INFORMATION TO USERS

This material was produced from a microfilm copy of the original document. While the most advanced technological means to photograph and reproduce this document have been used, the quality is heavily dependent upon the quality of the original submitted.

The following explanation of techniques is provided to help you understand markings or patterns which may appear on this reproduction.

1. The sign or “target” for pages apparently lacking from the document photographed is “Missing Page(s)”. If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting thru an image and duplicating adjacent pages to insure you complete continuity.

2. When an image on the film is obliterated with a large round black mark, it is an indication that the photographer suspected that the copy may have moved during exposure and thus cause a blurred image. You will find a good image of the page in the adjacent frame.

3. When a map, drawing or chart, etc., was part of the material being photographed the photographer followed a definite method in “sectioning” the material. It is customary to begin photoing at the upper left hand corner of a large sheet and to continue photoing from left to right in equal sections with a small overlap. If necessary, sectioning is continued again — beginning below the first row and continuing on until complete.

4. The majority of users indicate that the textual content is of greatest value, however, a somewhat higher quality reproduction could be made from “photographs” if essential to the understanding of the dissertation. Silver prints or “photographs” may be ordered at additional charge by writing the Order Department, giving the catalog number, title, author and specific pages you wish reproduced.

5. PLEASE NOTE: Some pages may have indistinct print. Filmed as received.

University Microfilms International
300 North Zeeb Road
Ann Arbor, Michigan 48106 USA
St. John’s Road, Tyler’s Green
High Wycombe, Bucks, England HP10 8HR
WANG, JIN-TUU
DESIGN OF A MIXED VOICE/DATA TRANSMISSION
SYSTEM FOR COMPUTER COMMUNICATION.
THE OHIO STATE UNIVERSITY, PH.D., 1978

Copyright by
Jin-tuu Wang
1978
DESIGN OF
A MIXED VOICE/DATA TRANSMISSION SYSTEM
FOR
COMPUTER COMMUNICATION

DISSERTATION

Presented in Partial Fulfillment of the Requirement for
the Degree Doctor of Philosophy in the Graduate
School of the Ohio State University

By
Jin-tuu Wang, B.S.E.E., M.S.

* * * * *

The Ohio State University
1978

Reading Committee:
Dr. Ming T. Liu, Chairman
Dr. David K. Hsiao
Dr. Kenneth H. Breeding

Approved By

Adviser
Department of Computer
and Information Science
I would like to express my thanks to my adviser Dr. Ming T. Liu for his continuing effort of guidance and instruction, especially in the last three years of air-mail correspondence, and also to my committee members Dr. David Hsiao and Dr. Kenneth Breeding for their valuable comments and suggestions, especially during my first informal presentation of work in December 1975. I would also like to thank my classmate, Dr. Cecil Heames, for his patience in reading my first draft and giving me useful suggestions and comments.

I would also like to state my appreciation to the National Science Council, Republic of China, for its sponsorship of my two-year fellowship for advanced study at the Ohio State University, and also to the Directorate General of Telecommunication, Ministry of Communication, Republic of China, for its research assistantship during my period of work in Taiwan. Finally, but not the least, I would also like to thank my wife, Spring River, for her great effort in taking care of our children during my two-year study in the United States.
VITA

December 4, 1937...... Born - Kaohsiung, Taiwan, ROC

1960....................... B.S.E.E., Electrical Engineering
Dept., National Taiwan University,
Taipei, Taiwan, Republic of China

1960-1966................. Technician, Microwave Office, Chinese
Government Radio Administration,
Taipei, Taiwan, ROC

1968....................... M.S.E.E., Electrical Engineering
Dept., University of Hawaii,
Honolulu, Hawaii

1968-1972................. Engineer, Satellite Communication
Earth Station, Chinese Government
Radio Admin., Taipei, Taiwan, ROC

1974....................... M.S. in Computer and Information
Science, Ohio State University,
Columbus, Ohio

1974-1977................. Senior Engineer and Project Manager,
Public Telegraph Message Switching
System, Chinese Government Radio
Administration, Taipei, Taiwan, ROC

1978-................. Senior Engineer, Directorate General
of Telecommunications, MOC, ROC;
Project Manager, Public Telegraph
Message Switching System,
Chinese Government Radio
Administration, Taipei, ROC

PUBLICATIONS

"A Novel Model for a Mixed Voice/Data Transmission System
for Computer Communication," Proc. of the 1975
International Computer Symposium, Vol. I, pp. 458-468,
Taipei, Taiwan, ROC, August 1975.
(Ming T. Liu, coauthor)

"Analysis and Simulation of the Mixed Voice/Data
Rec. of the 1976 National Telecommunication Conference,
Vol. III, pp. 42.3-1 to 42.3-5, Dallas, Texas, Nov.
(Ming T. Liu, coauthor)
(Ming T. Liu, coauthor)

(Ming T. Liu, coauthor)


FIELDS OF STUDY

Major Field: Computer and Information Science

Studies in Computer Organization and Architecture
Professor Ming T. Liu

Studies in Operating Systems and Systems Programming
Professor David K. Hsião

Studies in Programming Languages
Professor Clinton R. Foulk
# TABLE OF CONTENTS

ACKNOWLEDGEMENTS ................................................................. ii
VITA ................................................................................................ iii
TABLE OF CONTENTS ................................................................. v
LIST OF TABLES ............................................................................. x
LIST OF FIGURES ........................................................................... xi

Chapter

1. INTRODUCTION TO THE MIXED VOICE/DATA (MVD) SYSTEM ............................................................... 1

1.1 Merits of Computer Networking ............................................. 1
1.2 Expensive Communication Link Cost of Computer Networking ........................................................................ 2
1.3 Approach to Lowering Link Cost ........................................... 2
1.4 Main Objectives of This Research ......................................... 4
1.5 Description of the Method ..................................................... 5
1.6 Main features of Mixed Voice/Data System ......................... 6
1.7 Expected Contribution of the Mixed Voice Data System ........ 9
1.8 Organization of This Dissertation ......................................... 10

2. MOTIVATION, CONCEPT, APPROACH AND THE PROPOSED MODEL ......................................................................................... 14

2.1 Motivation and Concept ......................................................... 14
2.1.1 Inefficient Utilization of Channel Bandwidth ..................... 14
2.1.2 Present Frequency Division Multiplex (FDM) Scheme for Long-Distance Telephone Circuits ......................................................... 17
2.1.3 High Cost of Leasing Exclusively Allocated Data Communication Channel ................................................................. 23
2.1.4 Resource Saving Principle ................................................. 25
2.2 Approach and Proposed Model ............................................. 26
2.2.1 Using Microcomputer for the Controller ............................ 29
2.2.2 Keeping Analog Voice-Grade Channel Unchanged ............ 31
2.2.3 Treatment of Data Packet Interference ............................. 33
2.2.3.1 Difference in Retransmission between the ALOHA and the MVD System .................. 33
2.2.3.2 Partial Packet Transmission (Preemptive-Repeat vs Preemptive-Resume) ............ 34
2.2.4 Using Voice Detector .................. 35
2.2.5 Recognition of Voice and Data ........ 36
2.2.6.1 Inband Signaling Tones (Data Escort Carriers) ........ 38
2.2.6.2 Common Channel Signaling .... 38
2.2.6.3 Signal-Pattern Recognition ........ 39
2.2.6.4 Bracket Signal .................. 41
2.2.6 Using Multiple Voice-Grade Channels ........ 43
2.2.7 Allowing Multiple Input/Output Data Line at Transmit/Receive Side ........ 44
2.2.8 Fixed-Size Packet vs. Variable-Size Packet ........................................ 45
2.2.9 Block Acknowledgement and Data Scrambling ........................................ 46
2.2.10 Data Packet Queueing and Dispatching ........................................ 47
2.2.11 Recent Microcomputer Hardware Technology ........................................ 48
2.2.12 Synchronous Data Transmission and Modulation ........................................ 49
2.3 Summary ........................................ 50

3. SOME RELATED SYSTEM AND THEIR DIFFERENCES TO THE PROPOSED MVD MODEL ........ 52

3.1 TASI System ........................................ 52
3.2 IDAST or AAVD System .................. 54
3.3 PCM-TASI or DSI System .................. 56
3.4 DUV, DIV, DAV and DAVID Systems ........................................ 58
3.5 TDMA System ........................................ 62
3.6 ITT System ........................................ 63
3.7 Teletex System .................. 64
3.8 Pitch Synchronous Data Interpolation Scheme ........................................ 65
3.9 Summary ........................................ 66

4. ANALYSIS OF THE MIXED VOICE/DATA (MVD) SYSTEM ........ 67

4.1 Introduction ........................................ 67
4.2 Observation of Talking and Silent Period Distributions ........................................ 67
4.3 Probabilistic Approach to the Interfering Mechanism ........................................ 75
4.3.1 Average Service Time of Data Packet ........................................ 77
4.3.2 Expected Throughput of Data Packets ........................................ 80
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.4</td>
<td>Queueing Model of a Single Server with a Preemptive Priority Class of Arrival</td>
<td>85</td>
</tr>
<tr>
<td>4.5</td>
<td>Preemptive-Resume Model with Constant Service Time for Data Packets</td>
<td>100</td>
</tr>
<tr>
<td>4.6</td>
<td>Optimum Packet Size, Minimum Overhead, and Maximum Throughput</td>
<td>110</td>
</tr>
<tr>
<td>4.7</td>
<td>Analysis of Multi-Channel MVD System</td>
<td>115</td>
</tr>
<tr>
<td>4.8</td>
<td>Summary</td>
<td>121</td>
</tr>
<tr>
<td>5.</td>
<td>SIMULATION OF THE MIXED VOICE/DATA TRANSMISSION SYSTEM</td>
<td>122</td>
</tr>
<tr>
<td>5.1</td>
<td>Reasons for Simulation</td>
<td>122</td>
</tr>
<tr>
<td>5.1.1</td>
<td>Use of Actual Operational Conditions</td>
<td>122</td>
</tr>
<tr>
<td>5.1.2</td>
<td>Obtaining Performance Figures</td>
<td>123</td>
</tr>
<tr>
<td>5.1.3</td>
<td>Simulating Mathematically Non-representable Operation</td>
<td>123</td>
</tr>
<tr>
<td>5.1.4</td>
<td>Discovery of System Bottleneck Regarding Throughput</td>
<td>124</td>
</tr>
<tr>
<td>5.2</td>
<td>Approach Taken in Building a Simulation Model</td>
<td>124</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Generation of Data Packets</td>
<td>125</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Selection of a Minimum-Length Queue or Finding a Silent Channel</td>
<td>125</td>
</tr>
<tr>
<td>5.2.3</td>
<td>Capture of a Voice-Grade Channel for Transmission Service</td>
<td>127</td>
</tr>
<tr>
<td>5.2.4</td>
<td>Entering a Waiting Line or Leaving a Waiting Line</td>
<td>127</td>
</tr>
<tr>
<td>5.2.5</td>
<td>Entering for Processing or Transmission Service</td>
<td>126</td>
</tr>
<tr>
<td>5.2.6</td>
<td>Interrupting by Voice Arrival</td>
<td>128</td>
</tr>
<tr>
<td>5.2.7</td>
<td>Representation of a Multiple-Channel MVD System</td>
<td>130</td>
</tr>
<tr>
<td>5.2.8</td>
<td>Simulation of Actual Talking and Silent Distribution</td>
<td>132</td>
</tr>
<tr>
<td>5.3</td>
<td>Simulation Model</td>
<td>133</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Multi-line Queueing Model</td>
<td>135</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Single-line Queueing Model</td>
<td>138</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Repeat Transmission versus Resume Transmission of Interrupted Data Packets</td>
<td>143</td>
</tr>
<tr>
<td>5.4</td>
<td>Table of Definition of GPSS Entities for the MVD System</td>
<td>144</td>
</tr>
<tr>
<td>5.5</td>
<td>Flowchart and Program Listings</td>
<td>147</td>
</tr>
<tr>
<td>5.6</td>
<td>Simulation Results</td>
<td>148</td>
</tr>
<tr>
<td>5.7</td>
<td>Summary</td>
<td>161</td>
</tr>
</tbody>
</table>
6. IMPLEMENTATION OF THE MIXED VOICE/DATA TRANSMISSION
SYSTEM ................................................................. 163

6.1 Introduction ......................................................... 163
6.2 Principle of System Operation and Functional
Requirements ............................................................... 163
  6.2.1 Design of the Protocol ............................................ 164
  6.2.2 Principle of System Operation ................................. 169
    6.2.2.1 Dynamic Buffering ........................................ 169
    6.2.2.2 Header Attachment and
             Data Scrambling ........................................ 171
    6.2.2.3 Request to Send .......................................... 172
    6.2.2.4 Route Selection and
             Outputting .............................................. 173
    6.2.2.5 Voice Arrival and Termination
             Interrupt Handling .................................... 174
    6.2.2.6 Packet Holding for
             Acknowledgement ....................................... 176
    6.2.2.7 Request for Retransmission
             of Erroneous Packets .................................. 177
    6.2.2.8 Multiple Copies Retransmission .......................... 178
    6.2.2.9 Data Escort Carriers ....................................... 178
    6.2.2.10 Voice Signal Delay ...................................... 179
    6.2.2.11 Data Escort Carrier
             Recognizing ............................................. 179
    6.2.2.12 Receiving Data Packet
             Process .................................................... 180
  6.2.3 Processor Status Transition ................................. 181
6.3 Summary of Functional Requirements of
the MVD System ....................................................... 185
6.4 Detailed Design on the Required Hardware
Components .............................................................. 187
  6.4.1 A Microcomputer .............................................. 187
  6.4.2 Voice Detector ................................................ 192
  6.4.3 Transmitting Status Change Detector ...................... 195
  6.4.4 Data Escort Carrier Generator ................................ 197
  6.4.5 Modulator ...................................................... 198
  6.4.6 Mixer ........................................................... 200
  6.4.7 Data Escort Carrier Recognizer ............................. 200
  6.4.8 Receiving Status Change Detector ......................... 203
  6.4.9 Data Escort Carrier Suppressor ............................. 203
  6.4.10 Demodulator .................................................. 205
  6.4.11 Modem Interface .............................................. 207
  6.4.12 Switching Devices ............................................ 214
  6.4.13 Analog Delay Device ........................................ 216
6.5 Software Implementation of the MVD System ................... 218
  6.5.1 Operating System of the MVD System ......................... 218
  6.5.2 Algorithms to Implement Route
          Selection Strategies ....................................... 221
6.5.2.1 Multi-Queue Minimum Queue Content (MINQL) ........................................ 221
6.5.2.2 Multi-Queue Less-Than-Threshold (LTT) ........................................... 224
6.5.2.3 Single-Queue Most Recently Released (MRR) ........................................ 225

6.6 Summary ......................................................... 227

7 CONCLUSION ..................................................... 228
7.1 Summary of Work ............................................. 228
7.2 Technical Maturity and Economical Justification ......................................... 230
7.3 Expected Contributions ....................................... 231
7.4 Follow-up Research .......................................... 232
7.5 Conclusion ..................................................... 236

References ....................................................... 237

Appendix A Flowchart of GPSS Simulation ......................................................... 242
Appendix B Program Listing of GPSS Simulation ................................................. 245
Appendix C Sampled Output of GPSS Simulation ................................................ 255
<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Monthly Rental of Leased Data Communication Circuits from/to Taiwan, Republic of China (1977)</td>
<td>24</td>
</tr>
<tr>
<td>2. Statistics of Sampled Observation on International Telephone Circuits to/from Taipei, Taiwan, ROC</td>
<td>74</td>
</tr>
<tr>
<td>3. Talking- and Silent-Period Statistics for Mixed Destinations</td>
<td>75</td>
</tr>
<tr>
<td>4. Simulation Results for Single-Queue First Silent Strategy (C=5, S=2400 bps, and Z = 1000 bits)</td>
<td>148</td>
</tr>
<tr>
<td>5. Simulation Results for Single-Queue First Silent Strategy (C=5, S=4600 bps, and Z=1000 bits)</td>
<td>149</td>
</tr>
<tr>
<td>6. Throughput of Data Packets in the System (Simulation Time: 4 min.)</td>
<td>160</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figures</th>
<th>Captions</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>One-line Schematic of Long-Distance Telephone Conversation</td>
<td>18</td>
</tr>
<tr>
<td>2</td>
<td>International Telephone Transmission Path through Satellite Circuits and the Position of the MVC Controller</td>
<td>18</td>
</tr>
<tr>
<td>3</td>
<td>Baseband Configuration of Frequency Division Multiplex (FDM) Telephony</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>Block Diagram of Earth Station Communication Equipment</td>
<td>22</td>
</tr>
<tr>
<td>5</td>
<td>Conceptual Model of the Simplex MVD System</td>
<td>30</td>
</tr>
<tr>
<td>6</td>
<td>State Diagram of a Voice-Grade Channel Used for MVD System</td>
<td>37</td>
</tr>
<tr>
<td>7</td>
<td>Common Channel Signalling, (a) FDM Common Channel Signalling, (b) and (c) TDM Common Channel Signalling</td>
<td>40</td>
</tr>
<tr>
<td>8</td>
<td>Hyperspace Shows Decision Cubes for Data Traffic</td>
<td>42</td>
</tr>
<tr>
<td>9</td>
<td>Spectrum of Data In Voice (DIV) in Baseband</td>
<td>59</td>
</tr>
<tr>
<td>10</td>
<td>Spectrum of Data Under Voice in Baseband</td>
<td>59</td>
</tr>
<tr>
<td>11</td>
<td>Spectrum of Data Above Voice (DAV) in Baseband</td>
<td>61</td>
</tr>
<tr>
<td>12</td>
<td>Spectrum of Data Above Video (DAVID) in Baseband</td>
<td>61</td>
</tr>
<tr>
<td>13</td>
<td>Probability Density Function of Talking Period</td>
<td>70</td>
</tr>
<tr>
<td>14</td>
<td>Probability Density Function of Silent Period</td>
<td>71</td>
</tr>
<tr>
<td>15</td>
<td>Probability Distribution Function of Talking Period</td>
<td>72</td>
</tr>
</tbody>
</table>
16 Probability Distribution Function of Silent Period
17 Probabilistic Approach of Interfering Mechanism
18 Probability of Requiring N-time Retransmission
19 (a) The Expected Number of Packets Transmitted per Silent Period and (b) The Percentage Overhead versus the Packet Transmitting Time
20 Queueing Model for a Single Server with One Preemptive Priority Class of Arrival
21 State Transition Diagram for a Single Channel MVD System
22 The Expected Number of Data Packets in Transmitting System ($L_2$) versus the Traffic Intensity ($\rho_1$) with Mean Arrival Rate ($\lambda_2$) of Data Packets as a Parameter (Preemptive-Repeat)
23 The Expected Number of Data Packets in the Transmitting System versus Mean Service Time of Data Packets ($1/\mu_2$) with Mean Arrival Rate ($\lambda_1$) as a Parameter (Preemptive-Resume)
24 The Expected Total Resident Time of Data Packets in the Transmitting System versus Traffic Intensity of Data Packets with Mean Arrival Rate of Data Packets as a Parameter (Preemptive-Repeat Case)
25 The Expected Total Resident Time of Data Packets in the Transmitting System versus Mean Service Time of Data Packets with Mean Arrival Rate as a Parameter (Preemptive-Repeat Case)
26 Service Interruption of Data Packets Due to the Arrival of Voice Signals for Preemptive-Resume Case
27 The Expected Number of Data Packets in the Transmitting System versus Traffic Intensity of Data Packets with Mean
Mean Arrival Rate as a Parameter (Preemptive-Resume)

28 The Expected Number of Data Packets in the Transmitting System versus Mean Service Time of Data Packets with Mean Arrival Rate of Data Packet as a Parameter (Preemptive-Resume)

29 The Expected Total Resident Time of Data Packets in the Transmitting System versus Data Traffic Intensity with Mean Arrival Rate of Data Packets as a Parameter (Preemptive-Resume Case)

30 The Expected Total Resident Time of Data Packets in the Transmitting System versus Mean Arrival Rate of Data Packets as a Parameter (Preemptive-Resume Case)

31 Solution of Optimum Packet Size

32 Optimum Packet Size to Achieve Maximum Information Rate

33 Multi-Channel Queueing Model of the MVD System

34 State Transition Diagram for Multi-Channel Single-Queue Model

35 State Transition Diagram for Three-Channel Single-Queue Model of MVD System

36 Probability Distribution of (a) Talking Period (curve 1) and (b) Silent Period (curve 2)

37 Multi-line Queueing Model for the MVD System

38 Single-line Queueing Model for the MVD System

39 Average Delay versus Average Number of Data Packets in the Transmitting System for Various Strategies

40 W-N Plot for Single-Queue MRR Strategy

41 N- P'd Plot for Various Strategies

42 W- P'd Plot for Various Strategies
43 N- $\rho$ d Plot for Two Strategies 154
44 W- $\rho$ d Plot for Two Strategies 154
45 N- $\rho$ d Plot for Comparison between Preemptive-Repeat and Preemptive-Resume Schemes for Single-Queue First Silent Strategy 155
46 W- $\rho$ d Plot for Comparison between Preemptive-Repeat and Preemptive-Resume Scheme for Single-Queue First Silent (FSLN) Strategy 155
47 N- $\rho$ d Plot for Single-Queue with Different Transmission Speeds 156
48 W- $\rho$ d Plot for Single-Queue with Different Transmission Speeds 156
49 N- $\rho$ d Plot for Single-Queue MRR Strategy with Different Speed 157
50 W- $\rho$ d Plot for Single Queue MRR Strategy with Different Speed 157
51 Average Delay of Data Packets Including Transmission Delay and the Sequence Check in Receiving Side 159
52 Communication Line Protocol of MVD System 166
53 Block Diagram of a Multi-Channel Mixed Voice/Data Transmission System 170
54 Processor Status Transition Diagram 183
55 Block Diagram of the MVD Controller 188
56 Configuration Diagram of MVD Controller and Its Interface 189
57 Block Diagram of Voice Detector 193
58 Improved Voice Detector 193
59 Block Diagram of Transmitting Status Change Detector 196
60 Circuit Diagram of Data Escort Carrier 196
xiv
Generator

61 Circuit Diagram of the FSK Modulator

62 Circuit Diagram of the Mixer

63 Circuit Diagram of the Data Escort
   Carrier Recognizer

64 Block Diagram of the Tone Decoder

65 Block Diagram of the Receiving Status
   Change Detector

66 Circuit Diagram of the Data Escort
   Carrier Suppressor

67 Circuit Diagram for the FSK Demodulator

68 Block Diagram of the Phase-Lock Loop

69 Block Diagram of the Modem Interface

70 Block Diagram of the INTEL 8251 USART

71 Intel 8251 Data Block

72 CD4066 Analog Switch

73 Circuit Diagram of the Voltage Converter

74 Circuit Diagram of the Analog Shift Register

75 The Operating System of the Mixed Voice/
   Data (MVD) Transmission System
CHAPTER I

INTRODUCTION TO THE MIXED VOICE/DATA (MVD) SYSTEM

1.1 MERITS OF COMPUTER NETWORKING

The technical accomplishments of linking geographically distributed computing centers by high-speed digital communication line to achieve program, database, resource and load sharing are well recognized both in the literature and in the field [33,41]. In particular, the pioneering ARPA Network [20,39] is a national network in the United States that uses terrestrial microwave data channels as its communication links, while the ALOHA System [3] is a regional network in the islands of Hawaii that uses random access VHF radio channels as its communication links [2]. In addition, both systems are interconnected by a high-speed satellite data channel [4]. Another interesting computer networking system under development is the SBS Network (Satellite Business System) [11], a joint venture of IBM and Comsat General, that uses roof antenna satellite communication links to connect all the major IBM users in the Continent of North America. In this way, any user of a computing center can access any other resource in the network that his local computing center does not provide, and the user need not even be aware of the remoteness of the
access, as the remote access mechanism is completely transparent to him.

1.2 EXPENSIVE COMMUNICATION COST OF COMPUTER NETWORKING

However, the cost of leasing or establishing these high-speed digital communication links is very expensive. For example, the ARPA Network paid over one million US dollars a year (US$1,200,000) in January 1973 for its 50 Kbps data communication links [4] while the ALOHA System has been paying about $10,000 per month to Comsat for a 50 Kbps satellite link to the ARPA Network. The cost of establishing the SBS Network is estimated at US$250 million through 1979 [11]. Clearly, if this link cost can be lowered significantly, then the merits or benefits of computer networking can be made more generally available.

1.3 APPROACH TO LOWERING LINK COST

There are many ways to lower the communication link cost. The approach taken in this research is to see whether there is still capacity available in existing communication links that could be used to carry digital information from one end of the links to the other. This capacity is not an exclusively allocated bandwidth as in the existing communication link; rather, it is shared with the already
allocated analog voice-grade channel.

In the conventional long-distance telephone transmission network, the signal from the calling party runs independently (in a different channel and in an opposite direction) from that of the called party. This is the "four-wire" transmission system. Therefore, each voice-signal transmission channel (in either direction) periodically has a talking period (signal state) followed by a silent period (silent state) during which the respective talker is listening to the conversation of the other end party and during which no information flows in the respective direction. The approach of this research is to make use of the silent periods to transmit digital signals in packets across the existing "four-wire" transmission system. Thus, more than half of the total channel capacity that is originally wasted can be made available for extra digital signal transmission. By collectively using several channels in a group for the same purpose, the aggregate data transmission rate can be fast enough to carry digital signals for computer communication.

This approach can certainly lower the digital communication link cost, since the digital signals only "nitch hike" across the existing transmission system. There is no need to allocate an exclusive wide-band channel for
the digital signal, nor does it need a newly established communication link. The only cost for providing extra digital service is in designing a microcomputer-controlled interface to the existing "four-wire" transmission system. This cost can become negligibly small if this interface controller is designed only once and is manufactured in large quantity. The controlling program in ROM would be the same if the channel configuration can be changed by strapable wires. Such a controller would function like a "digital adaptor" to/from which extra digital transmission service can be provided by the communication carriers within a group of analog voice-grade circuits. Since the cost of this controller is small, the expected access cost for such digital circuits would be very inexpensive as compared to conventional exclusively allocated digital wideband service.

1.4 MAIN OBJECTIVES OF THIS RESEARCH

The main objectives of this research can be summarized as follows:

(1) to lower the data communication link cost for computer communication;
(2) to improve the channel utilization of the existing voice-grade channels;
(3) to avoid exclusive allocation of bandwidth for
data communication;
(4) to save the bandwidth resource for the telecommunication facilities;
(5) to design a digital system that would merge "packet transmission" and "packet switching" into an integrated system;
(6) to design a microcomputer-based controller that would merge "digital transmission" and "analog transmission" into a hybrid or mixed system.

These objectives can be realized due to the advances in microcomputer hardware and firmware techniques. Without these techniques, it would not be cost-effective and justified.

1.5 DESCRIPTION OF THE METHOD

The realization of the objectives can be accomplished by designing a Mixed Voice/Data (MVD) System which uses a microcomputer system as a digital-data insertion and delivery controller at both ends of a long-distance four-wire communication link. The operating system and the application software can be implemented in the microcomputer's PROM or ROM. Certain amounts of RAM are also necessary for digital data buffering and working space. On the transmitting side, this microcomputer is responsible
for dispatching data packets onto a voice-grade channel when the line interface circuitry senses, by interrupt to the microcomputer, that this channel currently is undergoing a transition from a talking to a silent period. Also, it has the responsibility of performing some wrap-up operation for the channel when the line interface circuitry senses, also by interrupt to the microcomputer, that this channel currently is undergoing a transition from a silent to a talking period. The transmitting microcomputer is also responsible for accepting digital data streams from input lines and assembling them into data "packets". On the receiving side, another microcomputer is responsible for receiving digital data packets when the line interface circuitry senses, by interrupt to the microcomputer, that the channel is currently changing to a data transmission state. It also has the responsibility to stop receiving digital data packets when the line interface circuitry senses, by interrupt to the microcomputer, that this channel is currently changing to the analog transmission state. The receiving microcomputer is also responsible for delivery of these data packets as streams of data to the outgoing lines.

At both ends of the communication link, one microcomputer should be enough to take care of all transmitting and receiving functions for digital data packets. This organization makes it more convenient to
implement the necessary handshaking between the transmitting and receiving microcomputers. Otherwise, there would need to be a linkage between the receiving microcomputer and the transmitting microcomputer of each side to facilitate positive or negative acknowledgement of each packet and provide the supervisory functions of the system.

A packet-oriented protocol must be designed to carry both the digital information as well as these supervisory functions, such as "request to send", "receiving processor ready", "request for retransmission" of a packet, "break" during receiving, etc. These supervisory functions will add flexibility and packet accountability to the system operation.

On the transmitting side, the line interface circuitry detects the status of each voice-grade channel and determines whether it is in a talking period (voice state) or a silent period (available for use as a data state). This determination can be made by using voice detectors that have been used in many application, such as in the TASI (Time Assignment Speech Interpolation) System [8]. The requirement of this line interface circuitry is to generate an interrupt signal to the microcomputer at both transition to and from a talking period (or silent period).
On the receiving side, the line interface circuitry must "recognize" the signal in each channel to be either analog voice or digital data. There are many techniques to recognize the kind of signal:

1. Data escort carriers (Inband signalling tones)
2. Signal-pattern recognition
3. Orthogonal-phase signal
4. Bracket signal
5. Common channel signalling.

The detailed descriptions and merits of those various techniques are discussed in Section 2.2.5 of Chapter 2. In Chapter 6, the first method is chosen for its simplicity in logic design.

1.6 MAIN FEATURES OF THE MIXED VOICE/DATA SYSTEM

After implementing the MVD System, it is expected that it will have the following features:

1. Digital data streams or messages can be transmitted over a long-distance four-wire communication link without establishing a new link or exclusively allocating a wide-band channel.
2. The data streams or messages can be switched to any destined outgoing line in the receiving side, and the intermediate medium provides "packet
transmission" as well as "packet switching" capabilities.

(3) The communication protocol of the MVD System enables the receiving side to request the retransmission of any data packet from the transmitting side if any error is detected by the receiving-side error-checking algorithm. It also ensures the correct receipt of packets by sending positive-acknowledgement supervisory packets from the receiving side. The protocol also enables the the receiving side to request permission to transmit its own information packets by sending a "break" supervisory packet.

(4) The average total throughput of the system is an increasing function of voice-grade channels configured as packet-carrying channels.

1.7 EXPECTED CONTRIBUTION OF THE MIXED VOICE/DATA SYSTEM

The Mixed Voice/Data System is a resource saving telecommunication system that provides additional data service for computer communication in the existing analog voice-grade circuits. After implementing the system, it is expected to have the following contributions:

(1) The communication link cost of computer networking would be reduced, thus leaving more funds for
nardware or software resources.

(2) The telecommunication carriers would benefit themselves by being able to provide extra data service without expanding their existing telecommunication links.

(3) The users of data communication links would have the illusion that they are using an exclusively allocated high-speed data channel for their computer communication.

(4) The existing separated analog and digital transmission systems could be integrated into a Mixed Voice/Data hybrid system.

(5) The existing separated transmission and switching systems could be merged into an integrated system.

(6) The problem of inefficient utilization of the current analog voice-grade channels would be solved. This would extend the lifetime of analog type circuits, in which there has been enormous investment.

1.8 ORGANIZATION OF DISSERTATION

This first chapter has served as an introduction to the MVD System. Examined were the merits of computer networking, the expensive communication link cost of computer networking, the approach to lowering link cost, and
the main objectives of this research. Also presented were
the main features of the MVD System and a description of its
expected contributions. An outline of the remaining
chapters of this work will now be given.

Chapter 2 presents motivation for this research,
concepts and the proposed model, and the detailed approach
for realizing the model.

Chapter 3 is a survey of the literature of some related
systems and their basic differences to the proposed MVD
model. In this survey, TASI (Time Assignment Speech
Interpolation) [8], IDAST (Interpolated Data and Speech
Transmission System) or AAVD (Automatic Alternate Voice and
data) [6,26], PCM-TASI or DSI (Digital Speech Interpolation)
[32], DUV (Data under Voice), DAV (Data Above Voice) and
DAVID (Data Above Video) [17], TDMA (Time-Division
Multiple-Access) [42], TTT (PAM-TDMA with Time Preassignment
and TASI) [38], Teletext [27], and Pitch Synchronous Data
Interpolation Scheme [43] are examined and differences in
approaches and characteristics are discussed.

Chapter 4 presents analysis of the MVD System. First,
the talking- and silent-period statistics (density and
distribution) of the satellite communication channels are
observed. Then, the probabilistic approach to the voice
interfering mechanism is studied. Next, the queueing model of a single server (transmission channel) with a preemptive class (voice) of arrivals is treated. Then, the optimum packet size, minimum overhead and maximum throughput are found. Finally, the analysis of a multi-channel MVD System and the preemptive-resume model with constant service time for data packets are examined.

Chapter 5 presents simulation of the MVD System. First, the reasons for simulation are discussed. Then, the approach taken in building the simulation model using GPSS is explained, which is followed by the description of the simulation models. Next, the table of definitions of GPSS entities for the MVD System is defined. Then, the flowchart and listings are presented. Finally, the simulation results are given and discussed.

Chapter 6 presents the implementation of the MVD System. This chapter starts with a discussion of the principle of system operation and functional requirements of the MVD System. A packet-oriented communication protocol is designed to facilitate inter-processor handshaking. Following this, the detailed design of the required hardware components is presented, together with the detailed logic design of components of the line interface circuitry. These components include a microcomputer architecture, voice
detectors, transmitting status change detector, data escort
carrier generator, data modulator, mixer, data escort
carrier recognizer, receiving status change detector, data
escort suppressor, data demodulator, modem interface,
switching devices, and the analog delay devices. Next, the
software implementation of the MVD System is presented. This
implementation starts with the design of an operating system
for the MVD System, consisting of the system monitor,
base-level processing programs, interrupt processing and
analogue programs, and line control (or service) programs.
Then, the software implementation of various route selection
strategies are presented in algorithms.

