实验我们选择的是 OSPF 协议。

然后设计 IP 地址方案。设计合理的 IP 地址方案，确保子网和网络地址分配合理。

最后根据所选的协议和 IP 地址设计拓扑图。

### (2) OSPF 的动态路由如何实现?

首先配置 OSPF 协议，在每台路由器上配置 OSPF 协议。具体包括指定 OSPF 进程号、区域 (Area) 和路由器 ID。每个路由器需要加入 OSPF 进程，以便开始交换路由信息。



**其次定义网络接口。**在每个路由器上，需要将参与 OSPF 的网络接口指定为 OSPF 区域的一部分。这通过将接口配置为特定区域来实现。区域是 OSPF 中的基本单元，路由器只能属于一个区域，但一个区域可以包含多个网络。

**之后的步骤都是 OSPF 协议自动完成，不需要人为干预。**

然后邻居关系建立。当 OSPF 协议配置完成后，路由器将尝试与直接相邻的路由器建立邻居关系。这是通过发送 Hello 消息来实现的。路由器需要在同一区域中的接口上配置相同的 Hello 参数才能成功建立邻居关系。

然后交换 LSA 信息。一旦邻居关系建立，路由器开始交换 LSA（链路状态通告）信息。LSA 包含了路由器对自己所知的网络拓扑信息。邻居路由器接受 LSA 并更新它们的路由表。

然后计算最短路径。每个路由器使用 Dijkstra 算法计算最短路径树，以确定到达目标网络的最佳路径。这个路径将被添加到路由表中，以便进行数据包转发。

最后周期性更新。OSPF 协议定期发送 Hello 消息来维护邻居关系，并定期更新 LSA 信息。这有助于确保路由器能够及时了解网络拓扑的变化。

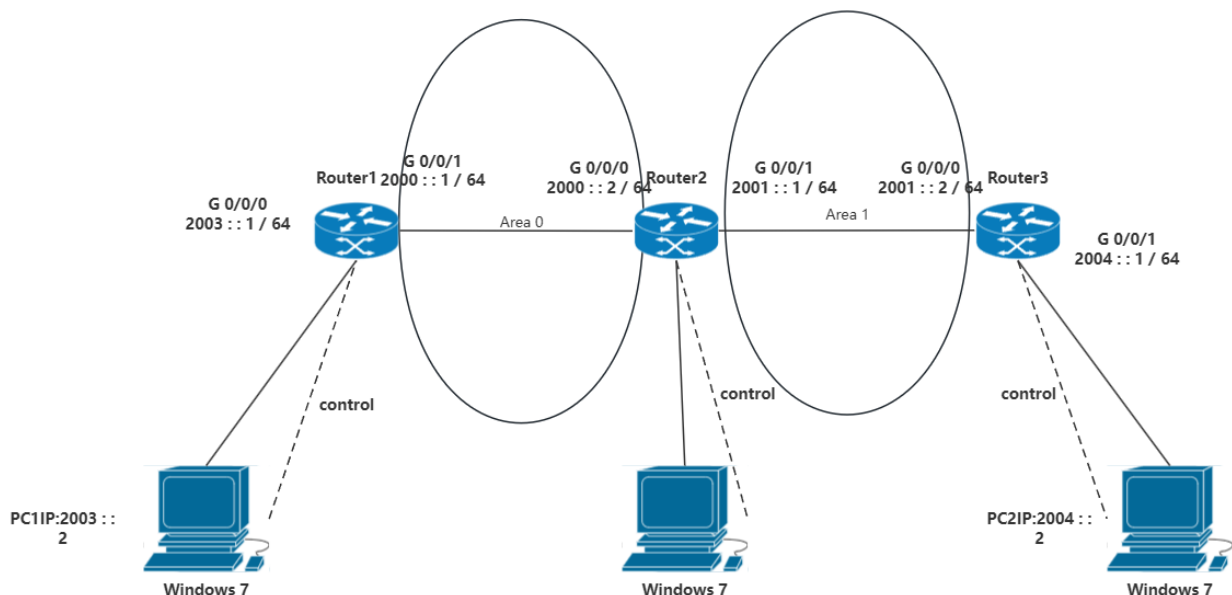
除此之外处理故障和变化。当网络发生故障或拓扑变化时，OSPF 路由器能够快速适应并更新路由表。它们发送通告以通知其他路由器，并重新计算最短路径。

