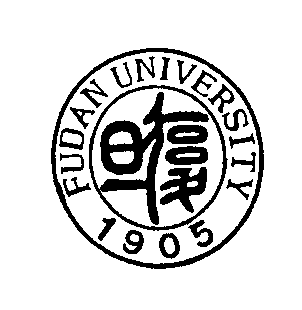
****

**Fudan Switch Simulator Tool**

**研制总结报告**

目录

[1. 引言 3](#_Toc45221882)

[1.1. 编写目的 3](#_Toc45221883)

[1.2. 背景 3](#_Toc45221884)

[1.3. 定义/术语 3](#_Toc45221885)

[1.4. 参考资料 3](#_Toc45221886)

[1.5. 小组分工情况 3](#_Toc45221887)

[2. 软件概述 3](#_Toc45221888)

[2.1. 主要功能/用途 3](#_Toc45221889)

[2.2. 性能要求 4](#_Toc45221890)

[2.3. 开发语言/工具 4](#_Toc45221891)

[2.4. 开发平台 / 运行平台 4](#_Toc45221892)

[2.5. 代码文件组织 4](#_Toc45221893)

[2.6. 软件设计亮点 4](#_Toc45221894)

[2.6.1. 现有实现方法回顾 4](#_Toc45221895)

[3. 软件研制过程 5](#_Toc45221896)

[3.1. 软件研制进度概述 5](#_Toc45221897)

[3.2. 交换机架构与原理调研 6](#_Toc45221898)

[3.3. 软件需求分析与设计 7](#_Toc45221899)

[3.3.1. 信息描述 7](#_Toc45221900)

[3.3.2. 功能描述 7](#_Toc45221901)

[3.3.3. 行为描述 8](#_Toc45221902)

[3.3.4. 测试 8](#_Toc45221903)

[3.4. 软件设计分析 9](#_Toc45221904)

[3.4.1. 整体结构 9](#_Toc45221905)

[3.4.2. 模块结构 9](#_Toc45221906)

[3.5. 编程 9](#_Toc45221907)

[3.5.1. 后端 10](#_Toc45221908)

[3.5.2. 前端 10](#_Toc45221909)

[3.5.3. 前后端对接 11](#_Toc45221910)

[3.6. 测试 11](#_Toc45221911)

[4. 研制成果 11](#_Toc45221912)

[4.1. 文档成果 11](#_Toc45221913)

[4.2. 软件成果 11](#_Toc45221914)

[4.2.1. FudanSwitchSimulatorTool v1.0 12](#_Toc45221915)

[4.2.2. FudanSwitchSimulatorTool v2.0 13](#_Toc45221916)

[4.2.3. FudanSwitchSimulatorTool v3.0 14](#_Toc45221917)

[5. 质量评价 16](#_Toc45221918)

# 引言

## 编写目的

开发本软件系统的目的是对网络交换机的工作过程进行仿真。

## 背景

本项目为复旦大学2020年春季软件工程课程的期末项目。

## 定义/术语

|  |  |
| --- | --- |
| **术语** | **描述** |
| MAC地址 | 一台设备的物理地址，每台设备的MAC地址是该网络中独一无二的。 |
| 冲突域 | 同一个网络上两个设备同时进行传输则会产生冲突，冲突域是会产生冲突的最小范围。 |
| 以太网 | 一种计算机局域网技术。在本系统中均指交换式以太网，与传统以太网相比，交换式以太网中交换机的各个端口隔离了冲突域，从而保证了各个端口的独立带宽。 |
| 以太网帧 | 数据链路层的协议数据单元。 |
| 客户机 | 与交换机进行交互的设备，通过交换机发送消息 |
| 以太网交换机 | 一种基于MAC地址识别，能够完成以太网帧封装与转发功能的网络设备。 |
| 端口 | 连接其他网络设备的接口。 |
| 转发器 | 处理转发逻辑的模块。 |
| DMA | 直接MAC地址表存取（Direct MAC address table Access），提供对MAC地址表的读写接口。 |