Chapter 7 is the conclusion of this dissertation. In
this concluding chapter, first a brief summary of the work
is described. Then, the technical maturity and economical
justification of implementing the MVD System are asserted.
Next, the expected contributions are emphasized. Finally,
several areas of follow-up research are discussed subjects
for future exploration.
CHAPTER 2

MOTIVATION, CONCEPT, APPROACH, AND PROPOSED MODEL

2.1 MOTIVATION AND CONCEPT

In view of the high cost of leasing exclusive wideband circuits for computer communication and the wasteful way the limited resource of bandwidth is used in existing long-distance voice telecommunication channels, a novel concept was developed hereby it would be possible to multiplex high-speed digital data streams for computer communication into existing multiple voice-channels during voice conversation. As a result of three years of research and the rapid advances in microcomputer technology, this idea was proved to be both economically justified and technically feasible. The implementation of this idea will, therefore, not only improve the utilization of bandwidth for long-distance telecommunication channels but will also provide an inexpensive alternative for computer communication.

2.1.1 Inefficient Utilization of Channel Bandwidth

According to field observation on the international voice-grade circuits, statistical research shows that the
voice signal's occupancy in each direction of the 4-wire circuit is less than 38% of the total time [48]. In other words, each direction of the 4-wire circuit is 62% of the time in a silent state during which there is no information transmitted from one side to the other. This is an extremely wasteful use of a resource as expensive and scarce as a telecommunication circuit. Furthermore, if we examine a voice channel from the viewpoint of information theory to see how much information has actually been transmitted over an individual 4 KHz channel, we find that it has been used at a surprisingly low rate.

For example, assume that a person talks *continuously* through a telephone channel, and that he has a vocabulary of 10,000 words. Furthermore, assume that he talks at a speed of 3 words per second and has the freedom of selecting his words randomly among the words in his vocabulary. For that case, the information rate conveyed by this "zero-memory information source" (assuming also that the waveform characteristic of the voice signal conveys no information content) is only

\[ 3 \log_2 10,000 = 40 \text{ bps}. \]

This information rate is even less if we consider the inter-dependence among neighboring words, since it is known
that an *m*-th order Markow source has less entropy (average information) than the corresponding *adjoint* information source [1].

Of course in normal conversation, the speaker does not talk *continuously*. He merely spends about 40% of his time in talking while waiting the other 60% in listening to his partner's conversation or preparing to speak again himself. Therefore, the average information conveyed is less than 16 bps (in a 4 KHz bandwidth channel).

Moreover, Shannon has shown that the maximum channel capacity for a noisy channel is [33]:

\[ C = W \log_2 (1 + S/N) \text{ bps} \]

where \( W \) is the channel bandwidth in Hz and \( S/N \) is the average signal to noise ratio.

For a voice-grade channel with \( S/N \) being 30 dB and \( W \) being 3700 Hz (300-4000 Hz), the maximum channel capacity of a voice-grade circuit in one direction is
\[ C = 3700 \log_2 \left( 1 + \frac{1000}{1} \right) \]

\[ = 36,879 \text{ bps}. \]

It should be readily apparent how wasteful it is to use the voice-grade channel to convey only spoken information!

### 2.1.2 Present Frequency Division Multiplex (FDM) Scheme for Long-Distance Telephone Circuits

In long-distance telephone communication, such as those through satellite communication links or the East-West terrestrial microwave links in the United States, one telephone channel always consists of four wires, two for each direction. Fig. 1 shows a one-line schematic of such a typical telephone conversation, while Fig. 2 shows a simplified block diagram of an overseas transmission path through various switching and maintenance centers and satellite communication facilities. From these figures, we see that the transmitting and receiving paths are divided by a hybrid circuit at the International Switching and Maintenance Center (ITMC). The purpose of dividing the paths is to increase the signal to noise ratio (S/N) using amplifiers and to suppress the echo which occurs 0.6 seconds after each spoken word in satellite communication.
Fig. 1. One-line Schematic of Long-Distance Telephone Conversation.

Fig. 2. International Telephone Transmission Path through Satellite Circuits and the Position of the MVD Controller.
The transmission of outgoing telephone signals for multiple simultaneous conversations is accomplished mostly by using a Frequency Division Multiplex (FDM) scheme. This scheme translates each 4-KHz voice signal into a 4-KHz frequency slot (channel) in frequency space and arranges all the translated voice channels to form a Basic Group (12 consecutive 4-KHz channels). Five consecutive Basic Groups are then translated to form a Basic Supergroup. This configuration is shown in Fig. 3 [36]. The arranged multi-channel voice signals then constitute a baseband. Therefore, each outgoing telephone voice signal occupies a 4-KHz slot in the baseband. This slot will be occupied by a voice signal whenever the person on this side of the circuit talks and will be empty whenever he stops talking.

The composite baseband signals are then fed to the input of an FM modulator which converts the composite baseband signals into a modulated carrier having the center frequency at 70 MHz Intermediate Frequency (IF). This IF modulated carrier is up-converted to an assigned microwave Radio Frequency (RF) and then after high power amplification transmitted to the satellite.

In the satellite transponder, each RF carrier carrying all the outgoing telephone-signal information from each satellite communication earth station occupies an RF
Fig. 3. Baseband Configuration of Frequency Division Multiplex (FDM) Telephony.
frequency bandwidth depending on the number of voice-grade channels compositd inside the baseband. For the INTELSAT-IV Global Beam System, the RF carrier for 60-channel baseband will occupy 5 MHz satellite bandwidth, while for 132 and 252 channels it will occupy 10 MHz and 15 MHz, respectively [36].

The incoming telephone signals are extracted in exactly the reverse order of the sequence described above. The RF carriers from those earth stations that have telephone circuits with this earth station will be filtered-out after low-noise amplification. Each RF carrier will be down-converted to the IF modulated carrier, which is then fed to the FM demodulator to extract the baseband composite signal. Finally, the composite signal is "de-multiplexed" to separate individual 4 KHz voice-grade channels. Such a system is called Frequency-Division Multiple-Access, Frequency Modulation, Frequency Division Multiplex (FDMA-FM-FDM) System. Fig. 4 shows a block diagram of the earth station communication equipment needed to perform all these actions.

From the above description, it can be seen that the bandwidth of communication channels is a valuable resource that has to be more efficiently utilized, since it is limited in baseband frequency space as well as in the
Fig. 4. Block Diagram of Earth Station Communication Equipment.
satellite frequency spectrum.

Although the PCM-PSK-TDM (Pulse Code Modulation - Phase Shift Keying - Time Division Multiplex) technique is currently available, most existing facilities are still using the FDMA-FM-FDM technique. Since very large amounts of money are invested in these communication facilities, they cannot be abandoned for another ten or twenty years. However, slight modification of them can improve their utilization of the bandwidth resource tremendously.

2.1.3 High Cost of Leasing Exclusively Allocated Data Communication Channels

In designing a distributed computer-communications network, one of the greatest recurring costs involved is the tremendous amount spent in leasing data communication links interconnecting computers and their remote terminals. This cost nearly always outweighs the attractive merits of computer communication, such as resource sharing and reliability enhancement. For example, the ALOHA System [2,3,4,28] has been paying about US$10,000 per month (including the ground link to COMSAT and Hawaiian Telephone Co.) for a 50 Kbps data communication circuit linking it to the ARPA Netork [4] while the ARPA Network [20,39,41] paid
over one million US dollars (US$1,200,000 in January 1973) a year for its 50 Kbps data communication links [4]. The former uses an SCPC (Single Channel per Carrier) channel in the SPADE System and occupies 45 KHz of satellite bandwidth; the latter uses a Group of 12 voice-grade channels which occupy 48 KHz in the terrestrial microwave baseband. These are both examples of exclusively allocated channels. One reason behind this high rental cost for exclusively allocated circuits is that data channels are paying a great penalty in cost due to the large amount of bandwidth inefficiently occupied by voice circuits, which results in a scarcity of available bandwidth. Another reason may be that the data channels have richer information content and better channel utilization.

<table>
<thead>
<tr>
<th>Speed</th>
<th>2400 bps</th>
<th>4800 bps</th>
<th>9600 bps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destination</td>
<td>US $</td>
<td>US $</td>
<td>US $</td>
</tr>
<tr>
<td>USA/Australia/Canada</td>
<td>12,105</td>
<td>15,132</td>
<td>21,184</td>
</tr>
<tr>
<td>Japan/Korea/other Asian Countries</td>
<td>8,474</td>
<td>10,579</td>
<td>14,816</td>
</tr>
<tr>
<td>UK/Italy/other European Countries</td>
<td>14,526</td>
<td>18,158</td>
<td>25,421</td>
</tr>
</tbody>
</table>

Table I. Monthly Rental of Leased Data Communication Circuits from/to Taiwan, Republic of China (1977).
Table 1 shows monthly rental costs of leased data communication circuits to/from Taiwan, Republic of China, via satellite communication routes*.

2.1.4 Resource Saving Principle

The energy crisis of this decade has given people a very useful lesson in showing that all kinds of resources are scarce and have to be efficiently utilized. It was mentioned that the existing voice-grade analog transmission circuits are very inefficiently utilized resulting in a great waste of the resource of telecommunication bandwidth. Therefore, the motivation is to improve the circuit utilization of the voice-grade channels.

As an analogy of how this can be accomplished, some people find it inexpensive to travel long distance by getting a ride with another person if there is space available in the other person’s car and his car is going to the same destination or even in the same direction. The rider need only share the gas expense. He may even ride in

several different cars before reaching his final destination. The same principle can be applied to a voice-grade channel in which the silent periods are available for carrying other information on the same voice-grade channel.

2.2 APPROACH AND PROPOSED MODEL

The idea is to insert digital data in packets under the control of a microcomputer and not to alter the voice-grade circuits in their analog form. To use an analogy with automotive traffic, the existing voice-grade channels function like parallel running freeways to which there are entrances to get on and from which there are exits to get off. Voice signals can be considered as the "through traffic" in the freeways, and data packets can be considered as cars waiting for a clear signal at the entrances to get on the freeways and driving for a long distance to find an exit to get off. At the entrances of the freeways there are semaphores that signal to the waiting cars when to get on which freeway. Because the approaching cars in the through traffic arrive in random fashion and to avoid possible collisions, the detection of an arriving car must be early enough for the waiting cars to decide whether to get on or to wait. Once getting on the freeway, each car will carry
two flags so that it can be identified at the exit. The above example is a very close conceptual model to the proposed microcomputer-controlled data packet insertion concept. Because it uses data packet insertion into the existing voice signal, the conceptual model is called the Mixed Voice/Data Transmission System and is abbreviated as the MVD System.

The proposed MVD System can be described as having the following characteristics:

(1) It uses a microcomputer to control the insertion of digital data packets;

(2) It keeps existing voice-grade circuits in analog form and considers voice signals as real-time through traffic having higher priority than data packets;

(3) It retransmits the data packet or the residual part of the data packet whenever an interference occurred due to the unexpected arrival of a voice signal;

(4) It uses voice detectors to sense the circuit activity status and to interrupt the microcomputer whenever there is a change of circuit status between voice and data;

(5) It uses a sentinel or data escort carrier to
accompany the data signal in transmission and uses a data signal recognizer to distinguish data from voice at the receiving side so that they can be switched accordingly by a semiconductor switch;

(6) It uses multiple voice-grade channels (e.g., 12 or 24 channels) to achieve higher data throughput and to provide flexible choice of data packet routing;

(7) It allows single or multiple input data lines at both transmit and receive side so that a "packet switching" function can be simultaneously realized in the microcomputer software;

(8) It considers fixed-size data packets as well as variable-size packets for data traffic;

(9) It applies clock acknowledgement and packet retransmission protocol for error control and uses data scrambling techniques for data security during transmission;

(10) It considers data-packet queueing principle and dispatching strategies to achieve maximum throughput and minimum delay;

(11) It applies recent microcomputer hardware technology and design to store all the control program in a Read Only Memory (ROM);

(12) It uses synchronous-type data transmission and applies the most efficient modulation scheme (e.
g., PSK modulation) to achieve the highest data rate obtainable in each voice-grade channel.

Fig. 5 shows a simplified signal flow diagram of a multi-channel Mixed Voice/Data (MVD) Transmission System for simplex operation. In this diagram a microcomputer plays an important role in both the transmit side and receive side. Some of the characteristics will be explained in detail to show now and why the proposed model is derived.

2.2.1 Using Microcomputer for the Controller

Microcomputer technology has improved very rapidly in the last five years. Many powerful microcomputers are available today that are gradually replacing what previously only minicomputers and hardware logic could do. Among them the INTEL 4004, INTEL 4040, INTEL 8008, INTEL 8080, MOTOROLA 6800, NATIONAL IMP-10, FAIRCHILD F-8, TEXAS INSTRUMENT S8P 8400, GENERAL INSTRUMENTS CP-1600 and SIGNETICS 2650 are most widely known [34]. Recently, INTEL announced its 8085 and 8086 and ZILUG its Z-80. The Z-80 with its powerful instruction set, including bit handling and block transfer instructions, perhaps represents the most advanced microcomputer technology in 1977 [23]. An MVD controller built using these kinds of products will cost only a few thousand US dollars. Therefore, the approach using a
Fig. 5. Conceptual Model of Simplex MVD System.
microcomputer for the MVD System is economically justified. Furthermore, the development expense for control software is a non-recurrent cost and it can be spread over any number of mass-produced systems. However, hardware, which must be duplicated in every system, is a recurrent cost item, and each system must be assembled and tested separately*.

The design of the MVD controller in this research is based on the INTEL 8080A microcomputer system which seemed to be a good choice two years ago when the 8085, 8086 and Z-80 had not been announced. While one of the latter microprocessors might be a better choice if the MVD controller were redesigned today, the INTEL 8080A is still more than adequate to perform the desired functions and it is also cheaper.

2.2.2 Keeping Analog Voice-Grade Channel Unchanged

Analog voice-grade circuits, although very inefficient as far as information carrying is concerned, constitute a major investment in telecommunication facilities for the companies that provide them. Even the most advanced digital TDM-PCM technology cannot completely replace their

---

* See page 17 of Reference [34]
existence. Such a replacement involves very large new investments and immediate suspension of operation, even though some facilities have not been in service for their lifetime yet. Another reason for maintaining analog circuits is that an analog voice signal, if digitized, requires a 64 Kbps data rate to transmit without noticeable distortion, which requires a wideband channel (40 KHz) for transmission using the conventional 8-bit PCM technique. Although a major breakthrough to lower the transmission rate to 9.6 Kbps has recently been achieved by using a Residual-Excited Linear Prediction Vocoder [46], practically the compressed digitized voice signal can only be transmitted at the 20-30 Kbps range [15]. Therefore, analog voice-signal transmission finds many reasons for its continuing operation in competition with its digital counterpart.

The approach for the MVD controller is to avoid major modification to the analog voice-grade system, thereby eliminating the excessive cost involved in re-implementation. The only change required is the insertion of an MVD controller by tapping to the appropriate points in the existing analog voice-grade channels. This approach not only maintains the continuous operation of the analog voice-grade channels but also provides additional inlets and outlets for data service.
2.2.3 Treatment of Data Packet Interference

The treatment of data packet interference by the random arrival of voice signals is the major task of the MVD controller. Interference occurs when a voice signal arrives while a data packet is still being transmitted. Several techniques can be used for dealing with interference.

The ALOHA System solved the mutual interference of data packets transmitted from remote terminals by packet retransmission whenever positive acknowledgement is not received in the predetermined period of time [3]. The idea of retransmission can also be applied to the MVD System. However, there is an important difference in retransmission between the ALOHA System and the MVD System.

2.2.3.1 Difference in Retransmission between the ALOHA and the MVD Systems

In the MVD System, the reason for a packet retransmission is due to the unexpected arrival of a voice signal in a voice-grade channel during the transmission of a data packet in that channel. Therefore, the transmit-side CPU knows whether the transmitting data packet is being freely sent or is being interfered with by the arrival of voice signals, since the Voice Detector constantly informs
the CPU regarding the status of all voice-grade channels. If a data packet is being transmitted in a channel and if the Voice Detector reports that there is an arrival of a voice signal in this channel, the CPU will be interrupted to cause it to stop transmission of the data packet in this channel and to arrange for its later retransmission. In other words, the transmitting CPU itself can decide whether to retransmit a data packet, since it has a centralized Line Status Board in its memory and does not depend on the acknowledgement from the receiving-end microcomputer. In the ALOHA System, the transmitting packet is always broadcasted to the end of a packet, no matter whether it encounters interference or not, and the transmitting CPU depends on the information coming from the receiving CPU to decide whether to retransmit or not. In this regard, the MVD retransmission scheme is different.

2.2.3.2 Partial Packet Transmission
(Preemptive-Resume vs. Preemptive-Repeat)

For more efficient transmission, an alternative method of handling the voice arrival interrupt is to append a packet tail to the partially transmitted packet to complete it successfully and then later to transmit only the residual part of the packet instead of abandoning the already
transmitted partial packet and restarting from the beginning. This method of handling partial packets is
called the preemptive resume technique, while the one that
only handles full packets is called the preemptive repeat
technique. Intuitively, the data throughput of the
preemptive resume technique will be greater than that of the
preemptive repeat technique. However, the software control
logic may be simpler in the latter case, as it does not have
to deal with variable-length and partial packets.

2.2.4 Using Voice Detector

The microcomputer CPU shall be constantly informed
about the states of each voice-grade channel. It is the
responsibility of the Voice Detector and its associated
logic to sense the state changes and to tell the CPU so it
can update its Line Status board. When the beginning of a
new silent period in a channel is detected, this channel may
be selected by the CPU as a candidate for transmitting a
data packet. On the contrary, when the arrival of a voice
signal in a channel that is transmitting a data packet is
detected, an immediate wrap-up procedure must be undertaken
for this channel by the CPU to stop that packet
transmission. Therefore, the Voice Detector and its
associated logic shall be responsible for interrupting the
CPU at both the beginning and the end of each talking period, and the CPU will stop packet transmission and start packet transmission accordingly. Fig. 6 shows the state transition diagram of an MVD channel.

To minimize the probability of a voice arrival interrupt during a packet transmission, a small silent period shall not be used for data packet transmission. Thus, a suitable time constant shall be added to the Voice Detector to obtain a lagging edge of the detection response. On the other hand, to minimize voice clipping at the beginning of each talking period, the Voice Detector shall be sensitive enough to obtain a fast detection leading edge of the detection response. It was proposed by Fariella that a threshold of -40 dBmO is generally used to avoid noise triggering but to detect voice without noticeable clipping [16].

2.2.5 Recognition of Voice and Data

The voice-grade circuits are designed suitably for AC signal transmission. Thus, data packets in DC bit streams must modulate an AC carrier for their transmission. Therefore, the modulated carrier will provide similar AC characteristics as for the voice signal. It thus becomes a difficult problem for the receive side to separate the data
STATE DIAGRAM OF A VOICE CHANNEL USED FOR MVD SYSTEM

Fig. 6. State Diagram of a Voice-Grade Channel Used for MVD System.
signal from the voice. Various methods can be used to solve this problem as described below.

2.2.5.1 Inband Signaling Tones (Data Escort Carriers)

Whenever the modulated data carrier appears in the voice-grade line, there is always a signaling tone accompanying the data carrier inside the 4 KHz bandwidth. This signaling tone is sometimes called a sentinel signal or data escort carrier. At the receive side, the detection of this signal will signify that the signal currently in the channel is the data packet modulated carrier. By using this detected signaling tone, a fast actuating semiconductor switch will take the modulated carrier to the exit path, leaving the voice signal going to the other "through traffic" path. Dual or multiple signaling tones can be used to reduce the probability of misinterpretation of the existence of a data carrier, thereby improving the operational reliability of the MVD System.

2.2.5.2 Common Channel Signaling

A separate channel can be assigned as a common signaling channels dedicated for all the channel in a group. The signaling information in this common channel can be
multiplexed using FDM or TDM techniques as shown in Fig. 7. This method is justified only if there are a large number of channels used in the MVD System.

2.2.5.3 Signal-Pattern Recognition

There is also the feasibility of using a signal-pattern recognition method to determine whether the current signal in the voice-grade channel is either data or voice. The concept of this method is based on the differences in statistical characteristics of voice and data signals in amplitude, frequency spectrum and energy variation inside the channel. For example, the voice signal appears rather random in the above three aspects, while the data signal carrying only binary signal appears rather regular. These features can be used as clues to distinguish data from voice.

This method involves special selection of data carrier frequency as well as the modulation scheme, so as to create greater discrepancies in characteristics with respect to the voice signal. A signal-pattern recognition network can be designed using integrators, differentiators, detectors, and combinatorial logic circuits. The delay in recognition can be tolerable if it is less than 100 msec.
(a) Channels 1, 4, 6, 7, 10, and 11 are in data state; Channels 2, 3, 5, 8, 9 and 12 are in voice state.

Frequency Division Common Signalling Channel

(b) Time Division Common Signalling Channel
(Adjacent pulses may be combined).

(c) Time Division Common Signalling Channel
(Adjacent pulses may not be combined).

Fig. 7. Common Channel Signalling. (a) FDM Common Channel Signalling, (b) and (c) TDM Common Channel Signalling.
To be practically implemented, this method would involve extensive research on signal analysis, spectrum analysis and pattern recognition and thus will not be explored further in this research. The advantage of this method, however, is that the total bandwidth of a channel can be used to transmit data packets, thereby providing a higher data rate for each channel. Fig. 6 shows a three-dimensional hyperspace in which two decision cubes are characterized as data traffic.

2.2.5.4 Bracket Signal

Bracket signals can also be used to delimit the data signal in time space. For example, an "open bracket" signal can be inserted at the beginning of a new silent period before any data packet is sent to the line, while a "close bracket" signal can be inserted at the beginning of a new talking period. These bracket signals appear on the line as short bursts which will be used to control the outgoing paths at the receive side. The reliability of this method is much inferior to the inband signaling tones, however, since a false detection of a bracket signal could lead to a serious mistake of signal flow to the wrong path.
Fig. 8. Hyperspace Shows Decision Cubes for Data Traffic.
Among the methods presented above, the inband signaling method appears to be the simplest, the most reliable, and the easiest to implement. Thus this method has been adopted in the hardware implementation of an experimental MVD controller.

2.2.6 Using Multiple Voice-Grade Channels

Since one of the objectives of the MVD System is to provide a less expensive alternative for computer communication, a high-speed data throughput is desirable. From the current state of the art, a voice-grade channel can achieve 9600 bps medium speed when using a 4-phase PSK modulation scheme to transmit data [9]. If the entire silent periods that occupy about 60% of the time are used for data transmission using the same modulation scheme, a single MVD channel on the average can transmit data at 5760 bps. Therefore, ten MVD channels will be sufficient to accomplish 57 Kbps if all the MVD channels are busy with voice signals only 40% of the time. Thus, a group (12 VG channels) seems to be a suitable size to provide high-speed data service for computer communication. If higher speed is desirable, more groups can be used as MVD channels by a simple change in the channel configuration.
2.2.7 Allowing Multiple Input/Output Data Line at Transmit/Receive Side

The packet switching function can be implemented in the microcomputer software so that a message coming into one of the input lines in the MVD transmit side can be switched to one of the output lines in the receive side. The switching is of "packet" type rather than "message" type because each message is divided into packets for transmission, and each packet need not follow the same MVD channel for its transmission to the receive side [25,47]. This function can be accomplished simply by checking the destination line number in each packet header.

To facilitate multiple input/output lines, some kind of multiplexing or line scanning devices must be considered in the design. Recent development of USART (Universal Synchronous/Asynchronous Receive and Transmit) line interface chips can facilitate the functions. The USART will provide line interrupts to the CPU whenever its buffer is full or empty. One interface chip can be used for each input/output line. A software driver including the interrupt response and interrupt processing routines must be available to handle the input/output of these multiple data lines.
2.2.8 **Fixed-Size Packets vs. Variable-Size Packet**

The reason for using a fixed-size packet frame is that the data message transmission protocol can be greatly simplified. However, a variable-length transmission system could experience less average transmission delay [30,40]. In the MVD System, if the fixed-size packet is used, then the retransmission of the whole packet shall be necessary whenever an unexpected voice signal arrives and the packet has not finished its transmission. As it has been discussed in Section 2.3.3, a fixed-size packet is split into two variable-size packets only at the time of voice arrival, the first one of which has already been transmitted up to the moment of voice arrival, and the second one of which is the residual packet that must be transmitted urgently when a new silent period is detected. For a variable-length packet, the length and the checksum information are essential to be stored in the packet tail for the receive side to count the characters and to check for possible transmission errors. In the implementation, it was chosen to use fixed-size packet and to allow variable-length packets to be transmitted before and after a voice arrival interrupt.
2.2.9 Block Acknowledgement and Data Scrambling

In the MVD System there is an arrangement for a "block acknowledgement" from the receiving CPU to the transmitting CPU, giving it the reception condition of each data packet in the block so it can check for transmission errors on the communication channels. In satellite communication, due to the long transmission delays (about 0.6 seconds round trip), the transmit side cannot afford to wait for the immediate acknowledgement of each transmitted packet from the receiving CPU. Such an immediate acknowledgement greatly reduces the data throughput rate. Due to the high quality of these communication lines and their very low error rate, only a small percentage of data packets will encounter errors and need retransmission. Therefore, a means shall be provided for these erroneous data packets to be recovered on the transmit side and be retrieved for their retransmission.

The block acknowledgement functions as follows: Each data packet received at the receive side will be positively or negatively acknowledged if the checksum computation is successful or failed, respectively. This acknowledgement is sent through the reverse channel using supervisory packets. On the transmit side upon reception of these acknowledgements, the transmitting CPU will either delete the proper packet previously transmitted from its storage or
retrieve it from the storage for immediate retransmission. The packet-holding storage can be either large amounts of RAM or diskette secondary storage with its file management system.

Data security can be maintained during transmission by using a template to be logically operated with the data packets and using the same template as a key to recover the scrambled data. Various pre-assigned templates can be stored in the memory so that the one specified by the packet header will be taken to scramble the data packet at the transmit side and to descramble it at the receive side. This measure is to prevent the data packets being deliberately intercepted during their transmission. The assignment of the templates shall be accomplished through a bilateral agreement so that the third party will not know which template to use for the packet descrambling.

2.2.10 Data Packet Queueing and Dispatching

Unlike analog voice signals, digital data packets can be stored in computer memory and then forwarded. Therefore, data packets are assigned a lower priority level than voice signals to use the voice-grade lines. On the transmit side, data packets after their formation will wait for being queued for their transmission service. Since multiple
channels or servers are provided for such service but that service must occasionally be interrupted, multi-queues are set up in front of each server or a single "quicky line" is set up for all the servers [44]. The relative merit of using both types of queues are analyzed more extensively simulated in Chapter 5. For the single queue, the strategy of dispatching the very front data packet to an appropriate channel is also a complicated problem to be solved. Various dispatching strategies can be studied to find out their affects on the data throughput and packet delay. For example, round-robin, first silence, most recently release, or least previous interruptions, could be used for channel selection strategies for the single queue. For the multiple queues, the strategy of selecting a queue could be according to the minimum queue length, round-robin or random assignment. These strategies will be examined and compared in Chapter 5. An analytic comparison will not be possible to evaluate the effectiveness of these strategies. However, using a discrete simulation language such as GPSS will provide some insight to their effectiveness [22, 44, 48, 49].

2.2.11 Recent Microcomputer Hardware Technology

As discussed in Section 2.2.1, the MVD controller will be implemented using recent microcomputer hardware technology. Without this technology, this controller would
not be economically justified. Also, LSI technology provides us with many low-cost logical and functional chips which make the design of the MVD controller simpler.

In order to store the control program, some low-cost semiconductor memory chips are used for ROM or RAM. The INTEL MOS ROM provides up to 2K x 8 capacity and MOS RAM provides up to 1K x 1 high speed static and up to 4K x 1 dynamic RAM capacities [24]. Recent development in microcomputer technology makes the MVD design technically feasible. The stored program controller makes the design more flexible to fit in various operational requirements and simplifies the hardwired logic.

2.2.12 Synchronous Data Transmission and Modulation

Synchronous data transmission does not require start and stop bits, thereby increases the transmission efficiency. However, bit timing information must be transmitted to the receive side for its synchronization. This information can be provided in the open flag of the communication protocol so that the receive-side line terminator can recover it for the succeeding sampling of each bit position. Moreover, synchronous data transmission complies with the modem specification when higher data rates (say 9600 bps) is required [13].
In order to obtain high throughput for the MVD System, the average data rate for each channel shall be as large as possible. For synchronous-mode operation in a voice-grade channel using current modems, the highest data rate is 9600 bps using the four-phase Phase Shift Keying (PSK) modulation scheme [12,13,31].

However, in the experimental MVD controller described in Chapter 6, the simplest Frequency Shift Keying (FSK) modulation scheme was used. In this scheme one frequency represents the binary zero while the other frequency represents the binary one. This design is to avoid too much sophistication of the hardware during the initial attempt. However, the optimum design should adopt the four-phase PSK as described above.

2.3 SUMMARY

This chapter describes the motivation, concept, approach and proposed model of the MVD System. Of the major motivations, the inefficient utilization of the existing voice-grade channels, the high cost of leasing exclusively allocated data channels, and the resource saving concept that has been prevailing over the world since the energy crisis, are the most important reasons for this research. The concept is to make use of the silent periods in the voice
conversation to insert digital data in packets without having to allocate an exclusive data channel for digital communication. This will improve the utilization of the analog voice channels and at the same time reduce the cost of digital communication. The proposed model is a multi-channel, packet-oriented, packet-retransmission and voice interruptable system. The approach is to use the up-to-date microcomputer technology to control the data-packet insertion and delivery with the stored control programs in the microcomputer [MUM]. Such an approach will be very inexpensive and can provide an inexpensive link for computer communication.
CHAPTER 3

SOME RELATED SYSTEMS AND THEIR DIFFERENCES
TO THE PROPOSED MVD MODEL

Various other approaches have been tried in the past which can partially achieve some objectives of the MVD System. Its most important objective is to improve the utilization of bandwidth resource. Among these other approaches, TASI (Time Assignment Speech Interpolation) [6], IDAST (Interpolated Data and Speech Transmission System) or AAVD (Automatic Alternate Voice and Data) [6,26], PCM-TASI (Pulse Code Modulation Time Assignment Speech Interpolation) or DSI (Digital Speech Interpolation) [32], DUV (Data Under Voice) and DAV (Data Above Voice), DAVID (Data Above Video) [17], TDMA (Time-Division Multiple-Access) [42], TTT (PCM-TDMA with Time Preassignment and TASI) [38], Teletext [27], and Pitch Synchronous Data Interpolation Scheme [43] are widely known to the telecommunication world. Each system has its particular characteristics and is briefly described below.

3.1 TASI SYSTEM

The pioneering concept to improve voice channel utilization is perhaps the Time Assignment Speech
Interpolation (abbreviated as TASI) technique that was used in the early Atlantic and Pacific submarine cables [8]. Due to the limited number of telephone channels in the submarine cable, these channels would only be assigned to the active voice trunks. As soon as user stopped talking, the previously assigned channel would be reassigned to another active trunk. A common signaling channel was used to send trunk-channel connection information so that the other end of the cable would know the proper association at any instant. This system is analogous to merging the automotive traffic from a four-lane boulevard onto a two-lane narrow bridge without slowing down the traffic speed. If the traffic on the bridge is congested, it is expected that automobile accidents might happen. Similarly, when the number of active voice trunks is greater than the number of available serving channels, the TASI system will become overloaded. In this case, voice freeze-out will occur, which results in voice clipping that cannot be avoided because of the real-time nature of the voice conversation.

Unlike the MVL System, the TASI System has the following characteristics:

(1) It provides a switched path or channel for voice trunks;
(2) The number of trunks is greater than the number of channels;
(3) Voice signals are interpolated with those of other trunks;
(4) It employs common channel signaling; and
(5) Voice clipping will occur during overload condition.

However, it is related to the MVD System in the following sense:

(1) It improves voice-grade channel utilization;
(2) It uses the voice inactive (silent) period to provide additional service.
(3) It applies an interpolation scheme;
(4) The voice signal remains in analog form; and
(5) It is a resource saving system.

3.2 IDAST or AAVD System

ITT'S subsidiary, Standard Telephone Company (STC), England, developed an Interpolated Data and Speech Transmission System (IDAST) or Automatic Alternate Voice and Data (AAVD) in 1960 to provide facilities for the automatic mixing of speech and telegraph data traffic in a single voice-grade circuit [6,26]. This IDAST equipment delays speech signals on the transmitting side by recording them on a rotating magnetic drum and reading them off 60 milliseconds later, thereby achieving advanced awareness of
speech arrival so that data traffic can be cleared up and interference or clipping of speech will not occur. A 2930 Hz control tone was transmitted with data to permit the receiving IDAST unit to identify arrival signals as data or voice and to switch accordingly.