## 4. 实验设计：

首先在实验器材上，我们共需 3 台路由器及 2 台主机完成搭建。

其次是实验步骤：

- ① 首先重中之重根据所需设计 2 个 Area 并通过路由器将他们相连，并画出拓扑图清晰逻辑（很重要）。



- ② 配置 2 台主机的所属的局域网。根据拓扑图为每台主机配置 IP 地址和网关（一旦配错后边一定会出问题）
- ③ 其次配置路由的端口的 IPv6 地址，根据拓扑图
- ④ 配置动态路由信息，包括端口的所属的 Area 和路由器的 routerID
- ⑤ 使用网络工具如 ping 或 traceroute 来测试静态路由是否正确工作。
- ⑥ 向不同网络的计算机发送数据包，验证数据包是否能够正确路由到目标网络。

⑦ 若测试不通，则逐一对每一个子线路进行测试，排查故障点。

## 5. 实验过程：

### (1) 连接物理线路

根据之前设计的网络拓扑图连接物理线路



### (2) 配置主机的 IP 和网关

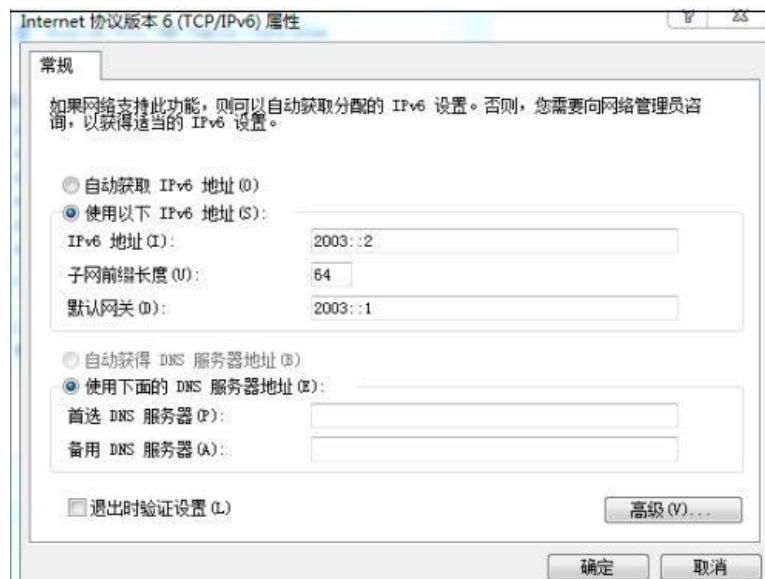
主机	IPv6 地址	子网前缀长度	默认网关
PC1	2003::2	64	2003::1
PC2	2004::2	64	2004::1

#### ① 配置 PC 1 的 IPv6 地址与网关





## ② 设置 PC2 IPv6 地址和默认网关



## (3) 配置路由器接口 IP

### ① Router1 配置接口 GigabitEthernet 0/0/0 开启 IPv6 功能并配置 IP 地址

```
Error:Too many parameters found at '^' position.
[Huawei]interface GigabitEthernet 0/0/0
[Huawei-GigabitEthernet0/0/0]ipv6 enable
[Huawei-GigabitEthernet0/0/0]ipv6 address 2000::1 64
Info: Please enable IPv6 function in system view.
[Huawei-GigabitEthernet0/0/0]quit
[Huawei]ipv6 enable
^
Error:Too many parameters found at '^' position.
[Huawei]int
[Huawei]interface G
[Huawei]interface GigabitEthernet 0/0/0
[Huawei-GigabitEthernet0/0/0]ipv6 enable
[Huawei-GigabitEthernet0/0/0]ipv6 address 2000::1 64
Error: The address has already existed on this interface.
[Huawei-GigabitEthernet0/0/0]quit
[Huawei]ip
```

### ② Router1 配置接口 GigabitEthernetd 0/0/1 开启 IPv6 功能并配置 IP 地址

```

[Huawei]in
[Huawei]inter
[Huawei]interface G
[Huawei]interface GigabitEthernet 0/0/1
[Huawei-GigabitEthernet0/0/1]ipv6 enable
Info: Please enable IPv6 function in system view.
[Huawei-GigabitEthernet0/0/1]quit
[Huawei]ipv6 enable
      ^
Error:Too many parameters found at '^' position.
[Huawei]interface GigabitEthernet 0/0/1
[Huawei-GigabitEthernet0/0/1]ipv6 enable
[Huawei-GigabitEthernet0/0/1]ipv6 address 2003::1 64
Info: Please enable IPv6 function in system view.
[Huawei-GigabitEthernet0/0/1]ipv6 enable
[Huawei-GigabitEthernet0/0/1]ipv6 address 2003::1 64
Error: The address has already existed on this interface.
[Huawei-GigabitEthernet0/0/1]quit
[Huawei]