## 参考资料

《软件工程》 钱秋乐，清华大学出版社,2007

## 小组分工情况

本软件系统一共由四人开发，分工情况如下：

需求规约说明书：

王子仪：功能描述

刘婧漪：行为描述

李纳川：信息描述

李泳桦：检验标准

设计规约说明书：

王子仪：体系结构设计，人机界面规格说明，出错处理设计

刘婧漪：转发器，数据设计

李纳川：端口，运行设计

李泳桦：引言，DMA模块设计

王子仪：体系结构设计，人机界面规格说明，出错处理设计

编程：

刘婧漪：后端转发器模块

李纳川：后端端口模块

李泳桦：后端DMA模块

王子仪：前端设计与开发，前后端对接，后端客户机模块，代码整合调试

测试报告：

刘婧漪：交换机转发数据帧白盒测试，定期清理MAC表项白盒&黑盒测试

李纳川：端口接受数据帧白盒测试与黑盒测试

李泳桦：交换机转发数据帧黑盒测试、定时清理MAC表项黑盒测试

王子仪：无

研制总结报告

刘婧漪：2.6软件设计亮点，交换机转发器模块，测试总结，质量评价

李纳川：3.2交换机架构与原理调研，3.3软件需求分析与设计，3.5.1.1交换机端口模块概述

李泳桦：软件设计分析，后端DMA模块概述，格式化文档

王子仪：引言，软件概述2,1-2.5,客户机模块概述3.5.1.4,后端设计概述3.5.2,研制成果4

交换机系统演示视频：刘婧漪

# 软件概述

## 主要功能/用途

本软件的主要功能是模拟网络交换机的工作过程，包括数据帧转发，地址学习，地址老化等等，主要用途为辅助教学，可以帮助计算机网络课程学生理解交换机工作过程。

## 性能要求

由于本软件的主要用途为教学工具，因此对性能要求不高，但仍需满足以下基本要求：

1. 能正确模拟交换机工作过程
2. 软件易于操作
3. 软件界面简洁美观
4. 软件运行过程无卡顿
5. 软件有完善的错误处理机制

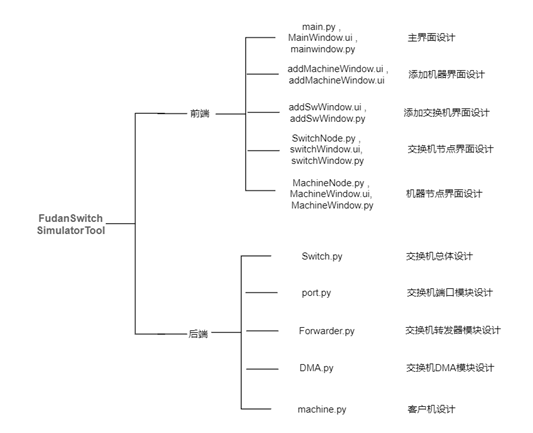
## 开发语言/工具

本软件前端部分使用PyQt-5.0开发，后端部分使用Python-3.6开发。

## 开发平台 / 运行平台

本软件开发平台为 Windows10，运行平台为Windows10。

## 代码文件组织



## 软件设计亮点

### 现有实现方法回顾

在前期，我们调研了现有的交换机的实现方法，发现现有的交换机的实现方法大多是基于switchyard库，这个库封装了很多交换机的实现方法，如net.send\_packet(port,packet)表示从端口port发送数据帧，调用table.update()可以直接更新MAC地址表。但是在我们想要实现的交换机仿真系统是希望可以用作教学的，所以更加关心交换机这些功能的实现原理。因此，本项目不会调用类似的包，而是亲手实现交换机的运作机制。

* + 1. 本项目的亮点

1. 模块化的设计

将交换机根据核心功能分为三个功能部件：接口，转发器，DMA，三个部件对内封装方法，对外提供接口，代码的层次结构清晰，可维护性强。即使改变，那么维护也只是在局部模块，所以维护起来是非常方便和较低成本的。

1. DMA的引入

DMA的设计是受计算机体系结构中DMA(DirectMemory Access)的启发：在外设与存储器之间建立数据通路，使它们可以直接传送数据，而不必通过运算器，从而减轻了运算器的负担。外设与主存的信息交换由DMA控制进行。而在交换机中，将MAC地址表的控制权全权交给DMA控制器，使DMA控制器可以控制MAC地址表无效表项的定期清理减轻了转发器的负担。