Unlike the MVD System, the IDAST has the following characteristics:

1. The mixing of speech and telegraph data traffic was only in a single voice-grade channel;
2. The achievable data rate was limited to low speed of 50 to 75 baud;
3. The voice signal was delayed for 60 milliseconds using a clumsy and expensive magnetic drum;
4. It was a message switching system for telegraph data traffic but not a packet switching system;
5. It did not retransmit the data traffic when encountering voice arrival interference;
6. Too much hardware switching logic was used instead of more flexible and efficient control software;
7. Speech interrupts took place whenever continuous speech was in progress longer than 10 seconds; and
8. Speech interrupts were warned to take place in 10 seconds when data storage reached 95% of its capacity.
Items 7 and 3 mean that voice signals do not have the absolute highest priority over data traffic.

However, the IDAST system is related to the MVD System in the following sense:

(1) It improves the utilization of a voice-grade channel;
(2) It is a voice and telegraph data interpolation system; and
(3) It uses a 2900-Hz control tone for telegraph data recognition.

3.3 PCM-TASI or DS1 SYSTEM

The PCM-TASI or DS1 (Digital Speech Interpolation) system [32] basically applies the same concept as that of TASI except the voice signal is digitized using the standard 8-bit PCM (Pulse Code Modulation) encoding before it is interpolated. Therefore, it is also called Digital Speech Interpolation. The data rate used to transmit voice signals is 64 Kbps, which requires a wideband circuits. An internal DSI signaling is necessary to inform the receiving side of the association of trunk to transmission channel. Two types of assignment can be used, namely, cyclical and differential assignments, each of which requires an additional 600 bps and 300 bps, respectively, for transmission of the
assignment message for the association. The interpolation
is achieved by using a Time Division Multiplexing (TDM)
scheme to insert the digital data into proper time slots
that do not have digitized voice signal. Various procedures
are used to minimize the effect of voice clipping during
speech freeze-out overload conditions. A "bit reduction
scheme" from 8-bit to 7-bit PCM encoding was found to be the
most promising; its degradation is unnoticeable even at
overload peaks of 50-60 %.

Unlike MVD System, the DSI System has the following
characteristics:
(1) Voice signals are digitized using PCM encoding;
(2) A single wideband channel at 64 Kbps is used;
(3) A TDM multiplexing scheme is used for voice signal
interpolation;
(4) A "bit reduction scheme" is used to relieve the
overload condition that results in voice
clippings;
(5) a TDM common channel is used to transmit the
assignment message; and
(6) No data traffic is interpolated.

However, the DSI System is related to the MVD System in
the following sense:
(1) It is an interpolation system that takes advantage
of the voice active and inactive periods;
(2) The utilization of the transmission channel is improved; and
(3) It is a resource saving system.

3.4 DUV, DIV, DAV AND DAVID SYSTEMS

Normal data channels are exclusively allocated in-between the FDM (Frequency Division Multiplex) voice channels in the baseband. The allocated bandwidth depends on the required data rate. For wideband operation, 50 Kbps needs 12 VG channels and 1.544 Mbps needs 120 VG channels. Each allocated bandwidth is sandwiched by FDM voice channels and such allocation is called Data In Voice (DIV) as shown in Fig. 9 [17].

Bell Laboratory implemented a Data Under Voice (DUV) System in October 1971 and put it into operation between New York and Chicago in 1973. This system made use of the low frequency range of the baseband in terrestrial microwave links for 1.544 Mbps data transmission. The original order-wire channels were moved to the upper end of the baseband as shown in Fig. 10. A Double Data Under Voice (DDUV) System was put into operation between Montreal and Toronto by using two different-path DUV systems. At the receiving side, a comparator is used to select a path having
Fig. 9. Spectrum of Data In Voice (DIV) in Baseband.

Fig. 10. Spectrum of Data Under Voice in Baseband.
a better S/N (Signal to Noise) ratio [17].

RCA developed a Data Above Voice (DAV) System that made use of the upper region of baseband in terrestrial microwave links to achieve 1.544 Mops data rate [17]. Its baseband configuration is shown in Fig. 11. Similar arrangements can be used for video baseband to implement Data Above Video (DAVID), as shown in Fig. 12.

As compared to the MVD System, these systems (except the DIV) have the following characteristics:

1. The data channels transmit simultaneously with voice channels; no interpolation is made with the voice signals;

2. It uses the low or high frequency region of the baseband to obtain an improved baseband utilization; and

3. It does not improve the utilization of individual voice-grade channels.

However, DUV, DAV and DAVID Systems are related to the MVD System in the following sense:

1. They create additional data channels for data communication in existing terrestrial microwave links;

2. The utilization of baseband is improved by these
Fig. 11. Spectrum of Data Above Voice (DAV) in Baseband.

Fig. 12. Spectrum of Data Above Video (DAVID) in Baseband.
system; and

(3) They are resource saving systems.

3.5 TDMA SYSTEM

COMSAT has developed a prototype Time-Division Multiple-Access (TDMA) System for use with INTELSAT IV satellite [42]. The Interface Module of the TDMA System converts analog voice signal into digital signals via PCM encoding on the transmit side and performs the inverse operation on the receive side. Then, it converts these digital signals from continuous signal streams to burst signals by compression buffers on the transmit side and performs the inverse operation via expansion buffers on the receive side. The data communication link is a 36 MHz wide RF carrier operating at 60 Mbps using a 4-phase PSK modulation scheme on a complete INTELSAT IV transponder (there are 12 transponders in an INTELSAT IV satellite). The frame length of the TDMA System is 750 micro seconds, consisting of "sync bursts" and a number of traffic bearing bursts from each earth station, called "data bursts". The Multiple-Access is achieved by transmitting the data bursts in a preassigned time slot relative to the sync burst.

As compared to the MVD System, the TDMA System has the following characteristics:
(1) It compresses the digitized voice signals during transmission;

(2) There is only a single RF bandwidth (36 MHz) being used in a time-division or time-sharing manner; and

(3) The compression process eliminates the necessity of transmitting the mostly redundant bit streams of the silent period of voice conversation.

However, it is related to the MVD System in the following sense:

(1) The utilization of RF bandwidth (i.e., satellite bandwidth) is improved;

(2) The system interpolates in TDM manner all the digitized voice-signal bursts; each of which comes from an earth station; and

(3) It is also a resource saving system.

3.6 TTT SYSTEM

International Telecommunication Carrier of Japan, KDD, has developed a 50 Mbps PCM-TDMA System with Time Preassignment and TASI feature called TTT System [29,38]. This system has a traffic capacity of 700 voice-grade channels using 4-phase PSK modulation and coherent demodulation at 50 Mbps. Its characteristics are similar to
those of the TDMA System except that the PCM-TASI scheme (Section 3.3) may be applied so that its capacity can be doubled. The PCM-TASI scheme will perform the PCM encoding whenever the voice trunk is active. Also, the "bit reduction scheme" is used to resolve the voice clippings during overload condition.

A comparison of the MVD and TTI Systems can be related to those of PCM-TASI and TDMA Systems as described in Sections 3.3 and 3.5 of this chapter.

3.7 TELETEX SYSTEM

London (England) Financial Time reports that a new communication medium called Teletext will find its way into millions of home in 1980 [27]. This system can provide the home television viewer with up-to-date information in digital form. This information service rides piggy-back on the normal television signal. The user can select any kind of information by using a keyboard, and the desired information will be displayed on the screen. The new service will bring computer technology into homes. In the United States, Reuters uses different but related technology to provide service for Wall Street traders and brokers over local cable television networks [14].
The Teletext System has the following characteristics as compared to the MVD System:

(1) It uses piggy-back technique to transmit digital information on television video signal; and

(2) It is not a packet-oriented, blanking-period insertion scheme.

However, it is related to the MVD System in the following sense:

(1) It uses the existing television channels for additional digital information broadcasting service;

(2) It improves the channel utilization; and

3.6 PITCH SYNCHRONOUS DATA INTERPOLATION SCHEME

Bell Laboratory developed a scheme to interpolate data with continuous speech signals [43]. This scheme created a "synchronous pitch" into the continuous speech signal to allow steady flow of data or signaling information in the created gaps, without the need for pitch detection. Average "off-time" ratio of 30% was achieved for continuous speech without audible degradation.

As compared to the MVD System, this scheme has the following characteristics:
(1) The insertion of data was accomplished by creating synchronous pitches in voice signals;
(2) It did not take advantage of the silent period; and
(3) It did not need the detection of circuit activity.

However, it is related to the MVD System in the following sense:
(1) It provided additional service for data in the existing voice-grade circuits; and
(2) It improved the circuit utilization.

3.9 SUMMARY

This chapter provides an overview of the existing or previously developed systems that are related to the concept of the proposed MVD System. For each system, its main characteristics are described and its major differences to the proposed MVD System are emphasized. The reasons that these systems are related to the proposed MVD System are the use of interpolation concept, the avoidance of transmitting "silent signal", and the provision of additional data service in the existing analog system. However, none of the aforementioned systems is exactly the same as the proposed MVD System.
CHAPTER 4

ANALYSIS OF THE MIXED VOICE/DATA (MVD) SYSTEM

4.1 INTRODUCTION

The design of the proposed Mixed Voice/Data System involves optimum selection of several parameters. Among these parameters, two are the optimum packet size of the fixed-sized packets and the required buffer size of the microcomputer RAM so as to achieve maximum throughput and minimum packet delay.

The optimum selection of packet size involves a statistical analysis of the voice signal in the voice-grade circuits. The buffering requirement involves a study regarding the queueing phenomena of the data packets in the microcomputer. This chapter provides detail analysis regarding the above two aspects.

4.2 OBSERVATION OF TALKING AND SILENT PERIOD DISTRIBUTIONS

Because of the early introduction of TASI, there have been numerous studies on the on-off pattern of speech in the telephone conversation. Brady built an apparatus to convert speech to on-off status and to record the digital data on
magnetic tape for data processing [7]. The processing procedure included "throwing-away" those spurts that are smaller than 10 milliseconds, "filling-in" those gaps that are less than 200 msec, and appending a hangover time at the end of each talkspurt or talking period. Using this speech measuring technique, he measured 16 speakers engaging in eight conversations over the standard circuits in a test room atmosphere. The result of the measuring showed that the mean talkspurt was 1.34 seconds with the median talkspurt length of 0.77 seconds and the mean pause was 1.67 seconds with a median pause of 0.72 seconds. The percent of time the average conversant talked was 44.3%. Since the above measurement was performed in a test room atmosphere, the conversants knew they were being recorded. These data do not necessarily represent those in the real telephone conversations.

Norwine and Murphy observed talkspurts and pauses in oscillograph recordings of telephone conversations on a special circuit between New York and Chicago [37]. Their result showed a mean talkspurts of 4.3 seconds for 2845 talkspurts. Since they ignored 2811 pauses which were inside the talkspurts, these pauses had a mean of 0.73 seconds. If these pauses were taken into consideration, the new mean talkspurt length should become 1.8 seconds. These data were measured in a special land line circuit.
Some measurements were also made by Miedema on calls placed on the Atlantic cable via TASI [35]. The TASI speech detector had an operate (pick up) time of 5 msec and a deferred hangover time of 60 msec to 240 msec depending on the length of talkspurt. The sensitivity of the speech detector was -40 dbm (refer to 0 Transmission Level). The TASI data indicated an activity period of 48 percent averaged over all 600 commercial calls. The mean TASI talkspurt length was 1.3 seconds.

In this research, voice activities were recorded on double-channel pen recorder on international satellite circuits to/from Taiwan. Three destinations were chosen for this observation, namely, USA, Europe and Japan. Silent periods that were shorter than 0.1 seconds were ignored because these periods will not be longer enough for the data packet insertion. Small talkspurts shorter than 0.1 seconds were ignored in the survey but they might constitute to be impulse noise of the NVD channels if they are undetectable by the voice detectors. Thirty-minute voice activity records were sampled for each busy channel. These records were sampled again and converted to digital forms. After processing by a digital computer, the talking-and silent-period density functions and their distribution functions were plotted in Fig. 13 to 16. Table 2 tabulates the result of these characteristics. Note here that the mean
Fig. 13. Probability Density Function of Talking Period
Fig. 14. Probability Density Function of Silent Period.
Fig. 15. Probability Distribution of Talking Period.
Fig. 16. Probability Distribution of Silent Period.
values deviate slightly from one to another for different destination. This phenomenon can be expected because of the inherent language pattern of each destination. For example, considerably shorter talking periods were observed in the circuits to/from Japan. These results show that the mean talking period to/from Europe (1.2587 seconds) is very close to TASI talkspurt (1.3 seconds) and that the mean silent period (1.56 seconds) is also very close to Brady's mean pause (1.67 seconds). The purpose of this observation is to see the talking- and silent-period distribution in real satellite circuits.

<table>
<thead>
<tr>
<th>Items</th>
<th>Percentage of Silent Period</th>
<th>Mean Silent Period (sec.)</th>
<th>Mean Talking Period (sec.)</th>
<th>Number of Sampled Observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>J.S.A.</td>
<td>57.45</td>
<td>1.5797</td>
<td>1.1698</td>
<td>639</td>
</tr>
<tr>
<td>Europe</td>
<td>55.72</td>
<td>1.584</td>
<td>1.2587</td>
<td>460</td>
</tr>
<tr>
<td>Japan</td>
<td>67.95</td>
<td>1.576</td>
<td>0.7441</td>
<td>421</td>
</tr>
</tbody>
</table>

Table 2. Statistics of Sampled Observation on International Telephone Circuits to/from Taipei, Taiwan, ROC

The above density functions are close to Erlang $k$ distribution with $k = 1.0$. In order to simplify the mathematical analysis, $k = 1$ is assumed, which is an exponentially distributed function and it represents the worst case assumption.
Table 3 shows the mean, variance and standard deviation of talking and silent periods for mixed destinations using more samples. The percentages of talking and silent periods then become 38.8% and 61.2%, respectively.

<table>
<thead>
<tr>
<th></th>
<th>Talking Period</th>
<th>Silent Period</th>
<th>Full Period</th>
<th>Number of Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>1.26 sec.</td>
<td>1.99 sec.</td>
<td>3.25 sec.</td>
<td></td>
</tr>
<tr>
<td>Variance</td>
<td>1.75</td>
<td>9.72</td>
<td>11.48</td>
<td></td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>1.33 sec.</td>
<td>3.12 sec.</td>
<td>3.39 sec.</td>
<td></td>
</tr>
<tr>
<td>Percentage</td>
<td>38.8%</td>
<td>61.2%</td>
<td>100%</td>
<td>2587</td>
</tr>
</tbody>
</table>

Table 3. Talking- and Silent-Period statistics for Mixed Destinations.

4.3 PROBABILISTIC APPROACH TO THE INTERFERING MECHANISM

Following assumptions are made to find out the effect of voice interference to the packet throughput:

1. The probability density function (PDF) of the silent period is $p(t) = a e^{-at}$, i.e. an exponentially distributed silent period with a mean silent period of $m_s = 1/a$;

2. The probability density function of the talking period
is \( q(t) = e^{-bt} \), i.e. an exponentially distributed talking period with a mean talking period of \( m_t = \frac{1}{\alpha} \); (3) \( T_s \) and \( T_t \) are random numbers of silent and talking periods, respectively, representing the durations of these periods, and they are mutually independent.

Then, the probability density function \( g(t) \) of inter-arrival time of voice signals may be found by convolving these two density functions, \( p(t) \) and \( q(t) \), i.e.

\[
f(t) = p(t) \ast q(t)
\]

\[
= a e^{-at} \ast b e^{-bt}
\]

\[
= \int_{0}^{t} a e^{-a \tau} b e^{-b(t-\tau)} d\tau
\]

\[
= \frac{a b}{b - a} (e^{-at} - e^{-bt})
\]

This probability density function can be used to generate the voice arrival transaction in the simulation model; however, it would be too complicated for the analytic model. Therefore, in the analytic model, a "worst case" Poisson arrival pattern is assumed, in which case the PDF was assumed to have

\[
g(t) = c e^{-ct}
\]

where \( 1/c = 1/a + 1/b \). This assumption is reasonable since the mean of \( g(t) \) is

\[
m_{p \ast q} = \int_{0}^{t} \frac{ab}{b-a} (t e^{-at} - t e^{-bt}) dt = \frac{1}{a} + \frac{1}{b}.
\]
4.3.1 Average Service Time of Data Packet

Fig. 17 shows the interfering situation of a data packet being transmitted on the line at time \( t \) from the beginning of a silent period. Part (a) shows that this packet is successfully transmitted and part (b) shows that the packet is interfered with the arrival of a voice signal before the packet has finished its transmission. Part (c) is the PDF of the silent period.

Let \( z \) be the packet size in bits,
\( s \) be the data rate in bps transmitted during the silent period on the voice line,
\( t_0 = z/s \) be the time in seconds required to transmit a packet on the voice line, and
\( S \) be the event that this packet is successfully transmitted without being interfered with by the arrival of voice signals.

Then,
\[
P(S) = P(T_B \geq t+t_0 \mid T_S \geq t)
\]
\[
= \frac{P(T_B \geq t+t_0 \cap T_S \geq t)}{P(T_S \geq t)} = \frac{P(T_B \geq t+t_0)}{P(T_S \geq t)}
\]
\[
= \frac{e^{-a(t+t_0)}}{e^{-at}} = e^{-at_0}
\]

This probability does not depend on \( t \) but only on packet size \( t_0 \). This is reasonable since the silent period
PROBABILISTIC APPROACH OF INTERFERING MECHANISM

(a) A Successfully Transmitted Packet

(b) An Interfered Packet

(c) Probability Density Function of Silent Period

\[ p(t) = ae^{-at} \]

Fig. 17. Probabilistic Approach of Interfering Mechanism.
is assumed to be exponentially distributed, i.e. the arrival or ending of a voice signal is independent of the arrival or ending of the next voice signal. This means that these events occur independently and that the system is memoryless.

The probability of a packet being interfered by the voice arrival is thus

\[ P(\overline{S}) = 1 - e^{-at_0} \]

This result can also be obtained from the following calculation:

\[
P(\overline{S}) = P( t \leq T_s \leq t+t_0 | T_s \geq t) = \frac{P( t \leq T_s \leq t+t_0 \cap T_s \geq t)}{P( T_s \geq t)} = \frac{\int_t^{t+t_0} a e^{-at'} dt'}{e^{-at}} = \frac{e^{-at} - e^{-a(t+t_0)}}{e^{-at}} = 1 - e^{-at_0}
\]

Let \( R_n \) be the probability that a packet needs retransmission for \( n \) times in order to get through the busy voice line, then

\[
R_0 = e^{-at_0} \\
R_1 = (1 - e^{-at_0}) e^{-at_0} \\
\vdots \\
\vdots \\
R_n = (1 - e^{-at_0})^n e^{-at_0}
\] (3)
Fig. 18 is a plot of these probabilities. Note that
\( R_n \to 0 \), as \( n \to \infty \).

The mean service time for a packet including retransmission can be calculated by using the above probability density function \( R_n \), assuming that the interfered packet has wasted \( t_0/2 \) on the average in transmission before it is interfered with.

\[
\frac{1}{\mu_d} = t_0 R_0 + (t_0 + t_0/2) R_1 + \ldots + (t_0 + n t_0/2) R_n + \ldots
\]

\[
\mu_d = \frac{t_0(1 + e^{-a t_0})/2}{2 e^{-a t_0}}
\]

In later derivation for the preemptive-repeat model, this \( \mu_d \) will be used as the mean service rate of data packets for an exponentially distributed service time.

4.3.2 Expected Throughput of Data Packets

Let \( N \) be the random number expressing the number of packets that can be transmitted successfully during \( T_s \). Then,

\[
E(N) = E \left[ \frac{T_s}{t_0} \right]
\]

where \( \left\lfloor \frac{T_s}{t_0} \right\rfloor \) denotes the maximum integer smaller than \( T_s/t_0 \). Therefore,

\[
E(N) = 0. P(0 \leq T_s \leq t_0) + 1. P(t_0 \leq T_s \leq 2t_0)
+ 2. P(2t_0 \leq T_s \leq 3t_0) + \ldots + n. P(nt_0 \leq T_s \leq (n+1)t_0) + \ldots
\]
Fig. 18. Probability of Requiring N-time Retransmission.
\[ \sum_{n=1}^{\infty} n \int_{a_{nto}}^{(n+1)a_{nto}} e^{-at} \, dt = \sum_{j=1}^{\infty} e^{-jat_0} \]

which is a geometric series. With \(0 \leq e^{-at_0} < 1\), \(\lim_{n \to \infty} (e^{-at_0})^n = 0\),

\[ E(N) = \frac{e^{-at_0}}{1 - e^{-at_0}} \] (5)

Fig. 19(a) is a plot of \(E(N)\) vs. \(t_0\). Note that the larger the \(t_0\), the smaller the number of data packets transmitted per silent period, and vice versa. This is due to the interference effect of the unexpected arrival of voice signals.

Equation (5) shows the expected number of packets transmitted through the voice active line during a silent period. This quantity is less than \(E(T_s)/t_0\), since the latter includes the portion of time during which the data packet is actually transmitted, but due to the arrival of a voice signal this period was wasted because it does not contributed to the packet throughput and a retransmission for this interfered packet is necessary. This contributes to overhead of the system. Therefore,

\[ t_{ovhd} = E(T_s) - t_0E(N) = 1/a - \frac{t_0}{e^{at_0} - 1} \] (6)

is the average portion of time spent in transmitting the last incompletely transmitted packet per average silent period, and

\[ \rho_{ovhd} = \frac{t_{ovhd}}{1/c} = \frac{c/\rho}{e^{at_0} - 1} \] (6a)
Fig. 19. (a) The Expected Number of Packets Transmitted per Silent Period and (b) The Percentage Overhead versus the Packet Transmitting Time.
is the utilization overhead by the data packet per average inter-arrival time of voice signals. Fig. 19(b) is a plot of percentage overhead of the silent period versus $t_o$.

The maximum throughput of data packets is then

\[
\lambda_{d_{\text{max}}} = \frac{\lambda}{1/c} = \frac{c}{e^{\lambda t_o} - 1} \quad \text{(packets/s)}
\]

\[
= \frac{cz}{e^{\lambda t_o} - 1} \quad \text{(bits/s)} \quad (7)
\]

To obtain a steady-state system the data arrival rate $\lambda_d$ should be less than or equal to $\lambda_{d_{\text{max}}}$; otherwise, the accumulation of data packets will cause the system queue to grow as time goes on. This will make the system operation unacceptable. Note here that

\[
\rho_v + \rho_{d_{\text{max}}} + \rho_{\text{ovhd}} = 1 \quad (8)
\]

where $\rho_v$ = utilization by voice signals

\[
= \frac{m_t}{m_t + \lambda_b} = \frac{1/b}{1/c} = c/b
\]

$\rho_{d_{\text{max}}}$ = maximum utilization by data packets

\[
= \frac{\lambda_{d_{\text{max}}}}{\mu_2} = \frac{c.E(N)}{E(N)} = \frac{ct_o}{e^{\lambda t_o} - 1}
\]

where $\lambda_{d_{\text{max}}}$ = maximum mean arrival rate of data packets

$\mu_2$ = mean service rate of data packets during the silent period without considering
interference effect

If \( k \) MVDO channels are used in parallel, and if all channels have identical characteristics, it is expected that the maximum achievable throughput will be at least \( k \) times greater than that of a single channel system. Thus,

\[
\text{dmax} = \frac{kc}{e^{at_0} - 1} \quad \text{(packets/s)} \quad (9a)
\]

\[
\text{e} = \frac{kc}{e^{at_0} - 1} \quad \text{(bits/s)} \quad (9b)
\]

4.4 QUEUEING MODEL OF A SINGLE SERVER WITH A PREEMPTIVE PRIORITY CLASS OF ARRIVAL

Gopinathan, Matra and Sondhi have done some work on queues in burst processes [18,45]. Their model considered a source feeding data to a buffer at a uniform rate. The buffer's access to a channel with maximum fixed rate of transmission is controlled by a switch: the buffer is only able to discharge when the switch is closed ("on"). Although this model can be applied to the "multiplexing of data with speech on a telephone channel", the essential schemes of the proposed MVDO System have not been treated. For examples the model is not packet-oriented, nor does it treat the service preemption and retransmission of data packets. Besides, the data input to the buffer is not
necessarily uniform. Hsu also had done some work on buffer behavior for a discrete system under geometrical input/output assumptions [21]. However, his work did not consider the retransmission and preemption characteristics that the MVD System provides. It is worthwhile to perform the analysis based on the MVD System model and its characteristics.

In order to derive various performance measures for the MVD System, such as average queue length, average waiting time of a packet, etc., a single server system is used to provide real-time transmission service for voice signals (preemptive priority class of arrival), and the data packet transmission service with lower priority. Fig. 20 shows such a queueing model. The assumptions for this model are as follows:

1. Talking- and silent-period distributions of voice signals are assumed to be exponential with PDF $q(t) = a e^{-at}$, and $p(t) = a e^{-at}$, respectively;

2. The data arrival pattern is Poisson with mean arrival rate $\lambda_2$, where $\lambda_2 \leq \lambda_d_{max}$;

3. No queueing is permitted for voice signals because of its real-time nature (i.e. as soon as a voice signal arrives, it preempts the use of server for its immediate transmission);

4. The data packet which is being served for its
Fig. 20. Queueing Model for a Single Server with one Preemptive Priority Class of Arrivals.
transmission is preempted by the arrival of the voice signal, and will hold temporarily in front of the data-packet queue station until it is re-served (the whole packet is retransmitted again--preemptive repeat).

Therefore, the service time of a packet should include the retransmission time, as derived in (4a), and exponentially distributed service time with mean service time of $1/\mu_2$ should provide a "worst case" assumption for the output transmission service pattern.

Let $p_{mnr}(t) = \Pr\{m$ units of priority 1 class (voice signals) in the system, $n$ units of priority 2 class (data packets) in the system, and type $r$ customer, $r = 1$ (voice) or 2 (data) being served at time $t\}$

where the priority 1 is higher than the priority 2. Due to the intrinsic characteristics of the MVD System, $m$ can only equal 1 or 0, i.e. the voice signal is either present or absent. Also, due to the preemptive assumption, if $m = 1$, $r$ must equal 1; and if $m = 0$, $r$ must equal 2 if $n > 0$ and 0 if $n = 0$. Therefore, the third subscript $r$ can be dropped and only $p_{1n}(t)$ or $p_{0n}(t)$ should be considered.
Let $E_{ln}$ and $E_{0n}$ represent the states in which voice is present and absent, respectively, and there are $n$ units of data packets in the system. Fig. 21 shows the state transition diagram of the neighboring states. The differential-difference equations for the state transition are set up as follow [19]:

\[
\begin{align*}
 p_{1n}(t+\Delta t) &= p_{1n}(t)(1-b\Delta t)(1-\lambda_2\Delta t) + p_{0n}(t)\Delta t + p_{1n-1}(t)\lambda_2\Delta t \\
 p_{0n}(t+\Delta t) &= p_{0n}(t)(1-a\Delta t)(1-\lambda_2\Delta t)(1-\mu_2\Delta t) + p_{1n}(t)b\Delta t + p_{0n+1}(t)\mu_2\Delta t \\
 p_{10}(t+\Delta t) &= p_{10}(t)(1-b\Delta t)(1-\lambda_2\Delta t) + p_{00}(t)a\Delta t \\
 p_{00}(t+\Delta t) &= p_{00}(t)(1-a\Delta t)(1-\lambda_2\Delta t) + p_{10}(t)b\Delta t + p_{01}(t)\mu_2\Delta t
\end{align*}
\]

Let $\Delta t \to 0$, \( \lim_{\Delta t \to 0} \frac{P_{mn}(t+\Delta t)-P_{mn}(t)}{\Delta t} = \frac{\partial P_{mn}(t)}{\partial t}, (m = 0, \text{ or } 1) \)

The above differential difference equations become

\[
\begin{align*}
 \frac{\partial p_{1n}(t)}{\partial t} &= -(b+\lambda_2)p_{1n}(t) + a p_{0n}(t) + \lambda_2 p_{1n-1}(t) \\
 \frac{\partial p_{0n}(t)}{\partial t} &= -(a+\lambda_2+\mu_2)p_{0n}(t) + b p_{1n}(t) + \lambda_2 p_{0n-1}(t) + \mu_2 p_{0n+1}(t) \\
 \frac{\partial p_{10}(t)}{\partial t} &= -(b+\lambda_2)p_{10}(t) + a p_{00}(t) \\
 \frac{\partial p_{00}(t)}{\partial t} &= -(a+\lambda_2)p_{00}(t) + b p_{10}(t) + \mu_2 p_{01}(t)
\end{align*}
\]

Note that in Equations (10a) and (11a), the second term on the right hand side expressing the arrival of a voice signal in $\Delta t$ causes the transition of states from $E_{0n}$ to $E_{ln}$. This implies that the preemption takes place after the arrival of the voice signal. Note also that the transition of states.
Fig. 21. State Transition Diagram for a Single Channel MVD System.
from $E_{1,n+1}$ to $E_{ln}$ can never occur since the presence of voice prohibits the service of data packets; however, the transition of states from $E_{1,n+1}$ to $E_{0,n+1}$ does occur upon completion of voice-signal transmission.

To obtain a steady-state system and to allow a finite number of data packets to be queued-up in the system, let

$$\frac{\partial p_{mn}(t)}{\partial t} = 0,$$

and $p_{mn}(t) = p_{mn}$. Therefore,

$$(b + \lambda_2) p_{1n} = a p_{0n} + \lambda_2 p_{1,n-1} \quad (n \geq 1) \quad (12a)$$

$$(a + \lambda_2 + \mu_2) p_{0n} = b p_{1n} + \lambda_2 P_{0,n-1} + \mu_2 p_{0,n+1} \quad (n \geq 1) \quad (12b)$$

$$(b + \lambda_2) p_{10} = a p_{00} \quad (12c)$$

$$(a + \lambda_2) p_{00} = b p_{10} + \mu_2 p_{01} \quad (12c)$$

Note that $p_{1n}$ and $p_{0n}$ can be solved from (12a) and (12b) by an iteration or operator method subject to the following boundary conditions:

$$p_{10} = \frac{a}{b + \lambda_2} p_{00} \quad (13a)$$

$$p_{01} = \frac{\lambda_2 + b \lambda_2 + a \lambda_2}{\mu_2 (b + \lambda_2)} p_{00} \quad (13b)$$

and,

$$\sum_{n=0}^{\infty} p_{0n} + \sum_{n=0}^{\infty} p_{1n} = 1 \quad (13c)$$

Then,

$$L_2 = E[\ell_2] = \sum_{n=0}^{\infty} n (p_{0n} + p_{1n}) \quad (14)$$
However, L2 can also be found by using the generating function method [19]. Let

\[ P_1(z) = \sum_{n=0}^{\infty} p_{1n} z^n \]  
(15a)

\[ P_0(z) = \sum_{n=0}^{\infty} p_{0n} z^n \]  
(15b)

Multiplying appropriate power of \( z \) to each term of (12a) and (12b), taking summation from \( n = 1 \) to \( \infty \) and combining (12c) and (12d), the following equations result:

\[ P_1(z) = \frac{a}{b + \lambda_2 - \lambda_2 z} P_0(z) \]  
(16a)

\[ P_0(z) = \frac{(\mu_2 z - \mu_2) p_{00}}{(a + \lambda_2 + \mu_2) z - \mu_2 - \lambda_2 z^2 - \frac{b a z}{b + \lambda_2 - \lambda_2 z}} \]  
(16b)

Note that

\[ P_1(1) = \frac{a}{b} P_0(1) \]  
(17a)

\[ P_0(1) = \frac{b \mu_2 p_{00}}{b \mu_2 - a \lambda_2 - b \lambda_2} \]  
(17b)

Also,

\[ P_1(1) = \sum_{n=0}^{\infty} p_{1n} = \frac{a}{b} = \rho_v \]

\[ P_0(1) = \sum_{n=0}^{\infty} p_{0n} = p_{00} + \sum_{n=1}^{\infty} p_{0n} = p_{00} + \rho_d + \rho_{ovhd} \]
From equation (13c),
\[ P_0(1) + P_1(1) = 1 \]  \hspace{1cm} (18)

Substituting (17a) and (17b) into (18),
\[ p_{00} = \frac{b \mu_2 - a \lambda_2 - b \lambda_2}{b \mu_2 + a \mu_2} = \frac{1 - \rho_v \rho_d - \rho_d}{1 + \rho_v} \]  \hspace{1cm} (19)

where,
\[ \rho_v = \frac{a}{b} \]
\[ \rho_d = \frac{\lambda_2}{\mu_2} = \frac{\lambda_2 t_0 (1 + e^{at_0})}{2} \]

Substituting (19) into (17a) and (17b),
\[ p_0(1) = \sum_{n=0}^{\infty} p_{0n} = \frac{b}{b + a} = \frac{1}{1 + \rho_v} \]  \hspace{1cm} (20a)
\[ p_1(1) = \sum_{n=0}^{\infty} p_{1n} = \frac{a}{b} p_0(1) = \frac{\rho_v}{1 + \rho_v} \]  \hspace{1cm} (20b)

\[ L_1 = E[l_1] = \text{Expected number of voice signals in the transmitting system} \]
\[ = 0. \sum_{n=0}^{\infty} p_{0n} + 1. \sum_{n=0}^{\infty} p_{1n} = P_1(1) = \frac{\rho_v}{1 + \rho_v} \]  \hspace{1cm} (21a)
\[ \text{Var}[l_1] = E[l_1^2] - E[l_1]^2 = P_1(l) - P_1^2(l) = \frac{\rho_v}{(1 + \rho_v)^2} \]  \hspace{1cm} (21b)
These results are proved to be independent of the existence of data packets. This is quite understandable because the higher priority voice signal always plucks the use of channel facility from the lower priority data packets. Equations (20a) and (20b) are proved to be identical to those of the M/M/1/I queueing system, which provides a service capacity of 1 [19]. This is realistic since the arrival of voice signals is orderly, i.e. the next voice signal will not arrive until the one in the system has completed its service and a silent period has elapsed.

