Contract No. DAHC04-71-C-0043
ARPA Order No. 1816
Program Code No. 62706D
Effective Date: 6/25/71
Expiration Date: 6/30/73
Date of Report: 6/30/72
Technical Reporting Period: 7/1/71 - 6/30/72
Amount of Contract: $509,531

Contractor: Regents of the University of California

Principal Investigators:
David O. Harris (805)961-2534
James A. Howard (805)961-3136
Roger C. Wood (805)961-2101

Project Engineer:
Roland F. Bryan (805)961-3933

Title: Research in On-Line Computation

REPORT CSL-22

COMPUTER SYSTEMS LABORATORY
UNIVERSITY OF CALIFORNIA

SANTA BARBARA, CALIFORNIA 93106

Approved for public release; distribution unlimited.

***

This research was supported by the Advanced Research Projects Agency under ARPA Order Number 1816 of the Department of Defense and was monitored by U.S. Army Research Office-Durham under Contract No. DAHC04-71-C-0043.

***

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Advanced Research Projects Agency or the U.S. Government.

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) University of California, Santa Barbara, California 93106	2a. REPORT SECURITY CLASSIFICATION Unclassified
	2b. GROUP

3. REPORT TITLE

Research in On-Line Computation

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
Scientific. Technical 7/1/71 - 6/30/72

5. AUTHOR(S) (First name, middle initial, last name)

David O. Harris
James A. Howard
Roger C. Wood

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 30, 1972	91	19

8a. CONTRACT OR GRANT NO. ARPA Order No. 1816 DAHC04-71-C-0043	9a. ORIGINATOR'S REPORT NUMBER(S) CSL-22
b. PROJECT NO. P10144-A	
c. Program Element Code 62706D	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
This research was supported by the Advanced Research Projects Agency.	U.S. Army Research Office-Durham, Box CM, Duke Station, Durham, North Carolina 27706

13. ABSTRACT

During the report period special software and hardware developments took place in support of five research goals as follows:

1) Optimization of 360/75 Availability as a Computing Resource.
2) Development of Network Potential.
3) Promotion of Local User Interaction with the ARPANET.
4) Consultation to Other ARPANET Sites.
5) Speech Recognition Research.

The Network Control Program, Data Reconfiguration Service, UCSB programs that allow Network graphics, Remote Job Entry Service, and access for Network users to all resources of the 360/75 have resulted from our software efforts.

Hardware development by CSL in these five areas has included design and implementation of mini-computer interfaces, a high-speed data link, a specialized keyboard in use at MITRE, and a production version of the 360 IMP-Host Interface. These are now in use at MIT, NASA-AMES, and USC.

The speech researchers have succeeded in phoneme recognition for single speakers through wavefunction analysis and have developed an interactive signal analysis system that has resulted in a new technique for speech data compression for transmission.

DD FORM 1 NOV 65 1473

Unclassified
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
User Telnet						
Server Telnet						
Remote Job Service						
Simple-Minded File System						
Data Reconfiguration Service						
Graphics						
IMLAC						
Multi-Line Controller (MLC)						
Wavefunction Analysis						
Speech Recognition						
Vowel Recognizer						
Phoneme Recognition						
Signal Analysis						
Data Compression						

Contract No. DAHC04-71-C-0043

ARPA Order No. 1816

Program Code No. 62706D

Effective Date: 6/25/71

Expiration Date: 6/30/73

Date of Report: 6/30/72

Technical Reporting Period:
7/1/71 - 6/30/72

Amount of Contract: $509,531

Contractor: Regents of the University of California

Principal Investigators:
David O. Harris (805)961-2534
James A. Howard (805)961-3136
Roger C. Wood (805)961-2101

Project Engineer:
Roland F. Bryan (805)961-3933

Title: Research in On-Line Computation

REPORT CSL-22

COMPUTER SYSTEMS LABORATORY
UNIVERSITY OF CALIFORNIA

SANTA BARBARA, CALIFORNIA 93106

***

This research was supported by the Advanced Research Projects Agency under ARPA Order Number 1816 of the Department of Defense and was monitored by U.S. Army Research Office-Durham under Contract No. DAHC04-71-C-0043.

***

LIST OF CONTRIBUTING PERSONNEL

Pfeifer, Larry Lee	Speech Project
Retz, David L.	Speech Project
Stoughton, Ronald M.	Software Development

SUMMARY

Report CSL-22 covers several areas of ARPA supported research and development carried out by the UCSB Computer Systems Laboratory. This work is detailed in Sections I through IV of the report as follows:

1) Section I deals with software, or computer programming, support of the ARPANET for both local and Network users.

2) Section II deals with hardware support of ARPANET users and speech researchers.

3) Section III details the speech recognition research project.

4) Section IV discusses an interactive system for signal analysis as an outgrowth of the speech project.

A brief summary of the work carried out in these four areas is given in subsequent text.

The principal effort of the Computer Systems Laboratory during the year covered by this report was directed toward development and enhancement of the ARPA Computer Telecommunications Network, or ARPANET. This work was carried out through investigation of network users problems, participation in working groups made up of personnel from each ARPANET site, stimulation of network use by helping users at UCSB and other sites to gain access to the Network, creation of new software to ease network operations, and fulfillment of the designated role of UCSB as a primary resource or "Server" site on the ARPANET.

In order to optimize the availability of the UCSB 360/75 as a computing resource on the Network and to promote local use of the ARPANET, the UCSB Network Control Program (NCP) was organized in a manner to link "Users" to "Resources". The group now includes both local users and users on the Network at large. All of the services of the UCSB Computer Center are now resources on the ARPANET. Appendix II-A discusses the organization of the UCSB site.

In order to develop Network potential, hardware and software support has been provided to other ARPANET sites to assist them in using the Network. Special interfaces have been implemented, consultation has been given, programs to allow ease in site-to-site communication and to allow Network graphics have been created, and several new terminals are now operational on the Network due to UCSB-CSL efforts.

Our second major research area, that of speech recognition, has resulted in the implementation of a system capable of identifying the phonemes of isolated words by a single speaker. Based upon a wavefunction representation of speech, techniques employed in this research have enabled us to perform recognition which has either never been done before or is classically difficult. Specifically, the recognizer can distinguish between the elements of a complete set of thirteen vowel-like sounds, including the similar sounds of /a/ (as in "tot") and /ɔ/ (as in "taut"). We have also achieved above-average scores on the identification of the unvoiced consonants /p,t,k/ and the voiced consonants /m,n,ŋ/.

Words spoken into the computer are segmented into a sequence of phonemes, then characteristic features are extracted from each. These recognition features have enabled us to distinguish one phoneme from another for 32 phonemes of general American English, with an accuracy of 92 percent for a single speaker. Results were obtained for a list of 258 words which contained 1170 phonemes. Processing time for any spoken word is faster than real-time once the wavefunction parameters have been generated.

The overall results of this research demonstrates that wavefunction analysis is an accurate time domain transformation technique and that this type of analysis can be successfully applied to speech recognition.

The third area of research was the investigation of techniques for wavefunction analysis of connected speech. Work in this area has brought about the development of an interactive system on a small processor that has proven a powerful tool in wavefunction analysis and other signal processing in general. Facilities provide for input and output of digitally sampled data, mathematical analysis of the data, and a means for graphic presentation of data and results. Use of this interactive system has also allowed researchers to study new methods for data compression of digital speech.

TABLE OF CONTENTS

LIST OF PUBLICATIONS AND REPORTS

Publications:

L.L. Pfeifer	"Feature Extraction From Speech With a Computer Applied for a Classification or Recognition Process", Digest of IEEE EUROCON 71, Lausanne, Switzerland (Oct. 1971).
L.L. Pfeifer	"The Application of Wavefunction Analysis to Single Speaker Phoneme Recognition", Report CRL-18, Computer Research Lab, University of Calif. Santa Barbara (March 1972).
L.L. Pfeifer	"Isolated-Word Phoneme Recognition Using Features Derived From Wavefunction Parameters", Digest of the 1972 International Conference on Speech Communication and Processing, Boston, Mass. (April 1972).
D.L. Retz & R.C. Wood	"An Interactive System for the Analysis of Connected Speech", Presented at the 1972 International Conference on Speech Processing and Communication, Boston, Mass. (April 1972).
B.J. Carey J.A. Howard	"A Method for Speech Analysis by a Wavefunction Representation", Digest of the 1972 International Conference on Speech Communication and Processing, Boston, Mass. (April 1972).

Reports:

J.E. White	NIC 7176, NWG/RFC #206, "A User TELNET-Description of an Initial Implementation", 9 Aug 71.
J.E. White	NIC 7546, NWG/RFC #216, "TELNET Access to UCSB's On-Line System", 8 Sept 71.
R.M. Stoughton E. Harslem	NIC 7624, NWG/RFC #225, "RAND/UCSB Network Graphics Experiment", 13 Sept 71.
CSL	First Quarterly Report, reporting period 7/1/71 - 9/30/71.