1. 面向对象技术的运用

使用了python面向对象技术来写端口，转发器，DMA三个模块。面向对象技术有以下好处：①易维护：易维护采用面向对象思想设计的结构，可读性高，由于继承的存在，即使改变需求，那么维护也只是在局部模块，所以维护起来是非常方便和较低成本的。②质量高：在设计时，可重用现有的，在以前的项目的领域中已被测试过的类使系统满足业务需求并具有较高的质量。③效率高：在软件开发时，根据设计的需要对现实世界的事物进行抽象，产生类。使用这样的方法解决问题，接近于日常生活和自然的思考方式，势必提高软件开发的效率和质量。④易扩展：由于继承、封装、多态的特性，自然设计出高内聚、低耦合的系统结构，使得系统更灵活、更容易扩展，而且成本较低。

1. socket编程的可行性和优点

可行性：我们实现的是交换机仿真系统，只需要模拟交换机的运作，不必拘泥于在链路层上传递消息。因此，即使socket是工作在应用层的也没有关系，我们只是借用它使端口可以接收到数据帧。比起这些底层逻辑，交换机仿真系统更注重的是模拟交换机的功能实现。

优点：socket可以实现进程间通信，让交换机的端口能够接收到外部网络的消息。这一点满足了我们“在端口接收数据帧，转发器和DMA再做相应的处理”的需求，并且使逻辑变得简单。

# 软件研制过程

## 软件研制进度概述

|  |  |  |
| --- | --- | --- |
| **工作内容** | **计划进度** | **实际进度** |
| 调研分析 | 2020.6.01 -2020.6.15 | 2020.6.01-2020.6.10 |
| 形成需求规格说明书 | 2020.6.15-2020.6.25 | 2020.6.11-2020.6.16 |
| 形成设计规格说明书 | 2020.6.26-2020.6.30 | 2020.6.26-2020.6.30 |
| 完成系统编写 | 2020.7.01-2020.7.04 | 2020.7.01-2020.7.04 |
| 完成测试报告 | 2020.7.05-2020.7.07 | 2020.7.05-2020.7.08 |
| 项目总结 | 2020.7.08 | 2020.7.09 |

## 交换机架构与原理调研

我们对于交换机的架构与原理进行了相关调研，并总结为以下四点：

1. 交换机概述：

交换机是一种基于MAC地址识别，能够完成数据帧封装与转发功能的网络设备，工作于数据链路层，其作用是快速高效，准确无误地转发数据帧。

1. 以太网基础知识：

以太网工作在数据链路层，是一种能够使计算机进行相互传递信息的介质，通过以太网帧传递信息。以太网帧的格式多种多样，但大多数以太网帧都包含目的MAC地址、源MAC地址、数据、FCS这几项。数据是网络层传来的数据。FCS是4个字节的帧校验序列，它采用32位CRC循环冗余校验对从"目标MAC地址"字段到"数据"字段的数据进行校验。MAC地址是一台设备的物理地址，并且每台设备的物理地址在该网络中是独一无二的。MAC地址有48位，6个字节，前三个字节是供货商标识（表示是哪个公司），后三个字节是供货商对网卡的唯一编号（序列号）。目的MAC地址与源MAC地址标示着数据帧的接受者与发送者。

1. 交换机架构：

交换机从外形上主要分为盒式交换机和框式交换机，两者部主要功能部件都一样，这里介绍的是框式交换机的架构。

交换机部件包括线卡/接口单板、交换网板、主控单板、背板等。交换机架构发展过程中，先后出现了共享总线交换，环形交换架构，共享内存交换，基于crossbar的交换架构以及CLOS架构，后两种为当前主流架构。

各个架构的性能特点基本可以由下图概括：



交换架构可以根据信元或者数据包在交换网中经历的交换次数分为单级交换架构和多级交换架构，也可以根据交换网和上行处理器交互的交换单元数目分为集中式交换架构和分布式交换架构。

1. 交换机工作原理：

交换机能保证数据帧进行有选择性的转发，而转发的依据就是MAC地址。交换机中有一个表，叫MAC地址表，它是交换机实现数据帧选择性转发的核心。交换机在接收从主机发出的数据帧后，若该数据帧为单播数据帧（即目的地址不为FF:FF:FF:FF:FF:FF的数据帧），便会查找MAC地址表,查看是否有此MAC地址，有则直转给相应的端口，若没有，交换机会把主机发出的数据帧中的源MAC地址与收到这个帧的端口关联起来记录到MAC地址表里面去，然后交换机把该数据帧从除了收到该数据帧的其它所有接口发送出去。MAC地址表中MAC地址的存在时间是有限的，这个时间称为地址表的老化时间。若交换机收到的数据帧为广播数据帧（即目的地址为FF:FF:FF:FF:FF:FF的数据帧），便将该数据帧从除了收到该数据帧的端口的其他端口都进行发送。