As for data packets, the expected number of data packets in the system, the expected number of data packets in queue, the mean time for a data packet to get through the system and the mean waiting time for a data packet to get into service can be derived as follows [48]:

\[ L_2 = E[l_2] = \text{Expected number of data packets in the transmitting system} \]

\[ L_2 = \sum_{n=0}^{\infty} \frac{n \cdot (p_0^n + p_1^n)}{\frac{dP_0(z)}{dz} \bigg|_{z=1} + \frac{dP_1(z)}{dz} \bigg|_{z=1}} \]

\[ \rho_d \left(1 + 2\rho_v^2 + \rho_v \frac{\mu_2}{b} \right) \]

\[ \frac{\rho_2 (1 + \rho_d \rho_v - \rho_d - \rho_v)}{(1 + \rho_v) (1 - \rho_v \rho_d - \rho_d)} \]

\[ (1 - \rho_v \rho_d - \rho_d) > 0 \]

\[ (22a) \]

\[ \frac{\lambda_2}{b + a} \cdot \frac{(b + a)^2 + a \mu_2}{b \mu_2 - (b + a) \lambda_2} \]

\[ (\rho_d < \frac{b}{b + a}) (22b) \]
\[ W_2 = \text{Mean time for a data packet to get through the transmitting system} \]

\[
L_2 = \left( 1 + 2 \frac{P_v}{\lambda_2} + \frac{P_v^2}{\lambda_2} + \frac{P_v \lambda_2}{\lambda_2} \right) \left( 1 + \frac{P_v}{(1 - P_v P_d - P_d)} \right), \quad (1 - P_v P_d - P_d > 0) \tag{23}
\]

\[
L_2 = \frac{1}{b+a} \left( \frac{(b+a)^2 + a \lambda_2}{b \lambda_2 - (b+a) \lambda_2} \right) \quad (P_d < \frac{b}{b + a}) \tag{24}
\]

where \( \lambda_2 \) is derived using Little's Formula.

If \( a = 0.5 \) and \( b = 3.5 \), Equations (22) to (24) can be simplified to

\[
L_2 = \frac{P_d + 0.295 \lambda_2}{0.615 - P_d} \quad (P_d < 0.615) \tag{25}
\]

\[
L_2 = \frac{1/\lambda_2 + 0.295}{0.615/\lambda_2 - 1/\lambda_2} \quad (\lambda_2 < 0.615 \lambda_2) \tag{26}
\]

\[
W_2 = \frac{(P_d - 0.295 \lambda_2)}{0.615 - P_d} \lambda_2 \quad (P_d < 0.615) \tag{27}
\]

\[
W_2 = \frac{1/\lambda_2 + 0.295}{0.615 - \lambda_2(1/\lambda_2)} \quad (\lambda_2 < 0.615 \lambda_2) \tag{28}
\]

Fig. 22 is a plot of \( L_2 \) against \( P_d \) using (25) and Fig. 23 is against \( 1/\lambda_2 \) using (26), with \( \lambda_2 \) as a
Fig. 22. The Expected Number of Data Packets in Transmitting System ($L_2$) versus the Traffic Intensity ($\rho_d$) with Mean Arrival Rate ($\lambda_2$) of Data Packets as a Parameter (Preemptive-Repeat).
Fig. 23. The Expected Number of Data Packets in the Transmitting System versus Mean Service Time of Data Packets ($1/\mu_2$) with Mean Arrival Rate ($\lambda_2$) as a Parameter (Preemptive-Repeat).
Fig. 24. The Expected Total Resident Time of Data Packets in the Transmitting System versus Traffic Intensity of Data Packets with Mean Arrival Rate of Data Packets as a Parameter (Preemptive-Repeat Case).
Fig. 25. The Expected Total Resident Time of Data Packets in the Transmitting System versus Mean Service Time of Data Packets with Mean Arrival Rate as a Parameter (Preemptive-Repeat Case).
parameter. Similarly Fig. 24 is a plot of \( w_2 \) against \( \rho_d \) using (27) and Fig. 25 is against \( \lambda w_2 \) using (28), with \( \lambda_2 \) as a parameter.

In Figs. 22 and 23, \( L_2 \) and \( W_2 \) increase as \( \rho_d \) increases for any value of \( \lambda_2 \). For constant \( \rho_d \), \( L_2 \) increases with increasing value of \( \lambda_2 \); however, \( W_2 \) decreases as \( \lambda_2 \) increases. This can be seen from (27) because the effect of increasing \( \lambda_2 \) results in more deduction in the numerator and more multiplying factor in the denominator. In Figs. 24 and 25, \( L_2 \) and \( N_2 \) are plotted against \( \lambda w_2 \), mean service time of data packets, which is independent of \( \lambda_2 \), the data-packet mean arrival rate.

In (22a), if the arrival of voice signals is temporarily blocked (i.e. let \( \rho_v = 0 \)), then

\[
L_{02} = \frac{\rho_d}{(1 - \rho_d)}
\]

which is identical to that for the \( M/M/1/\infty \) queueing system, i.e. a single server with infinite service capacity, and exponential inter-arrival time and service time.

4.5 PREEMPTIVE-RESUME MODEL WITH CONSTANT SERVICE TIME FOR DATA PACKETS
If the service discipline of data packets is preemptive-resume, \( M/D/1 \) model can be assumed for data packets; that is, instead of retransmitting the whole packet only the residual portion of the data packet is transmitted. This residual portion was unable to finish its transmission service at the time of voice arrival in the corresponding channel. Following assumptions are made:

1. \( M/M/1 \) model for voice transaction, i.e. the PDF of the silent period is \( e^{-at} \), \( t \geq 0 \), and that of the talking period is \( e^{-bt} \), \( t \geq 0 \);

2. \( M/D/1 \) model for data packets, i.e. the PDF of inter-arrival time of data packets is \( \lambda_2 e^{-\lambda_2 t} \), \( t \geq 0 \); and that of service time is \( \delta(t - \frac{1}{\mu_2}) \), that is, the service time of data packets is deterministic;

3. The service discipline is preemptive-resume with constant service time \( 1/\mu_2 \) for data packets;

4. The system has infinite capacity and has a single service channel.

Fig. 20 shows the interruptions due to the arrivals of the voice transactions when a data packet begins its service and up to the completion of its service. In this figure, all random variables \( M, S_i's \) and \( V_i's \) have the following meaning:

\( M = \) the random variable denotes the number of
Fig. 26. Service Interruptions of Data Packets Due to the Arrival of Voice Signals for Preemptive-Resume Case.
interruptions due to the arrival of voice transaction;

$S_i$ = the random variable denotes the $i$th service time of the data packet;

$V_i$ = the random variable denotes the service time of $i$th arrival of voice transaction after the data packet entering the service. Let

$S = S_1 + S_2 + \ldots + S_M + S_{M+1}$

= total service time of a data packet before leaving transmitting system.

Since a data packet must take $1/\mu_2$ to complete its service, the random variable $S$ is a constant, and is equal to $1/\mu_2$.

Similarly, let

$V = \sum_{i=1}^{M} V_i$

where all random variables $V_i$'s are independently and identically distributed with exponential distribution $\rho e^{-bt}$, $t\geq 0$, and independent of the random variable $M$.

Random variable $M$ denotes the number of interruptions due to the arrivals of voice transactions in the total service time $1/\mu_2$ of the data packet. Therefore, the probability mass function of random variable $M$ is

$Pr[M=m | (t=1/\mu_2)] = Pr\left[\text{there are n arrival of voice transactions during time } \frac{1}{\mu_2}\right]$
\[ e^{-a/\mu_2} \left( a/\mu_2 \right)^m \]

Therefore,
\[ E[M] = \frac{a}{\mu_2}, \quad \text{and} \quad \text{Var}[M] = \frac{a}{\mu_2}. \]  

(29)

with the above results, the mean and variance of random variable \( V \) can be calculated.
\[ E[V] = E[E(V|M)] = E[M E[V_1]] = E[M] E[V_1] \]
\[ = \frac{a}{\mu_2} \cdot \frac{1}{b} = \frac{a}{b\mu_2} \]  

(30)

\[ \text{Var}(V) = E[\text{Var}(V|M)] + \text{Var}[E(V|M)] \]
\[ = E[M\text{Var}(V_1)] + \text{Var}[M E[V_1]] \]
\[ = E[M] \text{Var}(V_1) + \text{Var}[M] \cdot E^2[V_1] \]
\[ = \frac{a}{\mu_2} \cdot \frac{1}{b^2} + \frac{a}{\mu_2} \cdot \frac{1}{b^2} = \frac{2a}{b^2\mu_2} \]  

(31)

Therefore, the mean and variance of random variable \( T \) that denotes the complete service time of the data packet is given by
\[ = E\left[\frac{1}{\mu_2}\right] + E[V] = \frac{1}{\mu_2} + \frac{a}{b\mu_2} \]  

(32)

\[ \text{Var}[T] = \text{Var}[S+V] = \text{Var}\left[\frac{1}{\mu_2} + V\right] = \text{Var}[V] = \frac{2a}{b^2\mu_2} \]  

(33)

Thus the second moment of random variable \( T \) is given by
\[ E[T^2] = E^2[T] + \text{Var}[T] = \frac{1}{\mu_2^2} (1 + \frac{a}{b})^2 + \frac{2a}{b^2\mu_2} \]
Thus, the $M/D/1$ model for data packets has been changed to the $M/G/1$ model* with complete service time $T$ having the mean and second moment shown above.

If there was only one class of transaction in the transmitting system, the expected steady-state system size $E[N]$ at any arbitrary point in time is given by the well known Pollaczek-Khintchine (P-K) formula in queueing theory**

$$E[N] = \rho + \frac{\rho^2 + \lambda^2 \sigma_s^2}{2(1 - \rho)} \quad (34)$$

In this case

$$\rho = \lambda E[T] \quad (35)$$
$$\sigma_s^2 = \text{var}[T]$$

therefore,

$$E[N] = \lambda E[T] + \frac{\lambda^2 E[T]^2}{2\{1 - \lambda E[T]\}} \quad (36)$$

By Little's formula, the total time in the transmitting system (system resident time, including waiting and service) is given by

$$W = \frac{1}{\lambda} E[N] = E[T] + \frac{\lambda E[T^2]}{2\{1 - \lambda E[T]\}} \quad (37)$$

* Pages 223-259 of Reference (19)

** Page 225, Ibid
The second term in the right hand side is the waiting time of a transaction before it gets into the service, i.e.

\[ W_q = \frac{\lambda E[T^2]}{2\{1 - \lambda E[T]\}} \]  

(38)

But there are two classes of transactions being served in the transmitting system and the data transaction must be served after any voice transaction has been completely served. That is, a data transaction must wait in queue on the average for a longer period during which any in-system voice transaction must be served first. Therefore, the expected waiting time of a data transaction in queue is given by

\[ W_{q2} = \frac{\lambda_2 E[T^2]}{2\{1 - \lambda_2 E[T]\}} + W_1 \]  

(39)

where the subscript 1 denotes the voice transaction and 2 denotes the data transaction, and from (21a) and Little's formula,

\[ M_1 = \text{the expected in-system time of a voice transaction} \]

\[ M_1 = \frac{1}{1 + \rho} = \frac{1}{\rho_v} = \frac{a/b}{b(1 + a/b)} = \frac{a}{b(a + b)} \]  

(40)

Therefore, four measures of system performance can be obtained as follows:

\[ W_{q2} = \frac{ab \mu_2^2 + \frac{1}{2} \lambda_2 (a + b)^3}{b(a + b)(b \mu_2 - a \lambda_2 - b \lambda_2) \mu_2} \]  

(41)
Lq2 = Expected number of data packets in queue
\[ Lq2 = \lambda_2 \frac{ab\mu_2^2 + \frac{1}{2}\lambda_2(a + b)^3}{a + b} \frac{b\mu_2(b\mu_2 - a\lambda_2 - b\lambda_2)}{b\mu_2} \] (42)

W2 = Expected in-system time of a data packet
\[ W2 = W_{q2} + \mathbb{E}[T] = \frac{1}{a+b} \cdot \frac{ab\mu_2^2 + (a + b)^2[b\mu_2 - \frac{1}{2}\lambda_2(a + b)]}{b\mu_2(b\mu_2 - a\lambda_2 - b\lambda_2)} \] (43)

L2 = Expected number of data packets in the system
\[ L2 = \lambda_2 W_2 = \frac{\lambda_2}{a+b} \cdot \frac{ab\mu_2^2 + (a + b)^2[b\mu_2 - \frac{1}{2}\lambda_2(a + b)]}{b\mu_2(b\mu_2 - a\lambda_2 - b\lambda_2)} \] (44)

For a steady-state system to exist, L2 must be finite, that is
\[ b\mu_2 - a\lambda_2 - b\lambda_2 > 0 \] (45)

This implies
\[ \rho_2 = \frac{\lambda_2}{\mu_2} < \frac{b}{a + b} \] (46)

Figs. 27 and 28 show the expected number of data packets in the system versus data traffic intensity and
Fig. 27. The Expected Number of Data Packets in the Transmitting System versus Traffic Intensity of Data Packets with Mean Arrival Rate as a Parameter (Preemptive-Resume).
Fig. 28. The Expected Number of Data Packets in the Transmitting System versus Mean Service Time of Data Packets with Mean Arrival Rate of Data Packet as a Parameter (Preemptive-Resume).
average service time of packets, respectively, with data arrival rate $\lambda_2$ as a parameter. Figs. 29 and 30 show the expected total resident time in the transmitting system versus data traffic intensity and average service time of packets, respectively, with data arrival rate $\lambda_2$ as a parameter. In these figures, $a$ and $b$ are chosen to be 0.5 and 0.8, respectively.

4.6 OPTIMUM PACKET SIZE, MINIMUM OVERHEAD, AND MAXIMUM THROUGHPUT

For easy reference, Equations (5), (6) and (7) are written herein:

$$E(N) = \frac{e^{-a t_0}}{1 - e^{-a t_0}}$$ (5)

$$t_{ovhd} = \frac{1}{a} - \frac{t_0}{e^{a t_0} - 1}$$ (6)

$$\lambda_{d_{max}} = \frac{c z}{e^{a t_0} - 1}$$ (7)

Assuming that a packet consists of $z_0$ bits of header and $z_1$ bits of information, i.e. $z = z_0 + z_1$, a condition is to be sought to transmit maximum information through the system. Let

$$I = \text{Total information transmitted in bps}$$

$$I = \frac{c z_1}{e^{a t_0} - 1}$$ (47)

where $t_0 = (z_0 + z_1)/s$, and

$s = \text{speed of the voice-grade line.}$
Fig. 29. The Expected Total Resident Time of Data Packets in the Transmitting System versus Data Traffic Intensity with Mean Arrival Rate of Data Packets as a Parameter (Preemptive-Resume Case).
Fig. 30. The Expected Total Resident Time of Data Packets in the Transmitting System versus Mean Service Time of Data Packets with Mean Arrival Rate of Data Packets as a Parameter (Preemptive-Resume Case).
To maximize $I$ with respect to $z_j$, a relation can be found from $dI/dz_j = 0$. This gives

$$1 - \frac{z_j a}{s} = e^{-a t_o}$$

(48)

where $t_o = (z_0 + \frac{z_j}{s})/s$.

Therefore, $z_j$, the optimum packet size for information part, can be solved graphically as in Fig. 31 for given values of channel length $z_0$ and line speed $s$. For $z_0 = 80$ bits and $s = 4800$ bps, $z_j = 0.1383$ s/a = 1049 bits. Then,

$$I_{max} = \frac{cs}{a} - c z_j$$

(49)

For the above case, $I_{max} = 2370$ bps. Fig. 32 is a plot of $I$ versus $t_o$. Note that with $t_o = 0.2352$ seconds, maximum information throughput of 2370 bps has been achieved.

With the optimum packet transmitting time $t_o$ found through the above procedures, the expected number of packets transmitted per silent period $E(N)$ can be found from Equation (5) or Fig. 19(a). $E(N)$ is found to be 6.25 packets per silent period, and $t_{ovhd}$ can be found from Equation (6) to be 0.1109 seconds per silent period. From Fig. 19(b) or Equation (6), it appears that less than 7% of the average silent period is wasted due to interference if the maximum packet size is used.
$$y = e^{-a(z_o + z_1)/s}$$

$$y = 1 - \frac{az_1}{s}$$

Fig. 31. Solution of Optimum Packet Size ($z_1$).

Fig. 32. Optimum Packet Size to Achieve Maximum Information Rate.
The above analysis is done for the single server case. This analysis can be extended to the k-channel case. Intuitively, if the statistical characteristics of all channels are identical and if the rotational priority scheme (round-robin) is applied to the distribution of the arriving data packets to each channels, then the maximum information throughput will be k times larger than that of the single server case. In this case, $E(N)$ and $t_{ovhd}$ will remain the same, while maximum data packet arrival rate as a whole can $k$ times larger than $\lambda_{d_{max}}$. It is expected that the average queue size $L_2$ and the average total resident time of a packet $W_2$ for each server will be the same as in the single-channel case. However, the buffer storage required for data-packet queues will be $k$ times larger.

### 4.7 Analysis of Multi-Channel MVD System

Fig. 33 represents a simplified model for the multi-channel MVD System. In this model, it is assumed that the time for assigning buffer and adding headers for data packets may be small enough as compared to the transmitting time of data packets or the waiting time of data packets before they are transmitted. Some other assumptions are made as follows:

1. The voice signal stream arrives in burst with Poisson arrival patterns;
Fig. 33. Multi-Channel Queueing Model of the MVD System.
(2) The talking period lasts for a duration that is exponentially distributed;
(3) The inter-arrival time of data packets is also exponentially distributed;
(4) The time required for a data packet to complete its transmission is also exponentially distributed.

For a two-channel system, let
\( \lambda_1 = \) mean arrival rate of voice signal bursts,
\( \lambda_2 = \) mean arrival rate of data packets,
\( \frac{1}{\mu_1} = \) mean talking period of voice signal,
\( \frac{1}{\mu_2} = \) mean transmission time of data packets,
\( P_{ij}(t) = \) the probability of \( i \) units of voice transactions and \( j \) data packets in the transmitting system at time \( t \), with \( i = 0, 1, 2 \) and \( j = 0, 1, 2, \ldots, n \).

If the single-queue round-robin or random selection strategy is used, then the state transition diagram of the system is shown in Fig. 34. Note that \( i \) ranges from 0 to 2 (for 2-channel case) because of the real-time nature of voice signals. There cannot be more than one unit of voice transaction in one channel of the system. On the other hand, the number of data packets in the transmitting system may be any non-negative integer depending on its input rate, the transmission service rate and the frequency of voice
\[ \Lambda_1 = 1 - \lambda_1 \Delta t \]
\[ \Omega_1 = 1 - \mu_1 \Delta t \]
\[ \Lambda_2 = 1 - \lambda_2 \Delta t \]
\[ \Omega_2 = 1 - \mu_2 \Delta t \]

Fig. 34. State Transition Diagram for Multi-Channel Single-Queue Model
interference. From the state transition diagram, the steady-state differential-difference equation are shown to be as follows:

\[
(2 \lambda_1 + \lambda_2) P_{00} = \mu_1 P_{10} + \mu_2 P_{01} \quad (50a)
\]

\[
(\lambda_1 + \mu_1 + \lambda_2) P_{10} = 2 \mu_1 P_{20} + 2 \lambda_1 P_{00} + \mu_2 P_{11} \quad (50b)
\]

\[
(2 \mu_1 + \lambda_2) P_{20} = \lambda_1 P_{10} \quad (50c)
\]

\[
(2 \lambda_1 + \lambda_2 + 2 \mu_2) P_{01} = \mu_1 P_{11} + 2 \mu_2 P_{02} + \lambda_2 P_{00} \quad (50d)
\]

\[
(2 \lambda_1 + \lambda_2 + 2 \mu_2) P_{0n} = \mu_1 P_{1n} + \lambda_2 P_{0,n-1} + 2 \mu_2 P_{0,n+1} \quad (n \geq 2)
\]

\[
(\lambda_1 + \mu_1 + \lambda_2 + 2 \mu_2) P_{1n} = 2 \lambda_1 P_{0n} + 2 \mu_1 P_{2n} + \lambda_2 P_{1,n-1} + \mu_2 P_{1,n+1} \quad (n \geq 1)
\]

\[
(2 \mu_1 + \lambda_2) P_{2n} = \lambda_1 P_{1n} + \lambda_2 P_{2,n-1} \quad (n \geq 1) \quad (50g)
\]

The Equations (50a) to (50g) of the simplified model are more complicated than Equations (12a) to (12d) of the single-channel model.

The idea of the simplified model may be easily extended to \( N \) channels. For instance, the steady-state differential-difference equations of such a simplified model for three channels may also be written from its state transition diagram as shown in Fig. 35.

The solution of these simultaneous difference equations \((30)\) or \((31)\) involves very length mathematical derivation. An attempt to solve equations for \( 2 \)-channel case can be found in [51].
Fig. 35. State Transition Diagram for Three-Channel Single-Queue Model of MVD System
\[
(3 \lambda_1 + \lambda_2) p_{00} = \mu_1 p_{10} + \mu_2 p_{01} \quad (51a)
\]
\[
(2 \lambda_1 + \mu_1 + \lambda_2) p_{10} = 3 \lambda_1 p_{00} + 2 \mu_1 p_{20} + \mu_2 p_{11} \quad (51b)
\]
\[
(\lambda_1 + 2 \mu_1 + \lambda_2) p_{20} = 2 \lambda_1 p_{10} + 3 \mu_1 p_{30} + \mu_2 p_{21} \quad (51c)
\]
\[
(3 \mu_1 + \lambda_2) p_{30} = \lambda_1 p_{20} \quad (51d)
\]
\[
(3 \lambda_1 + \lambda_2 + \mu_2) p_{01} = \mu_1 p_{11} + \lambda_2 p_{00} + 2 \mu_2 p_{02} \quad (51e)
\]
\[
(2 \lambda_1 + \mu_1 + \lambda_2 + \mu_2) p_{11} = 3 \lambda_1 p_{01} + 2 \mu_1 p_{21} + \lambda_2 p_{10} + 2 \mu_2 p_{12} \quad (51f)
\]
\[
(\lambda_1 + 2 \mu_1 + \lambda_2 + \mu_2) p_{21} = 2 \lambda_1 p_{11} + 3 \mu_1 p_{31} + \lambda_2 p_{20} + \mu_2 p_{22} \quad (51g)
\]
\[
(3 \mu_1 + \lambda_2) p_{31} = \lambda_1 p_{21} + \lambda_2 p_{30} \quad (51h)
\]
\[
(3 \lambda_1 + \lambda_2 + 2 \mu_2) p_{02} = \mu_1 p_{12} + \lambda_2 p_{01} + 3 \mu_2 p_{03} \quad (51i)
\]
\[
(3 \lambda_1 + \lambda_2 + 3 \mu_2) p_{0n} = \mu_1 p_{1n} + \lambda_2 p_{0,n-1} + 3 \mu_2 p_{0,n+1} \quad (n \geq 3) \quad (51j)
\]
\[
(2 \lambda_1 + \mu_1 + \lambda_2 + 2 \mu_2) p_{1n} = 3 \lambda_1 p_{0n} + 2 \mu_1 p_{2n} + \lambda_2 p_{1,n-1} + 2 \mu_2 p_{1,n+1} \quad (n \geq 2) \quad (51k)
\]

4.3 SUMMARY

This chapter provides analysis for the voice interference effect on data packets, the queueing phenomena of data packets in transmitting microcomputer for two retransmission schemes, namely preemptive-repeat and preemptive-resume, and optimum packet size and maximum throughput. Also, the state transition diagram and the differential-difference equations for multi-channel MVBD System are set up for comparison. In this analysis, the preemptive-resume scheme is shown to be better than the preemptive-repeat in the sense of less accumulation of data packets and less packet waiting time. The quantitative results can be used to estimate buffering and response time requirements.
5.1 REASONS FOR SIMULATION

It is very risky to implement a newly designed system without first evaluating its feasibility and predicting the system performance in real world environment. In most cases, it is more economical and advantageous to efficiently represent the physical system and its environment through the use of a "model". For the design of the MVD System, a model simulated by a digital computer is used to predict its system performance. Other reasons for this simulation can be described as follows.

5.1.1 Use of Actual operational Conditions

In the work of analysis for the MVD System, the Poisson arrival pattern for voice signal and the exponential distributions for the talking and silent periods are used; however, these are not exactly true according to the sampled observation. In the computer simulation, actual arrival pattern and talking- and silent-period distributions are used, so that the simulation model is operated in a real world environment.
5.1.2 Obtaining Performance Figures

In the work of analysis for the MVD System, tedious mathematical derivations are carried out in order to obtain closed-form form results for performance such as the average number of packets in the system and the average waiting time of a packet. However, these results are not always obtainable in a more complicated system configuration (e.g. in a multi-channel MVD System), because these mathematical derivations involve fairly advanced mathematics such as queueing theory and differential-difference equations. In the computer simulation these performance figures are always available for outputting any time during or after the simulation.

5.1.3 Simulating Mathematically Non-representable Operation

In the work of analysis for the MVD System, the "channel selection strategy" has not been considered in order to select the best channel for the next packet to be dispatched in the Route Selection Program. This operation can be very easily simulated in the computer simulation by selecting a channel that is the best candidate as far as the system throughput is concerned. In fact, many different strategies can be applied and their corresponding performance figures can be compared.
5.1.4 Discovery of System Bottleneck Regarding Throughput

In designing a transaction-oriented system, it is required to get as much throughput as possible. Likewise, in the simulation for the MVU System the flow condition of data packets must be visualized under the actual processing condition and under various system parameters and route selection strategies. In case that any bottleneck appears at the point of processing or a waiting station where a large amount of data packets are accumulated, the reason that causes such a bottleneck must be studied in order to find out a solution to smooth out such a flow condition.

5.2 APPROACH TAKEN IN BUILDING A SIMULATION MODEL

The purpose of the simulation is to predict the system performance under real operational conditions. In the case of MVU System, the observation of the queueing phenomena of data packets in the transmitting and receiving microcomputers is desirable under various system parameters and the channel selection strategies. Therefore, it is desirable to select a discrete simulation language that provides features of measuring such queueing characteristics.
The General Purpose Simulation System (GPSS) is a discrete simulation language that does provide features for simulating queueing systems and automatically measuring queueing characteristics [22,44]. Many features of GPSS can be very well applied to the MVD operation in the following aspects.

5.2.1 Generation of Data Packets

In the MVD System, data traffic arrives at the input lines of the transmitting side. These data are first framed into packets and then wait for transmission service. The GPSS GENERATE block creates transactions that simulate the data packets. The rate of transaction's creation is limited by the maximum throughput of data packets through the MVD channels. Considering the application of multiple data terminals to the MVD System, data packets are assumed to arrive in Poisson fashion with a mean arrival time. The first operand of the GENERATE block represents the mean arrival time and the second operand represents the exponential distribution.

5.2.2 Selection of a Minimum-Length Queue or Finding a Silent Channel
After formation of a data packet, the transmitting-side CPU has to decide which MVD channel is the best one to dispatch this packet. For a multi-line queueing model where each MVD channel has a waiting line in front of it, means should be available to select an MVD channel that has the minimum length of queue. This strategy can considerably shorten the packet delay. The GPSS SELECT block provides this feature that can select one channel among a set of channels to satisfy the specified minimum/maximum condition of certain Standard Numeric Attributes (SNAs). The selected number will then be assigned to one of the parameters of the transaction. For a single-line queueing model, one of the strategies is to use round-robin type scanning method to look for a silent channel and then to assign this channel as the packet's transmission server. The GPSS GATE LS and LOGIC S blocks working in different model segments will accomplish this task. In the voice-signal transaction model segment, one transaction for each channel will enter the LOGIC S block to turn on the "green light" (set the corresponding logical switch to "on" condition) at the appropriate time. In the data transaction model segment, a packet will check to see if the appropriate channel is in "green light" state or "red light" state (logical switch in "off" condition) when entering the GATE LS block. If this is in "green light" state, this channel will be assigned for its transmission service.
5.2.3 Capture of a Voice-Grade Channel for Transmission Service

In the MVD System all voice-grade channels are servers that provide transmission service. If a channel is not occupied by voice signals and this channel is assigned for data packet’s transmission, this channel should be captured immediately for the transmission service. After the completion of its transmission service, this channel must be made available for other data packets. The GPSS SEIZE and RELEASE blocks provide features that can capture or release a channel facility.

5.2.4 Entering a Waiting Line or Leaving a Waiting Line

In the MVD System, data packets must wait in one of the multiple waiting lines or on the single waiting line before they are served for output transmission in a "first-come-first-serve" basis. The GPSS QUEUE and DEPART blocks provide such a feature that can put a data-packet transaction to the back of a waiting line (queue) and automatically accumulate the waiting line statistics, such as average or maximum waiting time of a transaction (average or maximum packet delay), and average or maximum length of a waiting line (average or maximum number of data packets in
5.2.5 **Entering for Processing or Transmission Service**

In the MVD System, data packets must be processed by transmitting/receiving microcomputers or must receive transmission service provided by the voice-grade channels. The GPSS ADVANCE block can simulate the period of time during which the entering packet is processed or served. In the GPSS simulation only the time factor is essential. However, the processing time of a packet is comparatively shorter than the transmission service time of data packets. The former is in the order of 500 microseconds while the later is in the order of 400 milliseconds for the constant service time if the packet size is 1000 bits and the data rate is 2400 bps. In the event of being preempted by the arriving voice signal, the data packet will be taken away from the ADVANCE block and re-enter it at a later time when the preempted channel is once available.

5.2.6 **Interrupting by Voice Arrival**

In the MVD System, during the transmission-service period, any unanticipated arrival of voice signals might interrupt the present service of data packet transmission. Due to the urgent nature of the analog voice signal, the
present service of data packet has to be interrupted to give way to the voice signal. Therefore, a "preemption" will occur in that voice-grade channel and the previously served data packet will wait temporarily aside until the interrupting voice signal is over in that channel, or be taken away for a complete retransmission scheduling. The JPS5 PREEMPT block provides such features that the higher priority transaction is allowed to pluck the service facility from the lower priority one. In this PREEMPT block, the fifth (E) operand specifies that the preempted transaction is to be removed (RE) from the block for retransmission or to be resumed (blank) its service after the completion of the preempting voice signal. The third (C) operand of the PREEMPT block specifies the block location to which the preempted transaction is to be routed. The preempted transaction will be in contention for the use of the channel facility if the fifth operand does not specify the RE removal option. These two features of the PREEMPT block fit very well to the characteristics of handling the MVD voice interrupt, which allow interrupt-repeat transmission and interrupt-resume transmission. These two cases will be called "preemptive-repeat" and "preemptive-resume", respectively.
5.2.7 **Representation of a Multiple-Channel MVD System**

In the MVD System, one of the system parameters is the number of channels that will be configured to provide packet transmission service. This number greatly affects the data packet throughput of the system. To build the simulation model segments, one would like to keep the model segments unchanged regardless of the number of channels used in the configuration. This number should be conveniently changeable during each simulation run. The GPSS language provides such a feature that some operands can be represented indirectly by the Standard Numeric Attributes (SnAs), such as transaction's parameters and the save-value storage. In the approach taken, the first model segment simulates the data-packet transactions entering the MVD System and going through a serial service line, and the second model segment simulates the voice signal transaction entering the MVD System to interrupt the current data service if any service is still going on in that channel.

In the first model segment, the generated data packet transaction, when joining a queue in the multi-line queueing system or entering the transmission service in the single-line queueing system, is individually assigned a channel number that is kept in a specific parameter (say P2) of all the entering transactions. The content of this
parameter automatically categorizes these data-packet transactions into classes whose number is equal to the number of the MVD channels. When seizing a serving facility or joining a waiting queue, the transaction will simply enter the SEIZE or QUEUE block with the facility or queue number represented by the transaction's own parameter P2. Therefore, the corresponding facility or queue will be registered according to the content of the transaction's parameter P2. Similarly, if a transaction finishes using the facility or departs from a queue, it will simply enter the RELEASE or DEPART block with the facility or queue number represented by the transaction's same parameter P2.

In the second model segment, the generated transactions arriving randomly in time to each channel act as channel-availability controllers. After generated, each transaction is assigned a different channel number in its own parameter (say P1). Upon arrival to the respective channel, each transaction will turn on the "red light" by entering a LOGIC R block to reset the logical switch represented by the content of this parameter P1. It will immediately pluck the use of the channel represented by the content of this parameter P1 by entering a PREEMP block. After a talking period elapses, this transaction returns the plucked facility by entering a RETURN block with the returned facility number represented by the content of
parameter $P_1$, and then turns on the channels "green light" for data packets by entering a LOGIC $S$ block to set the logical switch represented by the content of parameter $P_1$. After a silent period elapses, this transaction goes back to the previously entered block that will perform the next preemption cycle for the associated channel represented by the transaction's parameter content.