Reports - con'd

J.E. White	NIC 7955, "IMPDIAG -- A Diagnostic Program for Hardware Checkout of System 360/IMP",21 Oct 71.
J.E. White	NIC 7812, NWG/RFC #264, "The Data Transfer Protocol", 15 Nov 71.
J.E. White	NIC 7813, NWG/RFC #265, "The File Transfer Protocol", 17 Nov 71.
J.E. White	NIC 8056, NWG/RFC #278, "Revision of the Mail Box Protocol", 17 Nov 71.
CSL	Second Quarterly Report, reporting period 10/1/71 - 12/31/71.
R.F. Bryan	NIC 9074, NWG/RFC #302, "Exercising the ARPANET", 8 Feb 72.
R.F. Bryan	NIC 9341, NWG/RFC #311, "New Console Attachments To the UCSB Host", 29 Feb 72.
CSL	Third Quarterly Report, reporting period 1/1/72 - 3/31/72.
CSL	Fourth Quarterly Report, reporting period 4/1/72 - 6/30/72.
R.F. Bryan	NIC 10531, "New OLS Keyboard for TIP Use", 16 June 72.
J.R. Pickens	NIC 11016, NWG/RFC #369, "Evaluation of ARPANET Services", July 72.

I. SOFTWARE

Nearly all software development carried out this contract year was devoted to Network-related projects. Implementation of a User/Server Telnet subsystem of OLS was completed during the first contract quarter. The major software accomplishment of the year was the development of a Data Reconfiguration Service (formerly called the Form Machine), a project conceived by members of the ARPA group at the Rand Corporation and developed jointly by UCSB and Rand personnel. A new Remote Job Service was developed for the ARPA Network which will eventually replace the rudimentary and less powerful Remote Job Entry service implemented the previous year. Other projects of major importance include the interfacing of the SEL 810B in Engineering to the IBM 360/75, implementation of many server graphics packages to support IMLAC and Tektronix terminals connected to OLS via the Network, a standard Network graphics server which supports level-0 of the proposed Network Graphics Protocol, and a miscellany of additions to OLS.

A. <u>Telnet</u>

A User Telnet was completed during this contract year which supports the Telnet Protocol as specified in NWG/RFC 158. As a consequence, all Culler-Fried On-Line System users have full access to all server Telnet Systems in the ARPA Network. The User Telnet subsystem of OLS supports full- and half-duplex transmission modes, character and line-at-a-time operation, shift-lock functions, and a facility for suspending output at the bottom of the display screen to permit "paging" of text. User Telnet is a subset of an OLS subsystem which provides status information on all Network sites, performs ICP's to Network sockets, performs low-level data transfers over Network connections, supports a command language for controlling the operation of our own NCP, and a miscellany of other Network service routines.

A Server Telent was also implemented during the same period which provides remote Network access to the Culler-Fried On-Line System. The major function of this software module is to allow simulation of an OLS function keyboard from a standard ASCII terminal. A <prefix> <name> <delimiter> notation is used to transmit function keys to OLS. Although this method of interfacing ASCII terminals to OLS is cumbersome to use, many Network sites have become consistent users of the UCSB On-Line System. In an effort to enhance the usage of OLS from remote Network sites, our hardware group has designed and fabricated an OLS keyboard which will attach directly to an input port on a TIP. One such keyboard has been delivered to MITRE Corporation for testing and evaluation and is now in use on their TIP.

B. Remote Job Service

As reported in our final report for 70/71, we developed a Remote Job Entry (RJE) service for Network users. Because there were Network sites who had expressed interest in such a service, we took the most expedient approach and used the internal card reader mechanism of HASP (our local spooling system which controls the flow of jobs through the batch processing system). However, this implementation provided no external controls over jobs as they flowed through the system as well as no facility to monitor the status of such jobs. As such, Network users often complained about the "unfriendly" nature of this service. The second half of this contract year thus found our group developing a complete Remote Job Service (RJS) facility which provides a full set of user controls over jobs submitted from remote Network sites. This service was also designed to comply with specifications being drawn up by the RJS subcommittee of the NWG for a Network standard RJS Protocol.

The proposed RJS protocol calls for a virtual operators console driven over a Telnet connection for interfacing the user to the RJS server system. Commands to the RJS server will be transmitted via this connection and responses to these commands will be returned to the user via the output path of the same Telnet connection. Input source files are to be retrieved by the RJS server from network file systems using the standard File Transfer Protocol (FTP) and in a like manner, printed and punched output will be deposited in Network file systems under full control of the RJS user.

HASP contains facilities for supporting standard IBM job entry terminals consisting of an operator's console typewriter, card reader/punch, and line printer. The method thus chosen for implementing a Network Remote Job Service was to simulate a HASP RJE terminal and include the necessary code in HASP to facilitate communication with the ARPA Network. We anticipated this to be a straight forward project and allocated two man-months for development. However, the end of the contract year found us still encumbered with debugging software related to this project.

Our major difficulty stemmed from a lack of detailed specifications for the Network RJS Protocol. Although many assumptions could be made while allowing a margin of flexibility, the problem was further compounded by the non-existence of an FTP upon which the RJS protocol is based. At this writing we have yet to receive detailed specifications for either of these protocols.

At this writing, implementation of RJS is nearing completion. Where uncertainties existed with respect to the proposed RJS Protocol, we implemented our own protocol. Should these areas be obsoleted by the finalized version of the Network standard, we will bring those areas into compliance.

C. Simple-Minded File System (SMFS)

It was anticipated that a Network standard FTP would be released sometime during the contract year to which SMFS would be interfaced. However, implementation of this project is being held in abeyance pending the adoption of a Network FTP by the NWG.

D. SEL Link

A high speed data link connecting the IBM 360/75 and SEL 810B is now operational. I/O routines effecting data transfer between the two processors have been debugged and software development is now being directed towards the creation of 360/75 service routines to augment the research facilities supported by the SEL speech system. These services include direct access storage for speech data, access to the ARPA Network, and compilation of SEL software by the 360/75. Some of the coding supporting these capabilities has been completed and checkout is in progress. Work in this area will continue into the ensuing contract year. Plans to make the SEL speech system available to Network users are now under consideration.

E. Data Reconfiguration Service

During the past year UCSB, in cooperation with the Rand Corporation, has been endeavoring to develop a Data Reconfiguration Service (DRS). There are three major components which comprise the UCSB-Rand implementation of

DRS: a compiler which reduces DRS source programs (forms) to a simpler, machine-independent instruction sequence (object program), an interpreter which executes the object program created by the compiler, and executive programs which interface the Network user to DRS. The DRS compiler was written at Rand in PL-1 with the remaining components being developed at UCSB in assembly language. All components of the DRS system have been designed to operate within the 360 system at UCSB.

As checkout of the DRS system nears completion, the only known bugs remaining are associated with the compiler. A recent review of these problems with Rand personnel indicate that these problems are attributable to the code generation phase of the compilation process. As such, it is felt that these bugs are only minor and a fix is anticipated in the very near future. A more detailed description of the DRS source and object languages can be found in the following documents available from the Rand Corporation:

1) "The Data Reconfiguration Service--An Experiment in Adaptable, Process/Process Communication", R-860-ARPA,

2) "Data Reconfiguration Service Compiler: Communications Among Heterogenous Computer Centers Using Remote Sharing", R-887-ARPA.

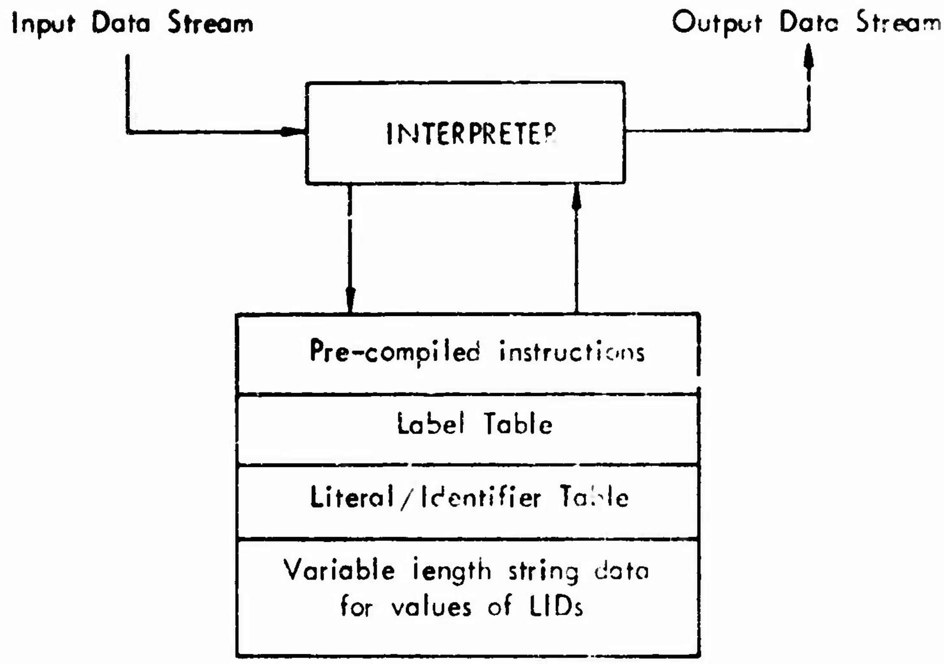
DRS is a time-shared service operating under control of the DRS Time-Sharing System (DRS/TSS). Network users of DRS communicate with DRS/TSS via a TENEX-like command language over a Telnet connection. In addition to a subset of TENEX executive commands (ATTACH, DETACH, LINK, etc.), a variety of commands are available for creating and editing forms, invoking the DRS compiler and interpreter, and displaying source and object listings of DRS forms. A full compliment of file system utilities has been included for storing, retrieving, and renaming DRS forms as well as listing directory items associated with the user's ID. DRS uses SMFS as its file system.