此外，交换机有三种交换方式：直通转发（Cut-through）、存储转发（Store-and-forward）、碎片隔离（FragmentFree）。直通转发不进行错误检查直接转发，存储转发对于每一个数据帧都会对FCS校验位进行计算检验，只有正常帧能够被转发给合适的目的端口。碎片隔离（FragmentFree）：这种方式是前两种方式的中和，它只检查接收的帧是否是残帧（长度小于64字节）。

## 软件需求分析与设计

本项目的目标是开发一款交换机仿真软件，能模拟交换机的运行。在软件需求规约中，我们描述了交换机的基本功能需求，如帧转发和地址学习，以及CRC校验，地址老化等额外需求。

### 信息描述

主要的数据对象有：数据帧、端口、交换机、MAC地址表，控制对象则是控制信号transmitCtrl。数据帧输入端口时，根据控制信号transmitCtrl的取值有3个不同的流向：transmitCtrl=存储转发，数据帧先后输入接受缓冲队列与残帧检测模块后端口输出完整的帧。transmitCtrl=碎片，数据帧输入残帧检测模块后端口输出完整的帧。transmitCtrl=直通，数据帧直接输出端口。交换机内部，数据帧输入解析帧模块，输出数据帧与源/目的地址，数据帧与源/目的地址输入过滤帧模块，输出合法的数据帧与源/目的地址，数据帧随后输入转发帧模块，数据帧的源/目的地址输入DMA，调用DMA的add接口向MAC地址表中添加表项，调用query接口输出端口号到转发帧模块，转发帧模块根据DMA返回的端口号输出数据帧发送到对应端口。

### 功能描述

交换机的主要功能包括接收数据帧，CRC校验，残帧检查，帧解析，过滤非法数据帧，转发数据帧，地址学习，地址信息维护，发送数据帧和避免回路。

* 接受数据帧：交换机的端口会实时监听网络中的数据帧，并根据控制信号transmitCtrl执行不同功能。
* CRC校验：对数据帧进行CRC（循环冗余码校验）检查，对错误帧进行处理。
* 残帧检查：检查数据帧的长度是否够64个字节，如果小于64字节，则说明为假帧，丢弃。
* 帧解析：对一个数据帧进行解析，获取数据帧的类型，源MAC地址和目的MAC地址。
* 过滤非法帧：若数据帧的源MAC地址和目的MAC地址网段相同，则该数据帧非法，丢弃。
* 转发数据帧：如果数据帧的目的MAC地址是广播地址或者组播地址，则向除数据帧来的端口外的所有端口转发；如果数据帧的目的地址是单播地址，且此地址不在MAC地址表中，那么向除数据帧来的端口外的所有的端口转发；如果数据帧的目的地址在MAC地址表中，那么就根据地址表转发到相应的端口。
* 地址学习：若解析到的数据帧的源MAC地址不在MAC地址表中，则将该源MAC地址及其对应的端口号加入MAC地址表中。
* 地址信息维护：交换机会为MAC地址表的每一个表项设定一个自动老化时间（如 300s），从一个地址记录加入地址表以后开始计时，如果在老化时间内各端口未收到源地址为该MAC地址的帧，交换机将自动把该MAC地址从地址表中清除。
* 发送数据帧：当某个转发帧从交换机内部到达端口时，交换机会将其加入发送缓冲区。交换机的端口会实时检查端口的发送缓冲区，若缓冲区不为空，则将取出缓冲区顶部的数据帧，并将其发送出去。