5.2.8 Simulation of Actual Talking and Silent Period

Distribution

In the MVD System, the voice signal arrives in a certain rate and elapses for a certain period of time, and then a silent period follows. In Chapter 4, the mean values of talking and silent periods and their distributions have been examined. Since the distributions are not exactly Poisson, the mathematical analysis can only assume that these distributions are Poisson for the purpose of obtaining a simple approximation. If the actual Erlang $k$ distribution with $k = 1.6$ were used, it is very difficult to get a closed-form result that is meaningful. However, in GPSS simulation the actual distribution can be used in the simulation run. Thus in this simulation these two periods are to be generated according to the actual distributions. The GPSS FUNCTION provides such a feature to generate these
two periods using its random number generator. First the probability distribution functions of talking and silent periods are defined graphically as in Fig. 36, with the abscissa representing the accumulated probability and the ordinate representing the talking (or silent) period in seconds. The value of the abscissa (between 0 and 1) is equal to the probability that the talking (or silent) period is less than or equal to its corresponding ordinate value. When it is required to find a talking (or silent) period in the simulation, one of the GPSS random number generators generates a fractional random number between 0 and 1, and then from the defined curve finds or calculates the corresponding talking (or silent) period. Although the generated period might not be exactly the same length as in the actual case at the specific instant, the "statistical characteristic" is the same as in the actual case. These generated talking (or silent) periods can be applied in the ADVANCE blocks of the voice-transaction model segment. Thus, the actual distribution of the talking and silent periods can be implemented in the simulation run.

5.3 SIMULATION MODELS

Two simulation models have been examined in this research; one is the multi-line queueing model and the other is the single-line queueing model, each of which in
Fig. 36. Probability Distribution of (a) talking period (curve 1) and (b) silent period (curve 2)
turn considers both the "preemptive-repeat" and "preemptive-resume" disciplines for voice interrupt handling.

5.3.1 Multi-line Queueing Model

In the multi-channel MVD System, data packets after their formation wait in the buffer of the transmitting microcomputer for their transmission service. The queueing discipline in the multi-line queueing model is to select one waiting line among all lines that may already form in front of each MVD channel. Fig. 37 shows such a multi-line queueing model.

The transmitting-side CPU should be intelligent enough to know to which queueing line this newly formed packet must be assigned. This is a matter of "queue selection" strategies. Five strategies have been examined:

1. Round-robin selection strategy (RRB),
2. Random selection strategy (RDM),
3. Minimum queue-length selection strategy (MINQL),
4. Minimum waiting-time selection strategy (MINWT),
   and
5. Less-than-threshold selection strategy (LTT).

The description of these five strategies and their relative merits are given below.
Fig. 37. Multi-line Queueing Model for the MVD System.
The RRB is to select a queue in a cyclic sequence among all the queues. This strategy is very easy to implement but may not be efficient since a long waiting line and an empty line will get an entry to its respective queue with equal opportunity.

The RDM is to select a queue according to a random number generated between one and the maximum number of channels with uniform distribution. This strategy needs a random number generator in the transmitting side. It is also easy to implement but may not be efficient for the same reason as the RRB.

The MINQL is to select a minimum length queue among all the MVD channels. This strategy is more complicated to implement since all the queue lengths must be compared against one another to determine which is the shortest. A doubly-linked list starting from the maximum length queue to the minimum one must be constructed in order to find the minimum element at the end of the doubly-linked list. This method needs an updating procedure for the doubly-linked list whenever one element is added or taken away from each queue. The algorithm to implement this strategy is described in Chapter 6.
The MINwT is to select a queue which has the minimum waiting time of previously departing transaction. This strategy involves recording of waiting time for each transaction in queue and construction of a doubly-linked list as in the MINQL. This requires much more core storage and CPU time to perform these tasks. The algorithm to perform the doubly-linked list is the same as that of the MINQL.

The LIT is to look for a queue on a rotating basis that its queue content is less than some pre-determined level or threshold. This strategy involves only updating the queue status by comparing to the threshold whenever a transaction is added to or taken away from the queue. The first encountered queue that has its queue content less than the threshold will be selected as the one for the next transaction to be added to. This strategy is probably the most promising one since it requires very few additional storage and CPU time to process or update the information. The algorithm to implement this strategy is also described in Chapter 6.

5.3.2 Single-line Queueing Model

The single-line queueing model resembles the "quicky line" policy in some banks where all the arriving customers
must join the end of a single waiting line and the customer at the very front of the line shall go to the teller immediately after the previously served customer finished his (or her) transaction and leaves the counter. Fig. 38 shows the single-line queueing model. However, in the MVD system because of the voice interrupting characteristics, it is necessary to select a channel that has the least probability of being interrupted by the voice signal. The following "channel selection" strategies have been examined to see which is the best to meet this requirement:

1. Round-robin selection strategy (RRB),
2. Random selection strategy (RDM),
3. First silent selection strategy (FSLv), and
4. Most recently released selection strategy (MRR).

The RRB is to select a silent channel in a cyclic order. This strategy is very easy to implement but may spend additional CPU time in searching for a silent channel. Besides, this choice does not assure that the found silent channel is the best candidate to insert a data packet.

The RDM is similar to the round-robin selection strategy except that the search for the next silent channel is not performed in a cyclic linear manner but on a uniformly random selection basis. This requires that a random number generator exists in the system. The drawback
Fig. 38. Single-line Queueing Model for the MVD System.
of this strategy is that the next selected voice-grade channel is not necessarily a silent channel and if it is, it is not assured that it is the best candidate for the next dispatching of a data packet.

The FSLN is to select a silent channel among all the MVD channels. The selection may start from the very beginning or in a cyclic order. If it is from the very beginning, it will result in selecting one of the first few channels and lead to non-uniform usage of channel capacity. This strategy is easy to be implemented but requires some CPU time in searching for a silent channel before a data packet is assigned to it.

The MRR is to select a silent channel that is the most recently released channel from voice occupancy. This strategy requires the construction of a channel status table with pointers forming a doubly-linked list to reflect the sequence of recentness. As soon as a voice-grade channel is released by voice occupancy, this channel is linked to the very front of the link list. It will become the next candidate for carrying data packets. Whenever a voice-grade channel is assigned for data packet dispatching or is plucked by the voice signal, this channel is taken out from the link list that requires minor adjustment to the link list.
The advantage of the MRR strategy is that the most recently released channel has the least probability of being interrupted again by the next arriving voice signal since this channel will probably stays silent for a while until the next voice signal arrives. This assertion can be explained using the probability theory as follow.

Given that the channel is in silent state and the channel is in a normal talking and silent sequence according to the observed statistical characteristic, the probability of being interrupted by the voice signal (same as silent period ends) at \( t_0 \) is the probability distribution function of the silent period at \( t_0 \), and this function is a monotonically increasing function from 0 to 1 with respect to silent duration \( t \).

During a silent period, several data packets can usually be sent through the voice grade channel without encountering voice interruption. If \( m \) packets can be sent during the mean silent period, \( m \) can be used as a threshold under which the probability of being interrupted by the voice signal to a packet is less than fifty percent. Therefore, if a channel is assigned for data dispatching and if it has dispatched less than \( m \) packets, this channel is still a good candidate to dispatch packets and it still stays in the MRR link list; otherwise, it is removed out of
the link list. The algorithm to implement the MRK will be described in Chapter 6.

5.3.3 *Repeat Transmission Versus Resume Transmission of Interrupted Data Packets*

If a data packet is interrupted by the voice signal during its transmission, there are two methods to handle the interrupted data packet:

1) to discard the partially transmitted packet and to *repeat* all over again;

2) to allow the partially transmitted packet to go through and to *resume* the packet transmission later starting at the point where interrupt has occurred.

In the simulation models, both methods have been implemented so that the results can be compared quantitatively. It is obvious that the resume method of handling the interrupted data packet will result in better throughput, shorter queue length and less packet delay as compared to the *repeat* method, because retransmission of the discarded portion of the partially transmitted packet constitutes the overhead of the system. However, the resume method shall be more complicated in software logic as well
as in the communication protocol.

5.4 TABLE OF DEFINITIONS OF GPSS ENTITIES FOR THE MVD SYSTEM

Time Unit: 100 microseconds

<table>
<thead>
<tr>
<th>GPSS Entities</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Transactions</strong></td>
<td></td>
</tr>
<tr>
<td>Model Segment 1</td>
<td>Data packets for transmission service</td>
</tr>
<tr>
<td>Model Segment 2</td>
<td>Voice signals interrupting data service</td>
</tr>
<tr>
<td>Model Segment 3</td>
<td>A timer</td>
</tr>
</tbody>
</table>

**Parameters of Transactions**

Model Segment 1

- **P1**: Packet sequence number
- **P2**: Assigned number of queue, facility, or switch
- **P4**: Packet arrival time at receive side
- **P5**: Preemption count
- **P7**: Input line number at transmit side
- **P8**: Output line number at receive side
- **P9**: Packet arrival sequence in RCV side
- **P12**: Packet processing time

Model Segment 2

- **P1**: Channel number associated with the interferer
<table>
<thead>
<tr>
<th><strong>GPSS Entities</strong></th>
<th><strong>Interpretation</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Facilities</td>
<td></td>
</tr>
<tr>
<td>CPU1R</td>
<td>Transmit-side CPU</td>
</tr>
<tr>
<td>CPU1RV</td>
<td>Receive-side CPU</td>
</tr>
<tr>
<td>1,2,3,...X$\times$CHnU</td>
<td>Transmission servers</td>
</tr>
<tr>
<td>Queue</td>
<td></td>
</tr>
<tr>
<td>TrPKT</td>
<td>Transmit-side system queue for statistics</td>
</tr>
<tr>
<td>X$\times$OUT</td>
<td>Transmit-side aggregated queue for statistics</td>
</tr>
<tr>
<td>1,2,3,...X$\times$CHnU</td>
<td>Multi-line separate queue</td>
</tr>
<tr>
<td>RVPKT</td>
<td>Receive-side system queue for statistics</td>
</tr>
<tr>
<td>OUTQ</td>
<td>Receive-side output queue</td>
</tr>
<tr>
<td>Storages</td>
<td></td>
</tr>
<tr>
<td>CHnLD</td>
<td>Number of available channels for data</td>
</tr>
<tr>
<td>CHnLV</td>
<td>Number of available channels for voice</td>
</tr>
<tr>
<td>Function</td>
<td></td>
</tr>
<tr>
<td>XPDIS</td>
<td>Exponential distribution function</td>
</tr>
<tr>
<td>SILEN</td>
<td>Silent period distribution function</td>
</tr>
<tr>
<td>TALK</td>
<td>Talking period distribution function</td>
</tr>
<tr>
<td>VIA1</td>
<td>Voice inter-arrival time distribution</td>
</tr>
<tr>
<td>EOFN</td>
<td>Expected number of packet transmitted during a silent period</td>
</tr>
<tr>
<td>DEST</td>
<td>Random assignment of packet's destination</td>
</tr>
</tbody>
</table>
**UPSS Entities**

**Interpretation**

**which**  Random assignment of outgoing channel

**Variables**

<table>
<thead>
<tr>
<th>i</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Quotient of channel number divided by 2</td>
</tr>
<tr>
<td>2</td>
<td>Residual of channel number divided by 2</td>
</tr>
<tr>
<td>3</td>
<td>Packet transmit time on voice-grade line ( (t) )</td>
</tr>
<tr>
<td>4</td>
<td>Average silent period overhead</td>
</tr>
<tr>
<td>5</td>
<td>A certain percentage of maximum mean arrival rate for the whole system (packets/second)</td>
</tr>
<tr>
<td>6</td>
<td>Inter-arrival time in 100 microseconds</td>
</tr>
<tr>
<td>7</td>
<td>Transmit time in 100 microseconds on receive-side output circuit</td>
</tr>
<tr>
<td>8</td>
<td>Maximum input speed in bps</td>
</tr>
<tr>
<td>10(BVARIABLE))</td>
<td>Providing sequence check logic</td>
</tr>
</tbody>
</table>

**Tables**

<table>
<thead>
<tr>
<th>1,2,3,...,10</th>
<th>Interpretation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channels 2,4,6,...20 queue content distributions</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TAB20</th>
<th>HCV-SIDE packet resident time distribution</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>TAB21</th>
<th>HCV-SIDE packet number distribution</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>TAB22</th>
<th>HCV-SIDE output queue distribution</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>TAB23</th>
<th>XML-T-SIDE packet number distribution</th>
</tr>
</thead>
</table>

<p>| TAb24  | XML-T-SIDE aggregated line distribution                                      |</p>
<table>
<thead>
<tr>
<th><strong>GPS Entities</strong></th>
<th><strong>Interpretation</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>TAB25</td>
<td>XMTI-SIDE waiting time distribution for transmission</td>
</tr>
<tr>
<td>TAB26</td>
<td>XMTI-SIDE resident time distribution</td>
</tr>
</tbody>
</table>

**Savevalues**

<table>
<thead>
<tr>
<th><strong>Variable</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>XH$CHNO</td>
<td>Number of output voice-grade channels</td>
</tr>
<tr>
<td>XH$SPEED</td>
<td>Data rate of output VG channel</td>
</tr>
<tr>
<td>XH$SIZE</td>
<td>Packet size in bits</td>
</tr>
<tr>
<td>LS1-LS2A</td>
<td>Logical switches simulating green light for data packet sending when turned on</td>
</tr>
<tr>
<td>XH$SEQCK</td>
<td>Sequence check counter</td>
</tr>
<tr>
<td>XH$PERCn</td>
<td>A percentage used to reduce the maximum mean arrival rate of data packets</td>
</tr>
</tbody>
</table>

**User's Chain**

<table>
<thead>
<tr>
<th><strong>Variable</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>HVPWQ</td>
<td>Receive-side waiting chain for sequence check</td>
</tr>
</tbody>
</table>

5.5 FLOWCHARTS AND PROGRAM LISTINGS

The flowcharts and the program listings for the multi-line and single-line models are provided in Appendices A and B, respectively. Some sampled program outputs are also provided in Appendix C.
5.6 SIMULATION RESULTS

For the sake of convenience in the discussion of the simulation results, the following symbols are used:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C$</td>
<td>Number of channels</td>
</tr>
<tr>
<td>$\lambda_d$</td>
<td>Mean arrival rate of data packets</td>
</tr>
<tr>
<td>$S$</td>
<td>Data speed of VG channels in ops</td>
</tr>
<tr>
<td>$Z$</td>
<td>Size of data packets in bits</td>
</tr>
<tr>
<td>$\mu_d$</td>
<td>Mean service rate of data packets in packets/s ($=S/Z$)</td>
</tr>
<tr>
<td>$W$</td>
<td>Average delay time of data packets in the transmit side</td>
</tr>
<tr>
<td>$N$</td>
<td>Average number of data packets in the transmit side</td>
</tr>
<tr>
<td>$\rho_d$</td>
<td>Traffic intensity of data packets ($=\lambda_d/C\mu_d$)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\lambda_d$</th>
<th>1.2 3 4 4.8 6 0.318 0.067 7.059</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>0.019 2.408 3.513 7.132 9.713 12.197 24.237</td>
</tr>
<tr>
<td>$W$</td>
<td>0.439 0.531 0.015 1.162 1.793 3.357</td>
</tr>
<tr>
<td>$N/W$</td>
<td>1.162 2.981 4.013 4.901 6.148 0.437 0.799 7.220</td>
</tr>
<tr>
<td>$N/\lambda_d$</td>
<td>0.985 0.994 1.003 1.021 1.025 1.019 1.020 1.023</td>
</tr>
</tbody>
</table>

Table 4. Simulation Results for Single-Queue First Silent Strategy
(C=5 S=2400 ops, and Z=1000 bits)
Table 4 and 5 list some simulation results for the single-queue "first silent selection strategy" (FSLN).

\[
\begin{array}{cccccccc}
\lambda_d & 2.4 & 0 & 8 & 9.0 & 12 & 12.623 & 13.333 & 14.118 \\
n & 0.541 & 1.076 & 2.312 & 3.109 & 5.904 & 9.112 & 9.560 & 22.156 \\
h & 0.224 & 0.273 & 0.286 & 0.332 & 0.492 & 0.726 & 0.722 & 1.075 \\
W/n & 2.415 & 0.147 & 8.004 & 9.545 & 12.00 & 12.551 & 13.241 & 14.067 \\
W/\lambda_d & 1.000 & 1.825 & 1.811 & 0.994 & 1.000 & 0.994 & 0.993 & 0.996 \\
\end{array}
\]

Table 5. Simulation Results for Single-Queue First Silent Strategy (C=5, S=4800 ops, and Z=1000 bits)

Fig. 39 shows the plot of the average delay time $W$ of data packets in the transmit side against the average number $n$ of data packets for (1) multi-queue "minimum waiting time" (MINWt); (2) multi-queue "round-robin" (RRB); (3) multi-queue "minimum queue length" (MINQL) and (4) single-queue "first silent" (FSLN) selection strategies (all in resume mode). Note that they come out to be straight lines and can be expressed by a linear equation $W = k (N+b)$, where $k$ is the slope of the straight line and $-b$ is the distance to the origin from its interception to the $N$-axis. The equation $W = k (h+b)$ and the Little's formula $N = \frac{\lambda_d}{W}$ may be combined to find $W$ and $N$ in terms of $\lambda_d$. Then

\[
W = \frac{kb}{1 - k\lambda_d} \quad \text{and} \quad N = \frac{kb\lambda_d}{1 - k\lambda_d}
\]
Fig. 39. Average Delay versus Average Number of Data Packets in the Transmitting System for Various Strategies.

Fig. 40. W-N plot for Single-Queue MRR Strategy.
with the restriction that $\lambda_d$ must not be greater than $0.012 \cdot C$, which is the residual capacity of the channels for data transmission (utilization by voice signal is $0.388$).

For example, the values of $k$ and $b$ for straight line #4 in Fig. 39 are found to be $0.12$ and $-2.0$, respectively. Hence

$$W = \frac{0.312}{1 - 0.12 \lambda_d}$$

$$N = \frac{0.312 \lambda_d}{1 - 0.12 \lambda_d}$$

for single-queue FSLh strategy with $S = 2400$ bps, $Z = 1$ kbps, $\omega$ and $C = 5$. The values of $k$ and $b$ may vary with respect to the number of channels, data speed of VG channels and the packet size. Fig. 40 shows the $W-N$ plot for the single-queue Mkh strategy with $C = 1$ and $3$. Notice that two lines are plotted for $C = 3$ but using $S = 24.360$ bps and 4800 bps.

Fig. 41 shows the average number $N$ of data packets in the transmit side against the traffic intensity of data packets for various strategies, and Fig. 42 shows the average delay time $W$ of data packets against the traffic intensity for the same strategies. We can see that the single-queue strategies provide greater data throughput than the multi-queue strategies since data packets experience shorter delay time under the former strategies.
Fig. 41. $N - \rho_d$ Plot for Various Strategies.

Fig. 42. $W - \rho_d$ Plot for Various Strategies.
Furthermore, the average number of data packets waiting for transmission service in the transmit side is also smaller than that for multi-queue strategies for the same parameters (C = 5, S = 2400 bps and Z = 1200 bits). Notice also that the curve of multi-queue MINQL strategy is close to those of the single-queue strategies.

Figs. 43 and 44 show similar comparisons for C = 3. Again, the single-queue strategies are better than the multi-queue strategies.

Figs. 45 and 46 show comparisons between the "preemptive-resume" and the "preemptive-repeat" transmission for the single queue "first silent" (FSLN) strategy. It is realized that the operation in the "preemptive-resume" model provides greater data throughput than that of the "preemptive-repeat" model since the corresponding delay time and the average number of data packets in the transmit side are smaller under the same operation parameters. These results also hold for multi-queue strategies.

Figs. 47 and 48 show plots of $N - P_d$ and $W - P_d$ for single-queue "first silent" strategy with data speeds varying from $S = 1200$ bps to $4800$ bps. Fig. 49 and 50 show plots of $N - P_d$ and $W - P_d$ for single-queue MRMR strategy with data speeds varying from $S = 1200$ to $9600$ bps. From these
Multi-Queue Minimum Queue Length (MINQL)
Single-Queue First Silent (FSLN)

C = 3
S = 2400 bps
Z = 1000 bits

Fig. 43. \( N - \rho_d \) Plot for Two Strategies

Fig. 44. \( W - \rho_d \) Plot for Two Strategies.
Fig. 45. $N - \rho_d$ Plot for Comparison between Preemptive-Repeat and Preemptive-Resume Schemes for Single-Queue First Silent Strategy.

Fig. 46. $W - \rho_d$ Plot for Comparison between Preemptive-Repeat and Preemptive-Resume Schemes for Single-Queue First Silent (FSLN) Strategy.
Fig. 47. $N - \rho_d$ Plot for Single-Queue with Different Transmission Speeds.

1. $S=1200$ bps
2. $S=2400$ bps
3. $S=4800$ bps

Fig. 48. $W - \rho_d$ Plot for Single-Queue with Different Transmission Speeds.
Fig. 49. $N - \rho_d$ Plot for Single-Queue MRR Strategy with Different Speeds.

Fig. 50. $W - \rho_d$ Plot for Single-Queue MRR Strategy with Different Speeds.
two figures it is found that $\hat{w}$ and $N$ increase as $P_d$ increases since $P_d = \lambda d / \mu d$. Also, it is known that $N$ and $\hat{w}$ both decrease as the average channel service rate ($\mu_d$) increases for the same traffic intensity ($P_d$).

Table 5 illustrates simulation results on the throughput of data packets at the point of receiving side including the sequence check. In this table it is also shown that the single-queue strategy is better than the multi-queue strategy from the viewpoint of data packet throughput. Since the preemption by voice signal is a major factor that degrades the data packet throughput, the frequencies of preemption occurring in each strategy is also included in the table. In general, more frequent preemptions result in less packets received in sequence in a fixed period of time by the receiving side for the same average arrival rate of data packets at the transmitting side. For single-queue strategy, the larger data throughput of first silent (FSLN) and of the "most recently released (MRR)" strategies must be related to the less frequent preemption as discussed in Section 5.2. Fig. 51 illustrates the average delay time against the traffic intensity $P_d$. This delay time includes sequence check and transmission delay of the satellite circuit.
Fig. 51. Average Delay of Data Packets Including Transmission Delay and the Sequence Check in Receiving Side.
<table>
<thead>
<tr>
<th>$\rho_d$</th>
<th>no. of arrived packets</th>
<th>single queue first empty</th>
<th>single queue MRR</th>
<th>multi-queue round robin</th>
<th>multi-queue min. queue</th>
<th>multi-queue min. aver. waiting time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>no. of preempt received</td>
<td>no. of preempt received</td>
<td>no. of preempt received</td>
<td>no. of preempt received</td>
<td>no. of preempt freq.</td>
</tr>
<tr>
<td>0.1</td>
<td>284</td>
<td>284</td>
<td>284</td>
<td>281</td>
<td>284</td>
<td>264</td>
</tr>
<tr>
<td>0.25</td>
<td>715</td>
<td>714</td>
<td>714</td>
<td>712</td>
<td>714</td>
<td>714</td>
</tr>
<tr>
<td>0.333</td>
<td>963</td>
<td>963</td>
<td>963</td>
<td>946</td>
<td>963</td>
<td>945</td>
</tr>
<tr>
<td>0.4</td>
<td>1176</td>
<td>1170</td>
<td>1119</td>
<td>1119</td>
<td>1161</td>
<td>1100</td>
</tr>
<tr>
<td>0.5</td>
<td>1476</td>
<td>1462</td>
<td>1441</td>
<td>1441</td>
<td>1457</td>
<td>1219</td>
</tr>
<tr>
<td>0.526</td>
<td>1545</td>
<td>1538</td>
<td>1408</td>
<td>1408</td>
<td>1532</td>
<td>1332</td>
</tr>
<tr>
<td>0.555</td>
<td>1632</td>
<td>1602</td>
<td>1534</td>
<td>1534</td>
<td>1604</td>
<td>1352</td>
</tr>
<tr>
<td>0.588</td>
<td>1733</td>
<td>1701</td>
<td>1667</td>
<td>1667</td>
<td>1667</td>
<td>1366</td>
</tr>
</tbody>
</table>

Table 6. Throughput of Data Packets in the System (Simulation Time: 4 min.).
5.7 SUMMARY

From the above simulation results, it is concluded that the single-queue "most recently released" (MRR) strategy with resume transmission is the best strategy that will achieve maximum data throughput and minimum overhead. The next one is the single-queue "first silent" (FSLN) strategy with resume transmission. In general, the multi-queue strategies are inferior to single-queue strategies from the viewpoint of packet delay and data throughput.

In the simulation run on the "preemptive-repeat" case, it was found that the bottleneck has appeared at the point where the packet sequence check is being made. This was because some earlier arrived packets must wait for certain missing packet that has encountered frequent voice interruption in the assigned channel. Therefore, it is suggested that multiple copies of data packets may be sent through different channels in order that at least one "interfere-free" copy of data packet is available early enough to resolve the accumulation of data packets at the sequence check station of the receiving side.

In the analysis of the MVD System, it was assumed that the arrival pattern of voice signal was Poisson. However, in this simulation, the actual statistical data and
distributions of voice signals are used. The major
difference of both results can be asserted as follows: In
the Poisson case, the probability of voice interference at
any instance is the same because of the memoryless nature of
the Poisson distribution. However, in the actual case, the
probability of voice interference right after the
termination of voice signal is less than that at the point
where voice signal has been lasting for over the mean value
of the silent period. This is why the single-queue "mostly
recently released" strategy is the best strategy in the
simulation results.

Due to the drastic growth of core memory during the
simulation and the limited amount of it, the simulation runs
for more channels, say 12, 24, etc. are very difficult and
expensive to perform. However, the principle of simulation
will be the same and their simulation results are
expectable. Also, there are many different strategies that
may be worth examining in the future; however, the major
ones have been done with satisfaction.
CHAPTER 6

IMPLEMENTATION OF THE MIXED VOICE/DATA TRANSMISSION SYSTEM

0.1 INTRODUCTION

This chapter describes the operating principle, the hardware design and the software implementation of the proposed Mixed Voice/Data Transmission System.

In order to implement the MVD System, the operation principle and the functional requirements are first described. The hardware to perform the required functions is then designed using existing LSI chips. The hardware design also takes into consideration INTEL SBC 80 microcomputer architecture as much as possible. For the chosen architecture, the operating system is designed so as to provide these control functions that are necessary for the MVD System operation.

0.2 PRINCIPLE OF SYSTEM OPERATION AND FUNCTION REQUIREMENTS

Before describing the principle of system operation, the communication line protocol of the MVD System shall be specified first. Assume that 16 voice-grade lines are to be configured as MVD channels to provide data packet
transmission service. Also, there are 16 input data lines at the transmitting side and 16 output data lines at the receiving side. Thus, data message coming from one input line of the transmitting side may be switched to one of the 16 destination lines by the built-in software logic so as to provide message switching capability for the leased circuits.

0.2.1 Design of the Protocol

A protocol is a set of mutual agreements on data format and the relative timing of message to achieve data transfer between two or more computers. In the MVD system, the transmitting and receiving microcomputers at both ends of voice-grade lines must communicate each other to handle data packet transmission. The designed protocol is not a complete full-duplex operation; rather it is a "three quarters" full-duplex operation [2,3,4,5]. This means that when one computer is transmitting information packets the other shall be responsible for positive or negative acknowledgement of every information packet received. These acknowledgement messages can be transmitted simultaneously in the reverse line, but no information packet may be transmitted in the reverse line at the same time. If the checksum computation at the receiving side is error-free, then a positive acknowledgement is sent; otherwise, a
negative acknowledgement is sent in which case the transmitting side shall be responsible for retransmission of the requested packet.

Fig. 52 shows the specified protocol for the above operation. Each field is explained below.

**Open Flag Field** (Start of Heading, SOH)

The open flag field is two unique 8-bit sequences, generated at the transmitting side. This flag indicates the beginning of a data packet, provides a synchronizing reference for the positions of the subsequent fields, and triggers the data-packet checking algorithm. The receiving-side CPU monitors the input line for digital signals in an idle state, and becomes active when an open flag is detected.

**Control Field**

The first bit of the control field signifies that this is an information packet (0) or supervisory packet (1). The assignment for other bits is described below.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Information packet</td>
</tr>
<tr>
<td>2</td>
<td>Single copy transmission</td>
</tr>
<tr>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td></td>
<td>Duplicate copy transmission</td>
</tr>
<tr>
<td></td>
<td>= 1</td>
</tr>
<tr>
<td>3</td>
<td>No SRC/DST I. D. byte</td>
</tr>
<tr>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td></td>
<td>with SRC/DST I. D. byte</td>
</tr>
<tr>
<td></td>
<td>= 1</td>
</tr>
<tr>
<td>4</td>
<td>Original packet</td>
</tr>
<tr>
<td></td>
<td>= 0</td>
</tr>
<tr>
<td></td>
<td>Retransmitted packet</td>
</tr>
<tr>
<td></td>
<td>= 1</td>
</tr>
<tr>
<td>OPEN FLAG</td>
<td>CNTRL FIELD</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>(1)</td>
<td>(2)</td>
</tr>
</tbody>
</table>

Legend:

(1) Open Flag (Two bytes unique sequences)
(2) Control Field
(3) Packet Number Field (0-4095)
(4) Route Number Field (Assigned channel)
(5) Source I. D. (Input Channel Number)
(6) Destination I. D. (Output Chnl Number)
(7) Information Field (Null if Supervisory Packet)
(8) Length Field (Less than $3F_X$ if Partial Packet)
(9) Cyclic Redundant Check Field
(10) Close Flag (Two-Byte Unique Sequence)

Figure 52. Communication Line Protocol of MVD System
bits 007

000 No data scrambling
001 One of the 7 data scramblers
or templates used for this packet
111

Bit 8 Spare

Bit 1 = 1 Supervisory packet

Bit 2 = 0 Single copy transmission
= 1 Duplicate copy transmission

Bit 3 = 0 No SRC/DST I. D. byte
= 1 with SRC/DST I. D. byte

Bit 4 = 0 Original packet
= 1 Retransmitted packet

bits 007

000 Received packet acknowledgement
001 Request to send
010 Processor ready (clear to send)
011 Processor not ready
100 Spare
101 Spare
110 Close specified route
111 Break and stop sending

Bit 8 = 0 Positive ACK
1 Negative ACK (request retransmission)

Packet number field

This number recycles every 4096 counts. It provides the packet dispatching sequence at the transmitting side and enables the receiving CPU to perform such a sequence check. If a packet does not comply with this check it will wait in queue until the missing packet arrives.

Route number field
This field denotes the voice-grade channel which the transmitting side chooses to transmit the packet (16 channels maximum).

**SRC/DST Line I.D. Number**

This field denotes one of the 16 data-input line numbers (source I.D.) at the transmitting side, and one of the 16 data-output line numbers (destination I.D.) at the receiving side.

**Information Field**

For the information packet (Bit 1 of Control Field = 0), this field contains the data packet to be transmitted. For maximum throughput, this field is chosen to be 128 bytes or 1024 bits at most. For the supervisory packet (Bit 1 of Control Field = 1), this field is null.

**Length Field**

Bit 1 = 0  Uninterrupted packet
       = 1  Continuation of the previously transmitted partial packet

Bit 2 to 7

= 3Fx  Full packet
< 3Fx  Partial packet

**CRC Field**

This is the checksum field. The binary addition of each bit position from the Control Field to the CRC Field should be 0.

**Close Flag Field** (End of Transmission Block, ETX)

This field is a unique bit pattern indicating the end of the transmission block. This field can be used to
identify the previous two fields (Length and CRC Fields).

6.2.2 **Principle of System Operation**

Fig. 53 shows a simplified block diagram of a multi-channel Mixed Voice/Data Transmission System for "three quarters" full-duplex operation. In this diagram microcomputers are playing an important rule for control of the data packet insertion and extraction.

6.2.2.1 **Dynamic buffering**

For easy description, assume that the transmit-side input data line #3 has strings of digital data in either synchronous or asynchronous form to be transmitted via the MVDT System to the receiving side output data line #7. When the character buffer is filled up in the line multiplexor (MUX), the latter initiates an interrupt to the microcomputer. The line control program (Driver and Interrupt Response Routine) reads in the character into the line buffer which has a total of 136 bytes in length including 128 information bytes and 8 header and tail bytes. For each input data line, there is a buffer pointer, INPTkn, which points to the current dynamic input buffering area. There are also two associated words to indicate the current
Fig. 53. Block Diagram of a Multi-Channel Mixed Voice/Data Transmission System.
character position and the character count. When an input buffer is filled with 128 characters or upon the detection of EOM (End Of Message), the line control program initiates a software interrupt requesting the immediate handling of this input line buffer and a new available buffer for the succeeding input. The Input Interrupt Processing Routine puts the pointer INPTK3 to the back of the Input Queue for packet formation processing. The Buffer Manager gets the next available pointer from the available buffer pointer stack, BUPSTK, and stores it into INPTK3. Now the input line #3 can continuously input its data stream into the line buffer pointed to by the new buffer pointer.

0.2.2.2 Header Attachment and Data Scrambling

Upon its turn, the packet pointer will be removed from the Input Queue by the Packet Processing Program for the base level processing. This base level processing includes appending packet header and data scrambling. The Control Field is 00101000 indicating that it is an information packet, single copy transmission, with SRC/DST I. D., and using number 2 data scrambling template. The packet number is now 001. The Route Field shall be determined by the Route Selection Program, and SRC/DST I. D. byte is then 37. The Destination information can be inputted to the
transmitting-side computer with the specified switching protocol for the leased circuit.

The Data Scrambling Routine will perform the specified logical operation using \#2 data scrambling template. The purpose of data scrambling is to assure the data security during transmission. Now the packet is almost formed except the tail which shall be attached by the Output Processing Program. The pointer of the processed packet is now put to the Transmit Queue for route selection and transmission.

