

基于宽带柔性转发器的高速交换技术研究

作者姓名 李彩萍 学校导师姓名、职称 田红心副教授

领 域 电子与通信工程 企业导师姓名、职称 王文军高工

申请学位类别 工程硕士 提交学位论文日期 2014 年 11 月

学校代码 10701

学 号 1201120480

分 类 号 TN92

密 级 公开

西安电子科技大学

硕士学位论文

基于宽带柔性转发器的高速交换技术研究

作者姓名：李彩萍

领 域：电子与通信工程

学位类别：工程硕士

学校导师姓名、职称：田红心副教授

企业导师姓名、职称：王文军高级工程师

提交日期：2014 年 11 月

High-speed Switching Technology Based on Broadband Flexible Transponder

A thesis submitted to
XIDIAN UNIVERSITY
in partial fulfillment of the requirements
for the degree of Master
in Electronics and Communication Engineering

By
Li Caiping
Supervisor: Tian Hongxin Wang Wenjun

November 2014

西安电子科技大学 学位论文独创性（或创新性）声明

秉承学校严谨的学风和优良的科学道德，本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果；也不包含为获得西安电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文若有不实之处，本人承担一切法律责任。

本人签名：_____ 日 期：_____

西安电子科技大学 关于论文使用授权的说明

本人完全了解西安电子科技大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属于西安电子科技大学。学校有权保留送交论文的复印件，允许查阅、借阅论文；学校可以公布论文的全部或部分内容，允许采用影印、缩印或其它复制手段保存论文。同时本人保证，获得学位后结合学位论文研究成果撰写的文章，署名单位为西安电子科技大学。

保密的学位论文在____年解密后适用本授权书。

本人签名：_____ 导师签名：_____

日 期：_____ 日 期：_____

摘要

宽带柔性转发器是一种具有星上处理能力的半透明转发器，采用基于非均匀滤波器组的数字信道化处理技术，兼具传统透明转发器和再生式转发器的优点，既具有灵活可靠的特点，又可以支持较小粒度的交换，还规避了物理层信号体制的约束，增加了系统容量，满足可变带宽业务的需求，是未来星载转发器的主要发展方向之一。多波束信道化路由交换技术是宽带柔性转发器具体实现过程中的关键技术之一。本文主要研究了基于宽带柔性转发器的高速交换技术，具体内容如下：

1、确定了交换方案为 T-S-T 交换网络，将 T-S-T 交换网络抽象为矩阵模型，T 交换抽象为矩阵行内变换，S 交换抽象为矩阵列内变换，调度问题抽象为寻找合适的中间矩阵。

2、研究了基于矩阵模型的程控电路交换网络的调度算法，在高冲突值行优先排列算法和列优先排列算法的基础上提出了一种基于列优先排列的改进算法，通过柱状图比较了三种算法的比较次数和交换次数，仿真结果表明，改进算法在时间复杂度上优于高冲突值行优先排列算法和列优先排列算法，但是空间复杂度高于其他两种算法。

3、设计了 T-S-T 交换网络的交换帧结构，在此基础上提出了基于 FDMA 多址接入方式的程控交换方案，实现了单播、组播和广播功能。

4、在基于 FDMA 多址接入方式的程控交换方案设计的基础上，给出了各个模块的 FPGA 设计和仿真，实现了单播、组播和广播功能，通过合理调整信号量化比特数和子信道数，实现了芯片容量的利用率最大化。

关键词：宽带柔性转发器，程控电路交换，调度算法，FPGA 设计

论文类型：应用基础技术

ABSTRACT

Broadband flexible transponder, which is capable of on-board processing, is a kind of semitransparent transponder. The digital channelized processing technology based on the non-uniform filter bank is applied to the transponder, thus it has the advantages of both traditional transparent transponder and regenerative repeater. Not only is it flexible and reliable, but also it can support the exchange of small particle size and avoid the constraints on signal system in the physical layer. Owing to the above advantages of broadband flexible transponder, the system capacity is increased, and what's more the demand for variable bandwidth is satisfied. For such reasons, broadband flexible transponder is expected to be one of the main development directions of onboard transponder in the future. Multi-beam channelized routing and switching technology is a key technology in the implementation process of broadband flexible transponder. The thesis focuses on the high-speed switching technology based on broadband flexible transponder, and the main contributions are as follows:

1. The T-S-T switching network is prescribed as the switching scheme and the matrix model is proposed. T switching is abstracted as the matrix inline transformation, S switching is abstracted as the matrix column transformation, and the scheduling algorithm in fact is to find a suitable intermediate matrix.
2. The scheduling algorithm for T-S-T switching network based on matrix model is discussed. Based on the line prioritize algorithm of highly conflicted value and column prioritize algorithm, an improved algorithm based on column prioritize arrangement is proposed. The number of comparison and that of switching of the three algorithms are compared in the histogram. The simulation results show that the improved algorithms performs better than in the time complexity than the line prioritize algorithm of highly conflicted value and column prioritize algorithm ,but worse in the space complexity.
3. The T-S-T switching frame structure is designed. And the program-controlled switching scheme based on FDMA is proposed so that the functions such as unicast, multicast , and broadcast are realized.

4. The FPGA design and simulation of each module is completed. Unicast, multicast and broadcast are accomplished, and the utilization of the chip capacity is maximized by adjusting the bit number of signal quantization and the number of sub-channels.

Keywords: Broadband flexible transponder, Program-controlled, Circuit Switching, Scheduling Algorithm, FPGA Design

Type of Dissertation: Applied Basic Technology

插图索引

图 2.1 基于柔性转发器的卫星通信系统框图	7
图 2.2 信道化器原理图	8
图 2.3 交换模块简单示意图	8
图 2.4 T 型交换器构成原理图	10
图 2.5 S 型交换器构成原理图	11
图 2.6 T-S-T 三级交换网络的工作原理	13
图 2.7 T-S-T 交换网络的矩阵模型	15
图 3.1 改进算法流程图	21
图 3.2 高冲突值行优先排列算法平均比较次数统计图	23
图 3.3 高冲突值行优先排列算法平均交换次数统计图	24
图 3.4 列优先排列算法平均比较次数统计图	24
图 3.5 列优先排列算法平均交换次数统计图	25
图 3.6 改进算法平均比较次数统计图	25
图 3.7 改进算法平均交换次数统计图	26
图 3.8 三种算法平均比较次数统计图	26
图 3.9 三种算法平均交换次数统计图	27
图 4.1 交换帧结构	32
图 4.2 端口内不同子信道之间的信号交换	35
图 4.3 不同端口之间的子信道信号交换	36
图 4.4 星上交换网络结构框图	37
图 4.5 交换网络的接续原理	37
图 4.6 端口内广播实现方式原理图	39
图 4.7 端口间组播实现方式原理图	40
图 5.1 串行转换模块图	42
图 5.2 时钟信号	43
图 5.3 输出信号	43
图 5.4 端口内子信道间交换的内部实现图	44
图 5.5 端口内子信道之间样点交换的仿真结果图	44
图 5.6 端口间交换模块的顶层模块实现图	45
图 5.7 端口间交换模块的内部实现图	46

图 5.8 s_switch 模块的内部实现图	46
图 5.9 端口间交换模块的仿真结果图.....	47
图 5.10 16*16 的模块资源占用图.....	47
图 5.11 8*8 的模块资源占用图	48
图 5.12 总体仿真结果图.....	48
图 5.13 初级 T1 接线器的输入输出结果图	48
图 5.14 初级 T8 接线器的输入输出结果图	49
图 5.15 S 交换器 1 的输入输出结果图	49
图 5.16 S 交换器 8 的输入输出结果图	49
图 5.17 次级接线器 T1 的输入输出结果图	49
图 5.18 次级接线器 T8 的输入输出结果图	50
图 5.19 基于组播功能的仿真结果图.....	50
图 5.20 基于广播功能的仿真结果图.....	51

表格索引

表 3.1 高冲突值行优先排列算法性能统计表	23
表 3.2 非回溯列优先排列算法性能统计表	24
表 3.3 改进算法性能统计表	25
表 4.1 有 N 个波束的 SS-TDMA 系统交换矩阵表	30
表 4.2 容量交换矩阵表	31
表 4.3 连接状态矩阵表	31
表 4.4 传输业务参数设置	32
表 5.1 m 和 n 取不同值时的芯片 IOBS 值	48

符号对照表

符号	符号名称
$h(n)$	实对称 FIR 滤波器
$g^2(n)$	低阶双通道滤波器
M	内插值
$y(n)$	原始信号
$x_{p-1}(n)$	重构信号
$H_{N-1}(z)$	非均匀滤波器组
T	时间交换器
S	空间交换器
n	存储器单元个数
c	每个存储单元比特数
TS_n	第那个时隙
A、B	用户
CM	控制存储器
SM	数据存储器
\hat{i}	中间信道
F	复用线上时隙数
mod	取余数符号
$n*m$	表示 n 行 m 列矩阵
in_port	输入矩阵
out_port	输出矩阵
middle_1	输入矩阵经过行内变换得到的中间矩阵 1
middle_2	中间矩阵 1 经过列内变换得到的中间矩阵 2
$i、j$	矩阵行号
v	middle_2 矩阵的前 $i-1$ 行元素在 in_port 矩阵的所在行值
h	out_port 矩阵第 i 行元素在 in_port 矩阵中的所在行值
$vRepeat[j]$	$v[j]$ ($j=i-1$) 中的元素在 h 中重复出现的次数之和
$hRepeat$	h 中的元素在 $v[j]$ ($j=i-1$) 中重复出现的次数之和
n	矩阵阶数
N	波束数

$i、j$	波束编号、信道编号
$m_{i,j}$	表示第 i 个波束覆盖区发送到第 j 个波束覆盖区的业务量
S_i	第 i 个波束覆盖区一帧内发送给自身及其他波束覆盖区的业务量总和
S_j	第 j 个波束覆盖区一帧内接收的来自自身及其他覆盖区的业务量总和
D	交换矩阵
D_i	状态矩阵
η	编排效率
T	帧长
k	模式数
$O(N^s)$	运算复杂度
$c_{i,j}$	表示从 i 信道到 j 信道的通信容量
clk_a	读时钟
clk_b	写时钟

缩略语对照表

缩略语	英文全称	中文对照
GEO	Geostationary Earth Orbit	地球同步轨道
IP	Internet Protocol	互联网协议
ETSI	European Telecommunication Standards Institute	欧洲电信标准协会
WINDS	Wideband Inter Networking Engineering Test and Demonstration Satellite	
WGS	Wideband Gap-filling Satellite	宽带全球卫星
DFT	Discrete Fourier Transform	离散傅里叶变换
IPTV	Internet Protocol Television	互联网电视
CWTDM	Continuous Wave Time Division Multiplex	连续波时分复用
FFT	Fast Fourier Transformation	快速傅里叶变换
FDMA	Frequency Division Multiple Access	频分复用
FIR	Finite Impulse Response	有限脉冲响应
T-S-T	Time-Space-Time Switching Network	时分-空分-时分交换网络
FPGA	Field Programmable Gate Array	现场可编程门阵列
NCC	Network Control Center	网络控制中心
CPU	Central Processing Unit	中央处理器
SS-TDMA	Satellite Switched-Time Division Multiple Access	卫星交换-时分多址
MSM	Microwave Switch Matrix	微波交换矩阵
DCU	Distribution Control Unit	分布控制单元
I/O	Input/Output	输入/输出
PLL	Phase Locking Loop	锁相环

目录

摘要.....	I
ABSTRACT	III
插图索引	V
表格索引	VII
符号对照表.....	IX
缩略语对照表.....	XI
目录.....	XIII
第一章 绪论.....	1
1.1 柔性转发器研究背景及意义	1
1.2 柔性转发器关键技术研究进展.....	3
1.3 论文主要研究内容与章节安排.....	4
第二章 柔性转发器以及交换网络基本原理	7
2.1 柔性转发器工作原理	7
2.2 系统参数介绍	8
2.3 T-S-T 交换网络的结构及工作原理	9
2.3.1 时间（T 型）交换器	9
2.3.2 空间（S 型）交换器	10
2.3.3 T-S-T 交换网络的结构及工作原理	12
2.4 T-S-T 交换网络数学模型的建立	14
2.4.1 问题描述	14
2.4.2 矩阵模型的建立	14
2.5 本章小结	15
第三章 程控电路交换网络调度算法的研究	17
3.1 调度算法的设计思想	17
3.1.1 问题的数学分析	17
3.1.2 算法的设计思想	17
3.2 调度算法	18
3.2.1 高冲突值行优先排列算法	18
3.2.2 列优先排列算法	19
3.2.3 改进算法	20

3.3 仿真结果	22
3.3.1 高冲突值行优先排列算法的性能统计	22
3.3.2 列优先排列算法的性能统计	22
3.3.3 改进算法的性能统计	22
3.3.4 三种算法的性能比较	23
3.4 本章总结	27
第四章 交换帧结构设计和交换方案	29
4.1 交换网络交换帧结构的设计	29
4.1.1 SS-TDMA 系统的工作原理	29
4.1.2 SS-TDMA 系统的时隙编排	29
4.1.3 基于 FDMA 多址方式的帧长方案设计	30
4.2 基于 FDMA 多址接入方式的程控交换方案设计	33
4.2.1 基于单播功能的程控方案设计	33
4.2.2 基于多播和广播功能的程控方案设计	38
4.3 本章总结	40
第五章 程控交换网络的 FPGA 设计	41
5.1 芯片的选择	41
5.2 各个模块的 FPGA 设计及仿真	42
5.2.1 数据输入模块设计	42
5.2.2 时钟模块设计	42
5.2.3 端口内样点交换模块设计	43
5.2.4 端口间交换模块设计	45
5.3 整个系统的 FPGA 设计及仿真	47
5.3.1 程控交换模块的 FPGA 设计	47
5.3.2 基于单播功能的仿真	47
5.3.3 基于组播和广播功能的仿真	50
5.4 本章小结	51
第六章 结论与展望	53
6.1 研究结论	53
6.2 研究展望	53
参考文献	55
致谢	59
作者简介	61

第一章 绪论

1.1 柔性转发器研究背景及意义

近几年，随着地面通信网络的飞速发展，资费降低，卫星通信的发展面临着巨大的挑战，但是卫星通信系统覆盖面广，资费没有地域差异，在竞争中又成为了当今通信领域的研究热点^[1]。在未来的卫星通信发展中，GEO 卫星仍然是主流，宽带多媒体业务走俏，军民一体化和天地一体化成为趋势，带宽和功率使用灵活性提高，宽带卫星通信系统具有很大的市场，而带宽资源紧张的问题导致高频段、多波束和全 IP 成为未来宽带卫星通信系统的发展方向^[1]。ETSI（欧洲电信标准化组织）设想未来的宽带卫星通信系统应该具有如下特点：第一，能够承载地面网络的所有业务尤其是全 IP 高速交互式的宽带多媒体业务；第二，高效利用卫星资源特别是频谱资源；第三，服务质量与地面网络类似；第四，降低终端价格，方便商业化生产。

我国“十二五”规划中将发展宽带卫星通信系统作为科研工作的重心^[1]，星载转发器^[2]是卫星通信系统的核心，卫星通信系统的性能优劣很大程度上取决于星载转发器。星载转发器决定信号的接收、处理和转发方式，系统的容量、功耗和可靠性等关键性能参数由星载转发器决定，所以星载转发器成为人们研究的热点。传统的星载转发器按照是否在星上进行信号处理分为再生式转发器和透明转发器。

再生式转发器^[3]采用再生式星上处理技术，卫星接收地面信号，先通过下变频、解调、译码技术变为基带信号，经过路由交换后，再对基带信号编码、调制、上变频，最后将信号发射到地面站。文献[3]对国外几种典型的再生式转发器的研究进展进行了简单概述，指出了每种转发器在信息处理和交换技术方面存在的一些问题并提出了对应的解决方法，比如针对多载波解调问题提出了合适的补偿技术，针对微波/光调制问题提出了光副载波调制技术，针对吞吐量较低问题提出了星上半透明交换以及波分复用等解决方法，为我国卫星转发器的研究设计提供参考。北美地区的 SpaceMux^[4]是星上分组交换处理器，欧洲的 Skyplex^[6]是星上多路复用处理器，日本研制的 WINDS（Wideband Inter Networking Engineering Test and Demonstration Satellite）采用了 3 种工作模式，分别是透明转发、星上交换以及混合模式，详细的转发器结构可以参考文献[7][8][9]。再生式转发器提高了卫星通信系统的容量和资源利用率，但是这种转发器对物理层有很强的依赖性，当地

面通信协议发生变化时，卫星转发器有可能不能正确接收数据，相比于透明转发器，灵活性不高。

透明转发器采用非再生式星上处理技术，只完成转发功能，先接收地面信号，再低噪声放大信号，然后进行变频和功率放大，最后把信号发射回地面站，转发灵活性高，与信号体制没有关系。但是透明转发器星上交换采用微波交换矩阵，属于硬件连接，固定的路由选择方式不能适应业务量的变化。此外，透明转发器通过中频交换矩阵和模拟滤波器来实现星上信号交换，带宽为转发器的带宽，所以交换矩阵的复杂度会随着星上交换粒度的减小而提高，微波开关矩阵不能支持复杂度很高的交换。透明转发器由于其结构简单、抗干扰能力强的特点而在军用卫星通信中占有重要的地位。

结合两种传统卫星转发器的优缺点，2005 年美军研制了一种具有星上处理能力的透明转发器，称为柔性转发器^[10]，这也是 WGS 系统的重要特点之一。文献 [10] 以 WGS 军用卫星为背景，详细分析了搭载在 WGS 卫星上的柔性转发器的工作原理，仿真了具体的实现过程，总结了柔性转发器实现过程中需要解决的关键技术。柔性转发器是基于非均匀滤波器组的数字信道化处理转发器，是采用多相滤波和快速傅里叶变换推导的复调制滤波器组调制法（DFT）调制的数字信道化技术的研究成果。柔性转发器对接收的地面信号只进行解扩，不进行解调，既承接了透明转发器灵活性高的特点，又具有较小交换粒度的优点，同时规避了两种传统转发器的缺点。

三网融合步伐加快，“天地一体化”技术政策的提出，军民共建共用趋势明显，星上高速数据处理和交换的需求不断增加，宽带柔性转发器将会是未来宽带卫星通信系统的发展重心之一。

在当今互联网飞速发展的时代，宽带正在向移动宽带方向发展。2007 年《全球卫星市场需求评估报告》指出欧洲卫星转发器租用市场中 80% 的收益来自视频业务，2012 年全球在线视频收入同比增长 10 倍，IPTV（基于 IP 的被管理的网络上传送的多媒体业务）在亚太地区主要是中国内地和印度快速发展。纵观全球卫星通信系统的发展现状，卫星通信在民用移动通信方面即宽带因特网应用的潜力还没有发挥出来。WGS 系统已经成为美军通信的主要手段，与国外宽带卫星通信蓬勃发展的趋势相比，我国还有很大的差距，急需加快军用技术民用化的步伐。

综上所述，课题组以 WGS 系统的关键技术为依据，结合 CWTDM（连续波时分复用）技术^[11]，构建了民用的多波束 GEO 卫星全 IP 宽带移动通信系统。GEO 卫星宽带移动通信系统不仅在实现很大用户容量的星上处理和星上交换方面存在很大问题，而且对支持因特网业务也有很多难点。鉴于多用户再生式星上处理的设备复杂度和功耗无法承受，一般采用非再生式星上处理以及相应的星上交换，

所以柔性转发器成为首选,同时文献[13]验证了柔性转发器支持 IP 业务的可能性。

1.2 柔性转发器关键技术研究进展

星载转发器的核心是星上信号处理和交换模块。星上数字信号处理方法不同,交换技术不同,卫星转发器的性能也会有差异。柔性转发器是将数字信道化技术作为星上数字信号处理方法的一种宽带卫星有效载荷。柔性转发器的性能取决于数字信道化方法的性能,而不同的数字信道化方法,调制滤波器组的设计方法不同,而不同的调制滤波器组的性能区别在于原型滤波器的设计方法不同。所以,星载柔性转发器的性能主要取决于原型滤波器的设计,原型滤波器的性能对非均匀分路和非均匀重构的性能有决定性意义。

因此,低损伤信道化器的设计是柔性转发器在实现过程中的关键技术之一。均匀滤波器组不能满足可变带宽信号的分析 and 重构,导致了非均匀滤波器组的出现。由于未来的宽带卫星通信系统具有多频段、多速率和多业务的特点,而且业务的实时性要求很高,各波束业务也很复杂,这就要求卫星转发器可以对信号进行近似精确地分析与综合,即需要近似精确的分析综合滤波器,进而要求滤波器的参数应该尽可能地得到优化。考虑滤波器设计的一些基本要求,我们需要研究具有线性相位特性、高阻带衰减以及频带非均匀划分的非均匀滤波器组,当信号进行交换后,非均匀滤波器组应该对信号进行准完全重构。为了保证信号达到极低的损伤程度,研究高效的抗干扰技术 also 具有很强的必要性。

2005 年美军提出的星载柔性转发器采用半再生式信号处理技术,即基于 FFT 的 FDMA 的分路和合路,结合程控交换,在星上只进行分路、交换和合路,回避了大量用户信息的星上处理和星上交换。这种星上非再生式处理和程控电路交换系统,解决了多波束卫星支持大量用户信息在星上进行交换的问题,与透明转发到地面中心站进行信号处理和用户交换后再经卫星链路转发到各个用户相比,将用户之间通信的双跳变为单跳,不仅可节省一半的信道资源,而且能缩短一倍的传输时延,这对于支持宽带实时视频业务来讲是一个非常难得的大优势。

由于 FIR 根升余弦滤波器具有良好的过渡带功率互补性、良好的阻带衰减性和设计简单性等优点,文献[14]中原型滤波器的设计选用了 FIR 根升余弦滤波器,子信道带宽为 2.6MHz,过渡带宽度为 0.325MHz,通带波动小于 0.5dB,阻带衰减小于-35dB。传统的设计方法是采用临界下采样,仿真结果显示合路信号发生了频谱混叠现象。文中通过改进,将临界下采样改为过采样,发现合路信号的频谱不会占满整个子信道,频谱混叠现象得到改善。