To execute a DRS form, the user supplies the DRS executive with the form name and Network connection data. Upon receipt of this information, DRS/TSS will establish the requested connections, fetch the form from SMFS, and instruct the DRS interpreter to commence execution of the form. As data flows between the connected processes, the interpreter performs transformations on the data in accordance with the rules specified by the form. The fact that DRS has instated itself as an intermediary process between user and server remains transparent to the connected processes. When execution of the form is terminated, DRS/TSS notifies the user and relays diagnostic data supplied by the interpreter.

The DRS interpreter applies a pre-compiled form to a real-time data stream to effect data transformations. (Figure I-a.) The compiler produces the instructions, label table, literals, and identifiers. The interpreter is a stack machine driven by a Polish postfix instruction sequence. It consists of an instruction decoder; instruction execution routines (called operators) for data fetching, storing, and conversions; an assemblage of state registers for control; and a run-time stack to house instruction operands (Figure I-b). Run-time-stack operands are used for arithmetic expression evaluation, concatenation, and comparison; they are also used as arguments to input and output instruction routines.

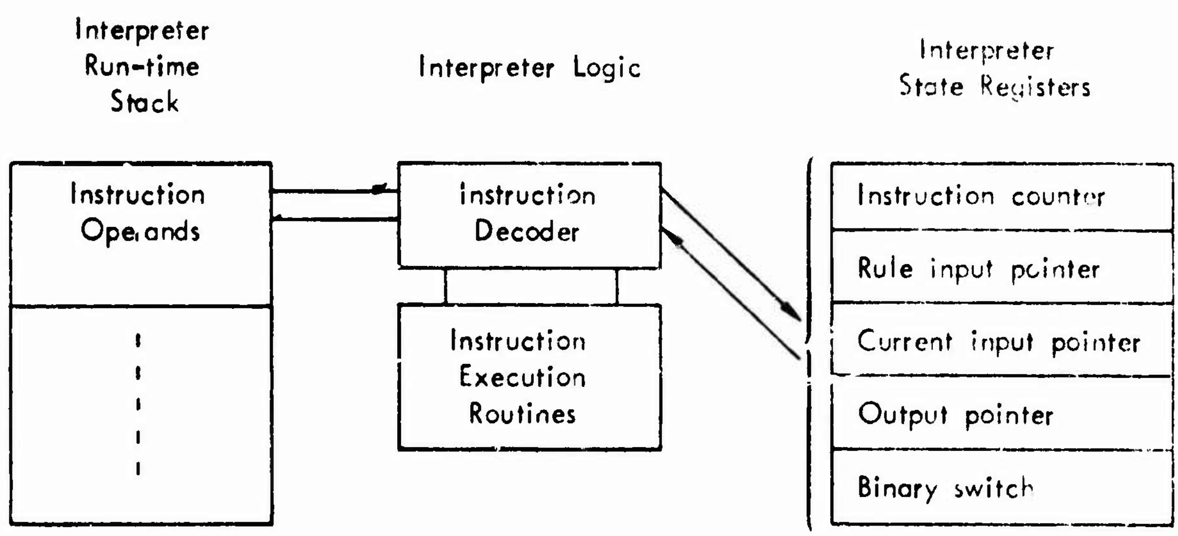
The Current Input Pointer addresses the next bit to be processed in the input stream. The Rule Input Pointer addresses the bit position of the input stream corresponding to the beginning of the current rule. Two input pointers are required: the Current Input Pointer moves along as each term is processed, but the Rule Input Pointer is not advanced unless the rule correctly describes the input. The Output Pointer addresses the next available bit position for inserting data into the output stream. The Instruction Counter points to the current instruction of the pre-compiled instruction sequence. The Binary Switch is a true-false indicator set by input call and compare instructions, and checked by test and branch instructions.

Personnel changes temporarily interrupted debugging of DRS software. However, checkout is now proceeding very rapidly and experiments are now being formulated to evaluate the effectiveness of this Network service.



Interpreter Interfaces

Figure I-a



Interpreter Components

Figure I-b

F. Graphics

Although OLS has been available to Network users for some time via Telnet, the effectiveness of this service has been severely limited due to the inability to support graphics within the Telent protocol. We have, therefore, extended the graphics capabilities of OLS by providing support for IMLAC and Tektronix terminals connected to OLS via the ARPA Network. We have also developed a Level-0 graphics server which conforms to the specifications of the proposed standard Network Graphics Protocol.

Several experiments were performed with MITRE Corporation to explore ways of providing OLS graphics to Network users equipped with IMLAC terminals. Software was written at UCSB utilizing the graphics features of the standard IMLAC Text and Edit software (modified slightly to allow an erase command to be generated remotely). This software which is resident in the IMLAC processor accepts graphic orders which draw 1-unit vectors in 8 directions and 2-unit vectors in 16 directions on an 80x80 display grid. However, the quality and resolution of the resulting displays was not considered acceptable for the intended graphic applications. An attempt to increase the resolution by modifying the horizontal and vertical gain settings to provide a grid of 113x113 was still not acceptable. This approach was thus abandoned although the software has been retained in OLS to provide graphics support for IMLAC users wishing to acquaint themselves with the UCSB system.

In an effort to obtain higher resolution graphics on the IMLAC, MITRE Corporation has developed special software for the IMLAC which is now available to other Network users. A graphics server was also developed at UCSB which interfaces to this special IMLAC software. The IMLAC graphics routines assume a display grid 1024x1024 and will accept the following commands: ERASE, ORIGIN(X,Y) and DRAW(X,Y). A Telnet connection is required over which keypushes are transmitted to OLS and alphameric output is returned to the user. Graphic orders are transmitted to the user via an additional simplex connection.

A graphics server supporting Tektronix 4002's has also been developed for OLS. A Telnet connection is required and output from OLS conforms precisely with the formats specified for the Tektronix terminal.

A rough draft of a standard Network Graphics Protocol was recently adopted by the Network Graphics Group. UCSB has implemented a server for OLS which supports Level-0 of this protocol. The services described in this section are available at the following socket numbers:

x'701' - IMLAC with MITRE software

x'703' - IMLAC with standard Text and Edit software

x'705' - Network Graphics Protocol (Level-0)

x'801' - Tektronix 4002.

G. Miscellaneous

As usual, many experiments and small projects were undertaken with personnel at other Network sites. Of particular importance was a project initiated by students in the Engineering Department at UCSB.

During the Winter and Spring quarters of last year a graduate course was offered by the Engineering Department (EE210) for the purpose of studying computer networks. Particular emphasis was given to the ARPA Network and projects were assigned to develop problem solving capabilities using Network resources. However, many efforts were hampered by the absence of file and data transfer facilities. As such, a group of these students were provoked into developing these facilities.

A program was written in PL-1 which utilizes Telnet to effect file and data transfers. The program runs in the 360/75 batch processing system and supports a simple command language which specifies the operations to be invoked. To effect a file transfer from a TENEX system to UCSB, for example, a command set is provided to the batch program which instructs it to log into the TENEX system, issue a COPY command which copies the desired file to the TTY associated with the logged in process, and log out after the transfer is complete. However, because the TTY is in actuality a process executing in the 360/75, the output received from the TENEX system can be written to a direct access file, listed on a printer, punched on cards, or transferred to another Network site in a similar manner. Half-duplex and line-at-a-time transmission modes are used to minimize overhead.

This method of effecting file transfers is admittedly inefficient. However, in the absence of a FTP, this is only of academic importance. The overriding significance of this project is the speed at which a file transfer facility was developed using existant resources and protocols. The potential of this facility is only beginning to surface and already Network sites are using this service to perform heretofore very difficult or impossible tasks.

II. HARDWARE

A. Technical Problems

Special hardware development carried out during the report period took place in support of five research goals as follows:

1) Optimization of 360/75 Availability as a Computing Resource.

2) Development of Network Potential.

3) Promotion of Local User Interaction with the ARPANET.

4) Speech Recognition Research.

5) Consultation to Other ARPANET Sites.

The hardware technical problems encountered in pursuing these areas are described below.