### 行为描述

* 数据结构：MAC地址表采用二级索引表，外层索引是供货商号，内层索引是序列号，每个表项还包含了老化时间。端口采用端口缓存队列调度数据帧。一台交换机的端口号是固定的，因此可以用一个数组记录端口。此外，解析帧，过滤帧，转发帧这三个转发器是交换机的核心部分，构成流水线结构。同时在交换机中引入DMA直接对MAC地址表进行操作，对外提供接口add, query，使MAC地址表对于交换机的其他部分透明。
* 具体行为描述：
  + DMA的转发帧模块遍历MAC地址表，查询目标地址项是否存在，若存在，返回该MAC地址映射到的端口号, 否则返回空值。
  + DMA的解析帧模块向MAC地址表中添加表项(MAC,n), 若已有该表项，则将它覆盖，同时将它的老化时间置零。端口的CRC校验模块对数据帧进行循环冗余码校验检查，并丢弃错误帧。
  + 端口的残帧检查模块检查帧长是否大于等于64字节，若不满足条件，端口将该数据帧丢弃。
  + 转发器的解析帧模块按照数据帧格式，解析出帧中的源MAC地址，目的MAC地址。
  + 转发器的转发帧模块根据目的MAC地址进行相应的转发：如果是广播地址，则向除接收端口外所有端口转发；如果目的地址如果不是广播地址，则查找MAC地址表，找到了，通过目的MAC地址对应的端口转发；找不到，通过除到来端口外，端口数组中记录的所有端口转发。
  + 转发器的过滤帧模块会判断数据帧的源MAC地址和目的MAC地址所处的网段是否相同，若相同，则丢弃该数据帧。
  + DMA的控制模块在对外提供的接口被调用或系统中的定时器经过了设定的时间间隔时执行相应的操作：
    - add接口被调用：add(mac,n)，若表中有相同项，说明MAC地址表中已有该表项，将它的老化时间置零。否则向MAC地址表中添加表项(MAC,n)。
    - query接口被调用：query(mac)，若表中能找到该MAC地址，返回对应的端口号，否则返回空值。
    - 系统中的定时器经过了设定的时间间隔：DMA以深度优先的方式遍历MAC地址表，找到老化时间超过阈值的表项，清除。若某一外层索引对应的所有表项都被删除，那么也将它从外层索引表中删除。

### 测试

* 功能测试：测试交换机接受发送数据帧等功能。
* 性能测试：测试交换机仿真软件的性能是否达标：软件的启动时间应该不超过10秒，需要能支持至少100个节点的交换网络，按键的响应时间应该不超过3秒，数据帧的最大往返时间应该不超过1秒。运行过程中不应该出现卡顿或卡死的情况，占用内存不应该超过1G。退出软件时应该及时回收用过的内存，避免内存爆炸。
* 可靠性测试：测试交换机在长期运行时的可靠性，期望在试过程中，合法数据包的丢失率不超过0.01%。如果发生丢包等异常，软件能及时处理异常，恢复正常的运作。
* 易用性测试：通过抽取志愿者试用软件，预测交换机仿真软件投入市场后的用户反馈。
* 兼容性测试：观察软件在不同的操作系统平台以及同一操作系统的不同版本上的运行效果。

## 软件设计分析

### 整体结构

经设计，本交换机仿真软件支持用户创建客户机和交换机两种机型并将它们组装成交换网络，研制的重心为交换机系统的设计。交换机系统被划分为三个模块：

* 端口模块：负责接收和发送数据帧
* 转发器模块：负责解析、过滤、转发数据帧
* DMA模块：负责存储、查询MAC地址和交换机端口的对应关系

交换机系统的工作流程为：数据帧通过端口模块进入数据帧，随后转发器模块解析帧的结构、调用DMA模块查询需要转发的端口、并将数据帧转交给对应的端口进行发送。

### 模块结构

端口模块

端口模块实时监听并接收网络中的数据帧，根据控制信号transmitCtrl的不同来选择存储转发、碎片隔离、直通操作之一。端口外设置接收缓冲队列和发送缓冲队列，一个端口在同一时刻只能同时发送或接收一个数据帧。该模块提供接收和发送数据帧的调用接口，供上层模块调用。

转发器模块

转发器模块接收来自端口的数据帧，利用其子模块来解析、过滤、转发数据帧，实行流水化的处理。其中，解析帧模块解析数据帧的源MAC地址和目的MAC地址；过滤帧模块过滤源MAC地址和目的MAC地址相同的数据帧；转发帧模块根据MAC地址表记录的信息，在相应的端口转发数据帧。该模块是交换机系统中的核心模块，提供三个子模块接口，供上层模块调用。

DMA模块

DMA模块直接与存储器相连，负责管理MAC地址的高效存储，支持添加MAC表项、查询目的MAC地址对应的端口号、定时清除老化项的功能。该模块提供添加表项和查询端口号两个接口，供上层模块调用。