0.2.2.3 Request to Send

Before the transmitting microcomputer sends any packet to the receiving microcomputer, it is necessary to check for the availability and the readiness of the receiving microcomputer. A supervisory packet is designed for such nanosnaking procedure. If a computer has not received acknowledgement from the other and wishes to transmit a packet to the destined computer, it will send a "request to send" supervisory packet in which Bit 1 of the Control Field is set and Bits 5-7 shall be set to 001. The destined computer after receiving such a request shall send a supervisory packet with a "clear to send" acknowledgement with 010 in Bits 5-7. Then upon reception of the
acknowledgement, the transmitting microcomputer now feels confident to send any packet to the receiving microcomputer. Note that the Information Field and the Length Field are both null in the supervisory packet.

0.2.2.4 Route selection and Outputting

Upon its turn, data packet is accessed through its pointer for route selection and outputting. These are done by Route Selection Program and Output Processing Program, respectively. Various route selection strategies have been investigated in Chapter 5 using GPSS simulation to find out the system performance such as the average number of packets and the average resident time of a packet in the system. It is found that the single-queue "most recently released" strategy will achieve maximum throughput, minimum buffering requirement and minimum packet delay [49].

According to the best strategy and the Channel Status Registration Board, the Route Selection Program selects a recently silent voice-channel and assigns this channel number, say 5, to the Route Number Field of the packet header. The Channel Status Registration Board is set to indicate the usage of Channel 5 by data packets.
An "up-to-now" checksum and byte count shall be kept in a dedicated core location and be updated as the Output Line Interrupt Response Routine sends out each byte from the transmit buffer into the character buffer in the multiplexor. At the end of the successful transmission of a full packet, the Output Processing Program attaches Length Field information (SF), and the checksum in the CRC Field. A Close Flag is also attached after the CRC Field.

6.2.2.5 **Voice Arrival and Termination Interrupt Handling**

During the period of packet transmission if an unexpected voice signal arrives, the Voice Detector will generate an output of 1 which in turn feeds to the Transmitting Status Change Detector. Whenever there is a change of status in the transmitting channels, there will be an interrupt signal feeding to the microcomputer CPU. After being interrupted, the Interrupt Service Routine identifies the interrupt source and checks that Channel 5 is now being used by data packets. The Interrupt Service Routine then passes this preemption order to the Output Processing Program which upon receiving the order will perform the wrap-up procedure for Channel 5. This procedure includes attaching the Length field information, checksum and a close flag to the already transmitted partial packet. In order to
avoid too much delay for the real-time voice signal, urgent
handling is required. The processing time involved here is
to attach the 3-byte tail for outputting. The residual
packet length and the starting byte location are recorded.
This residual packet will be queued at the very front of the
queue for the next dispatching. At this time the bit 1 of
Length Field should be set to 1 to indicate that this is a
continuous partial packet of the previously transmitted one.
The Route Selection Program at this time might select the
other recently silent channel, say #8, and replace the
original #5. All other fields of the header shall not be
changed so as to indicate that this is the same packet, same
SRC/DST I. D. and using the same data scrambling template.
This procedure is used to implement the Preemptive-Repeat
scheme of the proposed model.

If the partial-packet transmission is not allowed and
only the full-packet retransmission is permitted, the logic
for this interrupt handling becomes simpler. Since the
previously transmitted incomplete packet is to be discarded,
there is nothing to be done in the wrap-up procedure and the
Length Field is also useless. Only at the beginning of the
packet retransmission, bit 4 of the Control field must be
set to indicate that this is a retransmitted packet, and the
Channel Number field is altered if other recently released
cchannel is used for its transmission. The rest of the
header information shall not be changed. This procedure is used to implement the Preemptive-Resume scheme of the proposed model.

At the end of the talking period of Channel 5, the output of its Voice Detector is 0 which in turn feeds to the transmitting Status Change Detector for comparison to the previous state that is still latched in it. The discrepancy of the comparison between the two consecutive statuses will initiate an interrupt to the microcomputer CPU so that this channel can be linked to the very front of the "most recently released" chain for the next candidate to be selected for data packet dispatching. For a best route selection strategy, the chance of multiple preemptions of a data packet is very unlikely.

6.2.2.0 Packet Holding for Acknowledgement

Eventually, after the completion of the whole data packet transmission, this data packet is moved to a hold area until it is positively acknowledged by the other microcomputer. A large capacity RAM or a secondary storage like a diskette might be used for such a purpose to hold the transmitted data packets since the operation allows a long period of response time (such as one second) for the other
side to acknowledge the reception condition. In this case a file management program must be designed to provide fast access and retrieval to/from these storages.

6.2.2.7 Request for Retransmission of Erroneous Packets

If the checksum computation at the other end is erroneous, then a supervisory packet will be sent from the other microcomputer to request a retransmission of the specified packet. The Control Field of this supervisory packet will be 11111111 and the packet number will be the one required for retransmission. Upon reception of this negative acknowledgement packet, the transmitting microcomputer will retrieve the specified packet by using the specified packet number as a key. The retrieved packet will be linked to the very front of the single queue for outputting. The retransmitted packet will be flagged as such in Bit 4 of the Control Field. On the other hand, if the checksum computation at the other end is correct, a positive-acknowledgement supervisory packet is sent and upon reception of the acknowledgement, the transmitting microcomputer will delete the packet from the holo area.
0.2.2.8 Multiple Copies Retransmission

For the retransmitted packets, in order to reduce the chance of being interrupted again by the arrival of voice signal, two copies of retransmitted packet may be dispatched via two independent channels so that at the receiving side at least one interfere-free copy will be available for a sequence check.

0.2.2.9 Data Escort Carriers

One of the methods to recognize the data signal from the analog signal is to use two data escort carriers. In the experimental design two data escort carriers at 900 Hz and 3200 Hz are generated at the transmitting side to accompany the data signal on the voice-grade line. At the receiving side, a simple recognizer can detect the existence or absence of such carriers to decide that the receiving signal is data or voice and to switch the outgoing path accordingly. The use of two carriers instead of one is to reduce the chance of misinterpretation of voice signals as data when the given voice signal spectrum has some component of either frequency, thereby increasing the system operational reliability.
0.2.2.10 **Voice Signal Delay**

Upon the arrival of voice signals, time allowance should be available for the voice detector to sense its arrival (detector rising time is about 7 msec) and for the transmitting CPU to have sufficient time to handle the voice arrival interrupt and to append the tail (Length, CRC Fields and Close Flag). For the receiving side, time allowance should be available for the recognizer of the data escort carrier to function properly (response time is about 10 msec). Dense analog voice delay of 15 msec should be inserted into both the transmitting side and the receiving side to avoid voice clipping. This delay should be unnoticeable to the talkers since they are comparatively shorter than 300 msec one way transmission delay in the satellite communication.

0.2.2.11 **Data Escort Carrier Recognizing**

Upon arrival of data packet at the receiving side, data escort carrier recognizer shall respond to the data escort carriers and to control a switch which will lead data and voice signals to two different paths. The voice signal should follow the "through traffic" path while the data signal should follow the "exit" path for data escort carrier.
suppressing any data demodulation.

6.2.2.12 Receiving Data Packet Process

After demodulation, received data are transferred into receiving buffer byte-by-byte by the Input Line Interrupt Response Routine. The dynamic buffering scheme is also applied to the receiving buffer, received packet processing and output buffering. Each receiving buffer is pointed to by RVInPn. There are two associated words to indicate the current character position and the received byte count. When the receiving buffer is full with 136 bytes or encounters \texttt{Ei\&} (end of transmission of a packet), a software interrupt will be initiated requesting the immediate handling of this received data packet and a new receiving buffer for the succeeding data stream. The Input Line Interrupt Response Routine puts the pointer in RVInPS in the back of the Receive Packet Processing Queue for base level processing, while the Buffer Manager assigns a new available buffer pointer to RVInPS for succeeding data stream received. Upon its turn, the packet will be retrieved again via its pointer for base level processing. This processing includes checking of Control Field, extracting other information from the header for verifying the checksum field, and the descrambling of data packet information.
For a successfully received packet, the Length Field should be 3F and for an interrupted packet it should be less than 3F. If the checksum is verified then the data packet is descrambled using the template as specified in Bits 5-7 of the Control Field (template number 2 in this example). After descrambling, the Packet Number Field is checked against a packet sequence counter for a consecutive ascending sequence. If the sequence does not comply with the expected normal sequence, this packet is linked in the hold area until the missing packet arrives safely. At this circumstance, the receiving CPU must issue an urgent request for the needed packet using the supervisory packet. For an interrupted partial packet, it is held in the hold area until the remaining portion of the packet arrives. After passing the sequence check and extracting the destination circuit information LSI, the packet waits in the output queue of the destined circuit for delivery. Upon its turn, the packet is retrieved via its pointer by the Output Processing Program and is delivered to the destination circuit (#7 in this example). The Output Interrupt Response Routine is responsible for handling the buffer empty interrupt.

0.2.3 Processor Status Transition

The communication protocol of the MVD System is designed for a "three quarter" full-duplex operation between
two microcomputers. This means both processors are not allowed to transmit information packets simultaneously. However, if one processor is transmitting information packets the other one is allowed to transmit supervisory packets to acknowledge the reception condition of each information packet. Therefore, these two processors are designated as being in a "Master-Slave" relationship in which the Master processor has the right to transmit information packets while the Slave processor only has the responsibility of checking the correctness of the transmitted packets. Fig. 54 shows the processor status transition diagram with respect to the events of the communication protocol.

Assuming that both microcomputers are in an idle state (only idle for the matter of transmitting data packets, but still busy in updating the line activity status), either processor can grasp the right of being Master by sending a supervisory packet signaling a "Request to Send" (RQ2S). This puts the initiating processor (say Processor A) in a state TsHS which waits for the reception of a "Clear to Send" (CL2S) from the other processor (say Processor B). If Processor A does not receive a CL2S signal from Processor B within a pre-determined limit of time (time out situation) or Processor A receives a "Processor Not Ready" signal from Processor B, then Processor A will transfer back to the
Legend:

**Tx** : Transmit
**Ti** : Transmit Information Packets
**Ts** : Transmit Supervisory Packets
**Tr** : Retransmit Information Packets
**RQ2S**: Request to Send
**ACK** : Positive Acknowledgement
**Rx** : Receive
**Ri** : Receive Information Packets
**Rs** : Receive Supervisory Packets
**I** : Idle
**CL2S**: Clear to Send (Processor RDY)
**NAK** : Negative Acknowledgement

Figure 54. Processor Status Transition Diagram
original idle state. If Processor A receives a CL2S signal from Processor B, then it will transfer to state TiKs which allows Processor A to transmit information packets to Processor B. While in state TiKs, any reception of a negative-acknowledgement supervisory packet from Processor B will cause the state to transfer to TrKs which allows processor A to retransmit the requested packet and to continuously receive supervisory packets. During transmitting or retransmitting information packets if Processor A receives a "break" signal from Processor B, it will cause transfer of the state to a Slave state in which Processor A is only allowed to transmit supervisory packets while processor B can transmit information packets. If processor A wishes to regain the right of being Master, it will transmit a break supervisory packet.

This protocol is different from the conventional nanoshaking protocol in which each processor will change its state upon immediate reception of a special designated control character, such as SIx, DLE, ETx, CRC, etc. and each transmitted packet or message must be acknowledged upon reception so that the next packet or message can be sent from the transmitting processor. Such a protocol can only be used in a communication network whose transmission delay is negligible. However, the MVD protocol that provides "Block Acknowledgement" feature is intended to be used in a
communication network in which considerably long period of transmission delay exists, such as in the satellite communication network.

0.3 SUMMARY OF FUNCTIONAL REQUIREMENTS OF THE MVD SYSTEM

From the previous description of the operation principle of the MVD System, the following hardware components are required:

Transmitting Side
(1) A microcomputer with PROM or ROM, RAM, a priority interrupt chip, and an USART (Universal Synchronous & Asynchronous Receive and Transmit) chip;
(2) Voice Detector and Channel Status Change Detector;
(3) Analog Delay Device and Analog Switch;
(4) Modulator and its interface;
(5) Data Escort Carrier Generator;
(6) Mixer; and
(7) Line Multiplexor.

Receiving Side
(1) A microcomputer with PROM or ROM, RAM, a priority interrupt chip and an USART chip;
(2) Voice/Data Recognizer and Line Status Change Detector;
(3) Analog Delay Device and Analog Switch;
(4) Demodulator and its interface;
(5) Data Escort Carrier Suppressor; and
(6) Line Multiplexer.

And the following software components are required:

(1) Monitor (including Scheduler, Watchdog Timer);

(2) Line Control Programs:
   - Input Buffer Full Interrupt Handler;
   - Transmit Ready Interrupt Handler;
   - Receive Ready Interrupt Handler;
   - Output Buffer Empty Interrupt Handler; and
   - Buffer Manager.

(3) File Manager.

**Application Programs**

(1) Transmit Packet Processing (Formation, Scrambling) program;

(2) Transmit Route Selection Program;

(3) Transmit Output Processing Program;

(4) Receive Packet Processing Program (Disassembly, Sequencing, and Descrambling);

(5) Receive Packet Analysis Program;

(6) Receive Output Processing Program;

(7) Voice Arrival Interrupt Handler (Tx);

(8) Voice Termination Interrupt Handler (Rx); and

(9) Line Status Change Interrupt Handler (Rx).

Some hardware and software components will be described in the following sections.
0.4 DETAILD DESIGN ON THE REQUIRED HARDWARE COMPONENTS

0.4.1 A Microcomputer

The heart of the MVD System is a microcomputer with its associated PROM or ROM to store the control programs and its RAM for I/O buffering. Fig. 55 shows the block diagram of the MVD controller. Using INTEL standard chips, the controller configuration is shown in Fig. 56 in which 8080A is used as the CPU, 8224 as the clock generator and 8233 as the system controller to/from the CPU set. For simplicity in block diagram, one single data input port and 6 Transmit/Receive Ports are shown, the former using 8251 as a serial I/O Interface while the later using 8255 as a parallel I/O Interface. To accept external interrupts, one 8214 Priority Interrupt chip is used. Types of interrupts include the transmitting status change interrupt, receiving status change interrupt, MODEM Tx/Rx ready interrupt, data-input buffer full interrupt, and data-output buffer full interrupt. The 8212 I/O port is to incorporated with the 8214 Priority Interrupt chip to realize an 8-level priority interrupt. The System Bus Interface and External Bus Interface are also included.

The CPU set is the control center of the microcomputer. It performs all system processing functions and provides a stable timing reference for all other circuit in the system.
Fig. 55. Block Diagram of the MVD Controller.
Fig. 56. Configuration Diagram of MVD Controller and Its Interface.
The CPU generates all the addresses and control signals necessary to access memory and I/O ports. This CPU set is capable of fetching and executing any of the 8080's seventy-eight instructions and responding to interrupt requests from the Priority Interrupt Controller.

The System Bus Interface includes circuitry which gates Interrupt request, HOLD request, Ready inputs and the system reset input to appropriate pins of the CPU set. It also includes two bi-directional bus drivers which drive the memory data bus on the 80/10. Six 8226 devices drive the external system data and address bus.

The Random Access Memory (RAM) provides $1024 \times 8$ bits of onboard read/write storage. Its control logic generates the necessary acknowledgement and memory address decoding. Its capacity may be easily expanded up to 65 K bytes.

The Read Only Memory (ROM/PROM) provides $4096 \times 8$ bits of storage for the system control programs. Its control logic includes the necessary acknowledgement and memory address decoding circuitry.

The Serial I/O Interface, using INTEL 8251 USART device, provides a bi-directional serial data communication channels which can be programmed to operate with most
current serial data transmission protocols. Synchronous or Asynchronous, baud rate, character length, number of stop bits and the choice of even, odd or no parity are all program selectable. Each 8251 interfaces with a modem for both transmit and receive circuits. The parallel I/O Interface, using two INTEL 8255 programmable Peripheral Interface devices, provides 48 signal lines for the transfer and control of data to/from peripheral devices. Each line has an output latch/ouffer. Half of these lines are used for inputting the status of the Voice Detector and Data Escort Carrier Recognizer, and half of them are used for the Transmitting Status Change Detector and Receiving Status Change Detector in the experimental MV D System.

The Priority Interrupt Controller, using the INTEL 8214 Priority Interrupt Control Unit and an 8212 8-bit I/O port, provides eight-level priority structure for the 808 microcomputer. The priority levels of the MV D System are assigned as:

<table>
<thead>
<tr>
<th>Priority Levels</th>
<th>Interrupt Processing Levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Base Level Processing</td>
</tr>
<tr>
<td>1</td>
<td>MODEM Tx Ready,</td>
</tr>
<tr>
<td></td>
<td>Output buffer Empty Interrupt</td>
</tr>
<tr>
<td>2</td>
<td>MODEM Rx Ready,</td>
</tr>
<tr>
<td></td>
<td>Input buffer Full Interrupt</td>
</tr>
<tr>
<td>3</td>
<td>Receive Line Status Change Interrupt</td>
</tr>
<tr>
<td>4</td>
<td>Voice Termination Interrupt</td>
</tr>
<tr>
<td>5</td>
<td>Voice Arrival Interrupt</td>
</tr>
<tr>
<td>6</td>
<td>Clock Interrupt</td>
</tr>
<tr>
<td>7</td>
<td>Reset</td>
</tr>
</tbody>
</table>
0.4.2 **Voice Detector**

The function of the Voice Detector is to detect the absence/existence of voice signals in each transmitting circuit so that the microcomputer can decide when to insert/suspend the data packet transmission.

because of the AC characteristic of the voice signal, the detector circuitry basically is a fast AC to DC converter plus a voltage comparator as shown in Fig. 57. In this figure, C16 is an operational amplifier with the gain adjustable by VH1. C2A and C2b operate together as a full-wave rectifier and averaging filter. For negative signals, C2b functions as a simple unit gain inverter, and for positive signal C2B operates as a summing inverter both given positive going voltage. C1 and R9 with C2E form a low pass filter with a time constant of approximately 7 msec constituting the rising and falling times of the Voice Detector. C3 is a voltage comparator to reference to a voltage threshold. If the input filtered voltage is lower than the threshold, the output will be "1" (about +5 V); otherwise, the output will be "0" (below 0.4 V).

The Voice Detector so designed is very sensitive to the short pauses of taking and impulse noise, even as short as 10 msec, that will change the output state of the Voice
Fig. 57. Block Diagram of Voice Detector.

Fig. 58. Improved Voice Detector.
Detector. However, these pauses are too short to be useful in the MVD system for inserting data packets. In order to overcome this difficulty, additional delay circuitry must be added. Fig. 58 shows the improved Voice Detector. C4A is a retriggeable one-shot (monostable multi-vibrator). Its time width is

$$t_w = 0.28 R_{ext} C_{ext} \left(1 + \frac{0.7}{R_{ext}} \right)$$

The principle of the circuit operation is as follows:

At the instant of voice termination, the output of C3 changes state from 0 to 1. The positive-going edge triggers the one-shot to cause the output \( \overline{Q} \) to change to 0, and hence maintains the output of C5. After 7 second delay, the \( \overline{Q} \) 's output rises to 1, and because the other input of C5 is also 1, the output of C5 changes from 1 to 0.

However, at the instant of voice arrival, the falling edge (1 to 0) cannot trigger the one-shot. The delay device will not function; hence C5 changes the output state instantly from 0 to 1. Thus, the voice detector is still sensitive to the voice arrival.

Various designs Voice Detector have been proposed in IAS1 and Vocoder Systems [10]. The main problem involved was to select an appropriate detection threshold so that
this threshold would provide sufficient fast detection on voice signals but would also prevent noise triggering as much as possible.

0.4.3 Transmitting Status Change Detector

The purpose of the Transmitting Status Change Detector is to detect the status change of the transmitting circuit so that the appropriate interrupts can be initiated to the microcomputer CPU for starting/suspending the data transmission. Fig. 59 shows the implemented circuitry. It mainly consists of the INIÉL parallel I/O interface 8255 and an Exclusive-Or gate.

The principle of the circuit operation is as follows. Initially the state of line FBO is the same as line PAU. This may be done by inputting PAU to the microcomputer and then outputting to PBU which latches there. The output of the Exclusive-Or gate then becomes 0. When any change in the output of the Voice Detector occurs, the state of PAU is different from that of the PBU. The Exclusive-Or gate hence outputs a 1 state. After inverting, this signal then may be used to start the interrupt handling routine or data transmission routine. After the interrupt has been honored, the state of PAU is then outputted by the microcomputer to the PBU latch.
Fig. 59. Block Diagram of Transmitting Status Change Detector.

Fig. 60. Circuit Diagram of Data Escort Carrier Generator.
0.4.4 Data Escort Carrier Generator

The purpose of using the data escort carrier generator is to provide an easy means for the receiving side to recognize that the current signal is a data signal or voice signal. This is the simplest way to implement the data escort carrier generator in the experimental system, although various methods have been proposed in Section 2.3.5 of Chapter 2. Also, a "signal-pattern recognition method" according to the statistical characteristics of voice or data signals, which is feasible and more effective for data transmission, has been originally proposed [58].

The data escort carriers are inband "dual tones" selected to be 9000 Hz and 32000 Hz at the band-edges of the voice-grade circuit. The reason for using two frequencies is to reduce the chance of misinterpretation between voice and data signals so as to improve the system reliability. Fig. 60 shows the implemented circuitry consisting of two Wien-bridge oscillators and an analog adder. Proper selection or $R_1$ and $C_1$ values would achieve the desired frequencies. Diodes D1 and D2 are used for minimizing the output distortion.
0.4.5 Modulator

Various modulation schemes can be used to transmit digital information over the line. To achieve a high data rate, four-phase PSK modulation was originally proposed. However, in the experimental system, a simple Frequency-Shift-Keying (FSK) technique was used for data modulation, and 2000 Hz and 2600 Hz are selected to represent binary 0 and 1, respectively.

Fig. 61 shows the implemented circuitry of the modulator. Basically it is a Wien-bridge oscillator, but with a variable feedback resistance. The oscillating frequency is given by

\[ f_0 = \frac{1}{2\pi R_1 C_1} \]

If the value of \( R_1 \) is time-dependent, then so is \( f \). When a "0" (below 0.4 V) is outputted from the Tx line of 8231, the output of the voltage converter C19A will be closed to the negative power supply voltage \( V_- \). The negative voltage opens both switches. \( R_1' \) and \( R_2' \) are now separated from the oscillator. The frequency is then given by:

\[ f_0 = \frac{1}{2\pi R_1 C_1} = \frac{1}{2\pi \times 4.3 \times 0.02 \mu F} \approx 2000 \text{ Hz} \]

When a "1" appears on the Tx line, the output voltage of C19A will be closed to positive power supply \( V_+ \), which
Fig. 61. Circuit Diagram of the FSK Modulator.

Fig. 62. Circuit Diagram of the Mixer.
closes both switches. $R1'$ and $R2'$ are connected parallely with $R1$ and $R2$, respectively, to form the oscillating circuit. The frequency is then given by:

$$f_0 = \frac{1}{2\pi(R_1'/R_1)C_1} = \frac{1}{2\pi(15K //4.3K) \times 0.02\mu F}$$

$$= 2600 \text{ Hz}$$

The modulated output then passes through a buffer $C22A$ to a mixer for further processing.

0.4.0 **Mixer**

The purpose of a mixer is to mix the data escort carriers with the modulated data signal. It is implemented with an analog adder which sums up the above three signals. Fig. 62 shows an operational amplifier for the said purpose.

0.4.7 **Data Escort Carrier Recognizer**

The purpose of this recognizer is to distinguish voice signals from data signals by detecting the existence of the data escort carriers that was used to identify the status of the voice-grade line in the experimental system. The output of this recognizer will control a switch (SW4 in Fig. 55) to let the mixed data signals pass through the Data Escort Carrier Suppressor, and then to a Demodulator.
Two tone-decoders are used and the outputs of these two decoders are then ANDed to generate the data recognition signal. Fig. 63 shows such a configuration.

National Semiconductor (NS) LM507 is used as the tone decoder. It outputs a "0" when the signal is in the desired frequency bandwidth. Fig. 64 shows the block diagram of the tone decoder in which C1 and R1 are used for setting the center frequency of the current controlled oscillator (CCO).

The center frequency is given by

\[ f_0 = \frac{1}{2\pi R_1 C_1} \]

The circuit operation can be described as follows. In Fig. 63 when the input signal contains the expected frequency, the phase-locked loop will lock on the frequency. The two inputs of the quadrature-phase detector hence have the same frequency and phase. After comparing with the referenced voltage, the AMP2 outputs a "0". If the input frequency is out of the range of the desired bandwidth, the phase-locked loop cannot lock the input frequency and the average output of the quadrature phase detector causes AMP2 to output a "1".
Fig. 63. Circuit Diagram of the Data Escort Carrier Recognizer.

Fig. 64. Block Diagram of the Tone Decoder LM567.
0.4.5 Receiving Status Change Detector

The purpose of using the Receiving Status Change Detector is to detect the status of the receiving circuit by sensing the output change of the Data Escort Carrier Recognizer to identify the status of the voice-grade line. Its output is used to interrupt the receiving microcomputer for starting/suspending the data reception routine. Its circuit design and the operation principle are the same as those of the Transmitting Status Change Detector (see Fig. 65).

0.4.9 Data Escort Carrier Suppressor

The purpose of the Data Escort Carrier Suppressor is to suppress the data escort carriers that are mixed with data signals during transmission, and let the modulated data signals remain in the circuit for further demodulation.

The implemented circuitry consists of two Twin-Tee notch filters, one for tone 1 and the other for tone 2. Fig. 66 shows the Twin-Tee notch filter circuits. The notch frequency is given by \( f_0 = \frac{1}{2 \cdot R_1 C_1} \).
To Microcomputer Internal Bus

Parallel I/O Interface
8255

Latch

Exclusive-OR

Interrupt

Fig. 65. Block Diagram of the Receiving Status Change Detector.

Fig. 66. Circuit Diagram of the Data Escort Carrier Suppressor.
0.4.1 ø Demodulator

The demodulator consists of a phase-locked loop and a voltage comparator, as shown in Fig. 67. It operates as follows: In Fig. 68 when there is no input signal, the output voltage of the low pass filter controls the voltage controlled oscillator to oscillate at a center frequency \( F_0 \) (=1.2/4\( \alpha_1 C_1 \)). When the input frequency is lower than the center frequency \( f_0 \), the output of the phase detector increases the average voltage at the low pass filter. This higher voltage then controls the Vco to oscillate at a low frequency until locking. When the input frequency is higher than the center frequency, the opposite effect occurs. Therefore, the output of the low pass filter is linearly increasing with the decreasing of the input frequency. As the modulation scheme in the experimental system is FSK with 260 Hz representing "0" and 266.5 Hz representing "1", the center frequency \( f_0 \) of 029 must be chosen to be in the middle of the two frequencies, i.e. 263 Hz.

The demodulated voltage then passes through a three-stage R-C ladder filter to remove the carrier component. This output is then available for the receiver Data (HxD) of the 8251.
Fig. 67. Circuit Diagram of the FSK Demodulator.

Fig. 68. Block Diagram of the Phase-Locked Loop.
0.4.11 Modem Interface

The modem interface includes an INTEL 8251 (USAR'), a baud rate generator and a timing control circuitry, as shown in Fig. 69.

The function of the USAR' (Universal Synchronous/Asynchronous Receiver/Transmitter) is to perform the parallel-series conversion and to insert or delete bits or characters that are functionally unique to the communication protocol, e.g., the Close Flag and the Open Flag. Its block diagram is shown in Fig. 70.

Prior to starting data transmission or reception, the 8251 must be loaded with a set of control words generated by the CPU. These control words completely define the function of the 8251 and must immediately follow a RESR operation. These control words can be divided into two groups, as shown in Fig. 71.

(1) Mode Instructions;
(2) Command Instructions.

The mode instruction defines the general characteristics of the 8251. These characteristics include: synchronous or asynchronous, character length, parity, SYNC detect external or internal, number of stop bits, baud rate
Fig. 69. Block Diagram of the Modem Interface.
Fig. 70. Block Diagram of INTEL 8251 USART.

<table>
<thead>
<tr>
<th>RESET</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>C/5 = 1</td>
<td>Mode Instruction</td>
</tr>
<tr>
<td>C/5 = 1</td>
<td>SYNC Character 1</td>
</tr>
<tr>
<td>C/5 = 1</td>
<td>SYNC Character 2</td>
</tr>
<tr>
<td>C/5 = 1</td>
<td>Command Instruction</td>
</tr>
<tr>
<td>C/5 = 0</td>
<td>Data</td>
</tr>
<tr>
<td>C/5 = 1</td>
<td>Command Instruction</td>
</tr>
<tr>
<td>C/5 = 0</td>
<td>Data</td>
</tr>
<tr>
<td>C/5 = 1</td>
<td>Command Instruction</td>
</tr>
</tbody>
</table>

Fig. 71. INTEL 8251 Data Block.
factor, etc.

The command instructions control the actual operation of the selected format. These functions include Enable Transmit/Receive, Error Reset, Modem Control, Enter Hunt Mode and Internal Reset.

In data transmission systems, it is often necessary to examine the status of the active device to ascertain if Receiver/Transmitter Ready or other error conditions have occurred that require the processor's attention. The 8251 allows the programmer to read the status of the device at any time during the functional operation.

Asynchronous Mode Transmission and Reception

Once programmed, the 8251 is ready to perform its transmission functions. The Transmitter Ready (TxRdy) output is raised "high" to signal the CPU that the 8251 is ready to receive a character. This output is reset automatically when the CPU writes a character into the 8251. On the other hand, the 8251 receives serial data from the Modem and upon receiving an entire character, the Receiver Ready (RxRdy) output is raised high to signal the CPU that the 8251 has a complete character ready for CPU to fetch. RxRdy is reset automatically upon CPU's read operation. The 8251 cannot begin transmission until the TxE (Transmitter Enable) bit is set in the Command Instruction and it has
received a "clear to send" (CTs) input. The lxd
(transmitter Data) output will be held in the marking state
upon reset.

**Synchronous Mode Transmission**

The TxD output is continuously high until the CPU sends
its first character to the d251 which usually is a SYNC
character. When C15 line goes low, the first character is
serially transmitted out. All characters are shifted out at
the same rate as the lxc (transmitter Clock). Once
transmission has started, the data stream at lxd output must
continue at the lxc rate. If the CPU does not provide the
d251 with a character before the d251 becomes empty, the
SYNC characters will be automatically inserted into the Tx
data stream.

**Synchronous Mode Reception**

If the internal SYNC mode has been programmed, the
receiver starts in a HUNI mode. Data on the rxd pin is then
sampled in on the rising edge of rxc. The content of the
receiver buffer is continuously compared with the SYNC
characters until a match occurs. When both SYNC characters
have been detected, the USART ends the HUNI mode and is in
character synchronization.

In the MVU System, it has been proposed that
synchronous data transmission scheme should be used to
achieve higher data throughput than that can be obtained
from the asynchronous scheme. The bi-sync protocol of the uSAARI can be used for the Open Flag and the Close Flag of the MVD protocol.

**Baud Rate Generator for 8251**

At the left-hand upper corner of Fig. 69, there is a Baud Rate Generator to provide clock pulses for both TxC and RxC. In this experimental design, they were chosen to be 1200 Hz and were derived from the 80/10 bus clock (BCLK 9.261 MHz). C14 and C15 are both counter devices. Since C15 is a CMOS device, a driver must be used for its CMOS-TTL compatibility.

**Operation of Modem Interface**

Since the Data Escort Carrier must be added to the modulated data signal before transmission and the Data Escort Carrier recognizer cannot respond immediately but with a delay of about 10 msec, the Data Escort Carrier must be transmitted in advance for a leading time greater than the response time of the recognizer so that the data front will not get lost.

At the right-hand upper corner of Fig. 69, C46 (one-shot) and C5 (NAND gate) constitute a delay device for the Data Escort Carriers. Its circuit operation is the same as that of the Improved Voice Detector.
When the CPU starts sending data to the 8251, it must first send a command to set the Request to Send (RTS) line "low". This signal triggers the one-shot C4d for a time duration. The "low" RTS signal is also inputted to a voltage converter, closing the S.W.A. switch. The Data Escort Carrier hence passes through the S.W.A. switch to the mixer. Because CIS line is still "high", TxD line stays at "marking" state. Hence the second input of the Mixer has a modulated "1". After the delay period, the "low" output of C5 reaches the CIS. After this moment, serial data are shifted out from TxD. The first character to be transmitted must be a synchronous character. The first bit of the SYNC character should be "0". On receiving, the falling edge of this "0" bit from the demodulator triggers a one-shot C53 (set time width equal to 1 msec). The output of this one-shot then reset the counter C32 for the Receiver Clock (RxC) to be synchronous with the Receiver Data (RxD). Because the 8251 samples RXD on the rising edge of RxC and resets the counter C32 on the falling edge, it makes sure that the sampling will occur at the middle of each bit. Thus, the probability of erroneous data sampling decreases to a minimum.
0.4.12 **Switching Devices**

CMOS CD4066 analog switches were used for all the switches in the MVU Controller. Fig. 72 shows the functional diagram of the CD4066 quan-bilateral switch.

To use the switch for passing AC signals, Vcc and Vss must be supplied with the positive and negative voltage, respectively. If the control line voltage is near the positive power supply voltage the switch will be closed. Under this condition, the ON-resistance will decrease to several hundred ohms. On the other hand, if the control line voltage is supplied with a negative voltage the switch will be opened. Under this condition, the OFF-resistance is in the order of megaohms.