1) The UCSB site was required to develop interactive software to support graphics problem solving at MITRE and other sites. Terminals of the type used at other sites were not available at UCSB, and the technique for attachment to the 360 was not established. In addition, the MITRE group did not have an appropriate keyboard for use on the graphics system. Users were required to operate a sequence of keys to produce results that could be obtained through single key operation on a standard UCSB keyboard. The UCSB keyboards however, could not be operated on the TIP at MITRE. This had to be corrected.

2) Uninitiated network users encountered numerous difficulties in operating at network sites. It was necessary to assess the exact nature of the problems encountered and to pursue the limitations.

3) Local mini-computers had no way to obtain access to the ARPANET to pursue resources and inter-active connection with other sites. Small computers are located in Chemistry, Physics, and Poli-Sci. Researchers were unable to obtain details of network operation.

4) The speech recognition group was unable to gain access to the 360/75 or to the network. Speech acquisition requires at least 320 kilobits/second data-rate for transfer into storage in the 360. In addition the SEL 810B system lacked sufficient storage for programs and buffering of speech.

5) In some cases other ARPANET sites required assistance in connecting to the network. This is particularly the case for sites with IBM 360 equipment. In addition, several sites require connection as "Very Distant Hosts" and the interfaces at their Host terminations need specialized hardware to accommodate error detection and data buffering.

B. General Methodology

Support was provided as necessary to fulfill the commitments listed above. In most cases, hardware was specified, designed, and implemented to meet the need. In others, consultation was provided on an individual basis as well as in seminar form. Specifics are discussed below.

1) Tektronix and IMLAC terminals were obtained on loan from the vendors and hardware interfaces were implemented to allow them to operate on the 360 by way of our Multi-Line Controller (MLC). The programmers were thereby able to gain access to the ARPANET to assist MITRE and the other sites and to create the required graphics programs. To facilitate MITRE's use of the UCSB On-Line System a standard keyboard was equipped with logic to simulate a Teletype and loaned to MITRE so that they could use the UCSB system by way of the TIP.

2) A test group was formed to exercise the ARPANET and to evaluate operations at each site. Direction for the study as well as seminars on the ARPANET were provided by Computer Systems Laboratory personnel and Principal Investigators of the research project. NIC reports were issued to familiarize the network at large of this investigation.

3) To promote use of the network by local users the MLC was modified to allow mini-computers to gain access to the 360/75 and thereby to gain access to the ARPANET. Again, seminars and direct consultation was given to the potential network users.

4) A special, high-speed, data communications link has been implemented between the SEL in Electrical Engineering (Speech) and a Selector Channel of the 360. Disk and drum storage was added to the SEL system on a custom basis.

5) To support new attachments at other sites, the hardware group provided consultation in site planning and implemented new hardware for site use as needed. In particular, those sites with IBM 360 processors have been supplied interface hardware to accommodate their link to the ARPANET. These special controllers are in operation at MIT, NASA-AMES, and USC.

C. <u>Results</u>

Requirements have been met in all areas as described below.

1) MITRE is now operating the UCSB On-Line System with the new keyboard and with new IMLAC graphics software that was readily developed once the programmers were able to operate a local IMLAC terminal. Similar results can now be obtained on Tektronix or IMLAC terminals from any site on the network.

2) The network site evaluation and test group has produced a report that is soon to be released to the Network Information Center (NIC). Recommendations are made that will assist network users in the future.

3) Techniques for linking the mini-computers and the 360 have been developed in both hardware and software. These computers with their associated consoles will soon be fully operational on the network.

4) The SEL 810, speech system, is now able to transfer speech to the 360 for storage. The inter-active console on the SEL is also able to operate on the network. Speech files can now be sent to any site on the network and researchers can obtain access to other sites as well.

5) MIT, NASA-AMES, and USC were attached to the ARPANET by this special interface and are each operational. The controller itself is fully documented and a production version is available on short notice for use at any new 360 site that comes onto the network.

D. Important Equipment Developed

Each of the following items of new equipment are important in their own right.

1) Interfaces for new terminals to attach to the UCSB 360.

2) The new MITRE keyboard that acts as a simple Teletype while providing access to the UCSB On-Line System.

3) The high-speed communications link between the 360 and the SEL.

4) The special interface at NASA-AMES and USC for attaching their 360s to the ARPANET.

We look forward to the opportunity to provide the special keyboard and the 360 interfaces to many other sites. Both devices are fully documented and are readily implemented.

E. Recommendations for the Future

Since all areas pursued by the hardware group supported implementation of better user-resource inter-connection by way of the ARPANET, we forsee a continuing need for these services. The need for such consulting and implementation should grow along with the ARPANET. Each new site has certain unknowns and requirements that can best be resolved with the assistance of an experienced support group.

III. SPEECH PROJECT

A. Technical Problems

The accuracy of wavefunction analysis as a time domain transformation of speech has been demonstrated by both visual and aural comparison of an input acoustic wave with the synthetic version of that wave, derived from its corresponding wavefunction parameters. As a result, a certain amount of speculation has existed as to whether or not this representation could also be successfully applied to speech recognition.

Accordingly, the purpose of this research was to investigate the application of wavefunction analysis to single-speaker phoneme recognition. The primary objectives were:

1) New and simple methods for breaking speech into fundamental sound elements.

2) Wavefunction-derived recognition features which are accurate and consistent.

3) Implementation of a recognizer which is capable of identifying phonemes of a single speaker with an accuracy of at least 90 percent.

B. General Methodology

Based upon the belief that identification of the phonemic elements of speech will be an important part of automatic speech recognition, techniques for determining phoneme boundaries were devised so that representative features could be extracted from each fundamental sound unit. A set of reference data was obtained by speaking single words, each up to 609 msec in duration. Any word which did not contain two or more consecutive vowels or two or more consecutive voiced consonants was eligible. Words were chosen on the basis of phoneme complexity, phoneme location, and data base requirements. When a sufficient number of samples of each phoneme (at least 20 for most cases) were stored in files, the recordings were terminated.

1. Data Base

One of the purposes of a phoneme data base is to provide a cross section of the number of possible variations which can occur in the representation of a potential input sample.

If a large enough number of samples is taken, then certain predictions of expectations can be made regarding the behavior of future inputs. The examination of phoneme feature variations and the specification of limits on these variations permits decision making and thus forms the learning phase of a phoneme recognizer.

Our data base consists of feature lists representing the phonemes of 258 words by a single speaker. The list of words used is given in Table III-1. Recordings were made simultaneously onto tape and into the computer. The words were spoken within a time span of approximately two months. These words yielded 338 vowel-like phonemes, 372 voiced consonants, and 460 unvoiced consonants. The 460 unvoiced phonemes include 280 null phonemes. After the segmentation of each word, an operator-specified alphabetic identification code was assigned to each phoneme feature list and the feature lists stored in files for later use.

Table III-2 gives breakdown in the number of samples of each phoneme. The vowel-like set contains twelve vowels and the vowel-like occurrences of /ℓ/. The voiced consonant set contains thirteen phonemes, while the unvoiced consonant set contains seven true phonemes plus the null phoneme. The unvoiced fricatives /f/ and /θ/ were excluded from the unvoiced consonant set because they were nearly always detected as silence, and therefore the features for these phonemes were inadequate.

2. Feature Expansion

The pattern recognition technique for the phonemes is the same as for the embedded vowel recognizer, described in previous reports; i.e., a binary tree. Each decision node is based upon a two-dimensional crossplot of distinctive features of the various phonemes. Computations are performed on each basic phoneme feature list (consisting of the frequency and energy in each filter band) to obtain expanded feature lists suitable for crossplotting. An expanded feature list contains the ten original frequency and energy measurements as well as sum and difference frequencies, normalized energies, segment duration, and phoneme class (V, VC, or U). The sum and difference frequencies are obtained by either adding or subtracting the frequency terms of any two bands. The normalized energy data is computed by finding the maximum energy in a specific set of bands and then dividing each energy element of that set by the maximum. This normalization can take place on any combination of the five bands, thus yielding several sets of normalized energies.