## 编程

我们使用了python + pyqt进行开发，其中使用python开发软件后端，使用pyqt开发软件前端，并使用python将前后端对接。

### 后端

* + - 1. 端口模块

（待填写）

* + - 1. 转发器模块

交换机转发器模块封装了三个方法：帧解析，帧过滤和帧转发三者形成流水线结构。

帧解析功能实现如下：通过struct.unpack解析出数据帧的源MAC地址，目的MAC地址，帧类型，FCS和数据部分，并用utf-8解码 。

帧过滤功能实现如下：判断解析出来的源MAC地址和目的MAC地址是否相等，相等则返回False，不转发数据帧，否则返回True，指示可以进行后续的操作。

帧转发功能实现如下：调用帧解析方法，解析出数据帧的源MAC地址，目的MAC地址，帧类型，FCS和数据部分，并用utf-8解码 ；调用帧过滤方法，判断是否进行转发。判断目的MAC地址是否是组播/广播地址，如果是，直接返回除到来端口外的所有端口，表示所有端口都会转发该数据帧；如果不是，进一步判断目的MAC地址是否在MAC地址表中存在表项，如果存在，则返回目的MAC地址映射到的端口，表示只有该端口会转发数据帧。如果不存在，则返回除到来端口外的所有端口，表示所有端口都会转发该数据帧。

* + - 1. DMA模块

DMA模块可以分为存储模块和控制模块两个子模块。

在存储模块中，一个私有的Entry类被用来记录表项结构，包括present标志、MAC地址、端口号和老化时间。MAC表在存储模块中以一个Entry的列表来表示，列表的大小由参数指定，代表MAC表能存储的表项数量。此外，设置mac\_addr\_t.tab文件作为MAC表的备份，使之与内存中的表同步，便于实现断电恢复机制。

在控制模块中，添加表项add和查询端口号query两个接口函数被提供。由于最终实现系统中支持的端口数较少，网络不会太大，所以采用遍历的方法实现了查询就足够了，而不必采用之前设计的二级索引。在添加表项时也会进行查询，检查表中是否已经有该地址对应的项。若有，则更新表项中的属性；否则，重新遍历此表，在表中找到第一个空槽填上该项；若没有空槽，则采用LRU策略替换掉一个旧项。

模块中设置计时器AgingChecker，它继承了thread库中的Timer计时器，通过重写其run方法实现了循环计时。每隔固定时间，它将会运行DMA的私有函数\_\_checker\_\_，检查表中各项的老化时间，并清除掉老化项。

* + - 1. 客户机模块

每台客户机拥有独属于自己的Mac地址，用于标识客户机，同时每台客户机可以连接到某台交换机的一个端口。客户机模块的主要功能有两个：发送数据帧以及接收数据帧。

客户机发送数据帧功能实现如下：建立socket套接字，与目标端口的套接字建立连接，然后按照以太网数据帧格式对发送数据进行打包，生成数据帧（若发送数据的长度超过1500字节，需要将发送数据分段打包），然后通过套接字发送数据帧。

客户机接收数据帧功能实现如下：通过之前建立好的套接字接收数据帧，解析数据帧的目的地址，若目的地址为以下三种情况中的一种：组播地址 / 广播地址 /客户机Mac地址，则接收该数据帧，并发送响应，告知交换机自己的Mac地址。由于接收功能需要一直开启，因此为每台客户机都设置了一个专门的线程负责接收功能。

### 前端

* + - 1. 前端界面

前端界面使用PyQt进行设计，单独设计各个子界面，然后通过PyQt的信号功能将各个界面链接起来，同时也将按键与相应功能链接起来，包括添加机器节点，添加交换机节点，机器节点发送数据帧等等。

此外，还通过python代码对界面进行了美化，包括设置背景图片，调节字体，设置交换机/客户机节点图标等等。

具体的界面设计见下文 4.2软件成果展示。

### 前后端对接

由于后端交换机及客户机已经模块化，因此前后端对接较为容易。

令前端的每个交换机节点都拥有一个交换机对象，其各个端口初始处于不活跃状态，直到有节点连接到某端口，才会激活该端口，为该端口分配两个线程：一个负责接收工作，一个负责发送工作。