A voltage converter is needed if the switch is to be controlled with an TTL circuit. Since a "0" output of TTL logic is below 0.4 V, it needs to be converted to -5 V negative power supply voltage. This voltage converter is shown in Fig. 73. Basically, it consists of an operational amplifier with no feedback. If Vi = "1" (above 2.4 V), then Vi is greater than Vref and the output Vθ is thus near the positive supply voltage. On the other hand, if Vi = "0", then the output of Vθ is close to negative supply voltage.
Fig. 72. CD4066 Analog Switch.
0.4.13 Analog Delay Device

In the MVD system as shown in Fig. 53, delay circuitry \( D_t \) is needed to be inserted in the voice signal path to compensate for the rising time of Voice Detector (7 milliseconds) plus the response time of \( St \) (1 microsecond). Delay circuitry \( D_r \) is needed in the mixed voice and data path at the receiving side to compensate for the response time of Data Escort Carrier Recognizer (10 msec) plus the action time of \( Sr \) (1 microsecond). Hence a 15 msec time delay circuitry should be satisfactory for these two purposes. Using conventional means it is difficult to find an economical delay device in msec order; however, with recent mass production of analog shift registers, such as Matsushita Matsusnita \( \text{Matsusnita} \), it is feasible to obtain such a delay device with controllable delay time.

The principle of the circuit operation is shown in Fig. 74. Two out-of-phase clocks \( \phi_1 \) and \( \phi_2 \) are used to control the shift registers. While the first clock is high, the "odd" cells are dumped into the next consecutive "even" cells. When the second clock is high, the even cells are dumped into the next consecutive "odd" cells. In this manner, individual charges are transferred along the line one stage at a time.
Fig. 73. Circuit Diagram of the Voltage Converter.

Fig. 74. Circuit Diagram of the Analog Shift Register.
Fig. 74 shows a schematic representation of only four typical stages of the MN3001 analog shift register. Each MN3001 IC contains two 512-stage shift registers.

In the MVD controller, one MN3001 IC chip is needed (2 x 512 stages). The clock was chosen to be 35 KHz and the delay time became 15 msec.

\[
T_{\text{delay}} = \frac{(512 \times 1)}{(35 \times 10^3)} = 15 \text{ msec.}
\]

0.5 SOFTWARE IMPLEMENTATION OF MVD SYSTEM

0.5.1 Operating System of the MVD System

Fig. 75 illustrates the operating system of the MVD System. It consists of three categories of programs:

(1) System Monitor;
(2) Application Programs; and
(3) Line Control Programs.

The system monitor includes a program scheduler and a watchdog timer. The former is used to schedule the sequence of executing all base level programs, while the latter is used to monitor the execution of a program. In addition to the system monitor, there should be a file management system that provides access methods for the data packet storage and retrieval to/from a floppy disk, and all kinds of system
Fig. 75. The Operating System of the Mixed Voice/Data (MVD) Transmission System.
data and tables that provide interface to all the system software modules.

The application programs are divided into two classes: Base Level Programs and Interrupt Driven Programs.

For transmitting software, the base level programs consist of:
(1) Output Processing Program,
(2) Route Selection Program, and
(3) Packet Processing Program;
and the interrupt driven programs consist of:
(1) Voice Arrival Interrupt Handler, and
(2) Voice Termination Interrupt Handler.

For receiving software, the base level programs consist of:
(1) Packet Processing Program,
(2) Packet Analysis Program, and
(3) Output Processing Program;
and the interrupt driven program consists of:
(1) Line Status Change Interrupt Handler.

The base level programs are running under the control of the scheduler while the interrupt driven programs interrupt the normal execution of the base level programs in different levels of priorities. The priority level of Voice Arrival Interrupt should be higher than the Voice termination Interrupt.
The Line Control Programs handle all the character input/output of all the communication lines. For transmitting software there are:

1. Transmit Ready Interrupt Handler, and
2. Input Buffer Full Interrupt Handler;

and for receiving software there are:

1. Receive Ready Interrupt Handler, and
2. Output Buffer Empty Interrupt Handler.

The Buffer Manager is incorporated to achieve dynamic buffering capability and centralized control of buffers.

0.5.2 Algorithms to Implement Route Selection Strategies

0.5.2.1 Multi-Queue Minimum Queue Content (MINQ)

The algorithm to implement the minimum queue content strategy is to construct a queue content table that consists of queue content, upward and downward pointers as shown in the following table:
<table>
<thead>
<tr>
<th>Queue Station (i)</th>
<th>Current Content (QC)</th>
<th>Upward Pointer (UP)</th>
<th>Downward Pointer (DP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>n = 5</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
</tbody>
</table>

Where LEASI is a pointer pointing to a queue that has a minimum length of queue entries.

To add a queue content or to remove one, the queue content table must be updated to reflect the new upward or downward sequence. To add a content to a queue, the queue content must be incremented, and the doubly-linked list must be traced upward until the current content of this queue station is less than or equal to the queue content of other queue stations in the doubly-linked list. To remove a content from a queue, the queue content must be decremented, and the doubly-linked list must be traced downward until the current content of this queue station is greater than or equal to that of other queue stations in the doubly-linked list. For example, if an entry is added to the minimum queue station, say ith station, and if j represents the other queue stations, QC represents queue content, UP
represents upward pointers and UP represents downward pointers, the following algorithm should be applied to update the doubly-linked list:

\[(1) \quad i = \text{LEAST};
(2) \quad QC(i) = QC(i) + 1;
(3) \quad j = \text{UP}(i);
(4) \quad \text{If } j \neq \emptyset \text{ Then}
\]

\[(5) \quad \text{If } QC(i) \leq QC(j) \text{ Then}
(6) \quad \text{If } j \neq \text{UP}(i) \text{ Then}
\]

\[(7) \quad \text{LEAST} = \text{UP}(i); \quad UP(i) = j; \quad UP(k) = i;
(8) \quad DP(i) = UP(j); \quad DP(j) = i; \quad DP(\text{LEAST}) = \emptyset;
\]

\[(9) \quad \text{Otherwise halt};
\]

\[(10) \quad \text{Otherwise } k = j; \quad j = \text{UP}(j); \text{ go to (4)}
\]

\[(11) \quad \text{Go to (4)};
\]

If an entry in the input queue station is removed from the queue, using the same notion above, the following algorithm should be applied to update the doubly linked list:

\[
\text{Do while } QC(i) \neq \emptyset
\]

\[(1) \quad QC(i) = QC(i) - 1;
(2) \quad j = DP(i);
(3) \quad \text{If } j \neq \emptyset \text{ Then}
\]

\[(4) \quad \text{If } QC(i) \geq QC(j) \text{ Then}
(5) \quad \text{If } j \neq \text{DP}(i) \text{ Then}
\]

\[(6) \quad \text{/* update up-pointers}
\]

\[(7) \quad \text{TEMP} = \text{UP}(i); \quad UP(i) = \text{UP}(j);
(8) \quad UP(i) = i; \quad UP(\text{DP}(i)) = \text{TEMP};
\]

\[(9) \quad \text{/* update downwarf pointers}
\]

\[(10) \quad \text{DP}(\text{UP}(i)) = i; \quad DP(\text{TEMP}) = \text{DIEMP};
\]

\[(11) \quad \text{Otherwise halt};
\]

\[(12) \quad \text{Otherwise } j = \text{DP}(j); \text{ go to (3)};
\]

\[(13) \quad \text{Otherwise halt.}
\]
0.5.2.2 Multi-Queue Less-Than-Threshold (LTT)

The algorithm to implement the LTT is to construct a queue content table with columns representing queue content (QC), LTT flag, and a link pointer (LP) as shown in the following table:

<table>
<thead>
<tr>
<th>Queue Station Number(i)</th>
<th>Queue Content (QC)</th>
<th>LTT Flag (LTT)</th>
<th>Link pointer (LP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>φ</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>φ</td>
<td>φ</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1</td>
<td>φ</td>
</tr>
<tr>
<td>N = 4</td>
<td>1</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>

LTT: 4

To add an entry to a queue station, the LTT (Less-Than-Threshold) pointer is used to find a queue station whose content is less than the preset threshold. The old queue content (QC) of that station is incremented and then compared to the threshold. If the incremented queue content is greater than the threshold, this queue station's number is removed out of the LTT link. The LTT flag is then reset to φ. To remove an entry from any queue station (say ith station), the old queue content is decremented and the LTT flag of that queue station is checked. If the LTT flag is reset, the queue content of this queue station is compared to the threshold. If the decremented queue content is less than or equal to the
threshold, this queue station's number is linked to the LTT link list. The LTT flag is then set to 1. The following algorithm should be applied to update the queue content table.

Algorithm for adding an entry to a queue station:

1. \( i = \text{LIT} \)
2. If \( i \neq 0 \) then
3. \( \text{QC}(i) = \text{QC}(i) + 1 \)
4. If \( \text{QC}(i) > \text{THRESH} \) Then
   - \( \text{LIT} = \text{LP}(i) \)
   - \( \text{LP}(i) = \text{NLIT} \)
   - \( \text{NLIT} = i \)
5. Otherwise halt;
6. Otherwise \( i = \text{NLIT} \);
7. \( \text{QC}(i) = \text{QC}(i) + 1 \); halt;

Algorithm for removing an entry from a queue station (say ith)

1. Do while \( \text{QC}(i) \neq 0 \);
2. \( \text{QC}(i) = \text{QC}(i) - 1 \);
3. Do while \( \text{LIT}(i) = 0 \);
4. If \( \text{QC}(i) \leq \text{THRESH} \) Then
   - \( \text{LP}(i) = \text{LIT} \)
   - \( \text{LIT} = i \); halt;

4.5.2.3 Single-Queue Most Recently Released (MRH)

The algorithm to implement the MRH strategy is to construct an MRH table containing columns of voice/data usage flags, packet dispatch count and a link list pointer linking to the next MRH entry. This table is shown as follows:
The algorithm to add a recently released channel (i) is as follows:

1. FP(i) = Mkk; BP(mrr) = i; VO(i) = Ø;
2. Mkk = i; BP(i) = Ø;

And the algorithm to update the table when dispatching a data packet via Mkk = i channel is as follows:

1. DU(i) = 1; DC(i) = DC(i) + 1;
2. IF DC(i) > THRSHD then
   BP(i) = -1; Mkk = FP(i); FP(i) = -1;
3. Otherwise halt;

If the voice signal arrives to the ithn channel, the following algorithm should be applied:

1. VO(i) = 1; DU(i) = Ø; DC(i) = Ø;
2. IF i ≠ Mkk Then
   FP(BP(i)) = FP(i); FP(i) = -1;
   BP(FP(i)) = BP(i); BP(i) = -1;
3. Otherwise
   Mkk = FP(i); FP(i) = -1;
   BP(FP(i)) = Ø; BP(i) = -1;
SUMMARY

This chapter describes the operating principle of the MVU System, the detail hardware implementation and the software structure of the MVU System. The hardware implementation is based on the INTEL 8080A microcomputer architecture and uses existing functional chips for interface components. The hardware cost of such a controller depends on the number of channels configured to the system and the data speed to be achieved. For a small size configuration with few input and output data lines, the cost is estimated around US$500 excluding the modems. And the software development is estimated for two person-years. However, the actual cost of software could be reduced to negligible amount if the MVU System is to be produced in large quantities. Therefore, the ultimate cost of the MVU System is dominated by the cost of hardware, and this cost is gradually reduced with the advance of the microprocessor technology.
7.1 SUMMARY OF WORK

The Mixed Voice/Data (MVD) System was presented with respect to the following aspects: motivation, approach, analysis, simulation, and hardware and software implementation. In each aspect, detailed exploration was made and conclusions were drawn. These aspects can be summarized as follows.

The main motivation of this research is to find an inexpensive alternative for the communication links of computer networking and at the same time to solve the inefficient channel utilization of present analog voice-grade channels. The approach is to insert data in packets into the silent periods of analog voice conversation through the use of microcomputers which control the data packet insertions, repeat or resume transmission at the transmitting side, and perform data packet ordering and delivery at the receiving side.

In the analysis of this approach, the main effort was to find a closed form representation of the system average
queue size and the average waiting time for the data packet with respect to the utilization of silent periods (or called data traffic intensity). In the simulation, digital representations of the same variables were provided using the GPSS simulation language. In these simulation studies, considerations were given to actual statistics of talking- and silent-periods, to voice arrival patterns, and to channel selection strategies that are mathematically non-representable. Together, the analysis and simulation results were very useful for determining the minimum buffer requirements, minimum packet delay, optimum packet size and maximum throughput of data packets.

In the hardware design for the implementation, besides the microcomputer architecture and its associated peripherals, hardware-logic circuits were designed to interface between the microcomputer and the existing voice-grade telephone network. In the software implementation, the design of an operating system for a "three quarters" full-duplex microcomputer was provided, emphasizing the modular design of the software. The software implementation of the algorithms for various channel selection strategies was also presented.
During this microcomputer era, use of a microcomputer system is the most cost-effective way to implement the proposed MVD System. The multi-level interrupt architecture of such a microcomputer system especially helps inherent characteristics of the MVD System, in which the arrival and termination of analog voice signals in each circuit must interrupt the CPU to initiate the immediate processing of related actions. The Read-Only-Memory (ROM) also provides an inexpensive way of storing the control program of the MVD System so that they become the firmware of the system. The semiconductor RAM can be used for buffers and working areas that can be adjusted depending on the system size (number of configured channels) and incoming data speed. Such a system, if mass-produced, would be fairly reasonable in cost and would have prevailing applications in the existing long-distance telecommunication network. The conspicuous savings of telecommunication bandwidth and the additional provision of data service would soon greatly outweigh the relatively small investment.

Fully digital transmission systems for voice signals have come into being (such as PCM, TDM or SCPC systems), but they still require very wide bandwidths to carry undistorted
voice signals. Therefore, the analog voice-grade circuits will remain in use for quite some time; they cannot simply be scrapped because of the large investment involved. However, due to the inefficient utilization of their circuits, something must be done to improve them. The Mixed Voice/Data System provides the needed solution and will ease this analog to digital transition period.

7.3 EXPECTED CONTRIBUTIONS

The Mixed Voice/Data System is a resource saving telecommunication system that provides additional data service for computer communication in the existing analog circuits. After integrating the MVD System into the existing telecommunication network, the telecommunication carriers would benefit by being able to provide extra data service without expanding their existing resources, while the users of data communication links for computer communication would also benefit by having available less expensive data communication links. These users would have the illusion that they were using an exclusively allocated high-speed data channel for their computer communication.

If the MVD System is configured with multiple input lines for data in the transmitting side and multiple output
lines for data in the receiving side, it would also provide "packet switching" capability, in that messages could be switched from any of the input lines to any of the output lines by means of packets. The switching facility would consist of the transmitting microcomputer, the transmission lines, and the receiving microcomputer and would constitute a packet switching network.

Thus, the MVD System is not only a computer controlling "transmission" system but also a computer controlling packet "switching" system. This concept of simultaneous switching and transmission would be a major breakthrough from the conventional telecommunication system where these two functions are physically separated. It would also lead to hardware cost saving, as two functions would be integrated into one physical system.

7.4 FOLLOW-UP RESEARCH

Several areas can be listed where follow-up studies are desirable:

(1) Signal pattern recognition methods for voice and data;
(2) Input and output multiplexing using microprocessors;
(3) Multi-processor operating system;
(4) Variable-length packet protocols;
(5) Record Management System for packet retrieval;
(6) System reliability considerations and switchover software; and
(7) Message accountability considerations.

These areas can be elaborated briefly in the following paragraphs.

Detailed signal analysis can be performed to distinguish the analog voice signal from the digital data signal. Such analysis is a problem of signal pattern recognition, but the result would contribute to the alternative way of recognizing voice signals and data signals on an MVD channel. This recognition method would increase the utilization of the whole channel bandwidth for digital data, and thus could lead to even greater data throughput of the MVD channel.

Multiple input data lines and output data lines to/from the transmitting/receiving microcomputer can be implemented with a front-end microprocessor to take care of the character I/O in asynchronous operation mode or bit timing and signal framing interrupt processing in synchronous
operation mode. This extension would greatly relieve the host computer from processing the communication-related tasks, thereby giving it more time to perform packet transmission and switching tasks.

In order to increase data throughput, a multi-processor configuration should be studied and examined so that the data packet processing efficiency can be improved, as compared to the single-processor, serial processing environment.

For the sake of logic simplicity, fixed-size packets were considered for transmission but partial or residual packets is allowed for later transmission only at the time of voice interruption. A more general scheme would be to allow a variable-length packet protocol to reduce overhead. Such a variable-length packet protocol should be studied, and its effect on process logic determined.

In the transmitting side, a transmitted packet must be able to be retrieved upon request for retransmission by the receiving side. It is advisable that a record management system should be devised to retrieve any transmitted packet from RAM or the floppy disk in a certain period of elapsed time. This facility would enhance the packet's accountability during transmission.
For system reliability considerations, dual or redundant systems must be configured to safeguard against system hardware failure. A handsaking or positive acknowledgement protocol between the on-line and standby systems must be designed to facilitate solid and secure control of the system. Upon any inconfidence of the on-line processor, the on-line processor must issue an immediate take-over request to the standby processor, or the standby processor must assume control of the system when no positive acknowledgement from the on-line processor is periodically received. This kind of automatic switchover software design would enhance the system reliability.

To protect against accidental lost of any digital data message, message accountability must be maintained. Thus, any message must have at least two copies in the system at any one time. This arrangement can ensure recovery of any message if its storage hardware malfunctions.

There are undoubtedly many other technical problems which would require extensive investigation at the time of implementation. However, the aforementioned areas are those that have current academic value and should be explored in follow-up research.
7.5 CONCLUSION

The Mixed Voice/Data System indeed may become an inexpensive alternative for providing the computer communication link, especially if it is implemented in this microcomputer era. A small amount of investment on the microcomputer-based communication controllers can provide additional digital data service within the existing analog voice-grade channels. Such a controller, if implemented and manufactured in large quantity, will cost very little, but will provide a digital adaptor to/from the existing analog communication links. Furthermore, the communication carriers will find it very beneficial since additional data services can be provided for the long-distance data communication users without additional new links and the users can enjoy less rental charges of such digital data communication links with an illusion that they use an exclusively allocated data channels.

Also, the use of microcomputer-based controller not only provides the intelligence as to when to insert and to accept digital data packets to/from the analog channels, but is also capable of implementing additional high-level communication protocols for message or packet switching function in the future applications.
REFERENCES


Appendix A

Flowchart of MVD System Simulation

So many Interferers Come
COUNT NO. OF INTERFERES ENTERED
ASSIGN I.D. NO. TO EACH INTERFERER
SPREAD OUT FIRST ARRIVAL
RAISE PR LVL FOR PREVENTIVE USE OF FACILITY
INTERCEPT FOR VOICE ARRIVAL HANDLING
100 usec. FOR HANDLING INTERRUPT
FINISH USING CPU
PLUCK (P1) CHANNEL FOR VOICE TRANSMISSION
VOICE LASTS 1.170 sec. ON THE AVERAGE

Finishes Using CPU
SILENT PERIOD LASTS 1.58 sec. ON THE AVERAGE

Model Segment 2
VOICE ARRIVAL INTERFERENCE

Simulate for 10 min. Operation
DECREMENT SIMULATION COUNTER
LEAVE INPUT QUEUE

400 msec. FOR BASE LVL PROCESSING

FINISH PROCESSING

SELECT AN EMPTY QUEUE

JUMP OVER MINIMUM QUEUE SELECTION

SELECT A MINIMUM CONTENT QUEUE

RAISE LN LVL FOR OUTPUT BUFFER ASSIGNMENT

GET CPU FOR THE TASK

FINISH USING CPU

ADVANCE 5

RETURN

UPDATE CHANNEL OCCUPANCY

LEAVE TRANSMIT SIDK

GENERATE RECEIVE SEQUENCE

ASSIGN RECEIVE SEQUENCE TO EACH PACKET

REGISTER RECEIVE SYSTEM COUNT

RAISE LN LVL FOR INPUT BUFFER ASSIGNMENT

CAPTURE CPU FOR THE TASK

200 msec. FOR THE TASK

FINISH USING CPU

LOWER LN LVL FOR BASE LVL PROCESSING

RAISE LN LVL FOR BUFFER TRANSMISSION

GO BACK FOR RETRANSMISSION

SELECT 2,1,3,2,0,9

TRANSFER (NIRQ)

SELECT 2,1,3,2,0,9

TRANSFER
Appendix B
Program Listings of the MVD System Simulation

(A) Simulation Program for the Multi-line Queueing Model

```
// 9999,CLASS=C,REGION=252K
/*SETUP UNIT=TAPE9,ID=[MANG2],RO81,WRITE*/
//JOBLIB DD DSN=FLA950,LOADFLOW=DISP=SIN
//EXEC GSSRUN,TIME=CD=9,PARM=GO="C"*
//GO*OUTPUT DD DSN=MSL6,DISP=(NEW,APPEND),UNIT=TAPE9,LABE=16,SL *,X
//VOL=(PRIVATE,RETAIN,SER=MANG2),DCB=(RECFM=PB,RECL=133,X
//BLKSIZE=3990,DEN=21
//SYSIN DD *
REALLOCATE BLD=200,FAC,70,ST=5,QUE=70,LOG=70,FUN=FSV,10
REALLOCATE HSV=10,CHA=1,VAR=2,FSM=1,MS=1,TAB=1,MSV=10
REALLOCATE COM,93000

SIMULATION OF MULTI-LINE QUEUEING MODEL FOR THE MIXED VOICE/DATA MVD TRANS-
- MISION SYSTEM FOR COMPUTER COMMUNICATION

MODEL SEGMENT 1 SIMulates DATA PACKETS ARRIVE AND QUEUE AT MULTI-LINE QUEUE-
- STATIONS, WAITING FOR TRANSMISSION SERVICE. THE NUMBER OF OUTGOING LINES
- IS EQUAL TO THE NUMBER OF SERVERS PROVIDING SUCH SERVICE. EACH SERVER HAS
- A WAITING LINE IN FRONT OF HIM. THE NUMBER OF SERVERS IS SPECIFIED IN HALF-
- WORD SAVEVALUE XCHANG.

ASSURE THAT DATA PACKETS ARRIVE IN POISSON FASHION WITH INTER-ARRIVAL
- TIME EXponentially DISTRIBUTED AND WITH maximum MEAN ARRIVAL RATE determined
- BY THE maximum ALLOWABLE THROUGHPUT OF OUTGOING CHANNELS. THE maximum
- THRUPUT OF DATA PACKETS IS GIVEN AS:

THRUPUT = KZ/(EXPAT = 1) BITS/SEC

WHERE 'K' IS THE NUMBER OF OUTGOING CHANNELS;
'I' is the MEAN SILENT PERIOD = 1.58 SECONDS
'T' is the PACKET TRANSMISSION TIME

THIS maximum THRUPUT FUNCTION IS IMPLEMENTED WITH A REAL FUNCTION #5

TABLE OF DEFINITIONS

TRANSACTIONS: DATA PACKETS

PARAMETERS
P1: PACKET SEQUENCE NUMBER
P2: NUMBER FOR QUEUE, FACILITY, OR SWITCH ASSOCIATED
P3: CHANNEL ASSIGNMENT
P4: PACKET ARRIVAL TIME TO RECEIVE SYSTEM
P5: PREEMPTION COUNTER
P6: PACKET LENGTH FOR VARIABLE LENGTH PACKET OPTION
P7: INPUT LINE NUMBER AT XMIT SIDE
P8: OUTPUT LINE NUMBER AT RCV SIDE
P9: PACKET ARRIVAL SEQUENCE IN RCV SIDE

TIME UNIT = 100 MICROSECONDS

FUNCTIONAL DEFINITIONS

EXPONENTIAL DISTRIBUTION FUNCTION
0.03,1.14,106,31,222,38,359,31,41,69,6,91,6,71,2,75,1.38
0.81,6,94,1,83,31,88,2,12,99,3,92,2,52,49,4,2,81,95,2,99,90,3,2
49,3,5,90,3,93,9,99,4,6,99,5,3,99,8,2,99,7,99,8,8

TALK PERIOD DISTRIBUTION
0.028,1000,0.117,2000,0.233,3000,0.313,4000,0.378,5000,0.436,6000
0.488,7000,0.506,8000,0.534,9000,0.575,10000,0.608,11000,0.633,12000
0.657,13000,0.688,14000,0.716,15000,0.748,16000,0.785,17000,0.818,18000
0.851,19000,0.884,20000,0.921,21000,0.935,22000,0.943,23000,0.987,24000
```
<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>P2/2</td>
<td>ONLY EVEN NUMBER TO BE MONITORED</td>
</tr>
<tr>
<td>P292</td>
<td>USE TO TEST FOR EVEN OR ODD</td>
</tr>
<tr>
<td>10000×XHSSIZE/XHSSPEED</td>
<td>PACKET TX TIME ON VG LINE TO</td>
</tr>
<tr>
<td>1980-V3/9FNSEDN</td>
<td>AVERAGE SILENT PERIOD OVERHEAD</td>
</tr>
<tr>
<td>FNSEDN×XHSCNO/XHSPECN/100/27500</td>
<td></td>
</tr>
<tr>
<td>10000/V5</td>
<td>INTER-ARRIVAL TIME IN 100 USEC</td>
</tr>
<tr>
<td>V5×XHSSIZE/8</td>
<td>XMT TIME IN 100 USEC ON RCV OUTPUT CKTS</td>
</tr>
<tr>
<td>V5×XHSSIZE/5</td>
<td>MAXIMUM INPUT SPEED IN BITS/SEC</td>
</tr>
<tr>
<td>XHSSQCK×E×P1</td>
<td>PROVIDE SEQUENCE CHECK LOGIC</td>
</tr>
<tr>
<td>XHSCNO:</td>
<td>NUMBER OF OUTPUT VOICE GRADE CHANNELS</td>
</tr>
<tr>
<td>XHSSPEED:</td>
<td>OUTPUT CHANNEL DATA RATE</td>
</tr>
<tr>
<td>XHSSIZE:</td>
<td>PACKET SIZE IN BITS</td>
</tr>
<tr>
<td>LSI:</td>
<td>LOGICAL SWITCH #1 SIMULATES GREEN LIGHT FOR DATA PACKETS</td>
</tr>
<tr>
<td>XHSQCK:</td>
<td>PROVIDE PACKETS SEQUENCE CHECK IN RCV SIDE</td>
</tr>
</tbody>
</table>

**TABLE DEFINITIONS**

<table>
<thead>
<tr>
<th>TABLE</th>
<th>CHANNEL 2 QUEUE CONTENT DISTRIBUTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>TABLE</td>
<td>CHANNEL 6 QUEUE CONTENT DISTRIBUTION</td>
</tr>
<tr>
<td>TABLE</td>
<td>CHANNEL 8 QUEUE CONTENT DISTRIBUTION</td>
</tr>
<tr>
<td>TABLE</td>
<td>CHANNEL 10 QUEUE CONTENT DISTRIBUTION</td>
</tr>
<tr>
<td>TABLE</td>
<td>CHANNEL 12 QUEUE CONTENT DISTRIBUTION</td>
</tr>
<tr>
<td>TABLE</td>
<td>CHANNEL 14 QUEUE CONTENT DISTRIBUTION</td>
</tr>
<tr>
<td>TABLE</td>
<td>CHANNEL 16 QUEUE CONTENT DISTRIBUTION</td>
</tr>
<tr>
<td>TABLE</td>
<td>CHANNEL 18 QUEUE CONTENT DISTRIBUTION</td>
</tr>
<tr>
<td>TABLE</td>
<td>CHANNEL 20 QUEUE CONTENT DISTRIBUTION</td>
</tr>
</tbody>
</table>

**SAVEVALUE DEFINITIONS**

**INITIAL**

**XHSCNO,1/XHSSPEED,2400/XHSSIZE,1000/XHSQCK,1/LSI**

**STORAGE DEFINITION**

<table>
<thead>
<tr>
<th>CHAIN STORAGE</th>
<th>1</th>
</tr>
</thead>
</table>

**GENERATE**

V6×FNXPDIS,....F | DATA PACKET ARRIVES EVERY V6 T.U.

**ASSIGN**

8×FNXDEST | ASSIGNMENT OF PACKET DESTINATION CIRCUIT

**WRITE**

TRPKT | REGISTER PACKETS COUNT IN SYSTEM
SAVETEXT SEQNO*K1,M
ASSIGN 1,SEQNO
ASSIGN PACKET SEQUENCE NUMBER

SAVEVALUES EQNO*K1,M
PROVIDE SEQ. NO. FOR EACH PACKET

PRIORITY 1,BUFFER
RAISE PR LVL FOR INPUT BUFFER ASSIGNMENT

PRIORITY 1,BUFFER
DROP PR LVL FOR BASE LVL PROCESSING

SEQO = CPURV
CAPTURE RCV SIDE CPU

SEQNO = CPURV
PROVIDE PROCESSING TIME

SEQO = CPURV
FINISH USING RCV SIDE CPU

UNLINK SEQPR,SEQPR,*,BP1,BPU1,BPKR
RELEASE ONE FOR USER CHAIN

TRANSFER 2,CHNL0
TO PASS SEQUENCE CHECK

* FOLLOWING SECTION OF PROGRAM REPRESENTS A MULTI-SERVER MULTI-LINE WAITING SYSTEM FOR OUTPUT TRANSMISSION SERVICE. THE CHANNEL SELECTION CRITERIA IS:
* FIRSTLY TO FIND A LINE WHICH HAS ZERO CONTENT, AND IF IT CANNOT BE FOUND,
* THEN SECONDLY TO FIND A LINE WITH MINIMUM QUEUE CONTENT.

* SELECT E 2,1*XHSCHNO*2,QMIN
SELECT FROM CHNL 1 TO XHSCHNO FOR AN

* TRANSFER 1,SEQ0
EMPTY QUEUE

* MINO SELECT MIN 2,1*XHSCHNO*2,QMIN
SELECT FROM CHNL 1 TO XHSCHNO A MIN. LENGTH QUE

* WSKR = P5,K1,XMTIN
RECEIVE INPUT PACKET

* PQ = P2,K1,XMTIN
WAIT AT 1P2 I CHNL FOR XMISSION

* GATE LS P2
WAIT FOR GREEN LIGHT

* INSERT P2
CAPTURE PROCESSING FACILITY

* TEST P2 = V2 = NOTAB
TEST FOR EVEN CHNL OR ODD

* TABULATE V1
IF EVEN, TABULATE

* DEPART V2
LEAVE INDIVIDUAL QUEUE

* TABULATE VAB23
DEPART VAB23
LEAVE AGGREGATED QUEUE

* ENTER CHNLD
REGISTER CHNLS USAGE FOR DATA

* PRIORITY 2,SEQ0
HAS THE PACKET BEEN PREEMPTED BEFORE?

* PRIORITY 2,SEQ0
IF YES, RESTORE ITS PR LVL

* XMITIN = V3
PERFORM OUTPUT XMISSION

* CMTR = CHNL0
REGISTER CHNLS USAGE FOR DATA

* CMTR = CHNL0
REGISTER CHNLS USAGE FOR DATA

* TABULATE VAB23
DEPART VAB23
LEAVE AGGREGATED QUEUE

* ASSUME A COMPLETE RECEPTION OF DATA PACKET AT RCV SIDE IS REQUIRED

* ADVANCE 2500
PACKET XMISSION TIME TO RCV SYSTEM

* DATA PACKETS ARRIVE AT RECEIVING SIDE, FIRSTLY PERFORM INPUT BUFFER ASSIGN-
MENT SEQUENCE CHECK, ERROR RECOVERY, AND DETERMINE WHICH ONGOING LINE IT IS
TO BE DISPATCHED OR SWITCHED. BEFORE SEQUENCE CHECK AND ERROR RECOVERY, THE
ARRIVED DATA PACKETS ARE TEMPORARILY WAITING IN A USER CHAIN CALLED RVPRO.

* UNTIL IT IS SEQUENTIALLY UNLINKED BY THE PREVIOUS PACKET.