近似精确重构的原型滤波器是一种改进的 FIR 原型滤波器。文献[15]提出了一种设计方法,作者参考 Parks-McClellan 算法的思想,设计了实对称 FIR 滤波器

$h(n)$ 和低阶双通道滤波器 $g^2(n)$ ，然后用 $h(n)$ 对 $g^2(n)$ 进行 M 倍内插和 2 倍抽取，得到 $h^M(n)$ ，即近似精确重构的原型滤波器。FIR 设计方法中存在镜像分量，镜像分量会使系统性能下降，文献[15]中的设计方法很好地弥补了这一不足之处，宽带柔性转发器的技术发展又迈出了一大步。

宽带多媒体业务的实时性导致了卫星资源频繁调度的问题，随着系统通信容量增加，交换连接复杂，通信质量也会随之下降。星载柔性转发器需要根据地面用户申请情况实时地改变路由交换矩阵，为了保证用户信号之间能够灵活可靠地交换，降低交换时延，需要研究天地协同的路由刷新机制，因此多波束信道化路由交换技术也是柔性转发器实现过程中的一大关键技术。

多波束信道化路由交换技术分为基带交换和中频（射频）交换两大类。星上基带交换包括电路交换和分组交换两种形式。分组交换采用了存储转发机制，每个分组有一个分组头，分组头包含选路信息和控制信息。分组交换提供两种服务方式，分别是数据报方式和虚电路方式。虚电路指的是在用户信息传输前先建立一条端对端的虚电路，进而数据分组在这条通路上进行传输。从呼叫建立过程来看，虚电路交换方式与电路交换方式类似，二者之间的不同是，前者是逻辑连接，后者是物理连接。虚电路交换方式用户可以共享一条物理信道，即在一条物理链路上能够建立很多条逻辑链路，而电路交换方式用户只能单独使用一条物理信道，如果该用户此时没有数据需要传输，即使此时这条物理链路处于空闲状态，其他用户也不能使用，除非断开这条链路，重新分配给其他用户，才可以使用。

系统选用柔性转发器结合 CWTDM 技术，选用电路交换技术，主要研究了基于柔性转发器的星上交换模块的相关技术。

1.3 论文主要研究内容与章节安排

本文主要研究了基于宽带柔性转发器的高速交换技术，一共有五章，具体内容安排如下：

第一章：介绍了柔性转发器的研究背景与意义，概括了宽带柔性转发器实现过程中的关键技术研究进展，提出了本文的研究意义和主要内容。

第二章：介绍了柔性转发器的工作原理，给出了系统的相关参数，概括了 T-S-T 交换网络的基本知识，将 T-S-T 交换网络抽象为矩阵模型，调度问题抽象为矩阵置换问题。

第三章：在矩阵模型作为 T-S-T 交换网络的数学模型的基础上，研究了程控电路交换网络的调度算法。在高冲突值行优先排列算法和列优先排列算法的基础上提出了一种基于列优先排列的改进算法，给出了改进算法的流程图，通过柱状图比较了三种算法的比较次数和交换次数，仿真结果表明，改进算法在时间复杂

度上优于高冲突值行优先排列算法和列优先排列算法，但是空间复杂度高于其他两种算法。

第四章：首先设计了 **T-S-T** 交换网络的交换帧结构，然后提出了基于 **FDMA** 多址接入方式的程控交换方案，实现了单播、组播和广播的功能。

第五章：在基于 **FDMA** 多址接入方式的程控交换方案设计的基础上，给出了各个模块的 **FPGA** 设计和仿真结果，实现了单播、组播和广播功能，通过合理调整信号量化比特数和子信道数，实现了芯片容量的利用率最大化。

第二章 柔性转发器以及交换网络基本原理

2.1 柔性转发器工作原理

宽带柔性转发器是一种具有星上处理能力的半透明转发器，也叫半再生式转发器。宽带柔性转发器采用基于非均匀滤波器组的数字信道化处理技术，兼具传统透明转发器和再生式转发器的优点，既具有灵活可靠的特点，又可以支持较小粒度的交换，还规避了物理层信号体制的约束，增加了系统容量，满足可变带宽业务的需求，是未来星载转发器的主要发展方向之一。

如图 2.1 所示，柔性转发器包括信道化器、路由交换和星载路由交换控制器三部分。每个波束设置一个信道化器，信道化器接收信号，对信号进行分路和合路，星载路由交换控制器根据来自 NCC 的指令，对路由交换矩阵进行设置，实现多波束信号之间的自由交换。NCC 根据地面站用户申请情况，为每个用户分配带宽和信道资源，同时配置星载路由交换控制器，完成多波束信号交换。

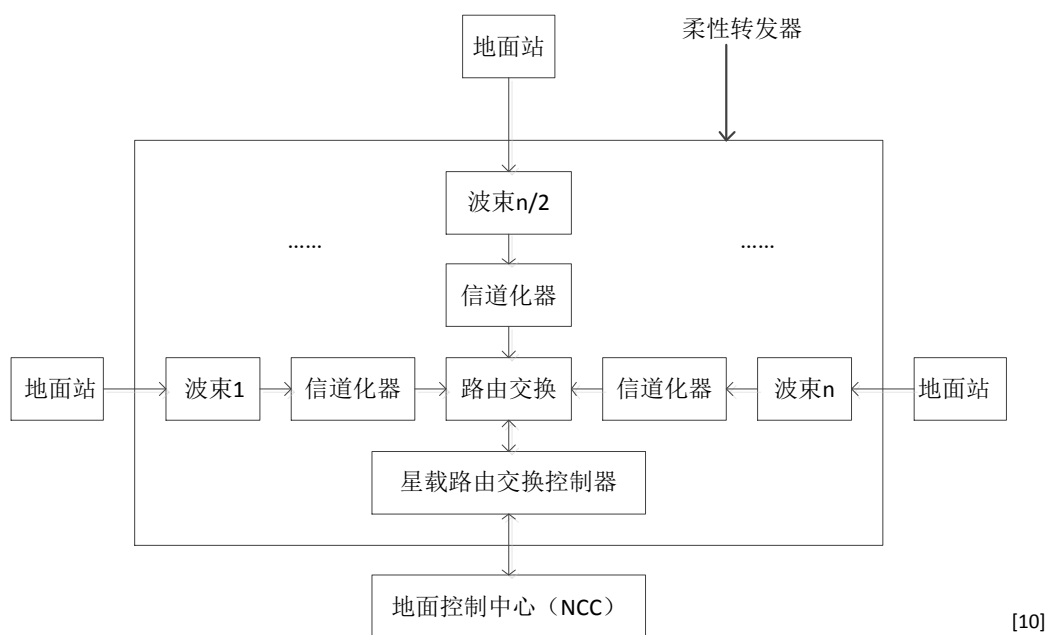


图 2.1 基于柔性转发器的卫星通信系统框图

星载柔性转发器采用非均匀滤波器组实现可变带宽信号之间的交换，在 WGS 系统中，每个波束有 39 个子信道，每个子信道带宽为 2.6MHz。假定，波束 1 接收的前三路信号，带宽分别为 2.6MHz、5.2MHz 和 10.4MHz，即占用子信道数分别为 1、2 和 4，第一路信号占用子信道 1，第二路信号占用子信道 2 和 3，第三路信号占用子信道 4、5、6、7，经过交换模块后，第一路信号占用子信道 15，第

二路信号占用子信道 8 和 9，第三路信号占用子信道 33、34、35、36。

如图 2.2 所示，简要描述了信道化器的实现过程，分析滤波器组先将非均匀带宽信号进行分路、抽取、交换和插值后，合成滤波器组再将每路信号进行重构。信道化器的设计是柔性转发器实现的关键技术之一，而非均匀分析和合成滤波器组是信道化器设计的关键。但是，信道化器的设计不是本文的研究重心所在，这里就不具体介绍了，详细工作原理和设计方案可以参考文献[10][14][15]。

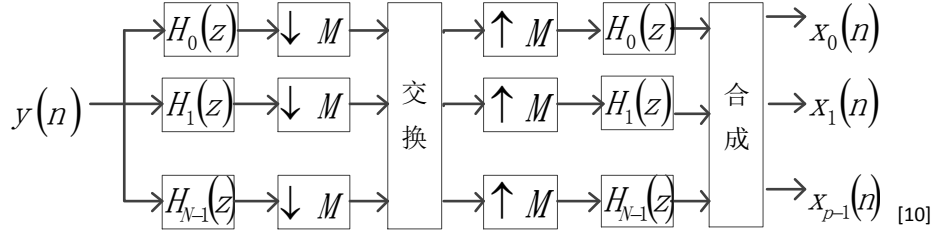


图 2.2 信道化器原理图

2.2 系统参数介绍

本文系统中包括 16 个端口，端口带宽为 125MHz，每个端口 48 个子信道，每个子信道带宽为 2.6MHz，每路信号可以占用一个子信道，称为窄带用户，也可以占用多个相邻的带宽为 2.6MHz 的子信道，称为宽带用户。

交换模块是本文的核心内容。图 2.3 是交换模块的简单示意图：



图 2.3 交换模块简单示意图

交换模块的实现包括三种情况：

- (一)端口内子信道之间的交换，例如图 2.3 中，端口 1 中子信道 10 和子信道 25 之间的交换；
- (二)端口间子信道之间的交换，例如图 2.3 中，端口 2 的子信道 36 和端口 3 的子信道 42 之间的交换；
- (三)端口间的交换，例如图 2.3 中，端口 15 和端口 16 之间的交换；

根据类比思想,可以把 16 个端口看做 16 个波束,每个端口的 48 个子信道可以看做是 FDMA 多址接入方式的 48 个子频带,子信道之间的交换可以类比为频带交换。所以问题转换为基于 FDMA 的多波束信道化路由交换。

基于柔性转发器中半再生式星上处理方式,星上处理模块只进行了采样和量化处理,没有进行编码,所以进入交换模块的是一系列的样点,程控交换模块需要实现样点交换的功能。样点交换又可以分为三种:多样点交换、单样点交换和单比特交换。但是,由于多样点交换一个时钟内需要处理的数据量比较大,对硬件的存储器容量要求比较大,单比特交换虽然需要的存储器容量小,但是需要的时钟频率很高,对硬件的要求也很大,综合二者的特点,我们采用单样点交换,既不需要很大的存储器容量,也不需要很高的时钟频率。

由于采用样点交换,我们借鉴 WGS 系统,也选择程控电路交换,而 T-S-T 交换网络是程控电路交换网络中应用最广泛的。2.3 小节主要介绍了 T-S-T 交换网络的结构和工作原理。

2.3 T-S-T 交换网络的结构及工作原理

电路交换网络根据不同的分类方法,包括模拟空分交换网络和数字时分交换网络,单级交换网络和多级交换网络等等。模拟空分交换网络能够等效为实现相同功能的数字时分交换网络^[6]。

T-S-T 交换网络是一种三级时空交换网络,包括两个时间交换器和一个空间交换器。

2.3.1 时间 (T 型) 交换器

时间交换器又称为 T 型交换器,适用于不同时隙位置,但是属于相同复用线之间的信号交换。时间交换器由数据存储器和控制存储器两部分组成,原理图如图 2.4 所示:

数据存储器的功能是存储用户的业务信息,输入/输出复用线上的时隙数等于数据存储器的单元数。

控制存储器用来存储用户所在的地址,从而实现对用户交换信息的控制。控制存储器的单元数等于数据存储器的单元数,同时,它的单元数也决定了每个单元的位宽。如果控制存储器有 n 个单元,每个单元占据 c 比特,则 n 和 c 之间满足如下条件: $2^c \geq n$ 。

对数据存储器的控制方式有两种:一种是顺序写入,控制读出,简称输出控制方式,如图 2.3 (a) 所示,一种是控制写入,顺序读出,简称输入控制方式,如图 2.3 (b)。在这里,顺序的含义是指控制存储器的读或写地址是由系统时钟电路提供的,这样可以保证控制存储器的单元号与输入/输出复用线上的时隙号一一

对应，这里的控制是指按照控制存储器中写好的内容控制数据存储器的读或写，进而完成时隙交换功能。

对控制存储器的控制方式只有一种，它总是由交换控制系统，例如 CPU 等控制写入，然后顺序读出。交换控制系统根据收到的用户呼叫请求，得到要对控制存储器的写入操作的地址和数据，即完成了链路的连接。

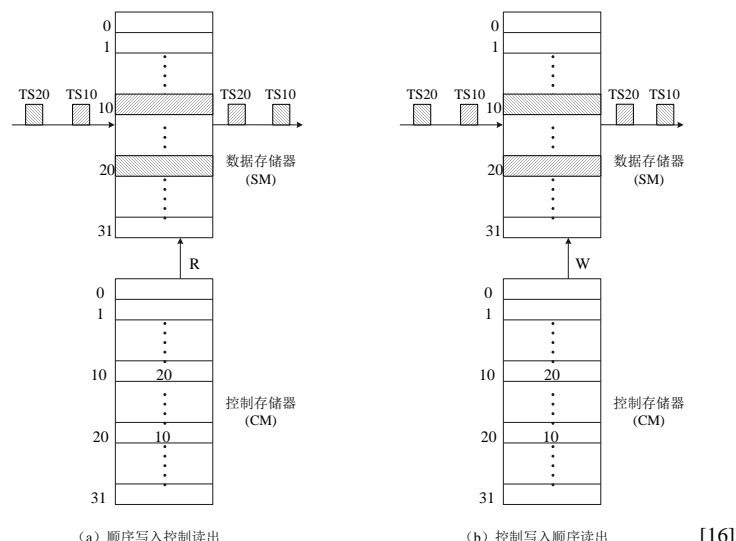
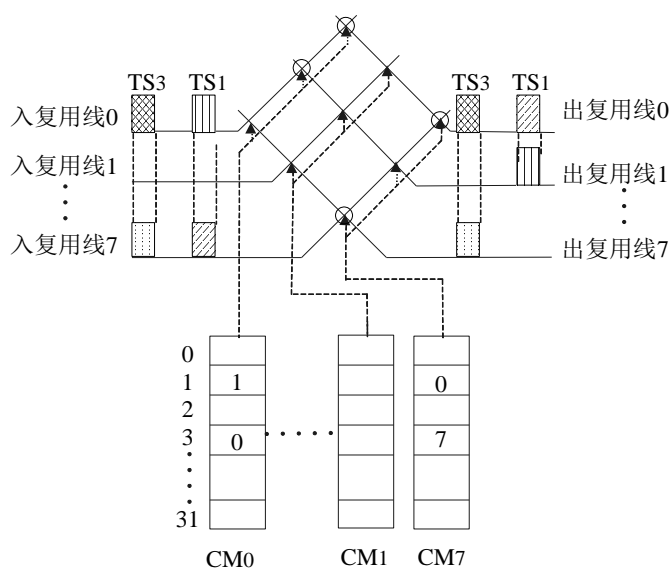


图 2.4 T 型交换器构成原理图

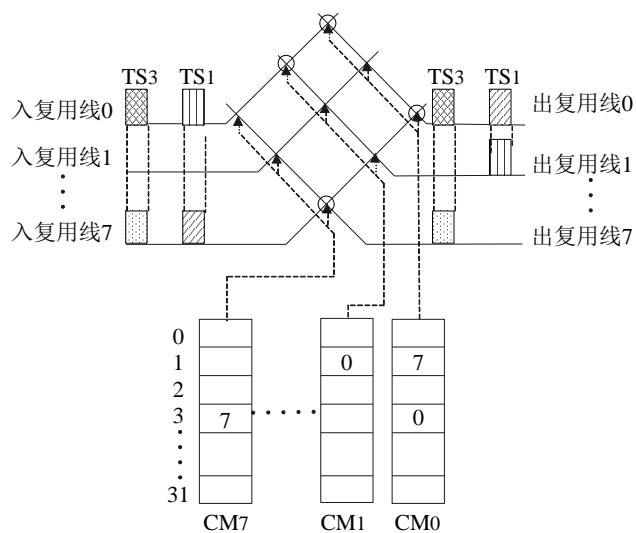
接下来，用两个用户之间的通信为例，详细说明一下时间交换器的工作流程。如图 2.3 (a) 所示，用户 A 占据 TS_{10} ，用户 B 占据 TS_{20} ，要想实现用户 A 和用户 B 之间的交换，需要把 TS_{10} 时隙中的内容和 TS_{20} 时隙中的内容进行互换，当控制存储器收到这样的请求时，交换控制系统控制控制存储器的 10 号单元写入 20，20 号单元写入 10。如果时间交换器采用输入控制方式，在 TS_{10} 时刻，控制存储器输出 20，把 10 号单元的内容写到了 20 号单元，同理，在 TS_{20} 时刻，控制存储器输出 10，把 20 号单元的内容写到了 10 号单元，然后数据存储器在时钟控制下顺序读出，在 TS_{10} 时刻输出 20 号单元的内容，在 TS_{20} 时刻输出 10 号单元的内容，这样就完成了用户 A 和用户 B 之间的通信功能。如果时间交换器采用输出控制方式，数据存储器在时钟控制下，把各个用户的信息顺序写入到数据存储器相应的单元内，在 TS_{10} 时刻，控制存储器输出 20，控制将 20 号单元的内容读出，同理，在 TS_{20} 时刻，控制存储器输出 10，控制将 10 号单元的内容读出，这样同样也完成了用户 A 和用户 B 之间的通信功能。

2.3.2 空间 (S 型) 交换器

空间交换器又称为 S 型交换器，适用于同一个时隙位置，但是属于不同的复用线之间的信号交换。如图 2.5 所示，空间交换器的组成部分包括交叉接点矩阵和控制存储器。



(a) 输入控制方式



(b) 输出控制方式

图 2.5 S 型交换器构成原理图

如图 2.5 所示，8 条输入复用线和 8 条输出复用线构成了一个 8×8 的矩阵，该矩阵叫做交叉接点矩阵，控制交叉接点工作的是 $CM0 \sim CM7$ 这七个控制存储器。其中的数量关系如下所述：复用线的条数决定了控制存储器的个数，比如图 2.5 中有 8 条输入复用线，所以有 8 个控制存储器；每条复用线上的时隙数决定了控制存储器的单元数，比如图 2.5 中每条复用线上有 32 个时隙，每个控制存储器有 32 个单元；由于控制存储器中的存储内容是复用线编号，控制存储器每个单元的位宽由复用线的条数决定，比如图 2.5 中，复用线编号为 0~7，所以控制存储器的单元位宽为 3。

空间交换器包括两种控制方式，一种是输入控制方式，在图 2.5 (a) 中，控制存储器是按输入复用线配置的，所以叫输入控制方式，控制存储器的个数与输入复用线的条数相同，控制存储器的单元个数与时隙个数相同，控制存储器的单元内容为输入复用线编号；一种是输出控制方式，在图 2.5 (b) 中，控制存储器是按输出复用线配置的，所以叫输出控制方式，控制存储器的个数与输出复用线的条数相同，控制存储器的单元个数与时隙个数相同，控制存储器的单元内容为输出复用线编号。

接下来，以图 2.5 (a) 为例，说明空间交换器的输入控制方式的工作原理。在 CM0 的 1 号单元写入了 1，3 号单元写入了 0，在 CM7 的 1 号单元写入了 0，在 3 号单元写入了 7。在 TS1 时刻，CM0 读出的内容为 7，CM7 读出的内容为 0，控制 0 号输出线的内容在 7 号输出线读出，7 号输出线的内容在 0 号输出线读出，在 TS3 时刻，CM0 读出的内容为 0，CM7 读出的内容为 7，控制 0 号输出线的内容在 0 号输出线读出，7 号输出线的内容在 7 号输出线读出。同理可以推出输出控制方式的工作原理。输出控制方式可以实现组播，也就是在某一时刻，某一条输入复用线的内容可以同时多条输出复用线上输出。

空间交换器的交叉节点闭合周期由时隙时间决定，当时隙时间发生改变时，输入和输出复用线的连通关系也会改变。显然，空分交换器的工作也是与时间有关系的。在大型的数字交换网络中，空间交换器必须和时间交换器结合起来才可以使用。

2.3.3 T-S-T 交换网络的结构及工作原理

在 T-S-T 三级交换网络中，第一个时间交换器叫做初级 T 接线器，用来进行同一条输入链路上的时隙之间交换功能，第二个时间交换器叫做次级 T 接线器，用来进行同一条输出链路上的时隙之间交换功能，S 交换器可以完成不同链路之间的交换。其中，每个交换器的工作方式可以任意选择，但是实际应用中，为了方便控制，两个 T 接线器工作方式一般不同，S 接线器没有要求。图 2.6 是 T-S-T 交换网络的组成及工作原理：

接下来，T-S-T 三级交换网络的详细流程通过一个具体例子来介绍。

在图 2.6 中，SMA₀~SMA₃ 表示初级 T 接线器的数据存储器，CMA₀~CMA₃ 表示初级 T 接线器的控制存储器；SMB₀~SMB₃ 表示次级 T 接线器的数据存储器，CMB₀~CMB₃ 表示次级 T 接线器的控制存储器；CMC₀~CMC₃ 表示 S 接线器的控制存储器。图 2.6 实现了复用线 0 的第 1 个时隙（用户 A）和复用线 3 的第 9 个时隙之间的交换（用户 B），具体实现过程如下：

(1) A 到 B 的通路：A 到 B 先经过初级 T 接线器与中间时隙(假定中间时隙为 30)，因为初级 T 接线器为输入控制方式，如图 2.6 所示，交换控制系统先将

单元号 30 写入 CMA_0 的第 1 个时隙，当时钟到来时通过 CMA_0 将第 1 个时隙的内容控制写入 SMA_0 的第 30 个时隙，在时钟脉冲下顺序读出，完成时隙 1 与时隙 30 的交换。为了将时隙 30 从复用线 0 交换到复用线 3，根据 S 接线器的输出控制方式需要在 CMC_3 第 30 号单元内填入复用线号 0，当时钟到来时，对应的交叉点闭合，完成子信道的空间搬移。接下来的次级 T 接线器的动作与初级 T 接线器类似，完成时隙 30 和时隙 9 的内容交换，不再细述。