另一方面，令前端的每个客户机节点都拥有一个客户机对象，在其加入到拓扑网络中时，根据其连接到的端口，设置其套接字连接。并启动其负责接收的线程。

此外，前端还设置了查看数据流功能：点击交换机/机器节点图标，可以看到交换机/机器内部数据流/Mac地址表最新情况，这是通过将窗口对象作为参数传递给交换机类/客户积类实现的。

## 测试

我们选取了交换机实现的核心方法分别进行黑盒测试和白盒测试，将UI测试融入这两种测试中；交换机的方法有：端口接收数据帧；交换机转发数据帧；MAC地址表定期清理。在白盒测试中，我们覆盖了分支的条件，检查每一条执行路径的输出是否正确；在黑盒测试中，我们考虑了一些边界值和错误猜测，如自己发送给自己的情况，长度不满64字节的残帧，长数据。结果是，我们的交换机仿真系统可以通过大部分的测试，功能较为完备；但一些长数据帧的测试未能通过，说明我们的交换机仿真系统在这方面还有待改善。

# 研制成果

## 文档成果

本项目最终形成的文档包括：

1. 交换机架构与原理调研分析报告
2. FudanSwitchSimulatorTool需求规格说明书
3. FudanSwitchSimulatorTool设计规格说明书
4. FudanSwitchSimulatorTool测试报告
5. FudanSwitchSimulatorTool研制总结报告。

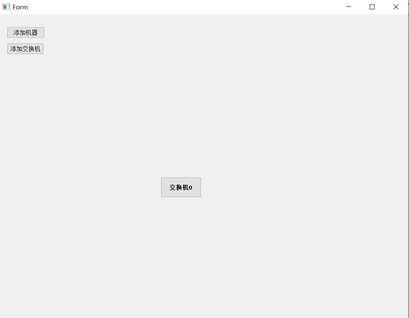
## 软件成果

FudanSwitchSimulatorTool软件的开发一共经历了三个版本，版本号分别为v1.0，v2.0，v3.0，其中v3.0为最终版本。

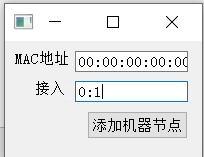
### FudanSwitchSimulatorTool v1.0

最初始的版本，仅完成了基本的消息转发功能，界面较为简陋。

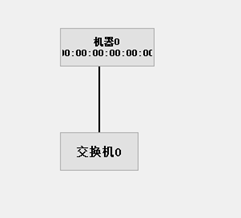
其初始界面如下：



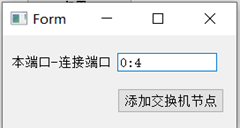
点击左上方“添加机器”按钮会弹出如下图所示界面：



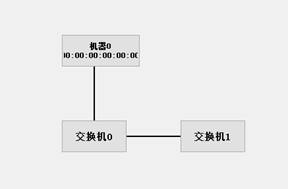
输入客户机的Mac地址以及连接到的端口，然后点击“添加机器节点”即可在主页面上添加一个新的客户机，如下图所示：