```

- ③ Router 2 配置接口 GigabitEthernet 0/0/0 开启 IPv6 功能并配置 IP 地址

```

Username:admin
Password:
<Huawei>system-view
Enter system view, return user view with Ctrl+Z.
[Huawei]sysname R2
[R2]it
[R2]int
[R2]interface Gi
[R2]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]ipv6 enable
Info: Please enable IPv6 function in system view.
[R2-GigabitEthernet0/0/0]ipv6 add
[R2-GigabitEthernet0/0/0]ipv6 address 2000::2 64
Info: Please enable IPv6 function in system view.
[R2-GigabitEthernet0/0/0]ipv6 address 2000::2 64
Error: The address has already existed on this interface.
[R2-GigabitEthernet0/0/0]quit

```

- ④ Router 2 配置接口 GigabitEthernet 0/0/1 开启 IPv6 功能并配置 IP 地址

```

[R2]interface GigabitEthernet 0/0/1
[R2-GigabitEthernet0/0/1]ipv6 enable
Info: Please enable IPv6 function in system view.
[R2-GigabitEthernet0/0/1]ipv6 address 2001::1 64
Info: Please enable IPv6 function in system view.
[R2-GigabitEthernet0/0/1]ipv6 address 2001::1 64
Error: The address has already existed on this interface.
[R2-GigabitEthernet0/0/1]quit

```

- ⑤ Router 3 配置接口 GigabitEthernet 0/0/0 开启 IPv6 功能并配置 IP 地址

```

[Huawei]interface GigabitEthernet 0/0/0
[Huawei-GigabitEthernet0/0/0]ipv6 enable
[Huawei-GigabitEthernet0/0/0]ipv6 add
[Huawei-GigabitEthernet0/0/0]ipv6 address 2002::2 64
[Huawei-GigabitEthernet0/0/0]
Dec 29 2023 12:29:20+00:00 Huawei IPv6/2/IF_IPV6CHANGE:OID 1.3.6.1.2.1.55.2.0.1 The stat
us of the IPv6 Interface changed. (IfIndex=3, IfDescr=HUAWEI, AR Series, GigabitEthernet
0/0/0 Interface, IfOperStatus=1, IfAdminStatus=1)
[Huawei-GigabitEthernet0/0/0]
Dec 29 2023 12:29:20+00:00 Huawei %%01IFNET/4/LINK_STATE(l)[0]:The line protocol IPv6 on
the interface GigabitEthernet0/0/0 has entered the UP state.
[Huawei-GigabitEthernet0/0/0]ipv6 address 2001::2 64
[Huawei-GigabitEthernet0/0/0]quit

```

- ⑥ Router 3 配置接口 GigabitEthernet 0/0/1 开启 IPv6 功能并配置 IP 地址

```
[Huawei]interface GigabitEthernet 0/0/1
[Huawei-GigabitEthernet0/0/1]ipv6 enable
[Huawei-GigabitEthernet0/0/1]ipv6 address 2004::1 64
[Huawei-GigabitEthernet0/0/1]interface GigabitEthernet 0/0/1
Dec 29 2023 12:30:35+00:00 Huawei IPV6/2/IF_IPV6CHANGE:OID 1.3.6.1.2.1.55.2.0.1 The status of the IPv6 Interface changed. (IfIndex=4, IfDescr=HUAWEI, AR Series, GigabitEthernet 0/0/1 Interface, IfOperStatus=1, IfAdminStatus=1)
[Huawei-GigabitEthernet0/0/1]quit
Dec 29 2023 12:30:35+00:00 Huawei %%01IFNET/4/LINK_STATE(1)[1]:The line protocol IPv6 on the interface GigabitEthernet0/0/1 has entered the UP state.
[Huawei-GigabitEthernet0/0/1]quit
```

#### (4) 配置 OSPFv3 协议

- ① 配置 Router1 的 RouterID 与 接口的 Area 分配

```
[Huawei]ospfv3 2
[Huawei-ospfv3-2]router-id 10.10.10.10
[Huawei-ospfv3-2]quit
[Huawei]in
[Huawei]inter
[Huawei]interface G
[Huawei]interface GigabitEthernet 0/0/0
[Huawei-GigabitEthernet0/0/0]osp
[Huawei-GigabitEthernet0/0/0]ospfv3 2 area 0
[Huawei-GigabitEthernet0/0/0]quit
[Huawei]os
```