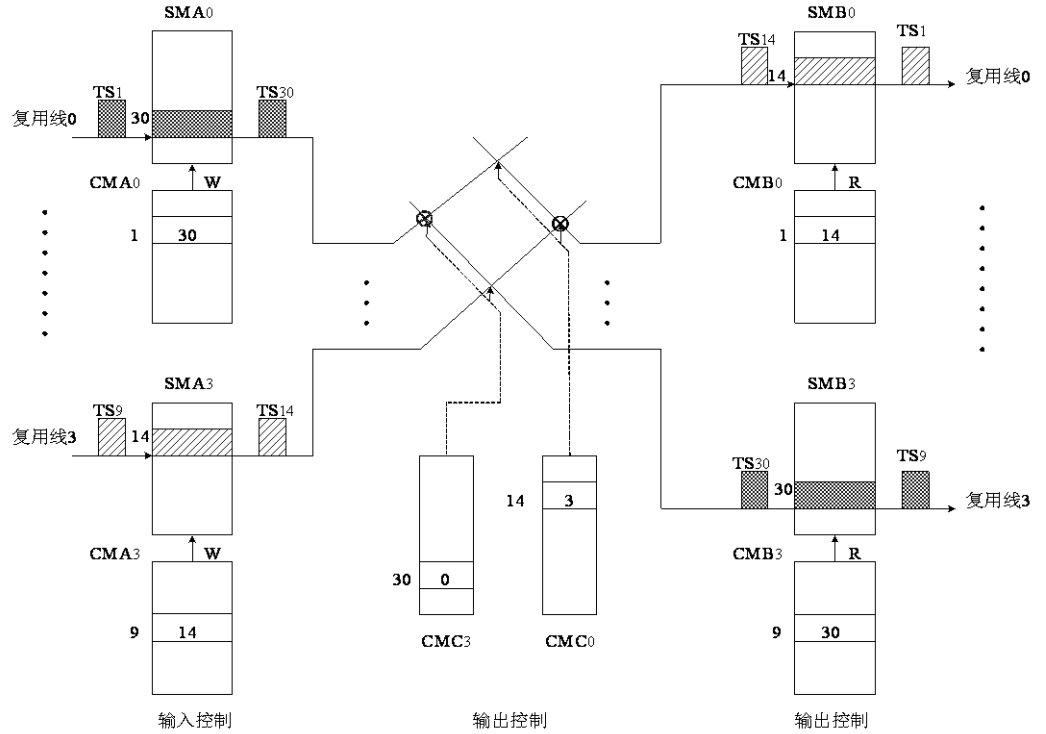


图 2.6 T-S-T 三级交换网络的工作原理

(2) B 到 A 的通路：B 到 A 的接续流程与 A 到 B 的一样，需要注意的是：A 到 B 的通信中，A 用户所占的时隙 1 为发话信道，B 用户所占的时隙 9 为受话信道；B 到 A 的通信中，B 用户所占的时隙 9 为发话信道，A 用户所占的时隙 1 为受话信道。用户占用的信道既是发话信道又是受话信道，只是发话信道总是在输入侧，受话信道总是在输出侧。

(3) 中间时隙的选择：分析可知，正方向的中间时隙和反方向的中间时隙有必然的对应关系，为了方便选择和简化控制，一般采用反相法选择中间时隙。在图 2.6 中就是采用的反相法，若正向连接的中间时隙选择 i ，则反向连接的中间时隙为 $(i + F/2) \bmod F$ ，其中 F 为复用线上时隙数。图 2.6 中 F 为 32，所以当正向路由的中间时隙为 30 时，反向路由的中间时隙为 14。

当双方通信结束后，处理器将各控制存储器相应单元的内容清除。

2.4 T-S-T 交换网络数学模型的建立

2.4.1 问题描述

根据 T-S-T 交换网络的基本原理，我们对其进行如下的数学抽象：

定义一个 $n*m$ 的矩阵， n 表示矩阵的行数， m 表示矩阵的列数。设想把输入数据流抽象为这样一个矩阵，即 in_port 矩阵，那么 T-S-T 交换网络中的链路数可以用矩阵的行数来表示，每一条链路中的时隙数可以用矩阵的列数来表示。T 交换是同一条链路中不同时隙之间的交换，所以 T 交换可以抽象为矩阵的行内变换。S 交换是不同链路之间同一时隙之间的交换，所以 S 交换可以抽象为矩阵的列内变换。数据流进入三级 T-S-T 交换网络中，先经过初级 T 接线器进行一次 T 交换，再经过 S 接线器进行一次 S 交换，最后经过次级 T 接线器进行第二次 T 交换。所以，下面定义几个矩阵：

in_port 矩阵：输入矩阵，表示上行链路（用户信息从地面站到达卫星）中每个用户的信道带宽使用情况；

out_port 矩阵：输出矩阵，表示下行链路（用户信息从卫星发射到地面站）中每个用户的信道带宽使用情况；

$middle_1$ 矩阵：中间矩阵 1，由 in_port 矩阵经过一次行内变换所得；

$middle_2$ 矩阵：中间矩阵 2，由 $middle_1$ 矩阵经过一次列内变换所得，它经过一次行内变换可以得到 out_port 矩阵。

2.4.2 矩阵模型的建立

通过分析 T-S-T 交换网络中各个模块的特点，T 接线器工作原理模拟为矩阵的行内变换，S 接线器工作原理模拟为矩阵的列内变换，T-S-T 交换原理模拟为矩阵置换问题。这样，2.4.1 小节的问题描述可以归纳为如下三个过程：

首先，给定的输入矩阵 in_port 经过合适的行内变换，进而获得 $middle_1$ 矩阵；

然后， $middle_1$ 矩阵经过合适的列内变换，进而获得 $middle_2$ 矩阵；

最后， $middle_2$ 矩阵经过合适的行内变换，最终获得符合要求的输出矩阵 out_port 。

图 2.7 是 T-S-T 交换网络的矩阵模型：

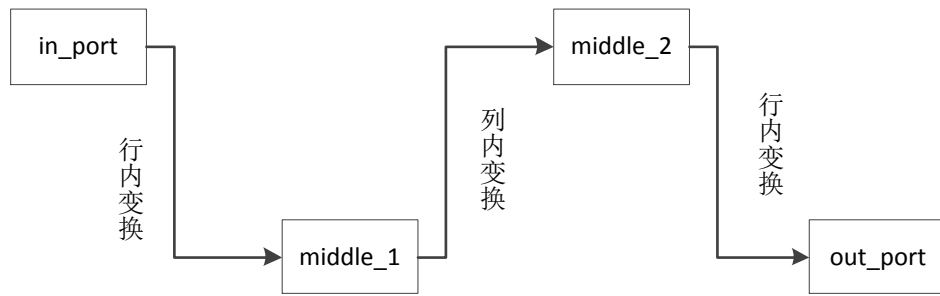


图 2.7 T-S-T 交换网络的矩阵模型

2.5 本章小结

本章首先介绍了柔性转发器工作原理，给出了系统参数，然后讲述了 T-S-T 交换网络的结构及工作原理，最后给出了一种 T-S-T 交换网络的矩阵数学模型，后续的工作都是围绕此模型展开的。

第三章 程控电路交换网络调度算法的研究

3.1 调度算法的设计思想

在 2.4 小节，我们已经建立了 T-S-T 交换网络的数学模型，具体的路径选择问题转化为纯数学矩阵置换问题。接下来，本文将从纯数学的角度出发，寻找一个调度算法，根据给定的输入矩阵，通过合理的矩阵置换，得到符合要求的输出矩阵。

3.1.1 问题的数学分析

仔细观察 T-S-T 交换网络的矩阵变换模型，我们发现，输入矩阵变换到输出矩阵期间，一共经历了两次时间变化和一次空间变换。

3.1.2 算法的设计思想

鉴于以上对问题的数学分析，我们可以总结出如下的算法设计思想：

- 1) 给定一个输入矩阵，根据用户的交换路由表，都会有一个输出矩阵与之相对应；
- 2) 对于每一对输入矩阵和输出矩阵，都存在两个中间矩阵满足上述要求；
- 3) 输入矩阵经过适当的行内变换可以得到第一个中间矩阵；
- 4) 第一个中间矩阵经过适当的列内变换可以得到第二个中间矩阵；
- 5) 第二个中间矩阵经过适当的行内变换可以得到最后的输出矩阵；

根据矩阵的行内变换和列内变换的数学性质，我们可以得出 $middle_2$ 矩阵具有如下的特征：位于输入矩阵 in_port 上的同一行中的元素不能出现在 $middle_2$ 矩阵的同一列中。这是因为从 in_port 矩阵到 $middle_1$ 矩阵是经过行内变换， in_port 矩阵同一行的元素只能位于 $middle_1$ 矩阵中的不同列，从 $middle_1$ 矩阵到 $middle_2$ 矩阵是经过列内变换，所以 $middle_2$ 矩阵的每一列元素不会存在 in_port 矩阵同一行中的元素。

如上所述，根据系统的用户接入情况和用户到达情况，我们可以得到一对输入矩阵 in_port 和输出矩阵 out_port 。所以，问题转换为，已知输入矩阵 in_port 和输出矩阵 out_port ，求中间矩阵 $middle_1$ 和 $middle_2$ 。

由前面描述的 $middle_1$ 和 $middle_2$ 矩阵的特点可以发现，从输入矩阵 in_port 变换到输出矩阵 out_port 的过程和从输出矩阵 out_port 变换到输入矩阵 in_port 的过程是对称的，所以从输出矩阵 out_port 变换到输入矩阵 in_port 所得的中间矩阵也是问题的解。因为输入矩阵对于输出矩阵 out_port 相对稳定，为了描述方便和

算法实现，论文采用了逆推法，过程如下：

首先，由输出矩阵 out_port 经过行内变换得到 $middle_2$ 矩阵；

然后， $middle_2$ 矩阵经过适当的列内变换得到 $middle_1$ 矩阵；

容易发现，从 $middle_2$ 矩阵变换到 $middle_1$ 矩阵只需要通过一个简单的排序就可以完成，所以，T-S-T 交换网络调度算法的关键是如何从输出矩阵 out_port 推导出 $middle_2$ 矩阵，即将一个矩阵经过适当的行内变换得到 $middle_2$ 矩阵，要求是 $middle_2$ 矩阵中同一列上不存在输入矩阵 in_port 中的同一行的数据。

3.2 调度算法

文献[17]中，提出了两种 T-S-T 网络调度算法，包括高冲突值行优先排列算法和列优先排列算法。其中，高冲突值行优先排列算法是从行出发，把 out_port 矩阵的每一行元素放到 $middle_2$ 矩阵对应行的适当位置上，保证 in_port 矩阵同一行的元素不会出现在 $middle_2$ 矩阵的同一列。列优先排列算法，是从列出发，调整 out_port 矩阵每一列的元素，使得每一列元素符合 $middle_2$ 矩阵的特点。本文针对两种算法的利弊，结合高冲突值行优先排列算法和列优先排列算法的思想，提出一种改进算法。

3.2.1 高冲突值行优先排列算法

高冲突值行优先排列算法^[17]的核心思想是，根据已知的 out_port 矩阵，对 $middle_2$ 矩阵进行逐行排列，设计算法，把 out_port 矩阵的每一行元素放到 $middle_2$ 矩阵对应行的适当位置上，保证 in_port 矩阵的同一行元素不会在 $middle_2$ 矩阵的同一列出现。

在介绍高冲突值行优先排列算法的具体步骤之前，首先给出一些如下概念：

假设 $middle_2$ 矩阵前 $i-1$ 行已经排列完毕，接下来要排列第 i 行，则

矩阵 v 表示： $middle_2$ 矩阵的前 $i-1$ 行元素在 in_port 矩阵的所在行值；

矩阵 h 表示： out_port 矩阵第 i 行元素在 in_port 矩阵中的所在行值；

$vRepeat[j]$ 表示： $v[j]$ ($j=i-1$) 中的元素在 h 中重复出现的次数之和；

$hRepeat$ 表示： h 中的元素在 $v[j]$ ($j=i-1$) 中重复出现的次数之和；

接下来给出算法的具体流程：

- 1) 把输出矩阵 out_port 的第一行直接赋给 $middle_2$ 矩阵， $middle_2$ 矩阵的元素安放从第二行开始；
- 2) 统计 $middle_2$ 矩阵的第一行元素在 in_port 矩阵中的所在行值，赋给 v ；
- 3) 统计 out_port 矩阵的第二行元素在 in_port 矩阵中的所在行值，赋给 h ；
- 4) 统计 $vRepeat[j]$ 中的最大值，将 $hRepeat$ 中最大的元素安放在最大值所在列；

- 5) 重新统计 $hRepeat$ 和 $vRepeat[j]$;
- 6) 如果这一行排列完成, 进入下一行排列, 如果这一行没有排列完成, 回到步骤 4 继续安放元素;

在实现过程中, 为了简化处理过程, 假设矩阵都是 n 阶方阵, 实际应用过程中, 矩阵的行和列可以不同。

3.2.2 列优先排列算法

列优先排列算法^[17]的核心思想是, 根据输出矩阵 out_port , 对 $middle_2$ 矩阵进行逐列排列。先把 out_port 矩阵赋给 $middle_2$ 矩阵, 设计算法, 调整输出 $middle_2$ 的每一列, 保证输入矩阵 in_port 的同一行元素不会出现在 $middle_2$ 矩阵的同一列。

在详细介绍列优先排列算法的步骤之前, 先给出如下一些定义:

物理行号: 输出矩阵 out_port 中的元素在输出矩阵 out_port 中的行号;

逻辑行号: 输出矩阵 out_port 中的元素在输入矩阵 in_port 中的行号;

重复元素: 输出矩阵 out_port 同一列元素中逻辑行号相同的元素;

缺少元素: 输出矩阵 out_port 一列元素中缺少的逻辑行号;

其他元素: 输出矩阵 out_port 中既不重复也不缺少的元素;

综上所述, 列优先排列算法的核心就是, 使得被处理的每一列既没有缺少元素, 也没有重复元素, 也就是说对每一列进行处理时, 把重复元素通过一系列的行变化转换为缺少元素, 达到该处理列包含所有的逻辑行号。接下来将详细介绍算法的具体流程:

- 1) 令 $middle_2$ 矩阵等于输出矩阵 out_port ;
- 2) 从第一列开始, 找重复元素和缺少元素, 记录重复元素和缺少元素信息, 即重复元素和缺少元素的位置和个数;
- 3) 对重复元素进行依次处理, 把重复元素置换成缺少元素;
- 4) 如果该重复元素所在行存在缺少元素, 则把重复元素和缺少元素直接交换;
- 5) 如果该重复元素所在行没有缺少元素, 则在重复元素所在行找其他元素, 检查该列的其他元素所在行有没有缺少元素, 如果该列的其他元素所在行存在缺少元素, 则把重复元素与所在行的其他元素交换, 然后将该列的其他元素与所在行的缺少元素进行交换;
- 6) 反复执行步骤 4 和步骤 5, 直到该列既没有重复元素也没有缺少元素, 然后列数加一, 对下一列进行同样的处理。

对于 n 阶矩阵来说, 经过有限次交换, 肯定会置换出满足条件的矩阵。

3.2.3 改进算法

我们通过置换过程中的交换次数和比较次数来衡量调度算法的性能优劣，对上述高冲突值行优先排列算法和列优先排列算法的性能统计表进行比较（比较结果见 3.3.3 小节），可以发现，列优先排列算法不需要回溯，算法复杂度小，唯一的不足就是增加了交换次数，鉴于以上分析，提出了列优先排列算法的改进方案。结合上述列优先排列算法的具体流程，提出改进算法与上述算法的不同之处主要在第五步：

- 1) 令 `middle_2` 矩阵等于输出矩阵 `out_port`;
- 2) 从第一列开始，找重复元素和缺少元素，记录重复元素和缺少元素的信息，即重复元素和缺少元素的位置和个数；
- 3) 对重复元素进行依次处理，把重复元素置换成缺少元素；
- 4) 如果该重复元素所在行存在缺少元素，则把重复元素和缺少元素直接交换；
- 5) 如果该重复元素所在行没有缺少元素，则找该列下一个重复元素，直到这个重复元素的所在行存在缺少元素，把重复元素和缺少元素进行交换；
- 6) 对该列的下一个重复元素进行相同的操作，直到改处理列既没有重复元素也没有缺少元素，列数加一，回到步骤 2 进行相同的操作；

改进算法流程图如图 3.1 所示：

3.2.4 调度算法的举例

用下面的例子来说明：

$$\text{in_port} = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 \\ 8 & 9 & 10 & 11 \\ 12 & 13 & 14 & 15 \end{bmatrix}$$

$$\text{out_port} = \begin{bmatrix} 2 & 1 & 6 & 9 \\ 10 & 12 & 4 & 8 \\ 3 & 13 & 14 & 15 \\ 7 & 0 & 5 & 11 \end{bmatrix}$$

统计 `out_port` 矩阵第一列，缺少元素为 4，即缺少逻辑行号为 4 的元素，重复元素为 2 和 3，即存在两个逻辑行号为 1 的元素，其他元素为逻辑行号 2 和逻辑行号 3。

文献[17]中：先选其他元素逻辑行号 2（`out_port` 矩阵中的元素为 7），在其所

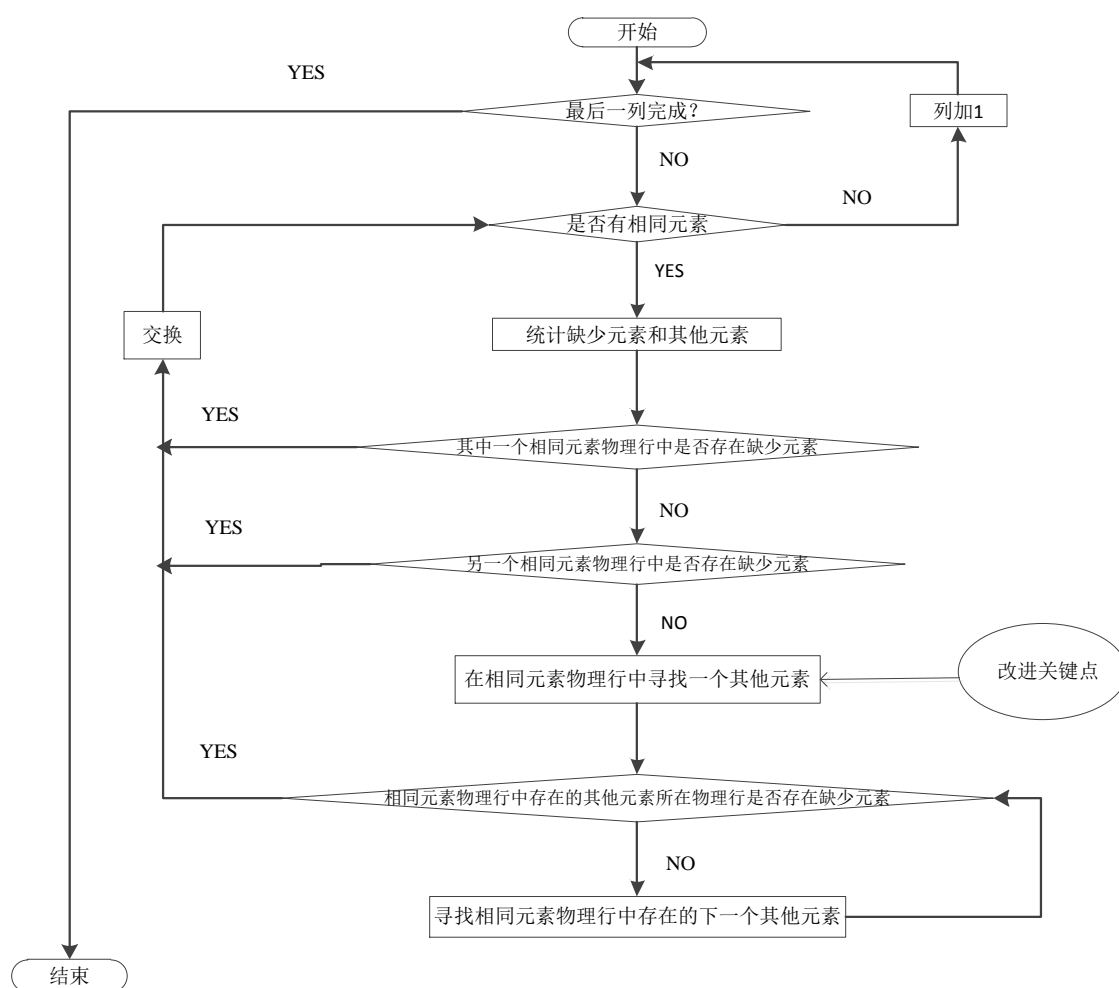


图 3.1 改进算法流程图

在的物理行内，看是否存在缺少元素 4，发现没有，继续在该物理行内寻找不同于本元素的其他元素逻辑行号 3 (out_port 矩阵中为 11 的元素)，再在第 1 列中逻辑行号为 3 的元素 (out_port 矩阵中的元素为 10) 所在行中找缺少元素 4 (out_port 矩阵中的元素为 12)，所以将 12 和 10 交换，11 和 7 交换，2 和 6 交换，这里进行了三次交换，得到了 middle_2 矩阵中第 0 列元素，符合要求。变换后的矩阵状

态为：

$$\begin{bmatrix} 6 & 1 & 2 & 9 \\ 12 & 10 & 4 & 8 \\ 3 & 13 & 14 & 15 \\ 11 & 0 & 5 & 7 \end{bmatrix}$$

依次类推，可以导出中间矩阵第二列为：

$$\begin{bmatrix} 6 & 1 & 2 & 9 \\ 12 & 10 & 4 & 8 \\ 3 & 13 & 14 & 15 \\ 11 & 5 & 0 & 7 \end{bmatrix}$$

第三列为：

$$\begin{bmatrix} 6 & 1 & 9 & 2 \\ 12 & 10 & 4 & 8 \\ 3 & 13 & 14 & 15 \\ 11 & 5 & 0 & 7 \end{bmatrix}$$