Table III-1
Word list for single-speaker phoneme data base.

see	seat	seed
seal	seam	gone
ghost	log	dog
gun	done	dumb
game	those	vein
them	van	veto
contract (N)	contract (V)	bid
bode	plaster	chief
sticks	splashed	thick
happiness	booed	bird
bud	bead	bod
baud	bade	bood
or	core	tore
easy	ooze	are
am	all	in
oven	owner	a
urban	f	you'll (1)
lose	wound	nule
zoo	booze	groom
stream	swim	slam
blend	chain	chin
vote	mother	father
snake	steak	boat
shame	faster	shamee
disk	the	up
down	limit	sum
bed	bad	butter
mustache	lion	alone
fat	learn	busy
rub	load	fast
them	wheel	bar
moving (1)	moving (2)	small
window	two	five
six	seven	shabby
display	oil	leaving
moon	redeemer	yellow
invent	ruler	digit
ream	ren	rain
room	root	roam
rim	ran	rung
wrong	rom	yeast
year	you'll (2)	yarn
yam	yell	yale

Table III-1 (continued)

leaf	lodge	lather
lazy	one	weed
wood	web	visit
free	three	four
away	yes	no
level	plus	minus
load	store	me
return	reset	zero
erase	eight	nine
give me	push	yawning
gap	follow	space
ping pong	awake	gather
heaven	choosing	fog
thing	pudding	soup
was	cook	with
began	should	penny
walk	place	good
cave	his	through
clap	karate	africa
sugar	gossip	amazing
panic	oath	obtain
pancake	afloat	wash
offer	obtuse	office
voo-doo	waiver	awkward
economy	ethnic	weather
cooking	hatching	you
echo	gawking	august
chosen	empathy	votive
awash	edifice	motion
checker	gauze	chauffer
yacht	audition	pocket
which	walking	yogurt
recover	young	wasp
season	youth	hung
woman	refresh	washing
butcher	topic	song
shove	tongue	long
hobby	not	lasar
shopping	feather	chop
leather	bother	nook
that	bush	give
thatch	book	gaze
theft	use	yet

Table III-2

Phoneme Data Base

Vowel-like			Voiced Consonants			Unvoiced Consonants		
Phoneme	As In	Samples	Phoneme	As In	Samples	Phoneme	As In	Samples
I	kick	43	b	bet	28	p	pay	24
i	sheet	33	d	dust	28	t	two	45
u	boot	21	g	go	20	k	key	32
æ	hat	23	v	vote	20	ʃ	shoe	18
ε	pet	24	z	zoo	16	tʃ	chain	11
a	pot	20	th	them	13	s	sit	45
ɔ	talk	22	m	mat	34	h	hay	5
ʊ	took	15	n	no	48	null		280
ʌ	shut	43	ŋ	sing	21			
o	coat	25	y	yes	20			
e	shake	24	ℓ	last	52			
ɝ	bird	38	r	red	50			
ℓ	you'll	7	w	wet	22			

The generalized form for computing normalized energy is

$$NBbEa = \frac{BbE}{EMAX} \tag{1}$$

where

BbE = Band b Energy

EMAX = MAX(BbE)

b = band number (any combination of 1 to 5)

NBbEa = Normalized Band b Energy, set a

a = set (A through Z) representing a particular combination of bands.

The expanded feature list contains 32 elements, as shown in Table III-3. There are five sets of normalized energy, some of which may be redundant. No attempt is made to eliminate these redundancies.

3. Phoneme Recognizer

The phoneme recognizer is made up of three separate recognition trees (tables), one for vowels, one for voiced consonants, and one for unvoiced consonants. Each node of the recognition tree is implemented with a linear two-dimensional threshold element. The form of the decision element is

$$Ax+By = C \tag{2}$$

where A,B and C are the constants defined by the location of the decision line in the two-dimensional plane. The crossplot facility (software) was expanded so that when the operator specifies the location of a decision line on a crossplot, the computer not only draws the line but calculates and displays the constants A,B and C at the bottom of the display. The complete set of crossplots for the phoneme recognizer is given in Appendix A.

A sample of one of the crossplots, displayed on a larger scale, is shown in Figure III-a. The upper and lower bounds of each dimension are displayed along the x and y axes. The two numbers in the lower left corner indicate which features have been crossplotted on the x and y axes respectively, and correspond to the feature numbers of Table III The three numbers separated by commas at the bottom of the figure are the decision-line constants A, B and C respectively. Thus a single crossplot of this type

Table III-3

EXPANDED FEATURE LIST

1	PCLAS	(Phoneme Class, V, VC, U)
2	B1E	(Band 1 Energy)
3	B2E	(Band 2 Energy)
4	B3E	(Band 3 Energy)
5	B4E	(Band 4 Energy)
6	B5E	(Band 5 Energy)
7	B1F	(Band 1 Frequency)
8	B2F	(Band 2 Frequency)
9	B3F	(Band 3 Frequency)
10	B4F	(Band 4 Frequency)
11	B5F	(Band 5 Frequency)
12	NB1EA	(Normalized Band 1 Energy, set A)
13	NB2EA	(Normalized Band 2 Energy, set A)
14	NB3EA	(Normalized Band 3 Energy, set A)
15	NB4EA	(Normalized Band 4 Energy, set A)
16	NB5EA	(Normalized Band 5 Energy, set A)
17	NB2EB	(Normalized Band 2 Energy, set B)
18	NB3EB	(Normalized Band 3 Energy, set B)
19	NB4EB	(Normalized Band 4 Energy, set B)
20	NB3EC	(Normalized Band 3 Energy, set C)
21	NB4EC	(Normalized Band 4 Energy, set C)
22	NB5EC	(Normalized Band 5 Energy, set C)
23	F3MF2	(Band 3 Frequency Minus Band 2 Frequency)
24	F4MF3	(Band 4 Frequency Minus Band 3 Frequency)
25	F2PF3	(Band 2 Frequency Plus Band 3 Frequency)
26	F3PF4	(Band 3 Frequency Plus Band 4 Frequency)
27	NB2ED	(Normalized Band 2 Energy, set D)
28	NB3ED	(Normalized Band 3 Energy, set D)
29	NB3EE	(Normalized Band 3 Energy, set E)

Table III-3 (Cont'd.)

30	NB4EE	(Normalized Band 4 Energy, set E)
31	DUR	(Phoneme Duration)
32	F3MOD	(Modified Band 3 Frequency, for /a,ɔ/)

contains all the information to define a node of the tree (one row of the recognizer table). The tables were generated by establishing the proper sequence of crossplots which leads to the eventual separation of the phonemes in each class.

Figures III-b, III-c and III-d are the recognition trees for the vowels, voiced consonants and the unvoiced consonants respectively. The vowel tree has 30 nodes, with the longest possible decision sequence being 11 nodes and the shortest three nodes. This tree is a little more complex than the embedded vowel tree (implemented on a previous ARPA contract), which is probably a reflection of the increased phoneme complexity. The voiced consonant tree is the largest with 40 nodes and the unvoiced consonant tree next with 38 nodes. The recognizer tables corresponding to these trees are in Appendix III-A.

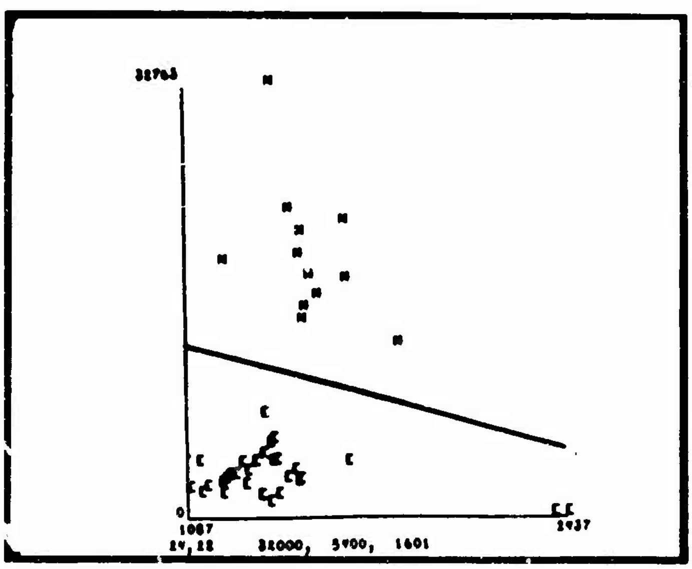


Figure III-a Sample crossplot of the vowels /i/ (E) and /a/ (H). A plot such as this contains a complete description of one node of the recognition tree.