- ② 配置 Router2 的 RouterID 与 接口的 Area 分配

```
[R2]ospfv3 2
[R2-ospfv3-2]router-id 20.20.20.20
[R2-ospfv3-2]quit
[R2]in
[R2]int
[R2]interface Gi
[R2]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]ospfv3 2 area 0
[R2-GigabitEthernet0/0/0]quit
[R2]interface GigabitEthernet 0/0/1
[R2-GigabitEthernet0/0/1]ospfv3 2 area 1
```

- ③ 配置 Router3 的 RouterID 与 接口的 Area 分配

```
[R2]ospfv3 2
[R2-ospfv3-2]router-id 20.20.20.20
[R2-ospfv3-2]quit
[R2]in
[R2]int
[R2]interface Gi
[R2]interface GigabitEthernet 0/0/0
[R2-GigabitEthernet0/0/0]ospfv3 2 area 0
[R2-GigabitEthernet0/0/0]quit
[R2]interface GigabitEthernet 0/0/1
[R2-GigabitEthernet0/0/1]ospfv3 2 area 1
```

- ④ 在 R1 配置直连路由



```
[Huawei]ospfv3 2
[Huawei-ospfv3-2]import
[Huawei-ospfv3-2]import-route di
[Huawei-ospfv3-2]import-route direct
[Huawei-ospfv3-2]quit
```

⑤ 在 R3 配置直连路由

```
[Huawei]ospfv3
[Huawei]ospfv3 2
[Huawei-ospfv3-2]im
[Huawei-ospfv3-2]import-route direct
[Huawei-ospfv3-2]quit
```

## (5) 验证连通性

① 查看邻居建立情况

```
[Huawei]display ipv6 routing-table protocol os
[Huawei]display ipv6 routing-table protocol ospfv3
Public Routing Table : OSPFv3
Summary Count : 1

OSPFv3 Routing Table's Status : < Active >
Summary Count : 0

OSPFv3 Routing Table's Status : < Inactive >
Summary Count : 1

Destination : 2000::: PrefixLength : 64
NextHop : :: Preference : 10
Cost : 1 Protocol : OSPFv3
RelayNextHop : :: TunnelID : 0x0
Interface : GigabitEthernet0/0/0 Flags :
```

```
[R2]display ipv6 routing-table protocol ospfv3
Public Routing Table : OSPFv3
Summary Count : 3

OSPFv3 Routing Table's Status : < Active >
Summary Count : 2

Destination : 2000::: PrefixLength : 64
NextHop : FE80::6A4A:AEFF:FE9D:FD1B Preference : 10
Cost : 2 Protocol : OSPFv3
RelayNextHop : :: TunnelID : 0x0
Interface : GigabitEthernet0/0/1 Flags : D

Destination : 2003::: PrefixLength : 64
NextHop : FE80::6A4A:AEFF:FE9D:FD1B Preference : 150
Cost : 1 Protocol : OSPFv3ASE
RelayNextHop : :: TunnelID : 0x0
Interface : GigabitEthernet0/0/1 Flags : D

OSPFv3 Routing Table's Status : < Inactive >
Summary Count : 1

Destination : 2001::: PrefixLength : 64
NextHop : :: Preference : 10
Cost : 1 Protocol : OSPFv3
RelayNextHop : :: TunnelID : 0x0
Interface : GigabitEthernet0/0/1 Flags :
```

② 查看联通情况

PC1:

Ping Router1 端口 0

```
C:\Users\lenovo>ping 2000::1

正在 Ping 2000::1 具有 32 字节的数据:
来自 2000::1 的回复: 时间=1ms
来自 2000::1 的回复: 时间=1ms
来自 2000::1 的回复: 时间=1ms
来自 2000::1 的回复: 时间=1ms

2000::1 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间<以毫秒为单位>:
    最短 = 1ms, 最长 = 1ms, 平均 = 1ms
```

Ping Router1 端口 1

```
C:\Users\lenovo>ping 2000::1

正在 Ping 2000::1 具有 32 字节的数据:
来自 2000::1 的回复: 时间=1ms
来自 2000::1 的回复: 时间=1ms
来自 2000::1 的回复: 时间=1ms
来自 2000::1 的回复: 时间=1ms

2000::1 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间<以毫秒为单位>:
    最短 = 1ms, 最长 = 1ms, 平均 = 1ms
```