最后的 `middle_2` 矩阵，期间一共交换了 5 次。

改进算法：选择其他元素逻辑行号 2 (`out_port` 矩阵中的元素为 7)，在其所在的物理行内，看是否存在缺少元素 4；接下来，我们再选其他元素逻辑行号 3 (`out_port` 矩阵中的元素为 10)，在其所在的物理行内，看是否存在缺少元素 4 (`out_port` 矩阵中的元素 12)，将 12 与 10 交换，将 2 与 9 交换，这里进行了两次交换，得到了 `middle_2` 矩阵中第 0 列元素，符合要求。变换后的矩阵状态为：

$$\begin{bmatrix} 9 & 6 & 1 & 2 \\ 12 & 10 & 4 & 8 \\ 3 & 13 & 14 & 15 \\ 7 & 0 & 5 & 11 \end{bmatrix},$$
 第二列满足条件，开始排第三列，

$$\begin{bmatrix} 9 & 6 & 1 & 2 \\ 12 & 10 & 8 & 4 \\ 3 & 13 & 14 & 15 \\ 7 & 0 & 5 & 11 \end{bmatrix}$$
 即最后的

`middle_2` 矩阵，期间一共交换 3 次。

3.3 仿真结果

在 3.2.4 小节，用一个 4*4 的矩阵进行实例分析，对三种算法进行了比较，接下来，分别取 $n=10,20,50,100,150,300$ ，通过计算机产生随机数值，在 n 相同的情况下，统计三种算法的平均比较次数和平均交换次数，通过对比三种算法的统计结果，针对时间复杂度和空间复杂度，对三种算法的性能进行分析。算法的实现是在 VS2008 编译环境下采用 C++ 语言实现的。

3.3.1 高冲突值行优先排列算法的性能统计

表 3.1 是高冲突值行优先排列算法性能统计表。

3.3.2 列优先排列算法的性能统计

表 3.2 是非回溯列优先排列算法性能统计表。

3.3.3 改进算法的性能统计

表 3.3 是改进算法性能统计表。

3.3.4 三种算法的性能比较

观察图 3.8 可以发现，在矩阵相同的情况下，当 n 比较小时，高冲突值行优先排列算法的平均比较次数明显高于列优先排列算法和改进算法，当 n 比较大时，改进算法明显低于高冲突值行优先排列算法和列优先排列算法，综上所述，随着 n 渐渐增大，改进算法的平均比较次数低于高冲突值行优先排列算法和列优先排列算法。

观察图 3.9 可以发现，在矩阵相同的情况下，随着 n 渐渐增大，高冲突值行优先排列算法的平均交换次数明显低于列优先排列算法和改进算法，但是，改进算法的平均交换次数虽然比高冲突值行优先排列算法高，但是比列优先排列算法低。

表 3.1 高冲突值行优先排列算法性能统计表

阶数 n	平均比较次数	平均交换次数
10	854	14
20	7013	67
50	150302	483
100	4590631	2179
150	18907518	3708
300	35876942	5947

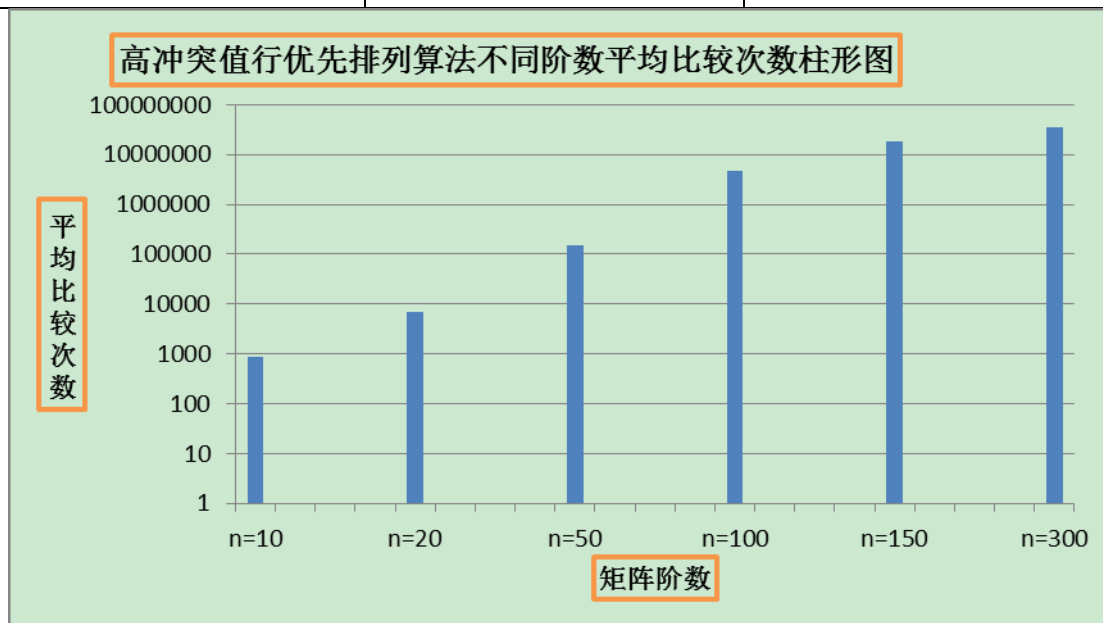


图 3.2 高冲突值行优先排列算法平均比较次数统计图

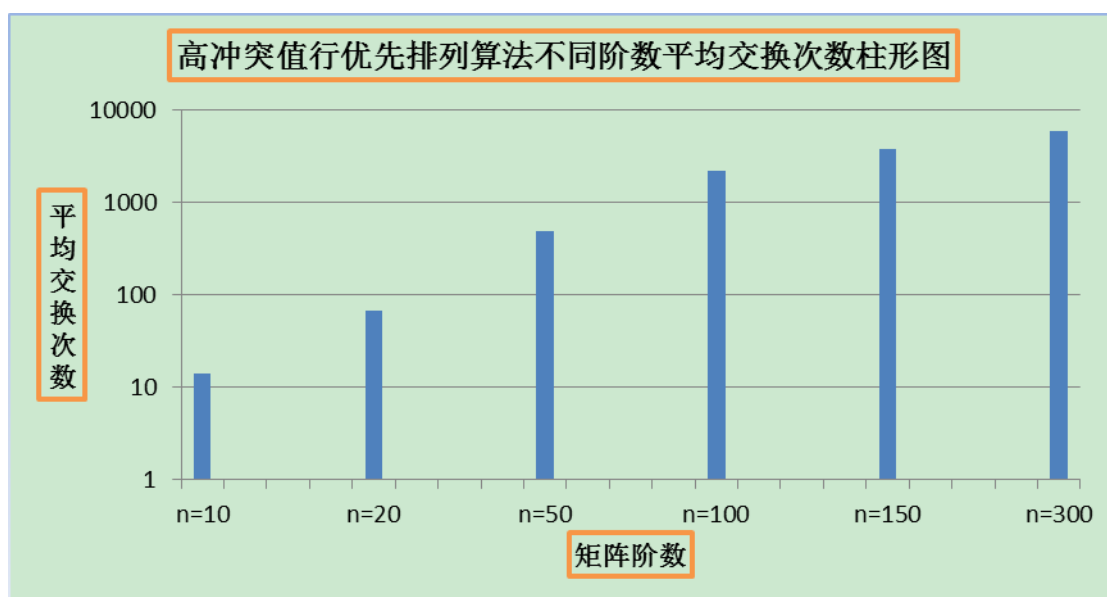


图 3.3 高冲突值行优先排列算法平均交换次数统计图

表 3.2 非回溯列优先排列算法性能统计表

阶数	平均比较次数	平均交换次数
10	434	33
20	4954	149
50	148907	977
100	4567307	3872
150	18568163	7055
300	35227187	8831

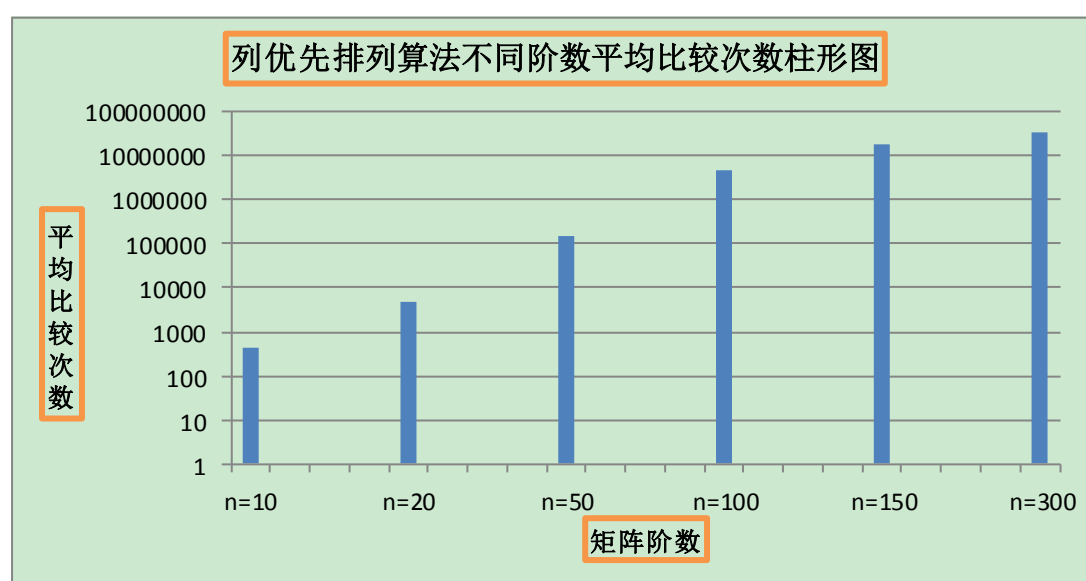


图 3.4 列优先排列算法平均比较次数统计图

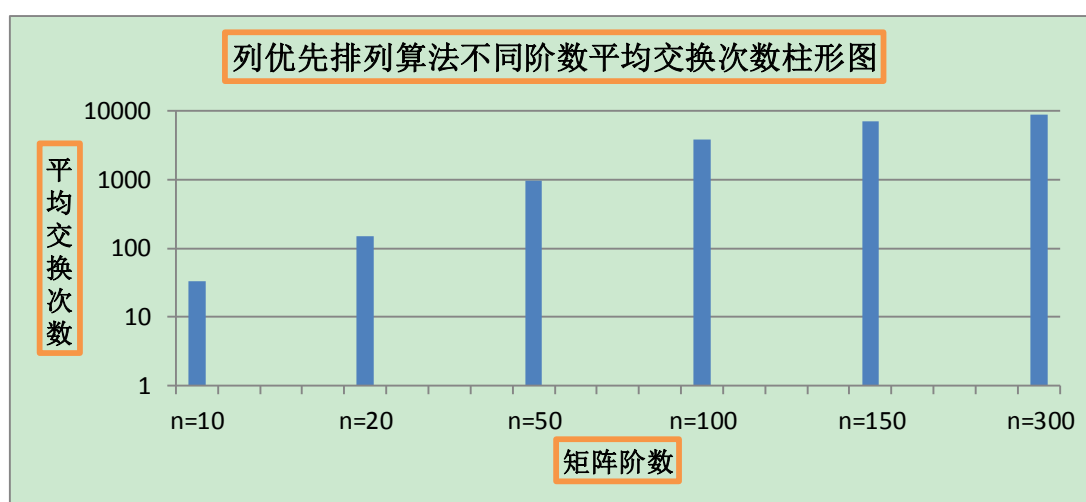


图 3.5 列优先排列算法平均交换次数统计图

表 3.3 改进算法性能统计表

阶数	平均比较次数	平均交换次数
10	336	5
20	4123	30
50	140855	456
100	4000284	2032
150	16245238	3211
300	30822347	5438

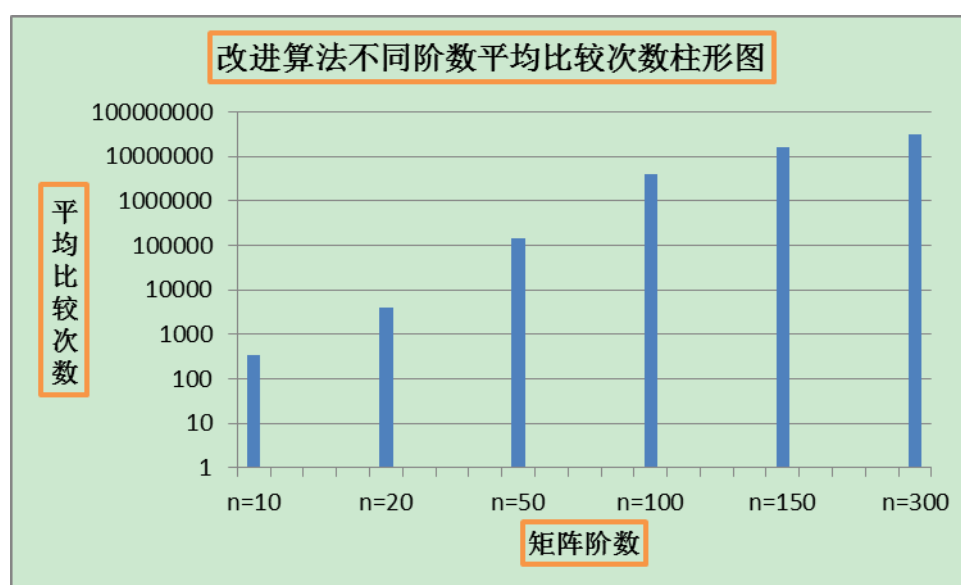


图 3.6 改进算法平均比较次数统计图

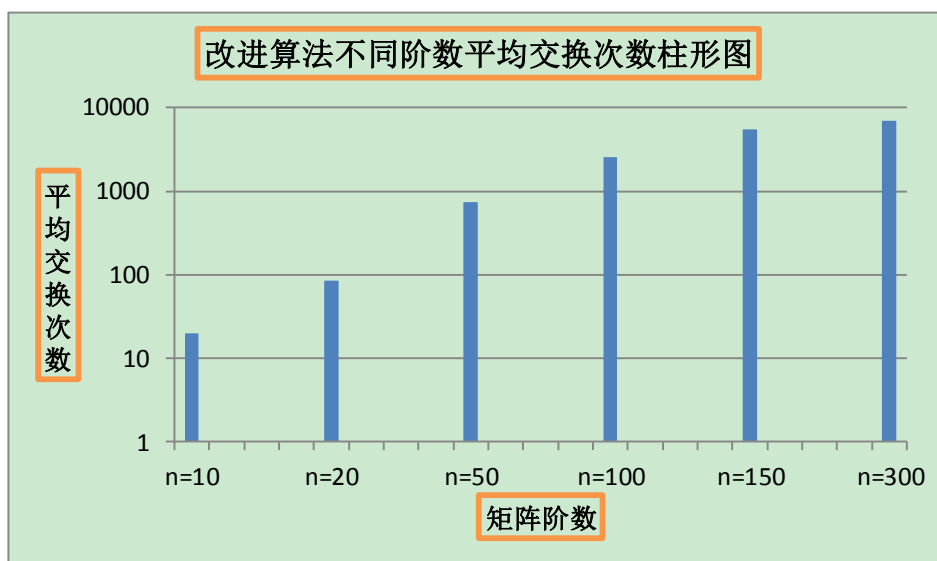


图 3.7 改进算法平均交换次数统计图

综合上述分析，可以得出如下结论，改进算法的时间复杂度低于高冲突值行优先排列算法和列优先排列算法，但是，改进算法的实现过程中用到链表这种数据结构来随时记录每一行中重复元素和缺少元素的信息，也就是说改进算法需要额外的内存空间来存储中间数据，所以，改进算法的空间复杂度高于高冲突值行优先排列算法和列优先排列算法。

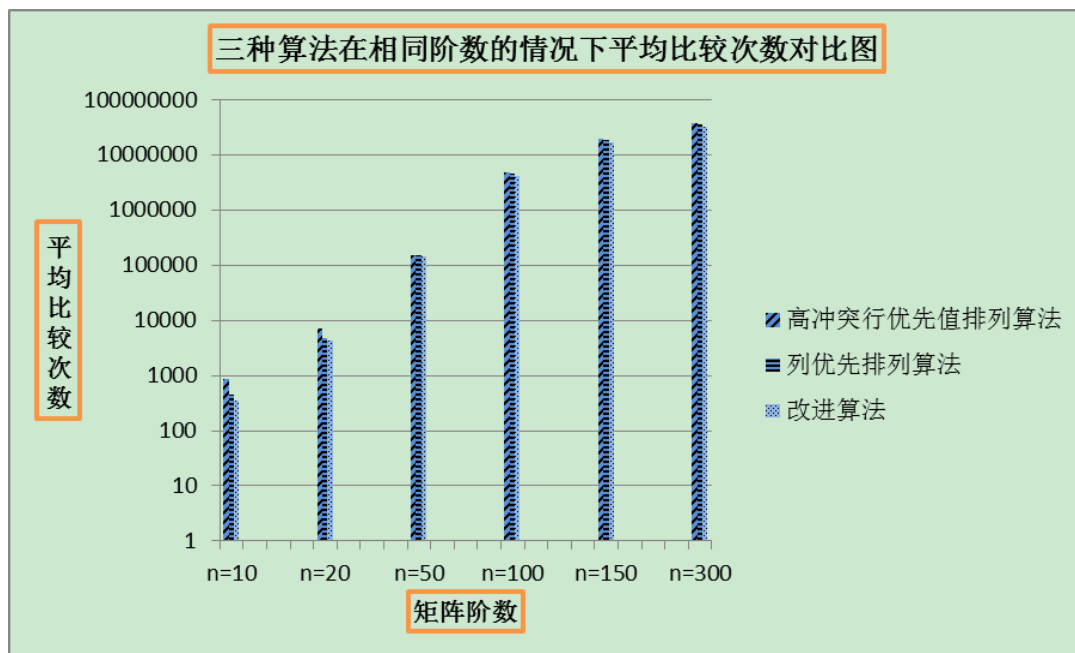


图 3.8 三种算法平均比较次数统计图

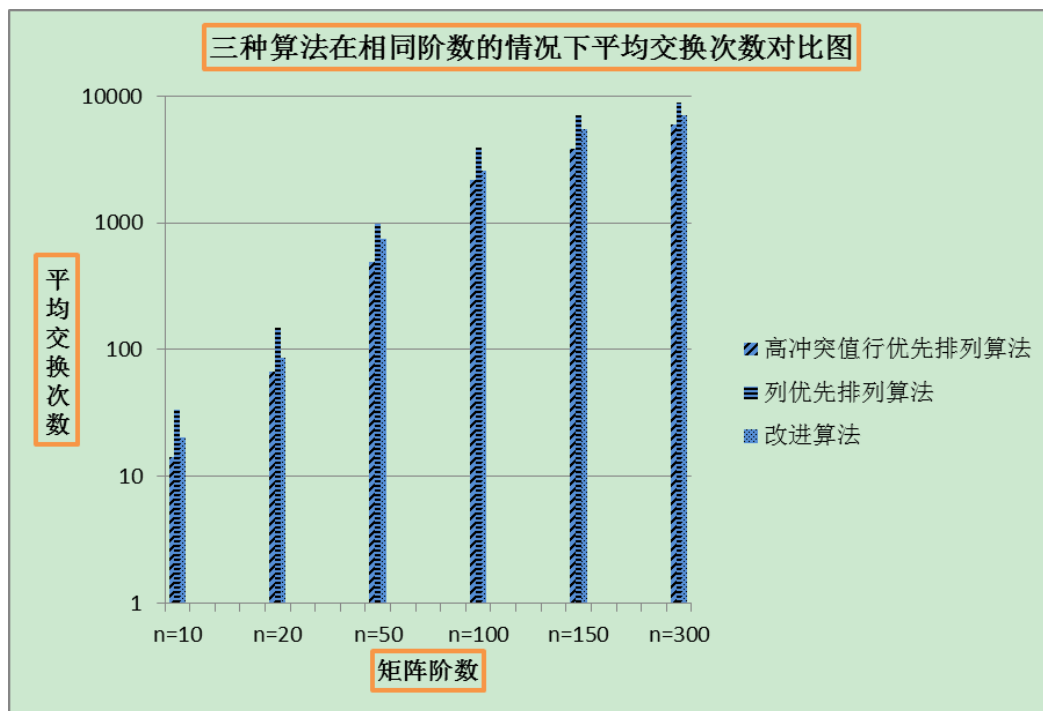


图 3.9 三种算法平均交换次数统计图

3.4 本章总结

本章在 2.4 小结建立的数学模型的基础上, 研究了程控电路交换网络的调度算法。首先回顾了文献[17]中提出的高冲突值行优先排列算法和列优先排列算法, 然后在已有算法的基础上提出了一种基于列优先排列的改进算法, 给出了算法的流程图, 通过柱状图比较了三种算法的比较次数和交换次数, 进而分析了三种算法的性能, 得出结论, 改进算法在时间复杂度上优于已有的两种算法, 付出的代价就是空间复杂度高于已有算法, 即通过牺牲存储空间而追求时间效率, 这对降低时延、支持天地一体化的路由刷新机制具有代表性意义。

第四章 交换帧结构设计和交换方案

4.1 交换网络交换帧结构的设计

在前一章介绍了 T-S-T 交换网络的调度算法，在建立的数学模型的基础上，给定一个输出矩阵 out_port ，根据一系列的数学方法，得到符合要求的中间矩阵 $middle_2$ 矩阵。这就说明，中间矩阵是随着输出矩阵的变换而变化，但是，输出矩阵什么时候变化呢，在什么情况下发生变化呢？问题随之而来，输出矩阵是随机变化的呢？还是按一定规律发生周期性变化呢？考虑到电路交换的特点，周期性变化易于控制，方便操作，性能高于随机变化。为了解决以上问题，下面将对如何设计 T-S-T 交换网络的交换帧长，顾名思义，就是 2.4.2 节建立的矩阵模型中，输出矩阵 out_port 发生改变的周期，即输出矩阵经过多长时间改变一次。

回顾所学知识，我们会发现 SS-TDMA（卫星交换-时分多址）系统中，分布控制单元也是周期变化的。下面，我们将简单介绍 SS-TDMA 系统的帧结构设计。

4.1.1 SS-TDMA 系统的工作原理

SS-TDMA (Satellite Switched-Time Division Multiple Access)，即卫星交换-时分多址卫星通信系统是一个多波束卫星通信系统，不同的波束覆盖不同的区域。为了实现波束之间的互相连通，星上部分设计了微波交换矩阵 MSM 和分布控制单元 DCU。在地面部分包括主控站和地面站，地面站位于不同的波束内且按时分多址方式工作。当系统工作时，首先，根据各波束覆盖区域内每个站的业务量需求，主控站要实时编排出一帧内微波交换矩阵 (MSM) 进行变换的连接状态序列和各地面站发送突发的时刻。然后，通过遥控遥测指令线路，主控站把编排结果送到分布控制单元的副寄存器，确认无误后再传送到主寄存器，与此同时，主控站也需将编排结果发送给各个地面站。最后，分布控制单元控制微波交换矩阵的二极管按编排好的序列变换连接状态，每个地面站则依照自己排定的时刻发送突发，继而实现波束之间的地面站信息传递。