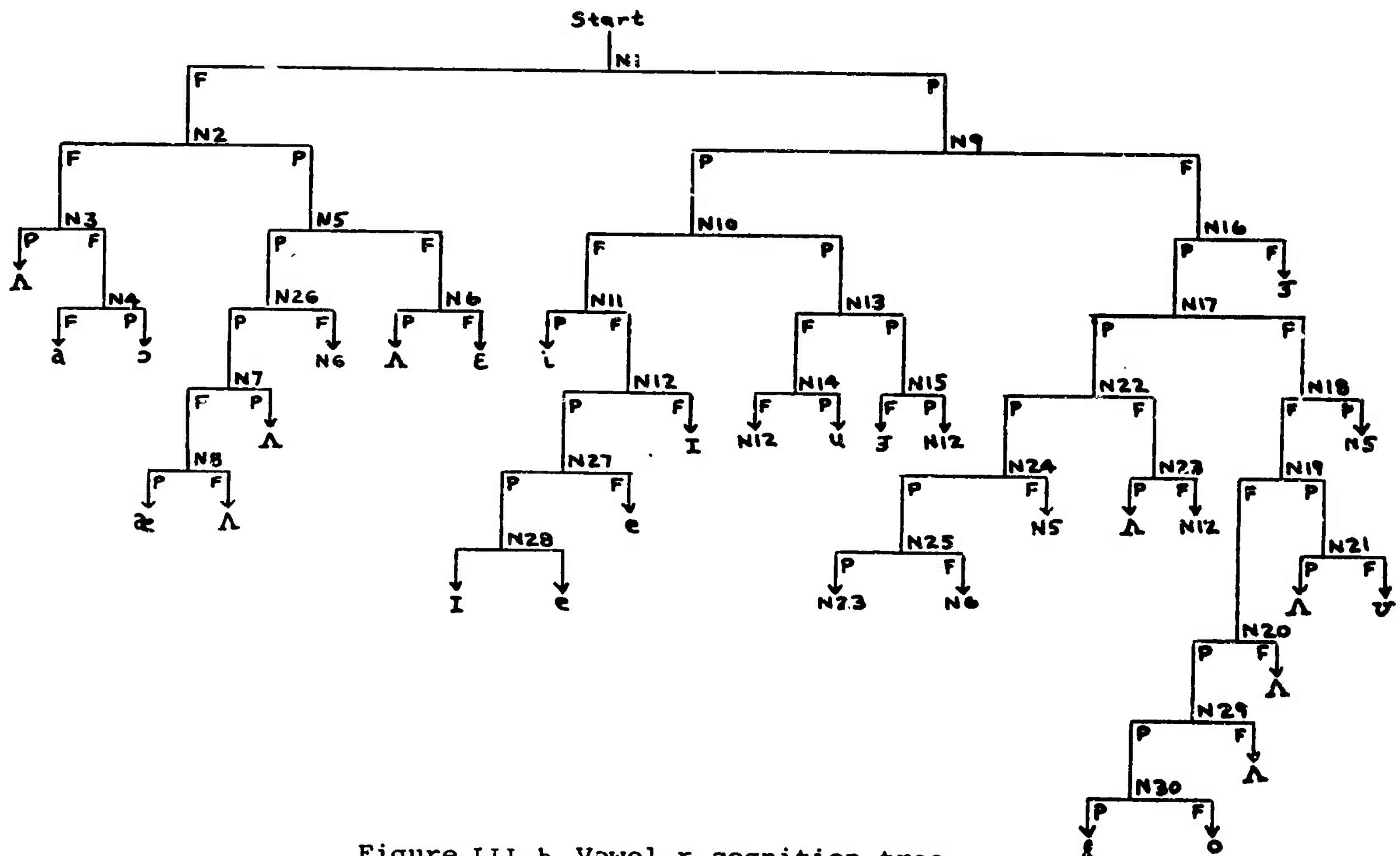


Figure III-b Vowel recognition tree.

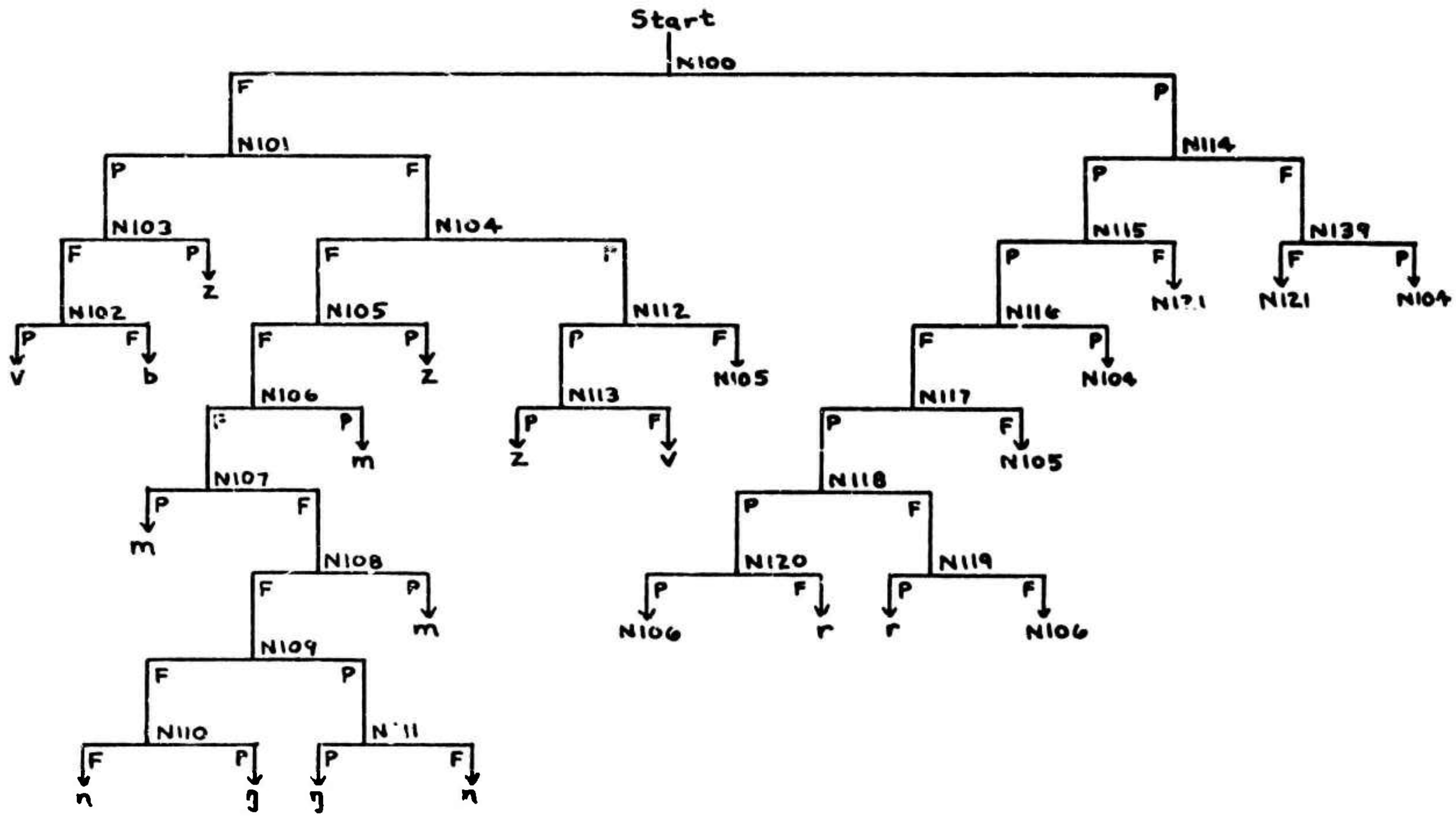


Figure III-c Voiced consonant recognition tree.

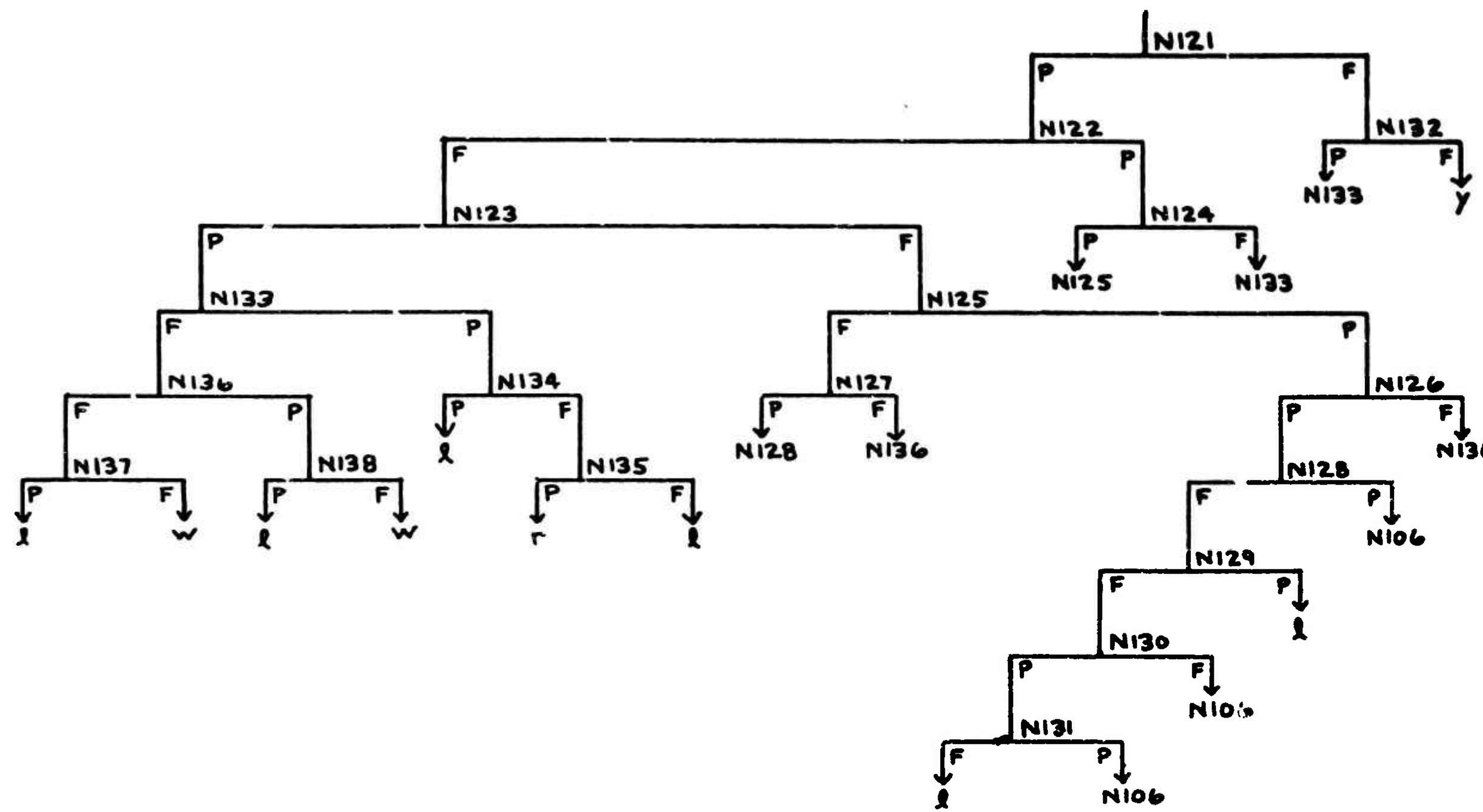


Figure III-c (Continued)