Ping Router2 端口 0

```
C:\Users\lenovo>ping 2000::2

正在 Ping 2000::2 具有 32 字节的数据:
来自 2000::2 的回复: 时间=1ms
来自 2000::2 的回复: 时间=1ms
来自 2000::2 的回复: 时间=1ms
来自 2000::2 的回复: 时间=1ms

2000::2 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
往返行程的估计时间<以毫秒为单位>:
    最短 = 1ms, 最长 = 1ms, 平均 = 1ms
```

Ping Router2 端口 1

```
C:\Users\lenovo>ping 2001::1

正在 Ping 2001::1 具有 32 字节的数据:
请求超时。
来自 2001::1 的回复: 时间=1ms
来自 2001::1 的回复: 时间=1ms
来自 2001::1 的回复: 时间=1ms

2001::1 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 3, 丢失 = 1 (25% 丢失),
往返行程的估计时间<以毫秒为单位>:
    最短 = 1ms, 最长 = 1ms, 平均 = 1ms
```

Ping Router3 端口 0

```
C:\Users\lenovo>ping 2001::2

正在 Ping 2001::2 具有 32 字节的数据:
来自 2001::2 的回复: 时间=1ms
来自 2001::2 的回复: 时间=1ms
来自 2001::2 的回复: 时间=1ms
来自 2001::2 的回复: 时间=1ms

2001::2 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
    往返行程的估计时间(以毫秒为单位):
        最短 = 1ms, 最长 = 1ms, 平均 = 1ms
```

Ping Router3 端口 1

```
C:\Users\lenovo>ping 2004::1

正在 Ping 2004::1 具有 32 字节的数据:
来自 2004::1 的回复: 时间=620ms
来自 2004::1 的回复: 时间=1ms
来自 2004::1 的回复: 时间=1ms
来自 2004::1 的回复: 时间=1ms

2004::1 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
    往返行程的估计时间(以毫秒为单位):
        最短 = 1ms, 最长 = 620ms, 平均 = 155ms
```

## 6. 结论与分析:

### (1) 遇到问题及解决办法:

问题: 尝试连接 4 台路由器然后配置动态路由发现不可行始终联不通。

分析原因: 首先我们需要明白一对概念, 对我们解决问题很有帮助, 就是 DR(Designated Router)和 BDR(Backup Designated Router)

说说为什么要有 DR? 当一个互连链路中, 链接了 N 个 router, 如果彼此都形成邻接关系, 将存在  $N(N-1)/2$  个邻接关系, 其实就是 full-mesh 结构, 大大加大了 LSA 流量发泛洪 (LSA 泛洪机制会在文章后面讨论), DR 的存在, 则让其余所有路由和 DR 交互即可, 作为整个多互联链路中的 LS 集散地。DR 就是相当于 Hub-Spoke 模式的中的 Hub, 所有其他路由器只和 DR (其实还有 BDR) 建立邻接关系, 彼此没有邻接关系, 但是邻居, 只不过邻接状态不是 Full。上面我们说过, hello 报文的 "active nb" 有自己的 rid 时, 就已形成邻居, 但邻接关系会处于 "Two-way" 状态。这多址网络中, 这个邻接状态将一直维持下去。简而言之, DR/BDR 机制将更好的控制完全邻接关系建立和后续 LSA 泛洪流量的处理。

那么为什么要有 BDR? 如果这单个 DR 挂了, 那是不是所有 router 又得重新来来一遍邻居的计算, 选出新的 DR, 这个计算的过程带来的网络失效将是不能容忍的。所以有了 BDR。

BDR 和 DR 一样, 所有路由器都会和 DR 和 DBR 分别建立邻接关系, 而且 DR 和 BDR 也会形成邻接关系, 当 DR 失效时, 也是无影响的。

而我们当时使用四台路由器连接时, 有一台路由器被分配为了 BDR 然后发挥作用, 只是在备用的, 这里还需要更多的知识来判断 DR 和 BDR 机制才能使四台路由器完全工作为 DR, 才能使四台路由器完全工作。



## (2) 实验感悟

本次实验相较之前的实验比较容易，但是花的时间却比之前多（课上没有做完课下又花了时间），思考其原因是因为我们在实验时并不知道 OSPF 动态路由的实现流程是什么，只知道输哪些指令可以让他工作，一旦出现问题，我们就很难找到问题所在，然后就会卡住，在四处像无头苍蝇一样，手足无措，如果我们知道他是如何工作的，他的原理是什么，那么我们会直接找到问题的根源。当然这是学习的必经之路，从手足无措到心里有数是需要时间不断学习和实践的，不可能是一蹴而就的。