4.1.2 SS-TDMA 系统的时隙编排

时隙编排，顾名思义就是编排一帧内 MSM 的连接状态序列。接下来给出时隙编排的数学模型。设 m_{ij} 表示第 i 个波束覆盖区发送到第 j 个波束覆盖区的业务量。对于一个有 N 个波束的 SS-TDMA 系统，交换矩阵如下表所示，行号表示系

统发射波束的覆盖区号，列号表示系统接收波束的覆盖区号， $\sum_{j=1}^N d_{ij} = S_i$ 表示第

i 个波束覆盖区一帧内发送给自身及其他波束覆盖区的业务量总和， $\sum_{i=1}^N d_{ij} = S_j$

表示第 j 个波束覆盖区一帧内接收的来自自身及其他覆盖区的业务量总和。

表 4.1 有 N 个波束的 SS-TDMA 系统交换矩阵表

	1	2	N	$\sum_j d_{ij}$
1	d_{11}	d_{12}		d_{1N}	S1
2	d_{21}	d_{22}		d_{2N}	S2
.....					
N	d_{N1}	d_{N2}		d_{NN}	SN
$\sum_i d_{ij}$	R1	R2		RN	

记交换矩阵为 D ， D_i 表示状态矩阵，每一个 D_i 对应 MSM 的一个连接状态，那么时隙编排的数学模型可以表示如下： $D = D_1 + D_2 + \dots + D_k$ ，这个等式表示将交换矩阵分解为一系列状态矩阵之和，式中的 k 值表示状态矩阵的个数，定义为模式数，分解过程满足如下 4 个条件：

- (1) 状态矩阵 D_i 的每一行和每一列至多有一个非零元素，保证 MSM 在第 i 个连接状态时波束之间的一一对应关系；
- (2) 模式数 k 值最小，保证 MSM 转换状态引入的过渡时间最小；
- (3) 各个 D_i 的最大元素之和最小，保证帧长最短，提高效率；
- (4) 分解速度尽可能快，保证信息及时传送；

衡量时隙编排算法的参数包括编排效率 η ($\eta = \frac{T_M}{T}$ ， $T_M = \sum_{i=1,2,\dots} \{S_i, R_i\}$ ， T 表

示帧长)、模式数 k 和运算复杂度 $O(N^s)$ 。

4.1.3 基于 FDMA 多址接入方式的帧长方案设计

首先，定义两个矩阵，容量交换矩阵和连接状态矩阵。

系统中有 16 个端口，每个端口 48 个子信道，一共有 768 (16*48) 路信号，

这样，第一个端口的信号编号为 1-48，第二个端口的信号编号为 2-96，以此类推，第 16 个端口的信号编号为 720-768。

容量交换矩阵如表 4.2 所示，其中 c_{ij} 表示从 i 信道到 j 信道的通信容量。

连接状态矩阵表如表 4.3 所示，其中 0 表示没有链路连接，1 表示有链路连接。例如，矩阵中，第一行第二列元素为 1，表示第一条链路和第二条链路接通，即第一条链路的用户信息交换到第二条链路。

表 4.2 容量交换矩阵表

	1	2	768	$\sum_j c_{ij}$
1	c_{11}	c_{12}		c_{1768}	S1
2	c_{21}	c_{22}		c_{2768}	S2
.....					
768	c_{7681}	c_{7682}		c_{768768}	S768
$\sum_i c_{ij}$	R1	R2		R768	

表 4.3 连接状态矩阵表

	1	2	768
1	0	1		0
2	1	0		0
.....	0			
768	0	0		1

程控电路交换的工作过程包括三个基本阶段：呼叫建立、连接通信和释放连接。当新的信号申请接入时，NCC 根据申请要求，先改变交换矩阵，建立链路连接，才允许新的信号接入，所以信号在进入网络之前，交换矩阵已经发生改变，新的交换矩阵已经形成。

在文献[19]中指出，GEO 卫星的传输业务主要分为话音业务、固定速率数据业务、多媒体业务、尽力而为业务和切换业务。其中，每种业务的呼叫请求到达速率服从泊松分布，持续时间服从指数分布，在四类业务中，业务呼叫请求的到达概率服从均匀分布，对于切换业务而言，假设四类业务发生切换的概率为 0.01，切换业务的等效带宽的持续时间由其他四类业务本身决定。传输业务的优先级权重、等效带宽和持续时间均值参数设置如表 4.4 所示：

在系统中，星上程控电路交换网络主要传输多媒体业务。从表 4.4 可知，其中多媒体业务的等效带宽为 100-500kbps,持续时间均值为 900s，所以多媒体业务

的通信容量均值为 90000-450000Kbit，即 8.8Mbit-439.5Mbit，计算出系统的传输速率，即可求出交换帧长。如果按带宽为 2.6MHz 为传输速率，则帧长取值范围为 3.38s-169.04s。用户信息从进入网络到到达星上程控交换模块大约需要 0.12(距

表 4.4 传输业务参数设置

业务类型	等效带宽 (kbps)	平均持续时间 (s)	业务优先级权重
话音业务	13	180	8
固定速率数据业务	20	200	5
多媒体业务	100-500	900	2
尽力而为业务	0-20	400	1
切换业务	——	——	10

离：3.6 万公里，光速：30 万公里/s，时间：3.6/30=0.12s)+0.02（星上处理时间）=0.14s=140ms。NCC 在允许新的用户接入时，新的交换矩阵已经形成，每条链路的用户信息陆续到达星上，在用户信息到达星上的这段时间，星上没有进行交换，这段时间即 0.14s 可以作为交换帧的帧头，用来每条链路进入程控电路交换网络之前的同步时间。程控电路交换需要用户发送释放连接信号时，链路连接才会断开，这段时间可以作为交换帧的预留时间。交换帧结构如图 4.1 所示：

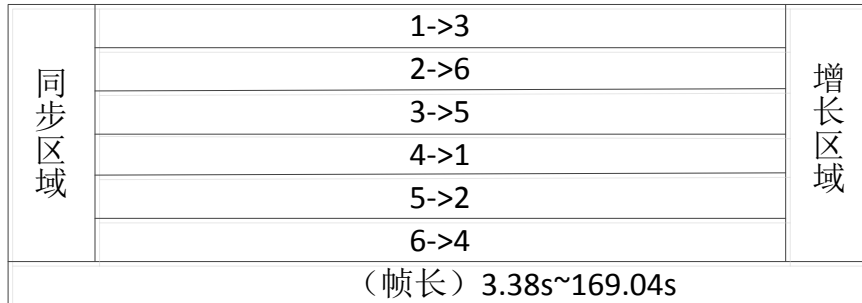


图 4.1 交换帧结构

其中，{1,2,3,4,5,6}分别表示进行交换的单元，可以表示一个波束，一个频带信息，也可以表示一个时隙的信息。

NCC 在交换帧期间，定期统计容量矩阵，利用文献[18]提出的 SDR 算法，以通信容量的多少为主要衡量准则，文献[19]给出的业务优先级为次要衡量要素，找出一组交换状态模式，得到连接状态矩阵，即输出矩阵，用于下一帧的交换。

下面详细介绍如何根据容量交换矩阵求得状态连接矩阵。

首先，文献[18]中提到的 SDR 算法的计算过程如下：定义 S_1, S_2, \dots, S_n 是有限集合 E 的子集，在这里， S_i 和 E 的元素为 $n \times n$ 的容量交换矩阵的行数或列数。如果子集 S_i 包含一个特别的元素 X_i ，使得当 i 不等于 j 时， X_i 不等于 X_j ，那么 X_1, X_2, \dots, X_n 叫做 S_1, S_2, \dots, S_n 的一组代表值即 SDR 值。

例如，对于一个 4×4 的矩阵， $E=\{1,2,3,4\}$ ，如果 $S1=\{1, 2\}$ ， $S2=\{2, 4\}$ ， $S3=\{2, 3\}$ ， $S4=\{1, 3\}$ ，那么取得 $\{2,4,3,1\}$ 为 $S1$ ， $S2$ ， $S3$ ， $S4$ 的一组 SDR 值，这表示，在容量交换矩阵中，取第一行的第二列元素，第二行的第四列元素，第三行的第三列元素，第四行的第一列元素，即交换状态为 $1 \rightarrow 2, 2 \rightarrow 4, 3 \rightarrow 3, 4 \rightarrow 1$ 。其中，在取 S_i 的元素时，遵循一个原则，取第 i 行中元素较大的列值。

NCC 得到状态连接矩阵（即输出矩阵）之后，根据调度算法，寻找 T-S-T 网络中的交换路径。

4.2 基于 FDMA 多址接入方式的程控交换方案设计

因为采用 FDMA 多址接入方式，所以没有时隙的概念。但是在 T-S-T 交换网络中，时隙数目代表了用户数，根据类比思想，时隙数目也就是端口的子信道数目，所以本方案中控制存储器的单元数等于每个端口的子信道数。

本文采用 FDMA 的多址接入方式，用户信号经过非均匀数字信道化处理，进入星上路由交换模块，实现频带之间的交换。系统中，可用频带带宽 125MHz，有 16 个端口，每个端口 48 个子频带，所以每个端口的带宽为 2.6MHz。

4.2.1 基于单播功能的程控方案设计

(一)端口内不同子信道之间的信号交换

图 4.2 实现了端口内第 2 路子信道和第 48 路子信道之间的样点交换。图 a 和图 b 的不同是，数据存储器的控制方式不同。

在图 4.2 (a) 中，数据存储器控制写入，顺序读出。每个端口的 48 路样点信号经过串并转换和复用后进入交换网络，交换网络时钟频率等于样点速率。控制存储器由交换控制系统控制写入，然后按顺序读出。交换控制系统先将单元号 48 写入控制存储器的 2 号单元，当时钟信号到来时，控制存储器将子信道 48 的内容写入数据存储器的 2 号单元，即实现了子信道 48 的内容在子信道 2 输出；同理也可实现子信道 2 的内容在子信道 48 输出。

在图 4.2 (b) 中，数据存储器顺序写入，控制输出。每个端口的 48 路样点信号经过串并转换和复用后进入交换网络，交换网络时钟频率等于样点速率。控制存储器由交换控制系统控制写入，然后按顺序读出。数据存储器由时钟提供写地址，样点信号顺序写入数据存储器，控制存储器控制子信道 2 输出子信道 48 的内容，即实现了子信道 48 的内容在子信道 2 输出；同理也可实现子信道 2 的内容在子信道 48 输出。

(二)不同端口之间的子信道信号交换

IDH1~IDH16 表示 16 个输入端口，ODH1~ODH16 表示 16 个输出端口，每个端口的 48 个子信道信号复用为 1 路信号。图 4.3 (c) 和图 4.3 (d) 都实现了端口

1 的子信道 3 输出到端口 2 的子信道 3、端口 2 的子信道 3 输出到端口 16 的子信道 3、端口 16 的子信道 3 输出到端口 1 的子信道 3、端口 1 的子信道 48 输出到端口 16 的子信道 48、端口 2 的子信道 48 输出到端口 1 的子信道 48 和端口 16 的子信道 48 输出到端口 2 的子信道 48。其中，图 4.3 (c) 采用了输出控制方式，图 4.3 (d) 采用了输入控制方式，下面以端口 1 的子信道 3 输出到端口 2 的子信道 3 为例说明原理。

在输出控制方式中，控制存储器号与输出端口号一致，控制存储器的单元号与子信道号对应，单元中的内容与输入端口号相对应。交换控制系统在 CM2 的 3 号单元写入 1，在时钟到来时，CM2 读出的内容为 1，控制端口 1 和端口 2 的交叉接点闭合，使端口 1 的子信道 3 输出到端口 2 的子信道 3。

在输入控制方式中，控制存储器号与输入端口号一致，控制存储器的单元号与子信道号对应，单元中的内容与输出端口号相对应。交换控制系统在 CM1 的 3 号单元写入 2，在时钟到来时，CM1 读出的内容为 2，控制端口 1 和端口 2 的交叉接点闭合，使端口 1 的子信道 3 输出到端口 2 的子信道 3。

可以发现，空间接线器主要用来实现端口之间的交换，但子信道位置不变。

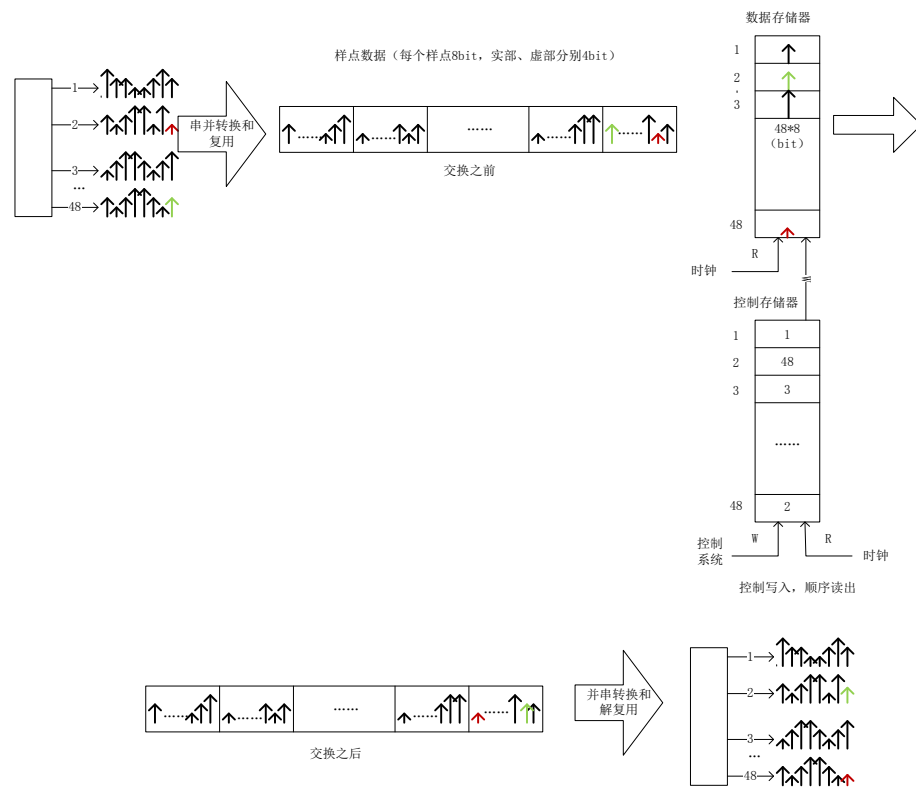
(三)不同端口之间的不同子信道信号交换

星上电路交换模块结构如图 4.4 所示：选择模块是 T-S-T 交换网络与 CPU 处理器(控制器)之间通信的桥梁与纽带，CPU 处理器接收通信双方的路由信令，通过选择模块填写相应的控制存储器，在 T-S-T 交换网络中为通信双方寻找一条空闲路由，通信双方之间的建立与释放即可完成。卫星系统有 16 个端口，先将每个端口的子信道信号都复用成一路信号，然后经过 T-S-T 交换网络。其中，T 接线器可以实现端口内的子信道之间的交换，S 接线器能够实现端口间的交换功能，整个 T-S-T 网络就可以实现输入端的任意一路子信道信号在输出端口的任意一个子信道输出。类比 T-S-T 电路交换原理，每来一个时钟，写入一个样点信号。方案中采用 16 个初级 T 接线器、16 个次级 T 接线器和 16 个中间级 S 交换器。

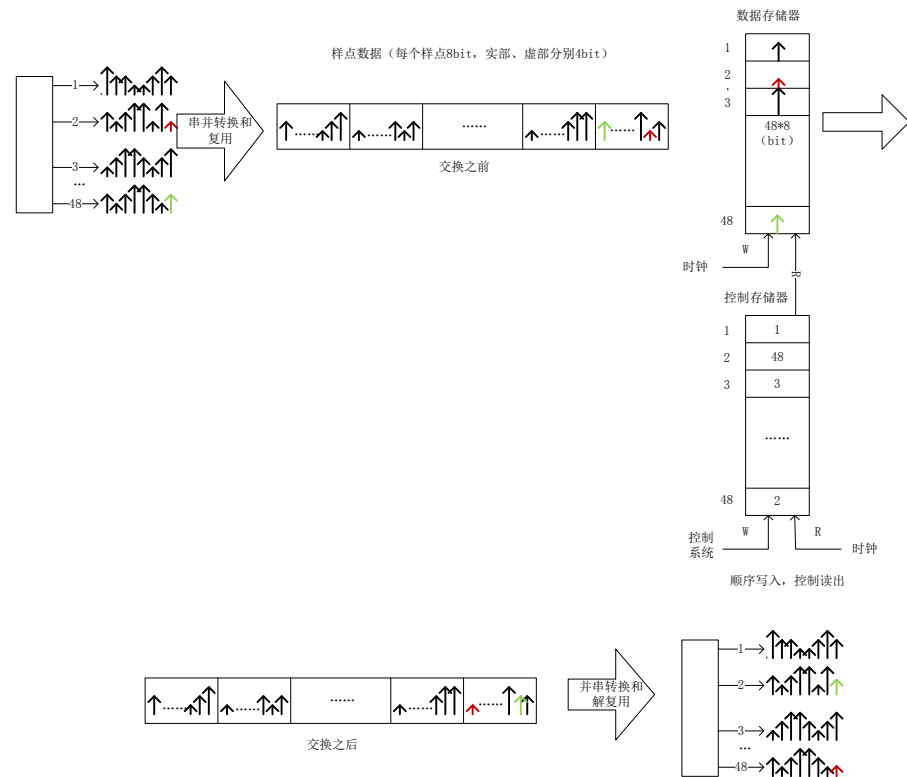
下面将用一个例子来说明星上交换网络的工作原理，如图 4.5 所示：初级 T 接线器和次级 T 接线器分别工作在输入控制方式和输出控制方式下，S 接线器工作在输出控制方式下。

在图 4.5 中，初级 T 接线器的数据存储器用 $SMA_1 \sim SMA_{16}$ 表示，控制存储器用 $CMA_1 \sim CMA_{16}$ 表示；次级 T 接线器的数据存储器用 $SMB_1 \sim SMB_{16}$ 表示，控制存储器用 $CMB_1 \sim CMB_{16}$ 表示；S 接线器的控制存储器用 $CMC_1 \sim CMC_{16}$ 表示。图 4.5 实现了端口 1 的第 5 路和端口 16 的第 20 路之间的交换，具体实现过程如下：