* MARK 4
MARK PACKET ARRIVING TIME

* QUEUE RVPR
ENTER RCV SYSTEM PACKET COUNT

* SAVETEXF RSEQX,K1,M
PROVIDE RECEIVING SEQUENCE

* ASSIGN P3,K1,RVSEQ
ASSIGN PACKET RECEIVING SEQUENCE

* PRIORITY 3
RAISE PR LVL FOR INPUT BUFFER ASSIGNMENT

* INSERT RSEQ,
PRIORITY 1,BUFFER
DROP PR LVL FOR BASE LVL PROCESSING

* CHECK TEST E KHSSEQP,PI,BPU1,BPKR
CHECK FOR SEQUENCE NUMBER

* SEQPR = SEQ0
INCREMNT SEQUENCE CHECK NUMBER

* SEQPR = SEQ0
PROVIDE PROCESSING TIME

* SEQPR = SEQ0
CAPTURE RCV SIDE CPU

* SEQPR = SEQ0
200 USEC FOR THE TASK

* SEQPR = SEQ0
FINISH USING RCV SIDE CPU

* SEQPR = SEQ0
RELEASE ONE FOR USER CHAIN

* TRANSFER 2,CHNL0
TO PASS SEQUENCE CHECK
MODEL SEGMENT 2 SIMULATES THE VOICE ARRIVAL INTERFERENCES WHICH PREEMPT THE USE OF CHANNEL FACILITIES IMMEDIATELY UPON THEIR RESPECTIVE ARRIVAL TO EACH CHANNEL. ASSUME THAT ALL CHANNELS ARE BUSY AND THE FIRST ARRIVAL OF VOICE SIGNAL TO EACH CHANNEL IS RANDOMLY SPREAD WITHIN THE MEAN VOICE INTER-ARRIVAL TIME AND ALSO EXPONENTIALLY DISTRIBUTED. LOGICAL SWITCHES ARE USED TO CONTROL THE MINIMUM USABLE SILENT PERIOD SO THAT THE INDIVIDUAL SWITCH WILL BE TURN ON OR GREEN FOR DATA PACKETS INSERTION AFTER THE CHANNEL IS RELEASED FROM VOICE USAGE FOR 5 MSEC FOR EXAMPLE.

transactions' parameters are defined as:

P1: CHANNEL ASSOCIATED TO THE INTERFERER

INTRF GENERATE ***XMCCHNO**** F SO MANY INTERFERERS COME

ADVANCE
ASSIGN 1NSINTRF ASSIGN INTERFERER'S I. D. NUMBER
ADVANCE PNSVIAT SPREAD OUT 1ST VOICE ARRIVAL TIME
PRIORITY 4 RAISE PR LVL FOR VOICE ARRIVAL INTERRUPT
NEXT PREEMPT CPUTR,PR INTERRUPT FOR VOICE ARRIVAL HANDLING
ADVANCE 3 ASSUME 300 USEC FOR VOICE ARRIVAL HANDLING
RETURN CPUTR FINISH USING CPU
LOGIC R P1 TURN ON THE RED LIGHT
PREEMPT P1,PR,BAKIN,RE REMOVE DATA PACKET FROM (P1) CHANNEL AND SEND TO BAKIN FOR RETRANSMISSION

ENTER CHNLV REGISTER VOICE OCCUPANCY STATISTICS
ADVANCE FNSTALK TALKING PERIOD LASTS
RETURN P1 VOICE TERMINATES, RELEASE CHANNEL FOR DATA PKTS
LEAVE CHNLV REGISTER VOICE OCCUPANCY STATISTICS
ADVANCE 500 CONTROL THE MINIMUM USABLE PERIOD OF TIME
LOGIC S P1 TURN ON THE GREEN LIGHT
ADVANCE FNSSTLEN SILENT PERIOD LASTS
TRANSFER +NEXT GO BACK TO BRING NEXT VOICE ARRIVAL INTERRUPT

MODEL SEGMENT 3 SIMULATES A TIMER TO CONTROL THE SIMULATION TIME

GENERATE 800000

TERMINATE 1
CONTROL CARDS
(B) Simulation Program for the Single-line Queueing Model

```c
// 9999:CLASS=C,REGION=252K
//JCLIB DD DSN=FGA950:LOADFLOW,DISP=SHR
// EXEC GSRRUN,TIME=0,PARM,GO='C'
//GO-DOOUTPUT DD DSN=SLMF24:DISP=(NEW,PASS),UNIT=TAPEN,LABEL=15:3L1,10
// VOL=PRIVATE,RETAIN,SER=WANG21,DCB=FRECPH=FR1,RECL=123'R
// SYSIN DD *
// BLOCK SIZE=3990,DE8N=21
REALLOCATE BLO,200,FAC,70,STO,5,QUE,70,LOG,70,FUN,9,F5V,10
RELOCATE HSY,10,CHA,1,VAR,20,FMS,1,MS,1,TAB,20,BVR,10
RELOCATE CON,93000

* SIMULATION OF SINGLE-LINE QUEUING MODEL FOR THE MIXED VOICE/DATA
* (AVO) TRANSMISSION SYSTEM FOR COMPUTER COMMUNICATION

* MODEL SEGMENT 1 SIMULATES DATA PACKETS ARRIVE AND QUEUE AT MULTI-LINE QUEUE
* STATIONS, WAITING FOR TRANSMISSION SERVICE. THE NUMBER OF OUTGOING LINES
* IS EQUAL TO THE NUMBER OF SERVERS PROVIDING SUCH SERVICE. EACH SERVER HAS
* A WAITING LINE IN FRONT OF HIM. THE NUMBER OF SERVERS IS SPECIFIED IN
* "WORD SAVE VALUE XCHTCHNO."

* ASSUME THAT DATA PACKETS ARRIVE IN POISSON FASHION WITH INTER-ARRIVAL
* TIME EXPONENTIALLY DISTRIBUTED AND WITH MAXIMUM MEAN ARRIVAL RATE DETERMINED
* BY THE MAXIMUM ALLOWABLE THROUGHPUT OF OUTGOING CHANNELS. THE MAXIMUM
* THRUPUT OF DATA PACKETS IS GIVEN AS:

THRUPUT = KZ / (EXPITA - 1) BIT/S
WHERE K IS THE NUMBER OF OUTGOING CHANNELS,
Z IS THE PACKET SIZE,
1/A IS THE MEAN SILENT PERIOD = 1.58 SECONDS
T IS THE PACKET TRANSMISSION TIME

THIS MAXIMUM THRUPUT FUNCTION IS IMPLEMENTED WITH A REAL FUNCTION 05

* TABLE OF DEFINITIONS
* TRANSACTIONS: DATA PACKETS
* PARAMETERS
  P1: PACKET SEQUENCE NUMBER
  P2: NUMBER FOR QUEUE, FACILITY, OR SWITCH ASSOCIATED
  P3: CHANNEL ASSIGNMENT
  P4: PACKET ARRIVAL TIME TO RECEIVE SYSTEM
  P5: PREEMPTION COUNT
  P6: PACKET LENGTH FOR VARIABLE LENGTH PACKET OPTION
  P7: INPUT LINE NUMBER AT XMIT SIDE
  P8: OUTPUT LINE NUMBER AT RCV SIDE
  P9: PACKET ARRIVAL 'SEQUENCE' IN RCV SIDE

* TIME UNIT = 100 MICROSECONDS

* FUNCTIONAL DEFINITIONS

* SIMULATE

** EXPONENTIAL DISTRIBUTION FUNCTION
0.0/1.0,1/1.024/2.0,2/2.222/3.0,3/3.359/4.0,4/4.69/5.0,5/5.69/6.0,6/6.95/7.0,7/7.15/8.0,8/8.35/9.0,9/9.64/10.0,10/10.34/11.0,11/11.64

** TALKING PERIOD DISTRIBUTION
0.028,1.0,0.117,2.0,0.233,3.0,0.313,4.0,0.378,5.0,0.430,6.0
0.468,7.0,0.506,8.0,0.536,9.0,0.575,10.0,0.609,11.0,0.633,12.0
```
QUO EQU 62,0  RCV SIDE OUTPUT QUEUE
CHNLD EQU 1+5  USE TO MEASURE CHANNEL UTILIZATION BY DATA
CHNLV EQU 1+8  USE TO MEASURE CHANNEL UTILIZATION BY VOICE
RPKMT EQU 83,0  RCV SIDE PACKET NUMBER STATISTICS
TRPKT EQU 64,0  XMT SIDE PACKET NUMBER STATISTICS
XMOUT EQU 65,0  XMT SIDE OUTPUT AGGREGATED QUEUE STATISTICS

* VARIABLE DEFINITIONS

1 VARIABLE P2/2  ONLY EVEN NUMBER TO BE MONITORED
2 VARIABLE P292  USE TO TEST FOR EVEN OR ODD
3 VARIABLE 10000*XHSIZE/XHSPD  PACKET XMT TIME ON VG LINE TO
4 VARIABLE isolate-V5*FN5EDPN  AVERAGE SILENT PERIOD OVERHEAD
5 VARIABLE FN5EDPN*XHCHNO*XHSPRCN*100/27500 AV. ARRIVAL RATE
6 VARIABLE 10000/V5  INTER-ARRIVAL TIME IN 100 USEC
7 VARIABLE V5*XHSIZE/8  XMT TIME IN 100 USEC ON RCV OUTPUT CKTS
8 VARIABLE XHSECK=V5  MAXIMUM INPUT SPEED IN BIT/SEC
10 VARIABLE XHSECK*V5  PROVIDE SEQUENCE CHECK LOGIC
11 VARIABLE XASNC=8*XHCHNO+1  PROVIDE ROTATING PRIORITY SCHEME

* TABLE DEFINITIONS

1 TABLE 02+0+1+10  CHANNEL 2 QUEUE CONTENT DISTRIBUTION
2 TABLE 0+4+0+1+10  CHANNEL 4 QUEUE CONTENT DISTRIBUTION
3 TABLE 0+8+0+1+10  CHANNEL 6 QUEUE CONTENT DISTRIBUTION
4 TABLE 0+8+0+1+10  CHANNEL 8 QUEUE CONTENT DISTRIBUTION
5 TABLE 010+0+1+10  CHANNEL 1 QUEUE CONTENT DISTRIBUTION
6 TABLE 012+0+1+10  CHANNEL 3 QUEUE CONTENT DISTRIBUTION
7 TABLE 014+0+1+10  CHANNEL 5 QUEUE CONTENT DISTRIBUTION
8 TABLE 016+0+1+10  CHANNEL 7 QUEUE CONTENT DISTRIBUTION
9 TABLE 018+0+1+10  CHANNEL 9 QUEUE CONTENT DISTRIBUTION
10 TABLE 020+0+1+10  CHANNEL 10 QUEUE CONTENT DISTRIBUTION

TAB20 QTABLE  IDENT TIME
TAB21 QTABLE  IDENT TIME
TAB22 QTABLE  IDENT TIME
TAB23 QTABLE  IDENT TIME
TAB24 QTABLE  LINE DISTRIBUTION
TAB25 QTABLE  TIME FOR TRANSMISSION
TAB26 QTABLE  IDENT TIME

* SAVEVALUE
  XHCHNO:
  XHSPRCE:
  XHSECK:
  LSI:
  LOGICAL:
  XHSECK:
  XHSPRCE:
  V5:
  XHCHNO:
  V5:

* STORAGE DEFINITION

CHNLD STORAGE 1
CHNLV STORAGE 1

GENERATE V6*FN5EDPN  DATA PACKET ARRIVES EVERY 5A TAK
ASSIGN B*FN5EDST  ASSIGNMENT OF PACKET DISTINATION CIRCUIT
QUEUE TRPKT  REGISTER PACKETS COUNT IN SYSTEM
SAVEVALUE SEQNO*+X1  PROVIDE SEQ. NO. FOR EACH PACKET
DOTS-CHNLO EQU 62,0

CHNLV EQU 1,5

RCV POCKET NUMBER STATISTICS

RVPKT EQU 63,0

XMIT SIDE PACKET NUMBER STATISTICS

TRPKT EQU 64,0

XMIT SIDE OUTPUT AGGREGATED QUEUE STATISTICS

VARIABLE DEFINITIONS

1 VARIABLE P2/2 ONLY EVEN NUMBER TO BE MONITORED

2 VARIABLE P22 USE TO TEST FOR EVEN OR ODD

3 VARIABLE 10000*XHSIZE/XHSPEED PACKET XMIT TIME ON VG LINE TO

4 VARIABLE 15860-V3*FNSEDPN AVERAGE SILENT PERIOD OVERHEAD

5 VARIABLE FNSEDPN*XHCHMLN*XHSPERA*100/27500 AV. ARRIVAL RATE

6 VARIABLE 10000/V5 INTER-ARRIVAL TIME IN 100 USEC

7 VARIABLE V5*XHSIZE/B XMIT TIME IN 100 USEC ON RCV OUTPUT CKTS

8 VARIABLE XHSIZE*V5 MAXIMUM INPUT SPEED IN BITS/SEC

9 VARIABLE XH5SEQCK*E*P1 PROVIDE SEQUENCE CHECK LOGIC

10 VARIABLE XH5SEQCK*E*P1 PROVIDE SEQUENCE CHECK LOGIC

TABLE DEFINITIONS

1 TABLE Q2,0,1,10 CHANNEL 2 QUEUE CONTENT DISTRIBUTION

2 TABLE Q4,0,1,10 CHANNEL 4 QUEUE CONTENT DISTRIBUTION

3 TABLE Q6,0,1,10 CHANNEL 6 QUEUE CONTENT DISTRIBUTION

4 TABLE Q8,0,1,10 CHANNEL 8 QUEUE CONTENT DISTRIBUTION

5 TABLE Q10,0,1,10 CHANNEL 10 QUEUE CONTENT DISTRIBUTION

6 TABLE Q12,0,1,10 CHANNEL 12 QUEUE CONTENT DISTRIBUTION

7 TABLE Q14,0,1,10 CHANNEL 14 QUEUE CONTENT DISTRIBUTION

8 TABLE Q16,0,1,10 CHANNEL 16 QUEUE CONTENT DISTRIBUTION

9 TABLE Q18,0,1,10 CHANNEL 18 QUEUE CONTENT DISTRIBUTION

10 TABLE Q20,0,1,10 CHANNEL 20 QUEUE CONTENT DISTRIBUTION

TAB200 XTABLE RVPKT,0,100,10 RCV SIDE PACKET RESIDENT TIME

TAB21 XTABLE SRVPKT,0,1,10 RCV SIDE PACKET NUMBER DISTRIBUTION

TAB22 TABLE QXSMOUT,0,1,10 RCV SIDE OUTPUT Q CURRENT CONTENT DISTRIBUTION

TAB23 XTABLE QTRPKT,0,5,30 XMIT SIDE PACKET NUMBER DISTRIBUTION

TAB24 XTABLE QSMOUT,0,5,30 XMIT SIDE AGGREGATED LINE DISTRIBUTION

TAB25 XTABLE XMOUT,0,10000,30 XMIT SIDE WAITING TIME FOR TRANSMISSION

SAVEVALUE DEFINITIONS

XHCHNO: NUMBER OF OUTPUT VOICE GRADE CHANNELS

XHSPEED: OUTPUT CHANNEL DATA RATE

XHSIZE: PACKET SIZE IN BITS

LST LOGICAL SWITCH IT SIMULATES GREEN LIGHT FOR DATA PACKETS

XHSEQCK: PROVIDE PACKETS SEQUENCE CHECK IN RCV SIDE

XHSPEED: VOICE INTERFERENCE OCCURRED SEQUENCE

INITIAL XHCHNO,1/XHSPEED;2400/XHSIZE;1000/LS1

INITIAL XHSSEQCK,1

STORAGE DEFINITION

CHNLV STORAGE 1

CHNLV STORAGE 1

GENERATE V6,FMSDPDIS*****K DATA PACKET ARRIVES EVERY V6 T.J.

ASSIGN B*FMDEST ASSIGNMENT OF PACKET DISTRICT CIRCUIT

QUEUE TRPKT REGISTER PACKETS COUNT IN SYSTEM

SAVEVALUE SEQNO*#114 PROVIDE SEQ. NO. FOR EACH PACKET
ASSIGN 1.XMSEGNO ASSIGN PACKET SEQUENCE NUMBER
PRIORITY 3 RAISE PR LVL FOR INPUT BUFFER ASSGNMT

FOLLOWING SECTION OF PROGRAM REPRESENTS A SINGLE-LINE MULTI-SERVER

CHANNEL BY ROTATING PRIORITY BASES 1 → 6. IF THE LAST ONE IS
UNAVAILABLE THE PROCESSOR WILL TRY THE FIRST CHANNEL TO SEE IF IT
IS DURING THE SILENT PERIOD.

THIS IS IMPLEMENTED BY INCREMENATING A FULLWORD SAVEVALUE AND
HAVE IT MODULOED BY THE CHANNEL NUMBER XmSCHNO AND THEN PLUS 1,
REPRESENTED BY THE VARIABLE Xi.

RETURN QUEUE XMOUT QUEUE AT OUTPUT STATION
TRY AGAIN LS V1.INCREE SEE IF THE NEXT CHANNEL IS SILENT?
ASSIGN 2+V1 IF YES, STORE THE CHNL NO. IN P2
SEIZE P2 LEAVE THE QUEUE STATION
DEPART XMOUT ENTER CHNLD REGISTER CHNL USAGE
PRIORITY 2+BUFFER WAIT FOR OTHERS
ADVANCE V3 PERFORM OUTPUT TRANSMISSION
RELEASE P2 FINISH USING THE CHANNEL FACILITY
LEAVE CHNLD REGISTER THE COMPLETION OF CHNL USAGE
TABULATE TAB23
DEPART TRPKT LEAVE XMIT SIDE SYSTEM

ASSUME A COMPLETE RECEPTION OF DATA PACKET AT RCV SIDE IS REQ'D
ADVANCE 25000 PACKET XMIT TIME TO RSV SYSTEM

DATA PACKETS ARRIVE AT RECEIVING SIDE, FIRSTLY PERFORM INPUT BUFFER ASSGN-
MENT, SEQUENCE CHECK, ERROR RECOVERY, AND DETERMINE WHICH OUTGOING LINE IT IS
TO BE DISPATCHED OR SWITCHED. BEFORE SEQUENCE CHECK AND ERROR RECOVERY, THE
ARRIVED DATA PACKETS ARE TEMPORARILY WAITING IN A USER CHAIN CALLED RVPRQ
UNTIL IT IS SEQUENTIALLY UNLINKED BY THE PREVIOUS PACKET.

MARK 4 MARK PACKET ARRIVING TIME
QUEUE RVPKT ENTER RSV SYSTEM PACKET COUNT
SAVEVALUE XMSEQ+K1+M PROVIDE RECEIVING SEQUENCE
PRIORITY 3 RAISE PR LVL FOR INPUT BUFFER ASSIGNMT
SEIZE CPURV CAPTURE XMIT CPU FOR SUCH TASK
ADVANCE P2 ASSIGNED PROCESSING TIME
ADVANCE . P2 ASSUME 400 USEC FOR SUCH TASK
RELEASE CPURV FINISH USING XMIT CPU
PRIORITY 1+BUFFER DROP PR LVL FOR BASE LVL PROCESSING
CHECK SEIZE RVSEQ+K1+M CHECK FOR SEQUENCE NUMBER
SAVEVALUE SEOCX+K1+M INCREMENT SEQUENCE CHECK NUMBER
SEUPK SEIZE CPURV CAPTURE RSV SIDE CPU
ASSIGN 122+ PROVIDE PROCESSING TIME
ADVANCE P2 200 USEC FOR THE TASK
RELEASE CPURV FINISH USING RSV SIDE CPU
PRIORITY 2 RAISE PR LVL FOR OUTPUT BUFFER ASSGNMT
UNLINK RVPRQ,SEQPR,11+BIVG+CHOK RELEASE ONE PR USER CHAIN
TRANSFER +CHECK TO PASS SEQUENCE CHECK
CHKUS TEST G CH1,0+NOCHS CHECK IF USER CHAIN HAS CONTENT
UNLINK RVPRQ,SEQPR,11+BIVG+NOCHS IF YES, RELEASE ONE PR UC
TRANSFER +CHECK GO BACK FOR HIS OWN CHECK
NOCHS LINK RVPRQ,11+BIVG+NOCHS IF YES, RELEASE ONE PR UC
CHKUS QUEUE OUTQ PASS SEQUENCE CHECK AND QUEUE
SEIZE OUTQ SEND TO DESTINATION CIRCUIT
TABULATE TAB22
DEPART OUTQ WAITING LINE
254

ADVANCE V7 PERFORM OUTPUT TRANSMISSION
RELEASE PB FINISH USING THE OUTPUT CIRCUIT
TABULATE TAB21 LEAVE RCV SYSTEM
DEPART RVPKT
TERMINATE TRANSACTION CORE GOES BACK TO POOL
MARK PREEMPTION COUNT
BAKIN ASSIGN 5K3
LEAVE CHNLND RAISE PR_LVL FOR IMMEDIATE RETRANSMISSION
TRANSFER +RETRAN SEND BACK FOR RETRANSMISSION
INCRE SAVEVALUE ASNCM+1.X INCREMENT TO THE NEXT CHANNEL
ADVANCE 500 SCAN LINE EVERY 50 MSEC
BUFFER - WAIT FOR OTHERS TO GET THRU
TRANSFER +TRYAG TRY THE NEXT CHANNEL AGAIN

MODEL SEGMENT 2 SIMULATES THE VOICE ARRIVAL INTERFERENCES WHICH PREEMPT THE
USE OF CHANNEL FACILITIES IMMEDIATELY UPON THEIR RESPECTIVE ARRIVAL TO EACH
CHANNEL. ASSURE THAT ALL CHANNELS ARE BUSY AND THE FIRST ARRIVAL OF VOICE
SIGNAL TO EACH CHANNEL IS RANDOMLY SPREAD WITHIN THE MEAN VOICE INTER-ARRIVAL
TIME AND ALSO EXPONENTIALLY DISTRIBUTED. LOGICAL SWITCHES ARE USED TO
CONTROL THE MINIMUM USABLE SILENT PERIOD SO THAT THE INDIVIDUAL SWITCH WILL
BE TURN ON OR GREEN FOR DATA PACKETS INSERTION AFTER THE CHANNEL IS RELEASED.
FROM VOICE USAGE FOR X MSEC. FOR EXAMPLE.

TRANSACTIONS' PARAMETERS ARE DEFINED AS:

T1 CHANNEL ASSOCIATED TO THE INTERFERER

INTER GENERATE ***.CHNO.3***F SO MANY INTERFERERS COME
SAVEVALUE CHNLNT+1.CHNT COUNT NO. OF INTERFERERS INTERED
ASSIGN 1.CHNLNT ASSIGN ID # TO EACH INTERFERER
PRIORITY 4 RAISE PR_LVL FOR VOICE ARRIVAL INTERRUPT
NEXT PREEMPT CPUPR,PR INTERRUPT FOR VOICE ARRIVAL HANDLING
ADVANCE CPUPR,PR ASSUME 100 USEC FOR VOICE ARRIVAL HANDLING
RETURN CPUPR FINISH USING CPU
LOGIC R P1 TURN ON THE RED LIGHT
PREEMPT P1,PR,BAKIN,RE REMOVE DATA PACKET FROM (T1) CHANNEL
AND SEND TO BAKIN FOR RETRANSMISSION

ENTER CHNLWT REGISTER VOICE OCCUPANCY STATISTICS
ADVANCE PNSLTALK VOICE SIGNAL LASTS
RETURN P1 VOICE TERMINATES, RELEASE CHANNEL FOR DATA PKTS
LEAVE CHNLWT REGISTER VOICE OCCUPANCY STATISTICS
ADVANCE 500 CONTROL THE MINIMUM USABLE TIME PERIOD
LOGIC S P1 TURN ON THE GREEN LIGHT
ADVANCE PNSILEN SILENT PERIOD LASTS
TRANSFER +NEXT GO BACK TO BRING NEXT VOICE ARRIVAL INTERRUPT

MODEL SEGMENT 3 SIMULATES A TIMER TO CONTROL THE SIMULATION TIME

GENERATE 800000
TERMINATE 1
CONTROL CARDS

START 1
### USER CHAIN STATISTICS

<table>
<thead>
<tr>
<th>RVPAQ</th>
<th>TOTAL ENTRIES</th>
<th>AVERAGE TIME/TRANS</th>
<th>AVERAGE CONTENTS</th>
<th>CURRENT CONTENTS</th>
<th>MAXIMUM CONTENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>191</td>
<td>3925</td>
<td>.434</td>
<td>.468</td>
<td>1</td>
</tr>
</tbody>
</table>

### STORAGE AVERAGE UTILIZATION

<table>
<thead>
<tr>
<th>CHINO</th>
<th>CHINL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>.460</td>
</tr>
<tr>
<td></td>
<td>.353</td>
</tr>
</tbody>
</table>

### HSV INCLUDE XH1-XH7 SYNTAX ERROR IN ABOVE CARD

### ALL QUEUE STATISTICS

<table>
<thead>
<tr>
<th>QUEUE</th>
<th>MAXIMUM CONTENTS</th>
<th>AVERAGE CONTENTS</th>
<th>TOTAL ENTRIES</th>
<th>ENTRIES</th>
<th>ZEROS</th>
<th>PERCENT</th>
<th>AVERAGE TIME/TRANS</th>
<th>AVERAGE TIME/TRANS</th>
<th>TABLE NUMBER</th>
<th>CURRENT CONTENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>INFQ</td>
<td>14</td>
<td>4.220</td>
<td>733</td>
<td>62</td>
<td>26</td>
<td>11.0</td>
<td>28735.941</td>
<td>32310.750</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUTQ</td>
<td>1</td>
<td>.000</td>
<td>196</td>
<td>194</td>
<td>2</td>
<td>100.0</td>
<td>.000</td>
<td>.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RVTQ</td>
<td>1</td>
<td>.487</td>
<td>192</td>
<td>192</td>
<td></td>
<td>94.7</td>
<td>7.221</td>
<td>140.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RVKT</td>
<td>14</td>
<td>4.778</td>
<td>194</td>
<td>194</td>
<td></td>
<td>100.0</td>
<td>29627.878</td>
<td>39620.878</td>
<td></td>
<td></td>
</tr>
<tr>
<td>XMKT</td>
<td>16</td>
<td>4.220</td>
<td>235</td>
<td>26</td>
<td></td>
<td>11.0</td>
<td>28735.941</td>
<td>32310.750</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SAVERAGE TIME/TRANS = AVERAGE TIME/TRANS EXCLUDING ZERO ENTRIES

### TABLE NUMBER CONTENTS

<table>
<thead>
<tr>
<th>TABLE</th>
<th>MEAN ARGUMENT</th>
<th>CURRENT CONTENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAB20</td>
<td>4065.286</td>
<td>NON-WEIGHTED</td>
</tr>
<tr>
<td>TAB21</td>
<td>1.500</td>
<td>NON-WEIGHTED</td>
</tr>
<tr>
<td>TAB22</td>
<td>1.000</td>
<td>NON-WEIGHTED</td>
</tr>
<tr>
<td>TAB23</td>
<td>5.134</td>
<td>NON-WEIGHTED</td>
</tr>
<tr>
<td>TAB24</td>
<td>4.634</td>
<td>NON-WEIGHTED</td>
</tr>
<tr>
<td>TAB25</td>
<td>20735.941</td>
<td>NON-WEIGHTED</td>
</tr>
<tr>
<td>TAB26</td>
<td>39615.976</td>
<td>NON-WEIGHTED</td>
</tr>
</tbody>
</table>
### XMIT-SIDE NO. OF PACKETS DISTRIBUTION

<table>
<thead>
<tr>
<th>TABLE</th>
<th>TABLE 23</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENTRIES IN TABLE</td>
<td>195</td>
</tr>
<tr>
<td>MEAN ARGUMENT</td>
<td>3.134</td>
</tr>
<tr>
<td>STANDARD DEVIATION</td>
<td>3.386</td>
</tr>
<tr>
<td>SUM OF ARGUMENTS</td>
<td>991,000</td>
</tr>
<tr>
<td>NON-WEIGHTED</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UPPER LIMIT</th>
<th>OBSERVED FREQUENCY</th>
<th>PER CENT OF TOTAL</th>
<th>CUMULATIVE PERCENTAGE</th>
<th>CUMULATIVE REMAINDER</th>
<th>MULTIPLE OF MEAN</th>
<th>DEVIATION FROM MEAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>111</td>
<td>57.31</td>
<td>57.31</td>
<td>-0.00</td>
<td>100.0</td>
<td>-0.000</td>
</tr>
<tr>
<td>1</td>
<td>48</td>
<td>25.23</td>
<td>82.54</td>
<td>37.5</td>
<td>45.4</td>
<td>0.973</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>6.33</td>
<td>88.87</td>
<td>-12.4</td>
<td>12.5</td>
<td>-0.516</td>
</tr>
<tr>
<td>3</td>
<td>13</td>
<td>6.74</td>
<td>95.61</td>
<td>13.2</td>
<td>13.2</td>
<td>-0.765</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>1.52</td>
<td>97.13</td>
<td>-10.4</td>
<td>-10.4</td>
<td>-2.812</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td>100.00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

REMAINING FREQUENCIES ARE ALL ZERO

### XMIT SIDE AGGREGATED OUTPUT QUEUE CONTENT DISTRIBUTION

<table>
<thead>
<tr>
<th>TABLE</th>
<th>TABLE 23</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENTRIES IN TABLE</td>
<td>235</td>
</tr>
<tr>
<td>MEAN ARGUMENT</td>
<td>4.634</td>
</tr>
<tr>
<td>STANDARD DEVIATION</td>
<td>3.328</td>
</tr>
<tr>
<td>SUM OF ARGUMENTS</td>
<td>1049,000</td>
</tr>
<tr>
<td>NON-WEIGHTED</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UPPER LIMIT</th>
<th>OBSERVED FREQUENCY</th>
<th>PERCENT OF TOTAL</th>
<th>CUMULATIVE PERCENTAGE</th>
<th>CUMULATIVE REMAINDER</th>
<th>MULTIPLE OF MEAN</th>
<th>DEVIATION FROM MEAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-0.000</td>
<td>-1.392</td>
</tr>
<tr>
<td>1</td>
<td>142</td>
<td>60.42</td>
<td>60.42</td>
<td>39.5</td>
<td>1.078</td>
<td>1.078</td>
</tr>
<tr>
<td>3</td>
<td>13</td>
<td>5.06</td>
<td>92.40</td>
<td>7.59</td>
<td>7.59</td>
<td>-1.114</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>1.26</td>
<td>93.66</td>
<td>-1.34</td>
<td>-1.34</td>
<td>-1.34</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td>100.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

REMAINING FREQUENCIES ARE ALL ZERO
### XMIT-SIDE PACKET WAITING TIME DISTRIBUTION

#### Table 10.24

<table>
<thead>
<tr>
<th>UPPER LIMIT</th>
<th>OBSERVED FREQUENCY</th>
<th>PERCENT OF TOTAL</th>
<th>CUMULATIVE PERCENTAGE</th>
<th>CUMULATIVE Remainder of Mean</th>
<th>NON-WEIGHTED Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>24</td>
<td>21.05</td>
<td>11.0</td>
<td>88.9</td>
<td>-0.00</td>
</tr>
<tr>
<td>10000</td>
<td>12</td>
<td>12.16</td>
<td>23.2</td>
<td>76.8</td>
<td>-1.92</td>
</tr>
<tr>
<td>20000</td>
<td>10</td>
<td>14.04</td>
<td>36.3</td>
<td>63.7</td>
<td>-1.92</td>
</tr>
<tr>
<td>30000</td>
<td>9</td>
<td>14.04</td>
<td>49.4</td>
<td>50.6</td>
<td>-0.88</td>
</tr>
<tr>
<td>40000</td>
<td>8</td>
<td>6.38</td>
<td>66.3</td>
<td>33.7</td>
<td>1.04</td>
</tr>
<tr>
<td>50000</td>
<td>7</td>
<td>6.38</td>
<td>72.0</td>
<td>27.0</td>
<td>1.27</td>
</tr>
<tr>
<td>60000</td>
<td>6</td>
<td>7.85</td>
<td>87.6</td>
<td>12.4</td>
<td>2.43</td>
</tr>
<tr>
<td>70000</td>
<td>1</td>
<td>6.80</td>
<td>94.4</td>
<td>5.6</td>
<td>2.78</td>
</tr>
<tr>
<td>80000</td>
<td>1</td>
<td>6.80</td>
<td>98.7</td>
<td>1.3</td>
<td>3.13</td>
</tr>
<tr>
<td>90000</td>
<td>1</td>
<td>6.80</td>
<td>100.0</td>
<td>0.0</td>
<td>3.82</td>
</tr>
</tbody>
</table>

REMAINING FREQUENCIES ARE ALL ZERO

### XMIT-SIDE PACKET RESIDENT TIME DISTRIBUTION

#### Table 10.26

<table>
<thead>
<tr>
<th>UPPER LIMIT</th>
<th>OBSERVED FREQUENCY</th>
<th>PERCENT OF TOTAL</th>
<th>CUMULATIVE PERCENTAGE</th>
<th>CUMULATIVE Remainder of Mean</th>
<th>NON-WEIGHTED Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0.00</td>
<td>0.0</td>
<td>-0.00</td>
<td>-1.12</td>
</tr>
<tr>
<td>20000</td>
<td>69</td>
<td>35.75</td>
<td>35.7</td>
<td>64.3</td>
<td>0.006</td>
</tr>
<tr>
<td>40000</td>
<td>34</td>
<td>20.20</td>
<td>55.9</td>
<td>44.1</td>
<td>-0.001</td>
</tr>
<tr>
<td>60000</td>
<td>15</td>
<td>16.04</td>
<td>72.0</td>
<td>27.9</td>
<td>1.54</td>
</tr>
<tr>
<td>80000</td>
<td>28</td>
<td>14.50</td>
<td>86.5</td>
<td>13.5</td>
<td>2.01</td>
</tr>
<tr>
<td>100000</td>
<td>29</td>
<td>10.36</td>
<td>100.0</td>
<td>0.0</td>
<td>3.07</td>
</tr>
</tbody>
</table>

REMAINING FREQUENCIES ARE ALL ZERO
<table>
<thead>
<tr>
<th>RELATIVE CLOCK</th>
<th>BLOCK COUNTS</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>2400000 ABSOLUTE CLOCK</td>
<td>2 4 0 0 0 0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>H</td>
<td>I</td>
<td>J</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>L</td>
<td>M</td>
<td>N</td>
<td>O</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>Q</td>
<td>R</td>
<td>S</td>
<td>T</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>V</td>
<td>W</td>
<td>X</td>
<td>Y</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>26</th>
<th>27</th>
<th>28</th>
<th>29</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>31</th>
<th>32</th>
<th>33</th>
<th>34</th>
<th>35</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>36</th>
<th>37</th>
<th>38</th>
<th>39</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>41</th>
<th>42</th>
<th>43</th>
<th>44</th>
<th>45</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>46</th>
<th>47</th>
<th>48</th>
<th>49</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
EXIT-SIDE PACKET NUMBER DISTRIBUTION IN %
XMRT-SIDE WAITING TIME DISTRIBUTION IN %

0

10.0K 30.0K 70.0K 90.0K 110.0K 130.0K 150.0K 170.0K 190.0K 210.0K 230.0K 250.0K 270.0K 290.0K 310.0K 330.0K 350.0K 370.0K 390.0K