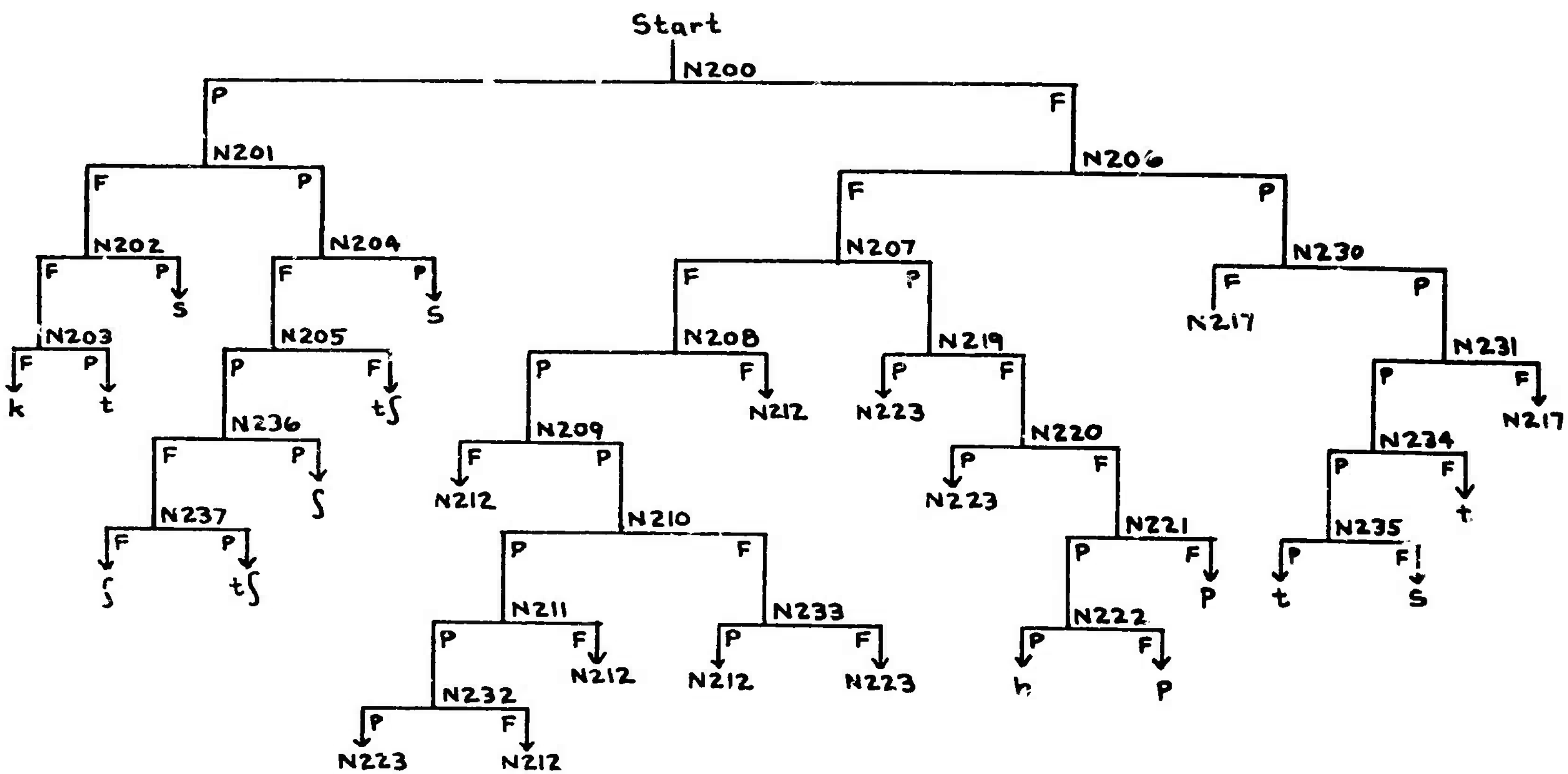


Figure III-d Unvoiced consonant recognition tree.

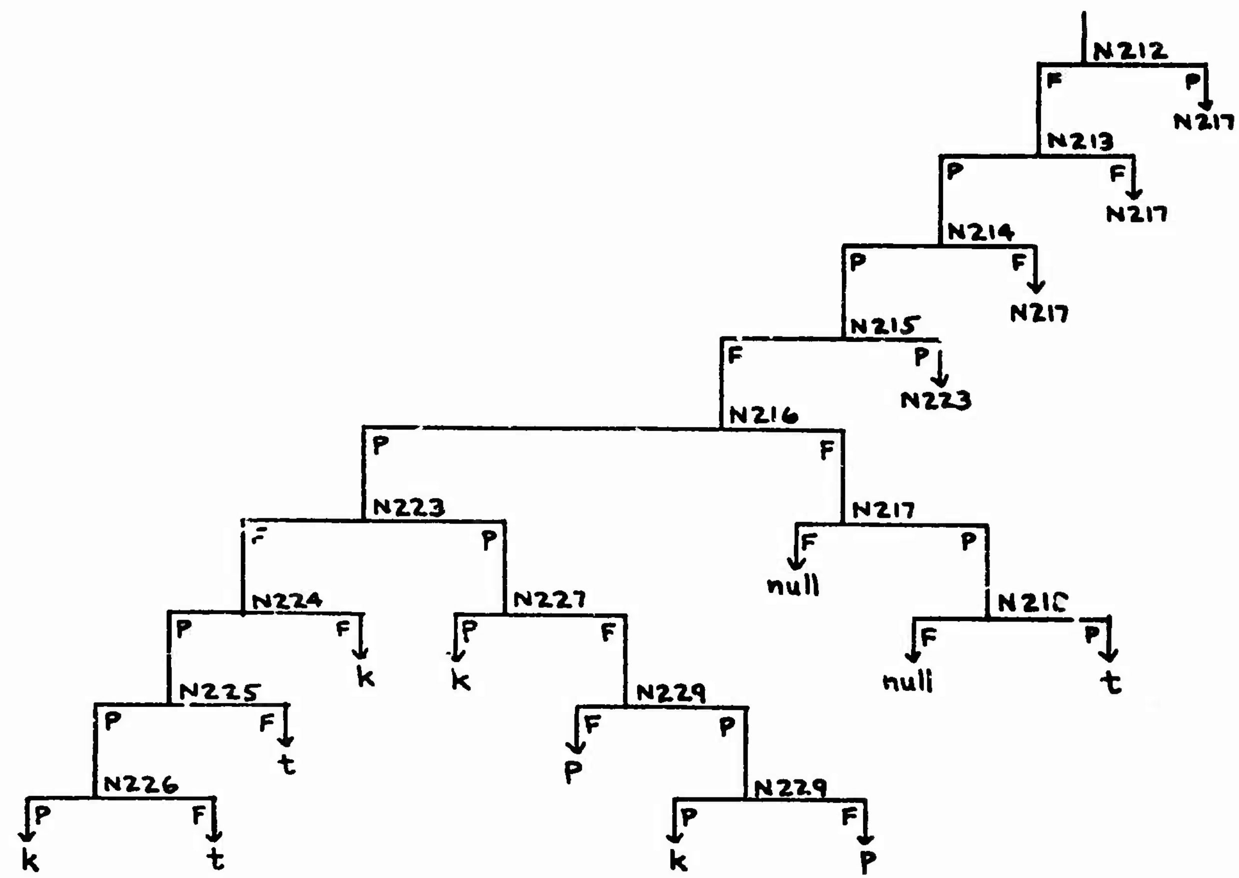


Figure III-d (Continued)

C. Results

For the entire set of 1170 phoneme samples, 92 did not fall within the proper pattern class, resulting in an overall accuracy of 92 percent. Table III-4 gives a breakdown of the overall results for the three major phonetic subclasses.