(一)A 到 B 的通路：端口 1 的第 5 路到端口 16 的第 20 路

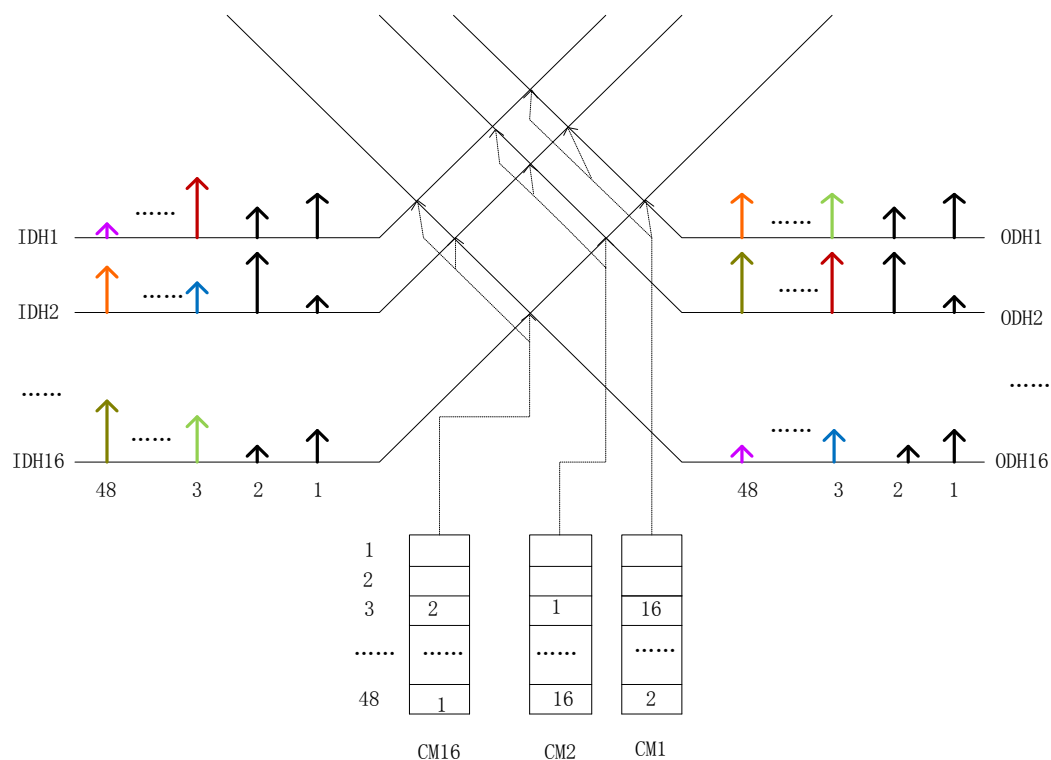


图a 控制写入，顺序读出

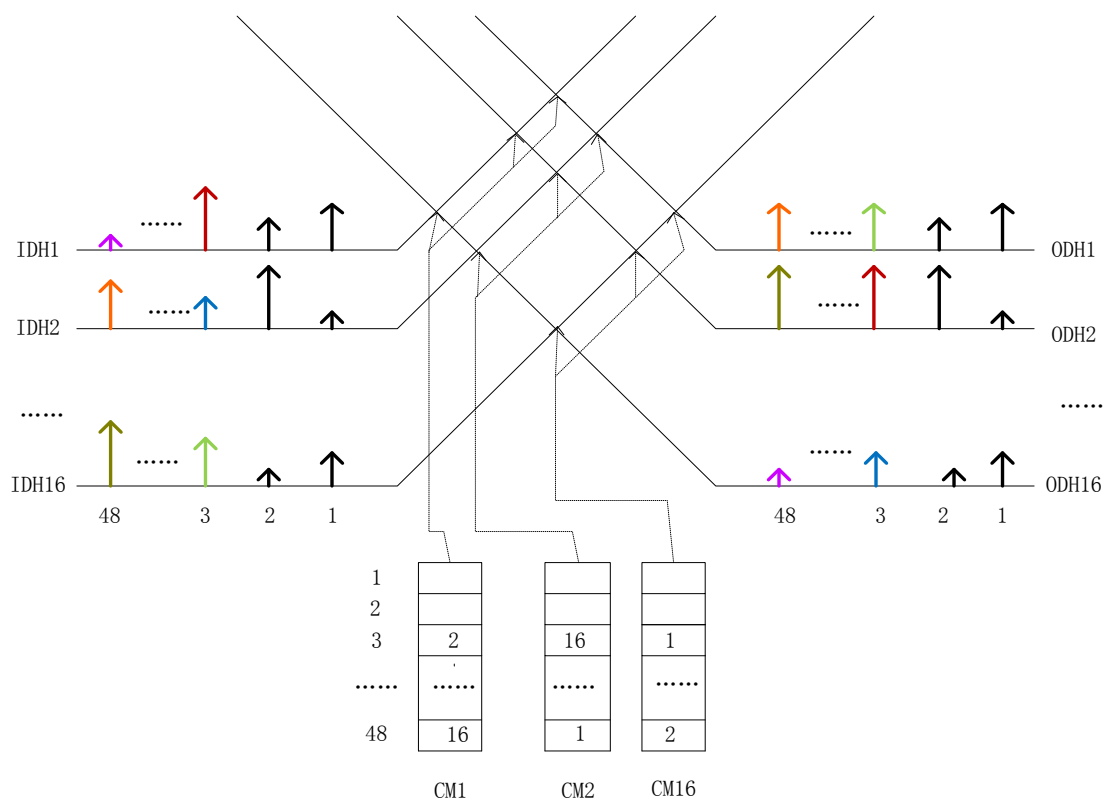


图b 顺序写入，控制读出

图 4.2 端口内不同子信道之间的信号交换

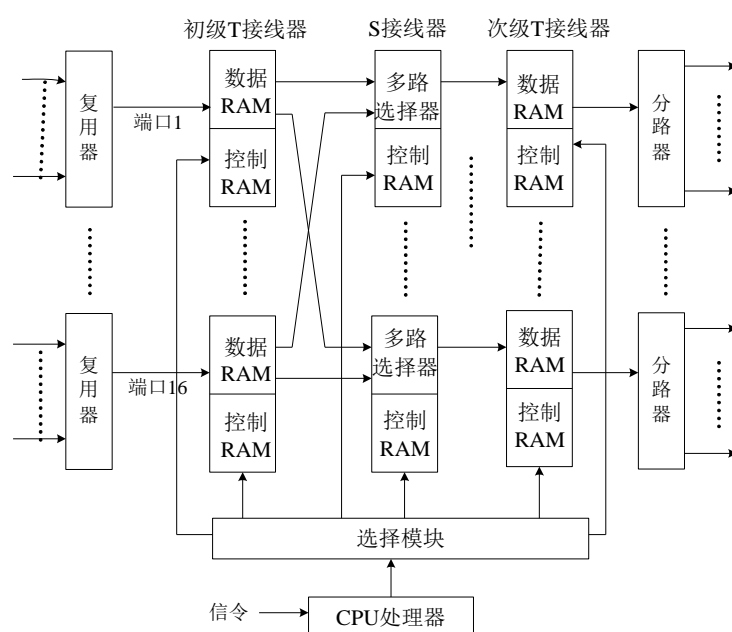


图c 输出控制方式



图d 输入控制方式

图 4.3 不同端口之间的子信道信号交换



[16]

图 4.4 星上交换网络结构框图

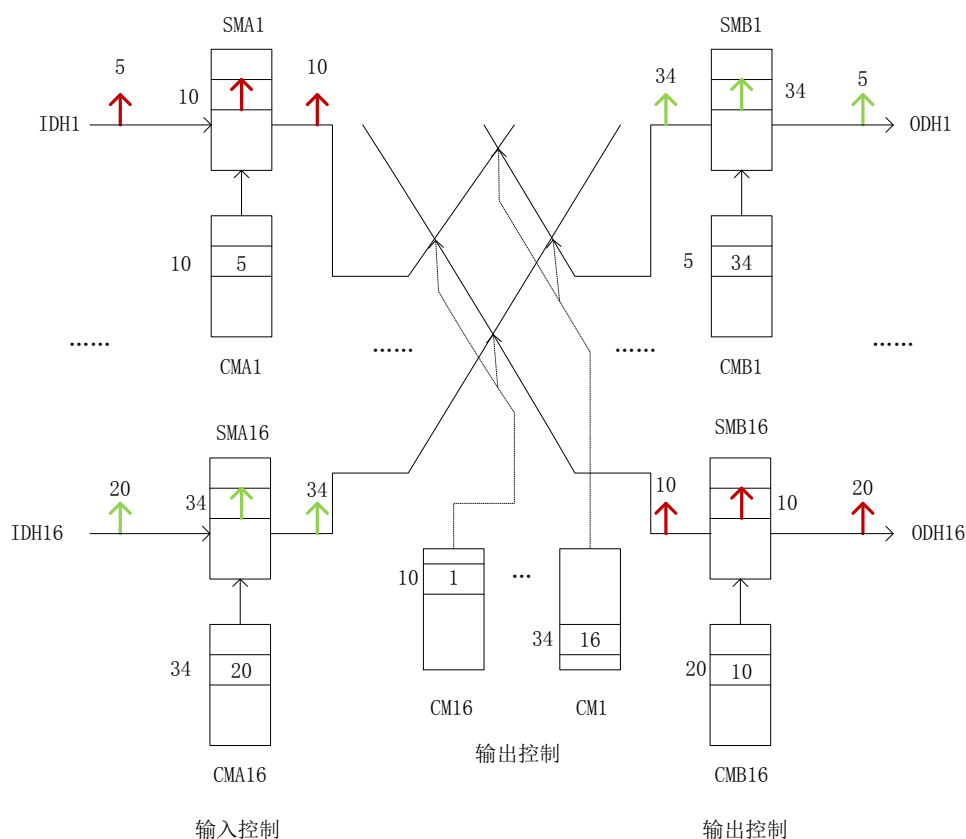


图 4.5 交换网络的接续原理

A 到 B 先经过初级 T 接线器与中间子信道(假定中间子信道为 10)，因为初级 T 接线器为输入控制方式，如图 4.5 所示，交换控制系统先将单元号 5 写入 CMA_1 的 10 号单元，当时钟到来时通过 CMA_1 将子信道 5 的内容控制写入 SMA_1 的第

10 号存储单元，在时钟脉冲下顺序读出，完成子信道 5 与子信道 10 的交换。为了将子信道 10 从端口 0 交换到端口 16，根据 S 接线器的输出控制方式需要在 CMC_{16} 第 10 号单元内填入端口号 0，当时钟到来时，对应的交叉点闭合，完成子信道的空间搬移。接下来的次级 T 接线器的动作与初级 T 接线器类似，完成子信道 10 和子信道 20 的内容交换，不再细述。

(二)B 到 A 的通路：端口 16 的第 20 路到端口 1 的第 5 路

B 到 A 的接续流程与 A 到 B 的一样，需要注意的是：A 到 B 的通信中，A 用户所占的子信道 5 为发话信道，B 用户所占的子信道 20 为受话信道；B 到 A 的通信中，B 用户所占的子信道 20 为发话信道，A 用户所占的子信道 5 为受话信道。用户占用的信道既是发话信道又是受话信道，只是发话信道总是在输入侧，受话信道总是在输出侧。

(三)中间信道的选择

为了方便选择和简化控制，正反两个方向的中间子信道一般都具有一定的对应关系，通常采用反相法或奇偶法。在图 4.5 中就是采用的反相法，若正向连接的中间信道选择 i ，则反向连接的中间时隙为 $(i + F/2) \bmod F$ ，其中 F 为复用线上子信道数。图 4.5 中 F 为 48，所以当正向路由的中间子信道为 10 时，反向路由的中间时隙为 34。

当双方通信结束后，处理器将各控制存储器相应单元的内容清除。

4.2.2 基于多播和广播功能的程控方案设计

在有些应用中，主机需要给其他多个或者所有主机发送信息。例如，用于发布天气预报的服务、股市行情最新报告或者现场直播节目等，它们的最佳工作方式是，将消息广播给所有的机器，然后让那些感兴趣的机器读取数据。同时给所有目标发送一个分组，这称为广播^[11]。

在分布式数据库系统中，位于不同主机的进程需要组合成一个整体协同工作。像类似于这样的应用要求，一个分组的进程需要给这个分组所有的成员发送相同的信息，当这个分组的规模比较小时，只需要用点到点的单播方式将消息发送给分组中的其他进程，当这个分组的规模比较大时，点对点的传播方式造成的系统代价太大，在有些情况下会选择广播方式进行通信。但是，如果一个网络有上百万个节点，使用广播机制来给 1000 台机器发送信息会非常低效，因为这个网络中绝大部分接收者对于接收到的消息并不感兴趣，或者更糟的是，即便他们有兴趣，实际上他们却不应该看到这些信息。如果有这样的一种方法，允许只给需要接收消息的分组进程发送消息，可以同时避免上述单播和广播方式的不足。像这样，只发送信息给某些有明确定义的组，这些组虽然有很多成员，但是与整个网络规模相比，其数量却是很小的，给这样一个组发送消息称为多播^[11]。

(1) 端口内组播、广播的实现

如图 4.6 所示，例如，需要将端口内第 2 路信号组播给其中几路用户，只需将对应路控制存储器内写入 2 即可。如果来实现广播功能，即将端口内所有控制存储器写入 2。

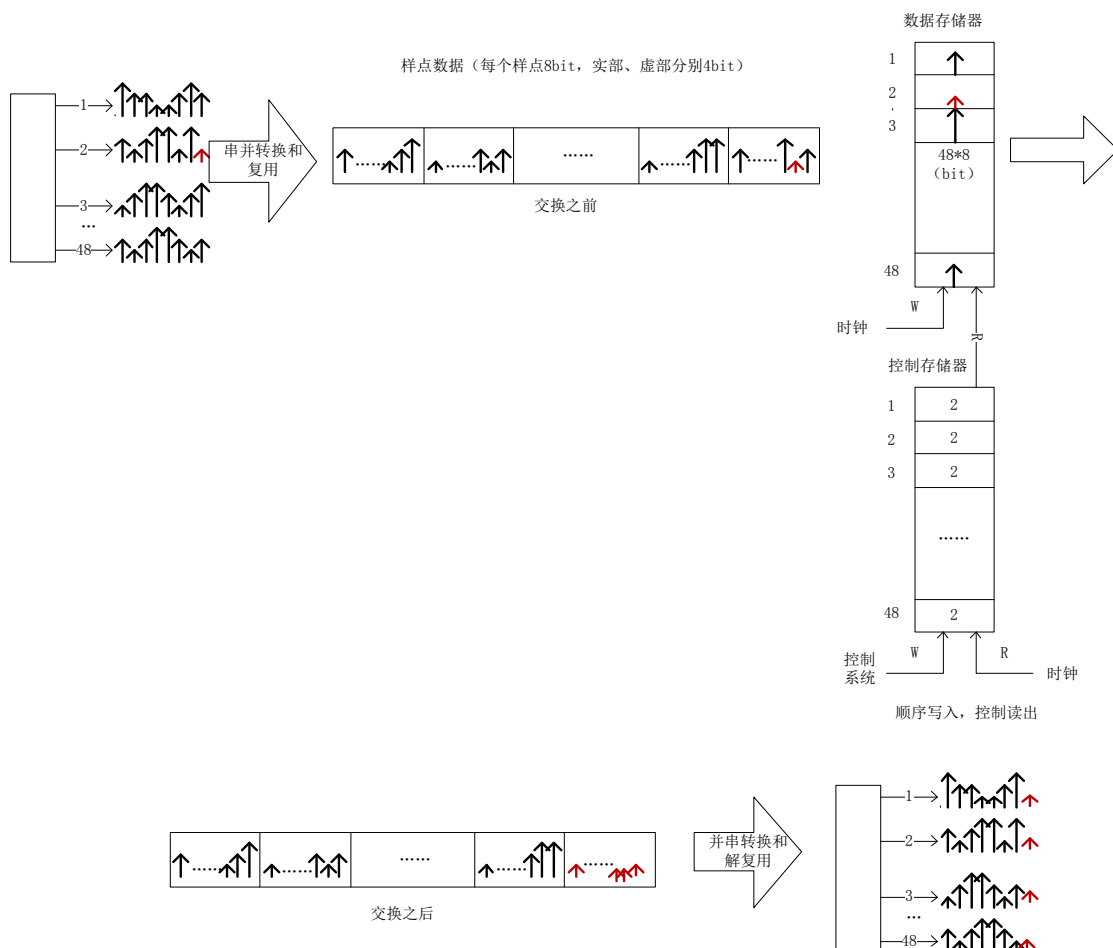


图 4.6 端口内广播实现方式原理图

(2) 不同端口间组播、广播的实现

要实现端口间组播和广播的功能，在 T-S-T 网络中，S 接线器只能采用输出控制方式。输出控制方式的优点是可实现组播，即某一端口的某一子信道的内容可以同时所需的几个端口的同一子信道输出。实际上，当要实现一个输入端口中相应子信道的组播功能时，需要将目的子信道的控制存储器在对应的存储单元内写入相同的子信道编号。

当一个时钟到来时，输入端口的对应子信道信号只能与目的端口的一个子信道交换，所以要实现组播，先通过 T-S-T 交换网络将要传送的信息交换到目的端口的其中一个子信道，然后在次级 T 接线器实现端口内组播功能。

同理，要实现广播，需要将所有子信道的控制存储器在对应的存储单元写入相同的子信道编号。

下面，通过一个实例具体说明实现方式，如图 4.7 所示：

在图 4.7 中，实现了端口 1 的第 5 路信号组播给所有用户。首先，先假设所有端口的中间子信道 10 处于空闲状态，初级 T 接线器将第 5 路信号交换到第 10 路信号，交换控制系统将 16 个 S 接线器的控制存储器的第 10 个单元全部写入 1，这样每个端口的第 10 路都输入端口 1 的第 5 路信号。最后，每个端口的第 10 路信号通过次级 T 交换器实现端口内的组播和广播功能。

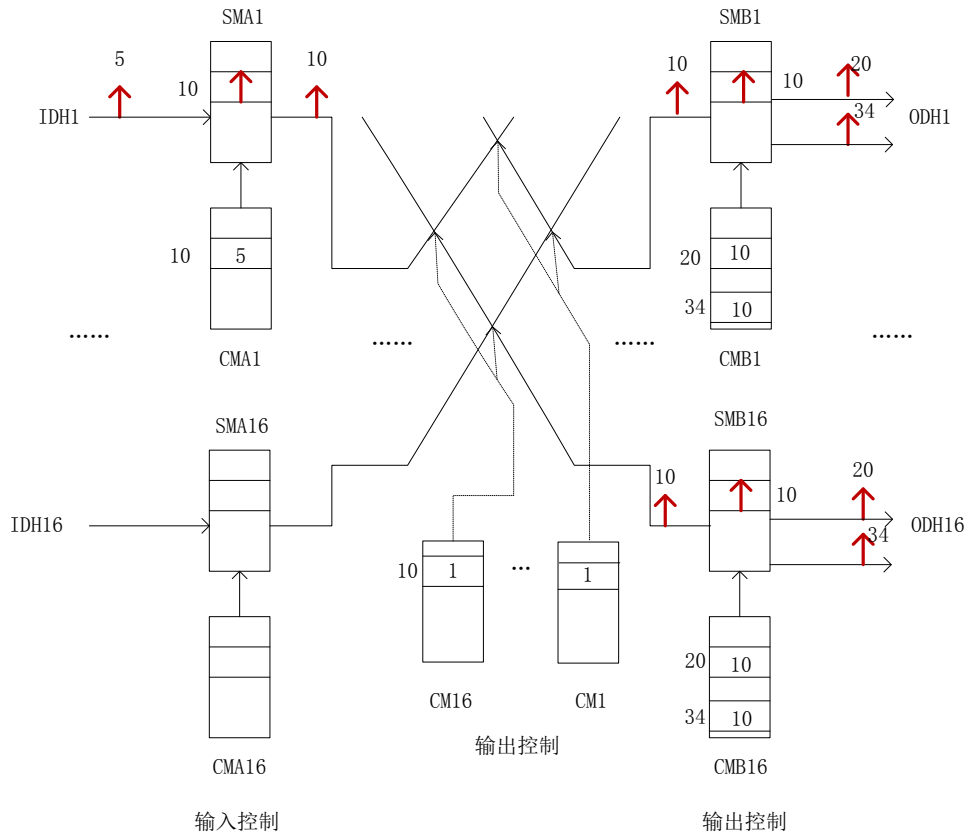


图 4.7 端口间组播实现方式原理图

4.3 本章总结

本章首先设计了 T-S-T 交换网络的交换帧，然后提出了基于 FDMA 多址接入方式的程控交换方案，实现了单播、组播和广播的功能。

第五章 程控交换网络的 FPGA 设计

5.1 芯片的选择

塞灵思公司在 FPGA 开发领域一直是领头羊, Spartan 类和 Virtex 类芯片是它的主流产品, 两者之间的差异主要在于芯片的规模和专用模块, 前者属于中低端应用, 成本在目前所有芯片中居于最低, 后者属于高端应用, 是行业中的顶级产品^[10]。

鉴于 Virtex-6FPGA 芯片在与同款 40nmFPGA 产品的比较中, 突出降低能耗、提高性能、压缩成本的特点, 在项目中, 我们采用了 Xilinx 芯片当中的 Virtex-6, 下面将简单介绍有关 Virtex-6FPGA 芯片的一些特点。

与之前芯片性能比较, Virtex-6FPGA 系列性能提高了 15%, 功耗降低了 15%, 可操作的内核电压为 1.0V, 此外还包括 0.9V 的低功耗版本。Virtex-6FPGA 芯片包括 LXT、SXT 和 HXT 三个系列, 其中, LXT 系列和 SXT 系列中嵌入的 DSP 具有超高性能, 串行连接能力与低功耗 GTX6.5Gbps 串行收发器相同, 但是前者还有高性能逻辑的优点, HXT 系列具有最高的串行连接能力, 比如 64 个 GTH 串行收发器能够提供的带宽有 11.2Gbps。

在系统中, 信号进入交换模块之前需要经过信道化器, 这就要求 DSP 模块性能不能太低, 所以我们采用了 SXT 系列。在 SXT 系列中, 包括 SX315T 和 SX475T, 数字代表了芯片的逻辑单元个数, 粗略估算系统的资源需求, 项目组决定采用 XC6VSX315T^[21]。

XC6VSX315T 的 SystemClock 分为全局时钟和局部时钟, Maximum Distributed RAM 为 5090kbits, Block RAM/FIFO (36kbits each) 为 704 kbits, Total Block RAM 为 25344 kbits, Mixed Mode Clock Manager 为 12 个, DSP48E1 Slices 为 1344 个, PCI Express Interface Blocks 为 2 个, 10、100、1000 Ethere MAC Blocks 为 4 个, GTX low-Power Transceivers 为 24 个, FF1750 封装类型, Maximum User I/O:Select I/O Interface Pins(GTX Transceivers)为 720 个。

综上所述, 本文采用 Vitrex-6 实现程控交换模块, 应用 Xilinx ISE14.1 作为集成开发工具, 用 ISE 自带的 ISim 和 Modelsim 作为仿真工具, 用 Verilog HDL 语言实现。

5.2 各个模块的 FPGA 设计及仿真

5.2.1 数据输入模块设计

用户信号采用 32bit 的复信号表示，其中实部和虚部分别为 16bit。由于芯片 I/O 口数量有限，所以信号采用串行输入方式，16bit 的实部信号和 16bit 的虚部信号分别经过移位寄存器和 D 触发器（起到时延的作用）产生 32bit 的复信号，图 5.1 是串行转换模块的具体实现：

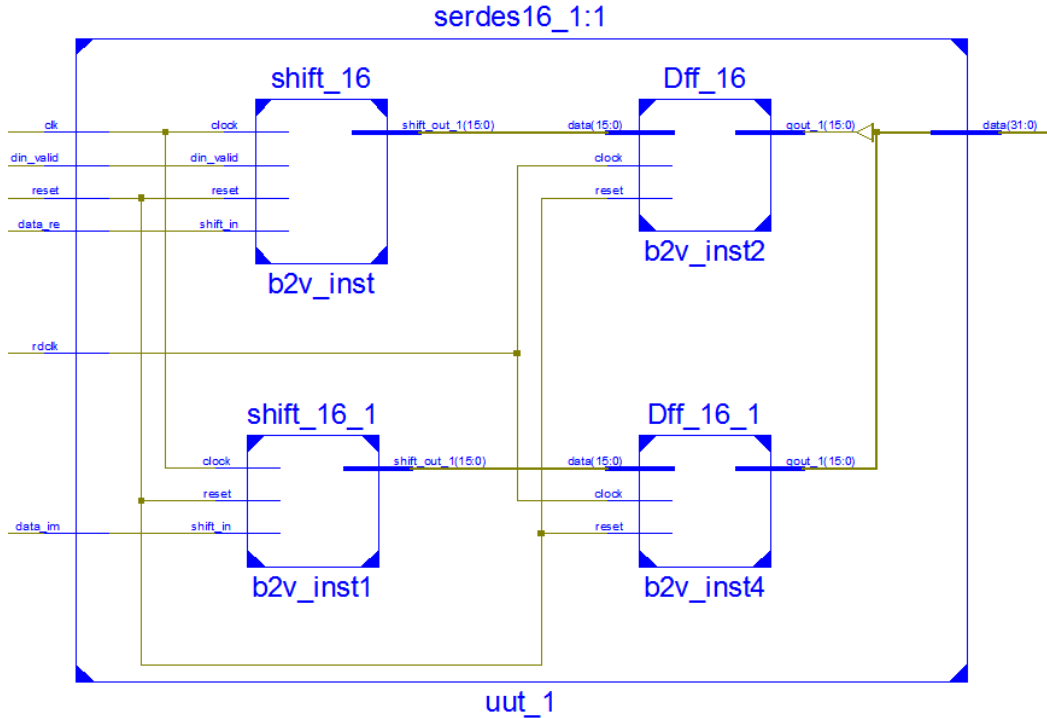


图 5.1 串行转换模块图

5.2.2 时钟模块设计

在 2.1 小节介绍的基于柔性转发器的卫星通信系统框图中，信号经信道化器处理之后，每一路的信号带宽为 5.2MHz，数值由信道化器的研究者提供，在此不做详细介绍。所以进入交换模块的时钟频率为 5.2MHz。在数据输入模块中提到，数据复信号的实部和虚部分别采用串行输入方式，所以数据输入模块的时钟频率为 $5.2\text{MHz} \times 16 = 83.2\text{MHz}$ ，用 clk 表示。数据输入模块的输出信号以并行方式进入数据存储器，所以并行信号的时钟频率为 5.2MHz，所以每个端口的 48 个子信道信号复用到一条干线上的时钟频率为 $5.2\text{MHz} \times 48 = 249.6\text{MHz}$ 。下文会提到，数据存储器的读操作和写操作不能同时进行，所以读时钟和写时钟用反相时钟，即相差半个时钟频率，分别用 clk_a 和 clk_b 表示。