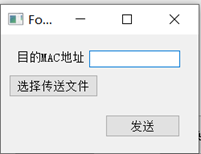
点击左上方“添加交换机：按钮会弹出如下图所示界面：



输入连接到的端口，然后点击“添加机器节点”即可在主页面上添加一个新的交换机，如下图所示：



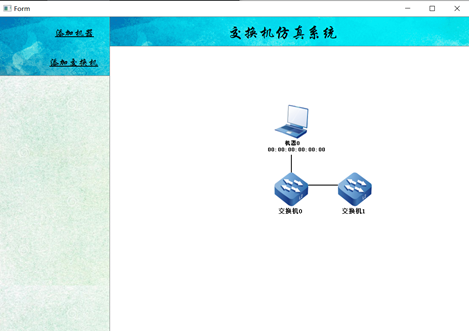
此外，点击机器节点图标，可以弹出发送窗口，如下图所示：



输入目的Mac地址，选择传输文件，点击发送即可将数据帧通过交换机发送到目的地址。

### FudanSwitchSimulatorTool v2.0

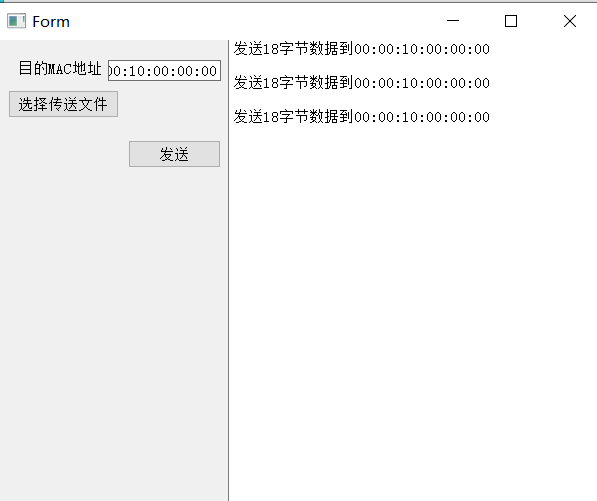
本版本中，对主界面进行了美化，效果如下：



### FudanSwitchSimulatorTool v3.0

本版本中，增加了数据流显示功能，点击交换机/客户机图标，可以查看对应节点的数据流。

点击客户机图标，会跳出如下所示界面，左侧为发送数据帧功能，右侧则显示该客户机节点的数据流。

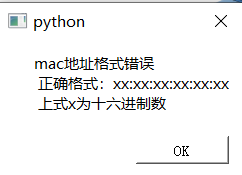


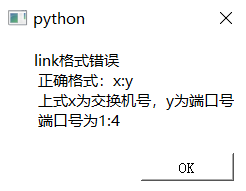
点击交换机图标，会跳出如下所示界面，该交换机的每个端口都有一个对应的消息窗口，显示该窗口的数据流。界面下方显示该交换机的MAC地址表。



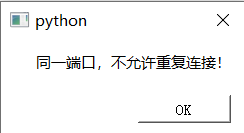
此外，在版本三中还加入了错误处理机制，包括：

规范输入：若输入的Mac地址格式，端口格式错误，会弹出提示窗口，如下图所示：





防止重复连接：若某个端口有重复连接的情况，会弹出相应提示窗口（如下图所示），并禁止此次操作。



# 质量评价

按照软件质量国家标准GB-T8566--2001G，软件质量可以用下列特征来评价：

1. **功能特征：与一组功能及其指定性质有关的一组属性，这里的功能是满足明确或隐含的需求的那些功能。**

在我们的交换机仿真系统中，预期实现的功能都已实现：端口可以通过三种方式接收到来的数据帧，并做相应的处理；交换机能够正确地转发数据帧；MAC地址表会定期清理无效的表项。

1. **可靠特征：在规定的一段时间和条件下，与软件维持其性能水平的能力有关的一组属性。**

本交换机仿真系统保证了一定的容错性，加入了一些错误处理机制，比如会判断MAC地址的合法性，判断端口是否重复连接，判断客户机是否重复连接等。当然，因为时间的原因，我们只进行了初步的错误处理，挑选了最重要的部分，在后续使用中如果还出现了其他错误，需要根据需求加入更多的错误处理方法。该交换机仿真系统有较好的易恢复性，MAC地址表设置了备份，这样如果程序异常中断的话重启还能恢复以前表里的内容

1. **易用特征：由一组规定或潜在的用户为使用软件所需作的努力和所作的评价有关的一组属性。**

本交换机仿真系统有图形界面，用户可以自行添加机器，交换机，指定机器和交换机如何相连，并可以通过点击机器获取机器发送消息的信息，通过点击交换机获取交换机的端口信息和MAC地址表的信息，十分用户友好。因此，本交换机仿真系统易用于教学，可以分步演示交换机的每一种功能，并且，每一步操作都有对应的文字解释。

1. **效率特征：与在规定条件下软件的性能水平与所使用资源量之间关系有关的一组属性。**

由于本交换机仿真系统是基于socket编程实现的，开启的线程较多，因此，响应时间较长。然而，socket编程也有其优越性，比如占用内存少，切换简单，CPU利用率高，可以共享进程数据，提高了资源利用率。

1. **可维护特征：与进行指定的修改所需的努力有关的一组属性。**

因为本交换机仿真系统的代码编写部分高度模块化：交换机被细分成了三个功能部分，每一部分封装了它的方法，对外提供接口；前端的界面也是分别设计的，相互独立，因此，本交换机仿真系统的可维护性好。

1. **可移植特征：与软件从一个环境转移到另一个环境的能力有关的一组属性。**

经过测试，本交换机仿真系统可以在windows os和 MAC os上通过源码编译运行，具有可移植性。

A 附录