方案中采用 PLL 模块产生三个时钟信号，PLL 模块的输入时钟为 5.2MHz，输出时钟为 clk、clk_a 和 clk_b。时钟信号的仿真结果如下图 5.2 所示：

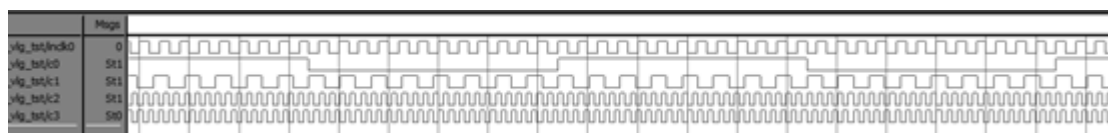


图 5.2 时钟信号

5.2.3 端口内样点交换模块设计

图 5.3 是初级样点交换模块的顶层模块电路图，位宽为 32bit 的复信号分实部和虚部各 16bit 进入端口内样点交换模块，经内部交换之后，输出信号。

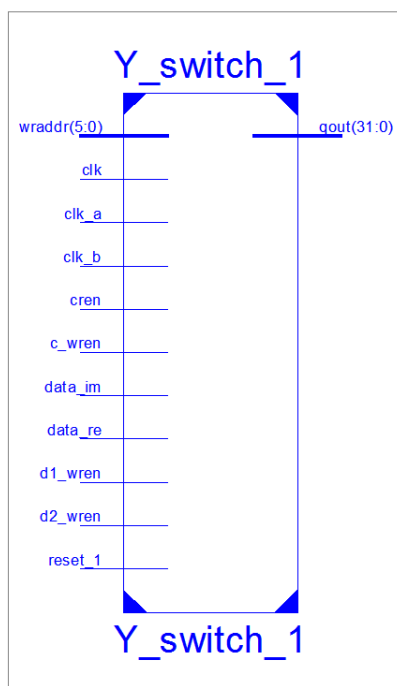


图 5.3 输出信号

图 5.4 是端口内子信道间交换的内部实现方案：核心是两个数据存储器和一个控制存储器。大家都知道，数据存储器是单工操作，即同一个时刻写操作和读操作只能进行一个，为了使得读数据流和写数据流没有间断，方案中采用两个数据存储器，用一个 96 位的计数器来实现此操作，把计数器的输出和数据存储器 1 的写使能端和数据存储器 2 的读使能端连起来，把计数器的输出通过一个非门和数据存储器 1 的读使能端和数据存储器的 2 的写使能端连起来，当计数为 0~47 或计数等于 95 时，输出 0，数据存储器 1 的读使能端有效，数据存储器 2 的写使能端有效，当计数为 48~95 时，输出 1，数据存储器 1 的写使能端有效，数据存储器 2 的读使能端有效，这样，两个数据存储器交替进行读数据和写数据，从左边看上去，数据源源不断地流入，从右边看上去，数据源源不断地流出。

图 5.5 是端口内子信道之间样点交换的仿真结果图：图中实现了子信道 47 和子信道 6、子信道 4 和子信道 12 之间的交换，这个从 qout[31:0]的信号可以读出，即{……43,44,45,46,6,0,1,2,11,4,5,47,7,8,9,10,3,12……}，由 dina[31:0]和 qout[31:0]

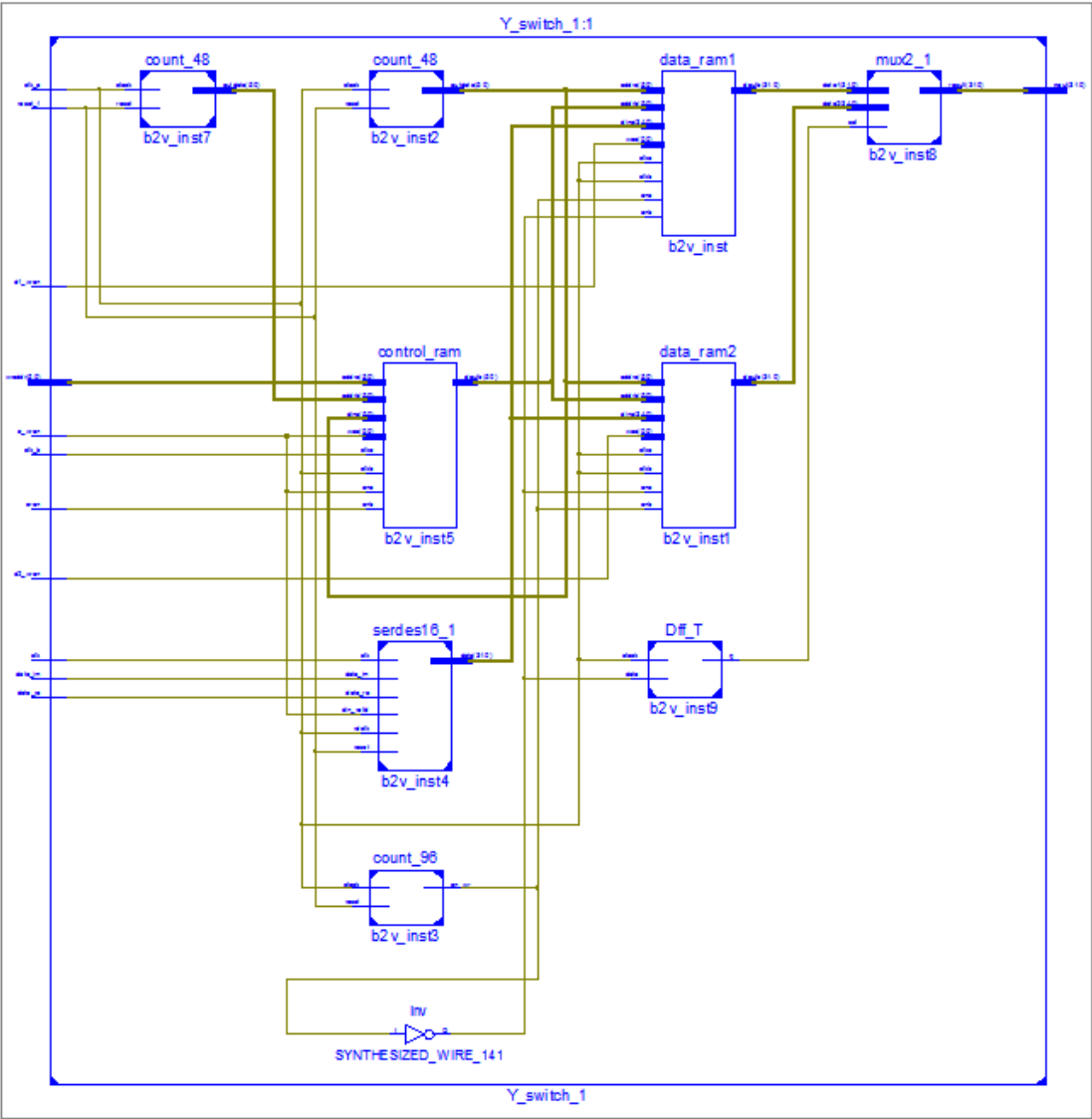


图 5.4 端口内子信道间交换的内部实现图

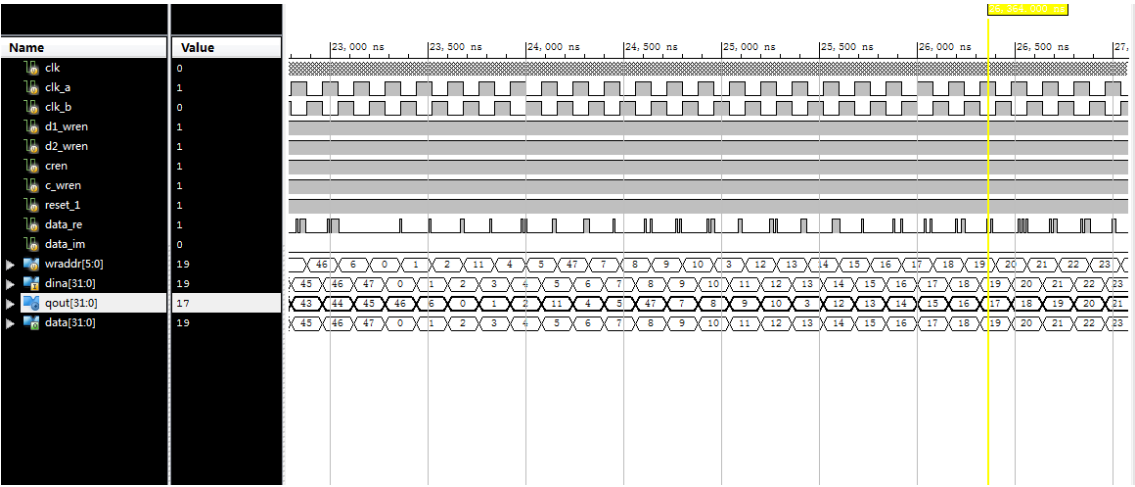


图 5.5 端口内子信道之间样点交换的仿真结果图

可以发现，端口内子信道之间的交换有两个时钟的时延。

5.2.4 端口间交换模块设计

图 5.6 是端口间交换模块的顶层模块实现图，如图所示，data_m1~data_m4 代表四个端口的信号虚部，data_re1~data_re2 代表四个端口的信号实部，data(3:0) 是选路信号，wraddr(5:0) 表示控制存储器 RAM 的地址信号，clk 控制信号串行写入，clk_a 和 clk_b 分别控制控制存储器的读和写操作。

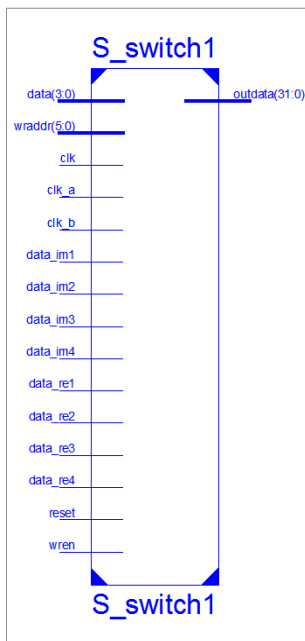


图 5.6 端口间交换模块的顶层模块实现图

图 5.7 是端口间交换模块的内部实现，图中 4 个串并转换模块代表 4 个端口产生 4 路位宽为 32bit 的复信号，由 data[3:0] 信号控制 outdata[31:0] 输出哪一路信号。

图 5.8 是 s_switch 模块的内部实现，它的核心是一个双端口的 RAM 存储器，48 位计数器控制 RAM 存储器的写入地址，四个端口的输入信号作为四选一路选择器的输入，控制存储器在下一个时钟到来时读出 dina[3:0] 写入存储单元的内容，作为四选一路选择器的选路信号，进而控制 outdata[31:0] 的输出。

图 5.9 是端口间交换模块的仿真结果，分析图 5.9 可知，wraddr[5:0] 信号给出了控制存储器的地址信息，控制存储器一共有 48 个内存单元，地址为 0~47，端口 1、端口 2、端口 3 和端口 4 写入的内容分别为 0~47、100~147、200~247 和 300~347，在写测试文件时，data[3:0] 的选路信号的值按照周期性依次变化为 1,2,3,4，这表示控制存储器周期性地输出端口 1、端口 2、端口 3 和端口 4 的信号，在仿真图中表示为 {……35,136,237,338,39,140,241,342,……}。图中 dina[3:0] 表示控制存储器的数据输入，即写操作，doutb[3:0] 表示控制存储器的数据输出，即读

操作,对比二者的信号可以发现控制存储器的读操作和写操作相差半个时钟周期。观察 doutb[3:0]信号和 outdata[31:0]信号,可以发现选路模块有一个周期的时延,综上所述,dina[3:0]和 outdata[31:0]之间有一个半周期的时延,即端口间交换模块的时延为一个半周期,这为后续分析整个系统的时延埋下伏笔。

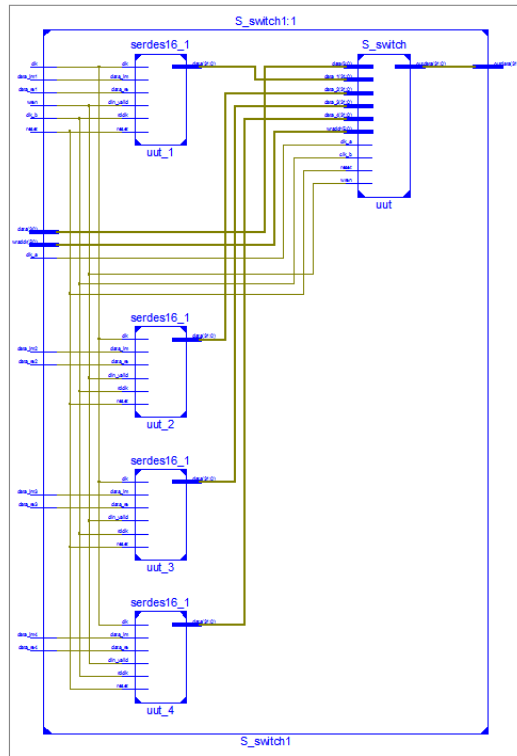


图 5.7 端口间交换模块的内部实现图

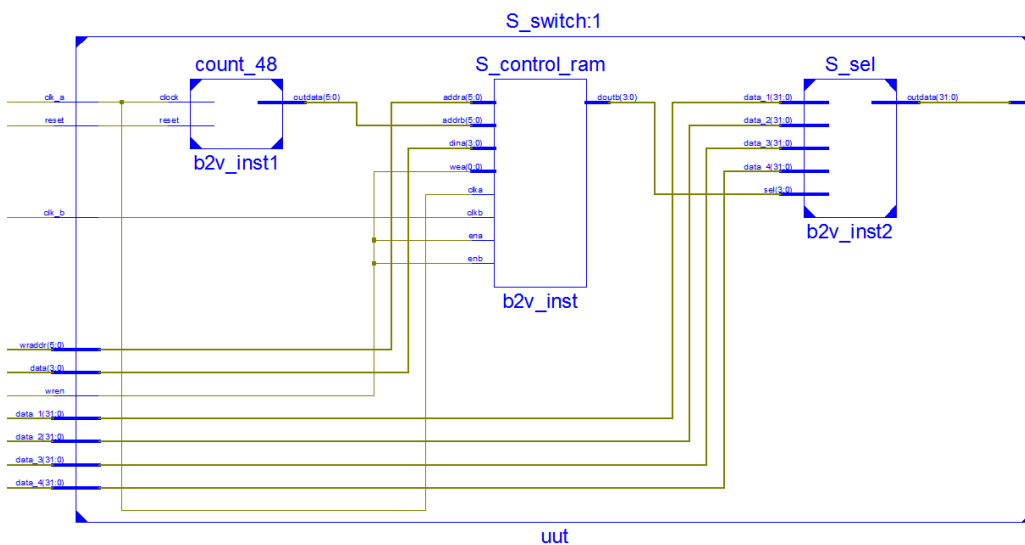


图 5.8 s_switch 模块的内部实现图

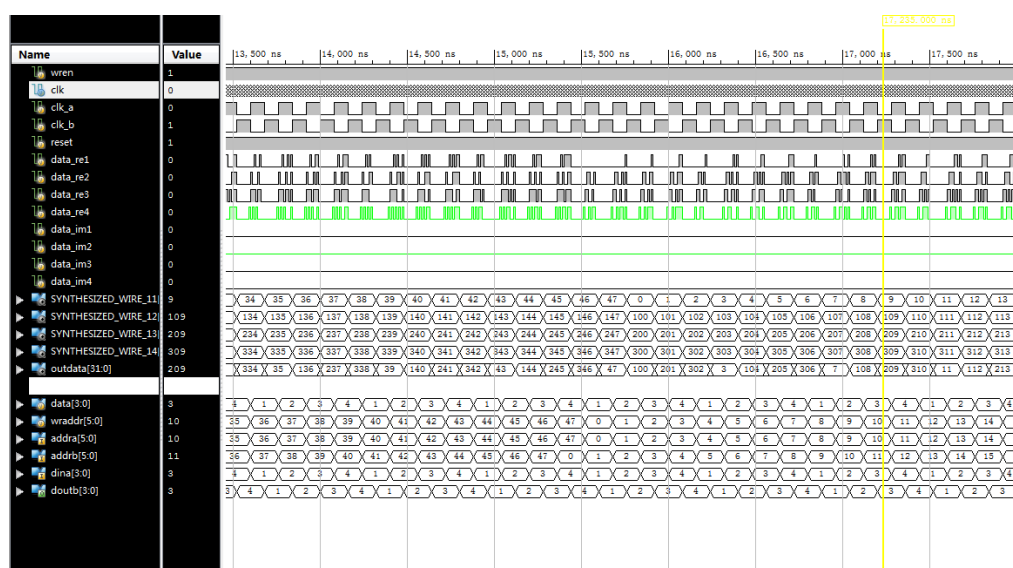


图 5.9 端口间交换模块的仿真结果图

5.3 整个系统的 FPGA 设计及仿真

5.3.1 程控交换模块的 FPGA 设计

当用户信号量化比特数为 32，端口数量为 16 时，XC6VSX315T 芯片的模块资源占用表如下：从图 5.10 中可以看出，芯片的可用 IOBS 为 600bit，实际使用为 904，使用率为 150%，明显资源不够用。

Device Utilization Summary (estimated values)				
Logic Utilization	Used	Available	Utilization	
Number of Slice Registers	1790	393600		0%
Number of Slice LUTs	3634	196800		1%
Number of fully used LUT-FF pairs	762	4662		16%
Number of bonded IOBs	904	600		150%
Number of Block RAM/FIFO	56	704		7%
Number of BUFGBUFCTRLs	2	32		6%

图 5.10 16*16 的模块资源占用图

设 m 表示用户信号的量化比特数， n 表示端口数量，则 m 和 n 取不同值时的芯片 IOBS 值统计如下表 5.1 所示：

从表 5.1 中可以得出如下结论，当 $m=32$ 时， n 可取的最大值为 10，当 $m=16$ 时， n 可取的最大值为 15。下面，取 $m=32$ ， $n=8$ 进行了综合和仿真，芯片模块资源占用表如图 5.11 所示：

5.3.2 基于单播功能的仿真

接下来以 4.2.1 小节中提到的端口间交换实现的例子为例，进行了仿真，即实现端口 1 的第 5 子信道和端口 8 的第 20 子信道之间的交换，图 5.12~5.18 是仿真结果。

表 5.1 m 和 n 取不同值时的芯片 IOBS 值

m（比特数）	n（端口数）	IOBS
32	16	904
32	8	456
32	10	580
32	11	634
16	16	692
16	15	594

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slice Registers	894	393600	0%
Number of Slice LUTs	1570	196800	0%
Number of fully used LUT-FF pairs	385	2079	18%
Number of bonded IOBs	456	600	76%
Number of Block RAM/FIFO	28	704	3%
Number of BUFG/BUFGCTRLs	2	32	6%

图 5.11 8*8 的模块资源占用图

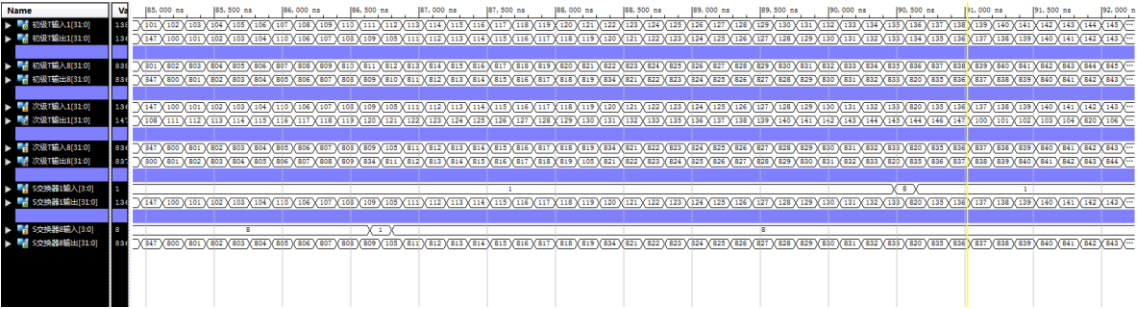


图 5.12 总体仿真结果图

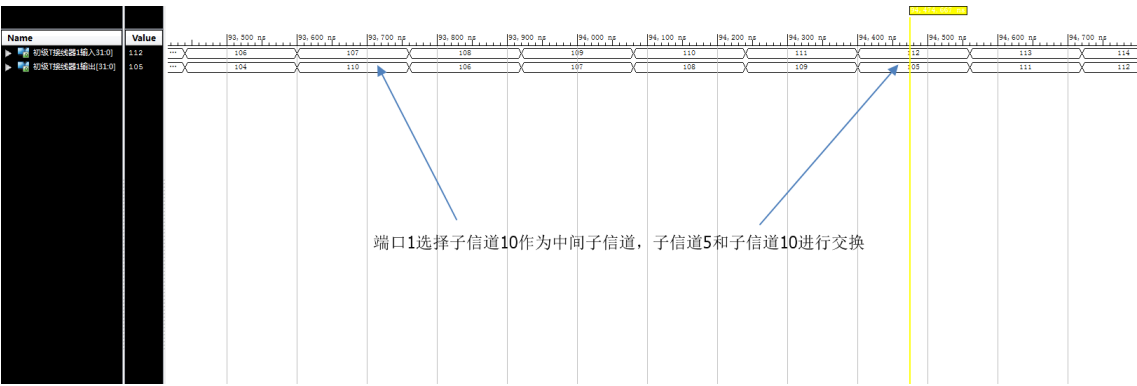


图 5.13 初级 T1 接线器的输入输出结果图

第五章 程控交换网络的 FPGA 设计

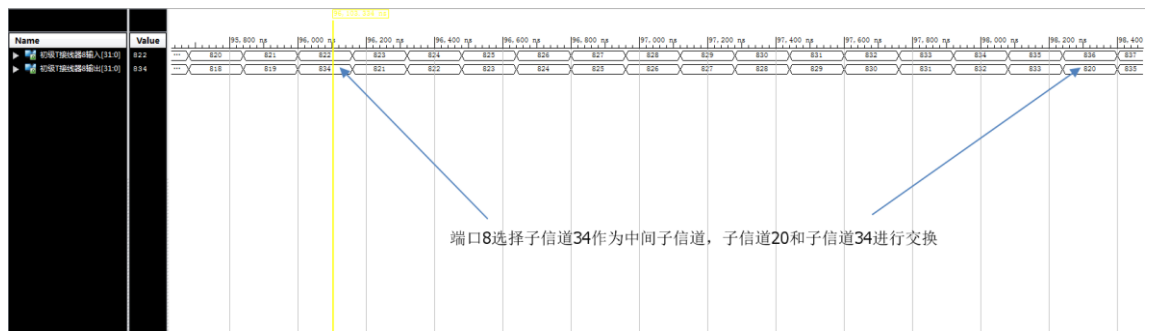


图 5.14 初级 T8 接线器的输入输出结果图

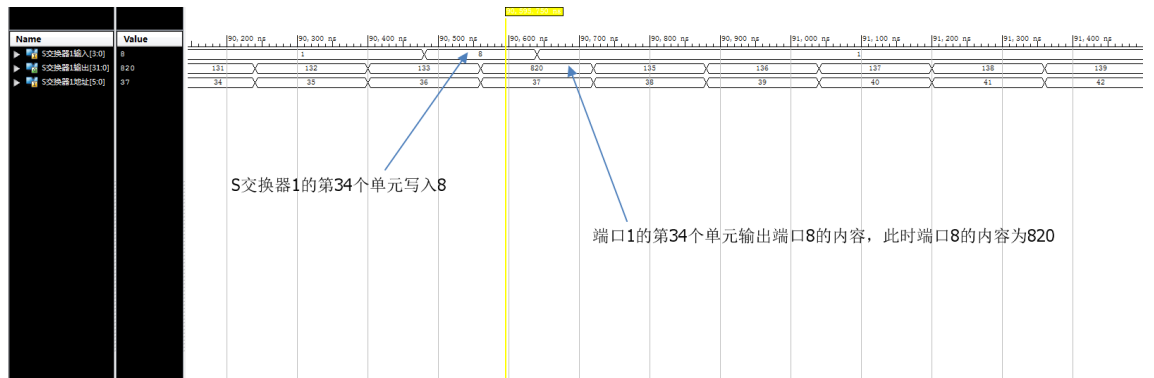


图 5.15 S 交换机 1 的输入输出结果图

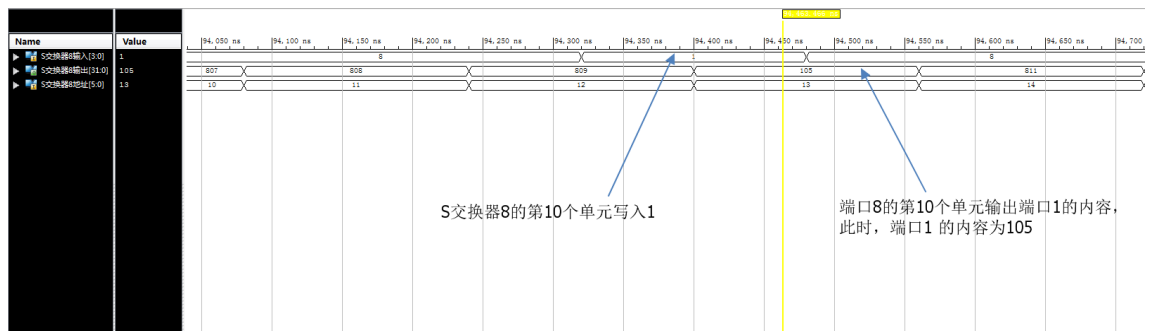


图 5.16 S 交换机 8 的输入输出结果图

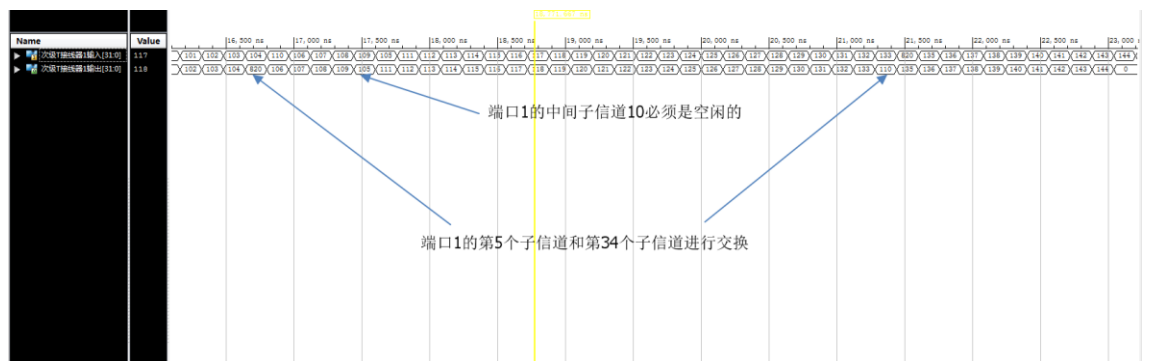


图 5.17 次级接线器 T1 的输入输出结果图



图 5.18 次级接线器 T8 的输入输出结果图

5.3.3 基于组播和广播功能的仿真

为了实现组播和广播的功能，需要把 S 接线器的工作方式设置为输出控制方式。例如，要实现端口 4 第 34 子信道的内容组播到端口 1、2、4、5 的第 34 个子信道输出，即把控制端口 1、2、4、5 的 S 接线器的第 34 个子信道写入 4 即可；要实现端口 1 的第 16 个子信道的内容广播到所有端口的第 16 个子信道，即把 8 个 S 接线器的第 16 个子信道都写入 1 即可。图 5.19~5.20 是组播和广播的仿真结果。

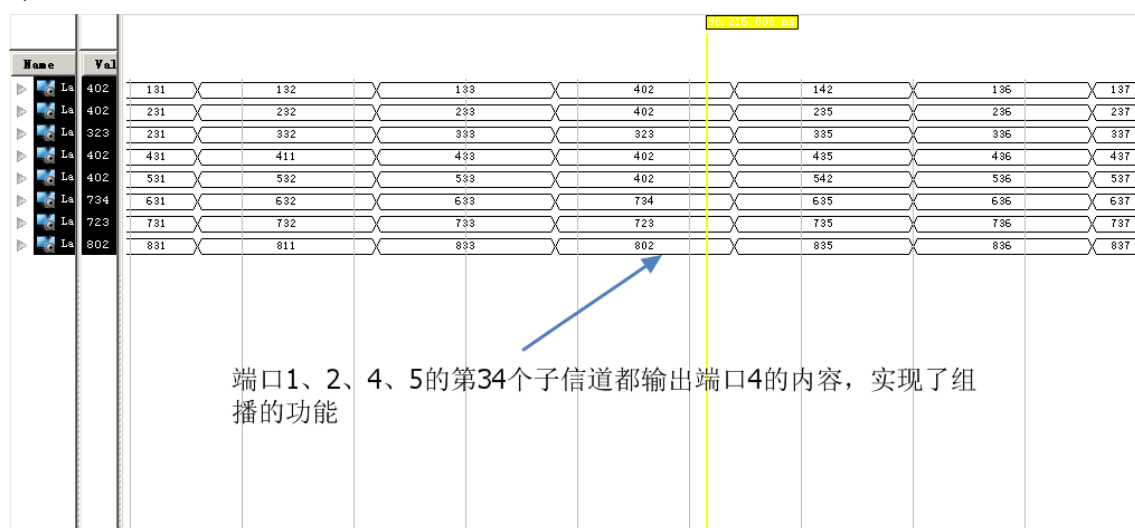


图 5.19 基于组播功能的仿真结果图

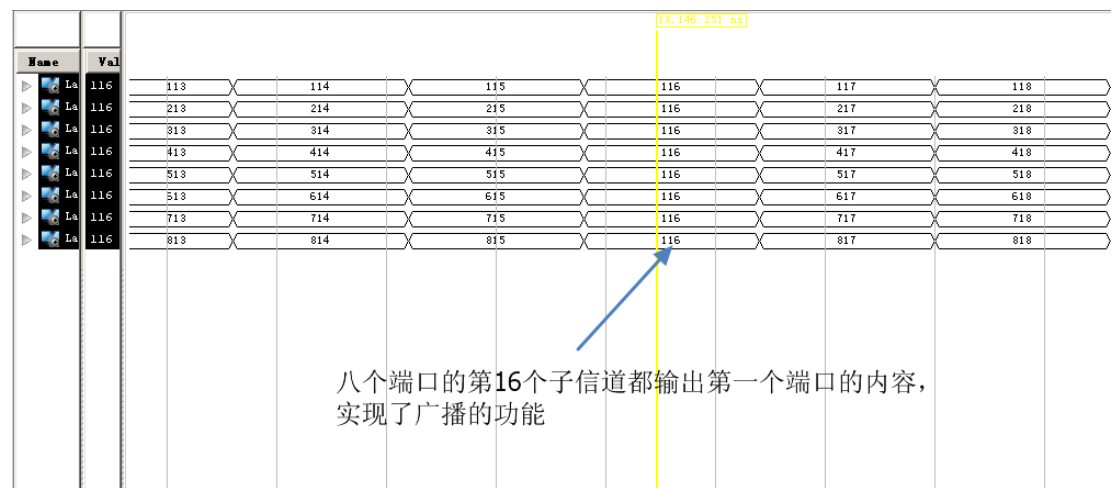


图 5.20 基于广播功能的仿真结果图

5.4 本章小结

本章在 4.2 小节中基于 FDMA 多址接入方式的程控交换方案的设计的基础上，给出了各个模块的 FPGA 设计和仿真结果，然后分单播、组播和广播功能，给出了系统的仿真结果及分析。

第六章 结论与展望

6.1 研究结论

星上处理技术在卫星通信系统中一直是研究的热点。宽带柔性转发器是一种具有星上处理能力的半透明转发器，也叫半再生式转发器。宽带柔性转发器采用基于非均匀滤波器组的数字信道化处理技术，兼具传统透明转发器和再生式转发器的优点，解决了传统转发器中存在的问题，既具有灵活可靠的特点，又可以支持较小粒度的交换，还规避了物理层信号体制的约束，增加了系统容量，满足可变带宽业务的需求，使卫星信号在任意频段和带宽之间实现了灵活交换，是未来星载转发器的主要发展方向之一。

本文的系统模型是一路复信号先经过宽带柔性转发器，得到非均匀带宽的多路子频带样点信号，然后经过高速交换，实现不同带宽之间的样点信号交换，最后进行非均匀重构，恢复原始复信号。接下来对本文的内容进行如下总结：

首先介绍了柔性转发器的研究背景与意义，概括了宽带柔性转发器实现过程中的关键技术研究进展，提出了本文的研究意义和主要内容。

其次介绍了柔性转发器的工作原理，给出了系统的相关参数，概括了 T-S-T 交换网络的基本原理，T 交换抽象为矩阵行内变换，S 交换抽象为矩阵列内变换，将 T-S-T 交换网络抽象为矩阵模型，调度问题抽象为矩阵置换问题。

然后重点研究了基于矩阵模型的程控电路交换网络的调度算法，在高冲突值行优先排列算法和列优先排列算法的基础上提出了一种基于列优先排列的改进算法，通过柱状图比较了三种算法的比较次数和交换次数，仿真结果表明，改进算法在时间复杂度上优于高冲突值行优先排列算法和列优先排列算法，但是空间复杂度高于其他两种算法。

接着设计了 T-S-T 交换网络的交换帧结构，在此基础上提出了基于 FDMA 多址接入方式的程控交换方案，实现了单播、组播和广播功能。

最后在基于 FDMA 多址接入方式的程控交换方案设计的基础上，给出了各个模块的 FPGA 设计和仿真，实现了单播、组播和广播功能，通过合理调整信号量化比特数和子信道数，实现了芯片容量的利用率最大化。

6.2 研究展望

当然，本文的研究工作还存在需要改进和完善的地方，主要包括：

- 1、本文只是提出了调度算法，没有与硬件实现结合起来。

2、本文的 FPGA 模块只进行了功能仿真，接下来还需要进行时序仿真和性能调试工作。

参考文献

- [1] 冯少栋, 李广侠, 张更新. 宽带卫星通信系统发展展望[J]. 卫星与网络, 2010 (4): 66-70.
- [2] 王景泉. 卫星转发器的新技术发展[J]. 中国航天, 1996, 12.
- [3] 朱子行, 赵尚弘, 李勇军, 等. 再生式通信卫星转发器的研究进展[J]. 电讯技术, 2011, 51(8): 147-153.
- [4] Whitefield D, Gopal R, Arnold S. Spaceway now and in the future: On-board ip packet switching satellite communication network[C]//Military Communications Conference, 2006. MILCOM 2006. IEEE. IEEE, 2006: 1-7.
- [5] Gopal R, Whitefield D, Arnold S. Technology readiness of future generation networks leveraging regenerative satellite mesh architecture-a spaceway perspective[C]//Military Communications Conference, 2006. MILCOM 2006. IEEE. IEEE, 2006: 1-7.
- [6] Switching I W O B. Regenerative Communication Satellites Developments in Europe, Past Present and Future[J]. 2007.
- [7] 卢珊珊, 曹文忠, 陈明辉. 日本宽带多媒体卫星通信系统“WINDS”技术详解[J]. 卫星电视与宽带多媒体, 2008 (20): 38-41.
- [8] 边东明, 何家富, 孙玉华. 宽带多媒体卫星通信系列讲座之四 WINDS 系统通信体制[J]. 数字通信世界, 2009 (5): 78-81.
- [9] 吕智勇, 汪宏武, 张健. 卫星转发器资源的选择[J]. 卫星与网络, 2007 (4): 36-41.
- [10] 张飞, 张更新, 王可青, 等. 卫星通信中柔性转发技术研究[J]. 空间电子技术, 2012, 3: 003.
- [11] 易克初. 带限信号时分复用传输方法. 中国发明专利[981128467]. 199.6, 2001 年授权
- [12] 易克初, 汪春霆. 非再生式星上信号处理技术及其应用[C]. 军事电子信息学术会议论文集, 2006: 695-701.
- [13] 杨焯, 田红心, 田斌, 等. 非再生式星上处理和程控交换及其支持 IP 业务的可能性[J]. 信号处理, 2013, 29(11): 1555-1559.
- [14] 程诚. 基于准正交时分复用的数字信道化技术研究[D]. 西安电子科技大学硕士学位论文, 2014.
- [15] 张飞, 边东明, 张更新. 星载柔性转发器中一种近似精确重构原型滤波器的设计[J]. 电子与信息学报, 2013, 35(3): 671-676.
- [16] 刘增基, 鲍民权, 邱智亮. 交换原理与技术[M]. 北京: 人民邮电出版社, 2007.
- [17] 陈腊梅. TST 交换网络矩阵数学模型和调度算法的研究[D]. 西安电子科技大学, 2005.

- [18] Inukai T. An efficient SS/TDMA time slot assignment algorithm[J]. Communications, IEEE Transactions on, 1979, 27(10): 1449-1455.
- [19] 周坡. 基于 OFDM 的 GEO 卫星移动通信系统关键技术研究[D]. 清华大学, 2011.
- [20] Tanenbaum A S, 计算机网络. 潘爱民译[M]. 计算机网络 (第四版), 2004.
- [21] 田耘, 徐文波. Xilinx FPGA 开发实用教程[M]. 清华大学出版社, 2008.
- [22] 杨焯. 基于 CWTDM 的宽带卫星星上交换技术研究[D]. 西安电子科技大学硕士学位论文, 2014.
- [23] Angelettil P, Gaudenzi R D. From "bent pipe" to "soft-ware defined payloads": evolution and trends of satellite communications systems[C]. Proc. of the 26th AIAA International Communications Satellite Systems Conference, San Diego, CA, 2008, 292~301.
- [24] Butash T C, Marshall J R. Leveraging digital on-board processing to increase communications satellite flexibility and effective capacity[C]. Proceedings of the 28th AIAA International Communications Satellite Systems Conference. 2010: 113-122.
- [25] Evans B, Thompson P. Key issues and technologies for a Terabit/s satellite[C]. 28th AIAA International Communications Satellite Systems Conference (ICSSC-2010). 2010.
- [26] 周宇昌, 李小军, 周诠. 空间高速数据传输技术新进展[J]. 空间电子技术, 2009, 6(3): 43-48.
- [27] 宋莉, 刘爱军, 马刘非. 基于 DiffServ 的星上交换设计与仿真[J]. 系统仿真学报, 2008, 20(6).
- [28] 肖丽萍, 顾学迈, 郭庆, 等. 基于蚂蚁算法的星上 ATM 交换结构[J]. 宇航学报, 2007, 28(4): 932-935.
- [29] 吕高峰. 星上交换和半实物仿真技术研究[D]. 国防科学技术大学, 2004.
- [30] 甘仲民, 张更新. 卫星通信技术的新发展[J]. 通信学报, 2006, 27(8): 2-9.
- [31] 糜正琨, 杨国民. 交换技术[M]. 北京: 清华大学出版社, 2006
- [32] Febvre P, Bouthors X, Maalouf S, et al. Efficient IP - multicast via Inmarsat BGAN, a 3GPP satellite network[J]. International Journal of Satellite Communications and Networking, 2007, 25(5): 459-480.
- [33] Gupta R. Network Centric Waveform operation over the WGS[C]. MILITARY COMMUNICATIONS CONFERENCE, 2011-MILCOM 2011. IEEE, 2011: 1690-1695.
- [34] 董启甲, 张军, 张涛, 等. 高效 MF-TDMA 系统时隙分配策略[J]. 航空学报, 2009, 30(9): 1718-1726.
- [35] 刘彩苗. 高空平台通信系统半再生信号电路交换技术研究 [D]. 西安: 西安电子科技大学, 2010.
- [36] Wajih A. Abu-Al-Saud, Student Member, IEEE, and Gordon L. Stüber, Fellow, IEEE. Efficient

- Wideband Channelizer for Software Radio Systems Using Modulated PR Filterbanks[J].IEEE Transactions on signal processing. 2004,8,10(52).
- [37] GARDNER F M Interpolation in digital modems, part I: fundamentals[J].IEEE Transactions on Com-munications, 1993, 41(3):501-507.

致谢

时光荏苒，稍纵即逝，转眼间，研究生学习生涯已经接近尾声，回顾两年的学习生活，五味俱全，有看到仿真结果的兴奋，也有找不到方案的苦恼，有老师的夸奖，也有老师的批评，在这样跌跌撞撞的两年中，我成长了，当然，这期间不乏周边老师、同学对我的教导和帮助。

第一个应该感谢的当然是我的导师田红心副教授，田老师知识渊博，诲人不倦，在学习上，对我们严格要求，一周一次例会，我们把每周的学习进展情况，做成 word 文档，汇报给老师，老师先给我们指点迷津，然后再针对我们中的不足提出自己的意见；生活中，非常和蔼可亲，时常嘱咐我们，学习要扎实，研究生重在积累，要好好规划好时间和安排。

接下来要感谢的是课题组负责项目的田斌教授，每周的例会，田老师准时出席，虽然我们不是田老师的学生，但是田老师一视同仁，每次开会，总是耐心指导我们。

这里要特别感谢易克初教授，在实验室学习的一年半里，易老师扎实的学识功底，良好的作息习惯，学而不殆的科研精神，使易老师一直走在学术领域的最前沿，正是有了这种耳濡目染，我的科研能力也在逐步提高。

还要感谢课题组的其他老师：刘祖军老师、孙永军老师，王杰令老师和孙德春老师，是他们营造了实验室浓浓的科研氛围，我们能够在这么一个这么好的学习环境中进步，养成好的工作和学习习惯。

感谢与我同届的硕士生赵惠惠、刘美娟、杨伦、苏志明、李晓菲、倪少平等同学，感谢他们陪我度过了充实的研究生生活，也感谢他们在科研过程中对我的帮助！感谢博士生李运华、李怡同学，感谢他们对我理论学习的指导！感谢刘慧杰、徐菲和许浩然等师弟师妹们和实验室的每一位同学，和大家一起学习的日子是我一生的财富！

最后感谢我的家人，我的爸爸、妈妈和两个弟弟，感谢他们对我的支持和帮助，让我从人生低谷走出来，重新面对生活！

作者简介

1. 基本情况

女，山西吕梁人，1989 年 7 月出生，西安电子科技大学通信工程学院电子与通信工程专业 2012 级硕士研究生。

2. 教育背景

2008.09～2012.07 就读于中北大学信息与通信工程学院电子信息科学与技术专业，获理学学士学位

2012.08～ 西安电子科技大学通信工程学院电子与通信工程专业硕士研究生

3. 攻读硕士学位期间的研究成果

2013 年 9 月到 2014 年 10 月参与了基于柔性转发器的高速交换技术研究，在 T-S-T 交换网络矩阵模型的基础上改进了调度算法，设计了交换帧结构，提出了基于 FDMA 多址接入方式的交换方案，并进行了 FPGA 设计。