1. Vowel Recognition

For the group of thirteen vowel-like sounds, 92 percent correct recognition was obtained. While this score is not as good as the 97 percent obtained by our previous vowel recognizer, it is at least as significant because of the increased phoneme complexity and the large input set.

A more detailed description of the recognizer's behavior with regard to vowel-like phonemes is available in Table III-5 in the form of a confusion matrix. Based upon the vowel confusion matrix, the following observations can be made.

Table III-4 Overall Recognition Results

Phoneme Group	Samples	Number Errors	Percent Correct
Vowel-like	338	26	92
voiced consonants	372	49	86
unvoiced consonants	460	17	96
TOTAL	1170	92	92

a) no errors occurred in the recognition of the vowels /ɔ/, /o/, and the vowel-like occurrences of /ℓ/,

b) the highest number of errors was recorded for /I/ but the highest percentage error (16.6%) is held by /e/,

c) the vowels /u/, /o/, /e/, and /ℓ/ were not confused with any other vowel,

d) the most frequent incorrect response was the vowel /ε/,

e) the highest confusion rate existed between /ae/ and /ε/.

Table III-5

Confusion matrix for vowel-like phonemes of single-speaker recognizer

VOWELS

INPUT \ RESPONSE	I	i	u	æ	ɛ	a	ɔ	ʊ	ʌ	o	e	ℓ	ɝ	# wrong
I	38	2			1				2					5
i	1	32												1
u			19										2	2
æ				20	2				1					3
ɛ				2	21								1	3
a						18	1		1					2
ɔ							22							0
ʊ								14	1					1
ʌ	1				1	1		1	39					4
o										25				0
e	2				2						20			4
ℓ												7		0
ɝ								1					37	1
# wrong	4	2	0	2	6	1	1	2	5	0	0	0	3	26

2. Voiced Consonant Recognition

An accuracy of 86 percent was obtained for the group of voiced consonants. The elements of this set were the most difficult to separate, yet favorable results were obtained for the subgroup of nasals and the subgroup of /y,ℓ,r,w/.

Within the set of voiced consonants no separation could be found between /b/, /d/, and /g/ in any two-dimensional plane. This is typical for these phonemes since they are classically difficult to recognize. As a result, all three are simply considered as one pattern class and treated as a /b/. Difficulty was also experienced in the separation of /v/ from /th/. This difficulty is related to the problems encountered with the detection of the unvoiced /f/ and /θ/. Since the fricative counterparts of /v/ and /th/ (/f/ and /θ/ respectively) could not be detected, only the low-frequency data remained and this did not provide sufficient information to distinguish between the two phonemes. Therefore, the /v/ and /th/ are also considered as one phoneme class and treated as a /v/.

The confusion matrix for the voiced consonants is given in Table III-6. Based upon this table, the following points can be presented:

a) only the phoneme /y/ was not misrecognized,

b) the largest number of misidentifications occurred for /b/, /d/ and /g/,

c) the phoneme /ŋ/ had the highest percentage (27%) error rate,

d) error rates for /v, th/ and /m/ (24 and 23.5% respectively) were also quite high,

e) the /y/ and /w/ had no incidents of confusion with the other unvoiced consonants,

f) the most frequent incorrect response was the /b,d,g/ set,

g) the greatest number of misidentifications occurred between the /v,th/ set and the /b,d,g/ set.

Table III-6

Confusion matrix for voiced consonant phonemes of a single-speaker recognizer

VOICED CONSONANTS

		RESPONSE										# wrong
		y	ℓ	r	w	m	n	ŋ	z	v,th	b,d,g	
INPUT	y	20										0
	ℓ		28	2				1			3	6
	r			31						2		2
	w		1		21							1
	m					26	4	2		1	1	8
	n					1	41	3	2		1	7
	ŋ						3	16			3	6
	z						1		15			1
	v,th						1	2		25	5	8
	b,d,g					4	2	1		3	66	10
	#wrong	0	1	2	0	5	11	9	2	6	13	49

3. Unvoiced Consonant Recognition

The recognition score for the unvoiced consonants was very favorable at 96 percent. No great difficulties were experienced with /p,t,k/ and the desired high accuracy in the discrimination of the null from the other phonemes was achieved.

Table III-7 gives the detailed results of the unvoiced consonant recognition. Some of the observations which can be made regarding these results are:

a) no errors were made in the recognition of /s/,

b) the phoneme /p/ was the most error prone,

c) no phonemes were misidentified as /h/,

d) the most frequent incorrect response was the null phoneme,

e) the greatest confusion existed between the null and /p/.

4. Word Evaluation

Another way of examining the results is to look at the list of words in which phoneme recognition errors were made. Table III-8 contains these words, the idealized phonemic transcription of each word, and the computer response. Out of the 258 words, 58 had nine single phoneme errors, 10 had two phoneme errors, and two had three phoneme errors. Many of the single-error cases could probably be corrected by using linguistic or contextual information. A word recognizer with a good set of rules for phonological combinations could still make sense out of the phoneme string which said the word "ecomomy" rather than "[illegible]nomy" or the word "heavuz" rather than the true word "[illegible]n".

Some of the errors re[illegible] the possible ambiguities which arise in word pronunciation. For example, the phonemic pronunciation for the word "limit" might be "/ℓ I m I t/" or "/ℓ I m ʌ t/". The word "bar" might be spoken as "/b a r/" or "/b ɔ r/". One might say the word "office" as "/ɔ f I s/" or "/ɔ f ʌ s/". Therefore, some of the errors may not be recognition errors at all, but rather the assignment of the wrong identification code to an ambiguous phoneme.

Table III-7

Confusion matrix for unvoiced consonant phonemes of a single-speaker recognizer

UNVOICED CONSONANTS

		RESPONSE								#
		p	t	k	s	h	ʃ	tʃ	null	wrong
INPUT	p	19		1					4	5
	t		42	1	1				1	3
	k	1		30					1	2
	s				45					0
	h			1		4				1
	ʃ						17	1		1
	tʃ						1	10		1
	null		3	1					276	4
	#wrong	1	3	4	1	0	1	1	6	17

Table III-8

List of words containing phoneme recognition errors.
*Commas are used for clarity

Spoken Word	Transcription	Response
gun	g,ʌ,n	v,ʌ,n
those	th,o,z	n,o,z
van	v,ae,n	v,ε,n
contract (N)	k,a,n,t,r,ae,t	k,a,n,k,r,a,t
contract (V)	k,ʌ,n,t,r,ae,t	_,ʌ,n,t,r,ae,t
bod	b,o,d	b,o,n
happiness	h,ae,p,i,n,ε,s	h,ae,k,i,n,ε,s
bade	b,e,d	b,I,d
am	ae,m	t,ae,v
f	ε,-	ae,-
you'll	y,u,ℓ	y,u,b
stream	s,t,r,i,m	s,t,r,i,ŋ
slam	s,ℓ,ae,m	s,ℓ,ʌ,n
snake	s,n,e,k	s,n,I,k
shame	ʃ,e,m	tʃ,e,ŋ
the	th,ʌ	b,ʌ
up	ʌ,p	a,p
limit	ℓ,I,m,I,t	r,I,m,ʌ,t
mustache	m,ʌ,s,t,ae,ʃ	m,ʌ,s,null,ε,ʃ
lion	ℓ,-,n	n,-,n
rub	r,ʌ,b	r,ε,b
bar	b,a,r	b,_,r
moving	m,u,v,I,ŋ	n,u,v,I,ŋ
display	d,I,s,p,ℓ,e	d,i,s,null,ℓ,ε
oil	-,ℓ	-,b
leaving	ℓ,i,v,I,ŋ	ℓ,i,b,I,ŋ
digit	d,I,-,I,t	d,I,-,i,t
ream	r,i,m	r,i,n

Table III-8 (continued)

Spoken Word	Transcription	Response
rain	r,e,n	r,e,b
room	r,u,[illegible]	r,ɜ,b
ran	r,ae,n	r,ae,z,t
rom	r,a,m	r,ʌ,m
lodge	ℓ,a,-	ℓ,a,-,t
visit	v,I,z,I,t	b,I,z,I,s
away	ʌ,w,e	ʌ,w,ɛ
level	ℓ,ɛ,v,ℓ	ℓ,ɛ,b,ℓ
store	s,t,o,r	s,t,o,v
me	m,i	n,i
return	r,i,t,ɝ,n	r,i,t,ɝ,ŋ
yawning	y,ɔ,n,I,ŋ	y,ɔ,n,I,n
heaven	h,ɛ,v,ʌ,n	h,ɛ,v,ʌ,z
choosing	tʃ,u,z,[illegible]	tʃ,u,z,I,n
began	b,i,g,ae,n	m,i,m,ae,n
penny	p,ɛ,n,i	p,ɛ,n,I
place	p,ℓ,e,s	null,ℓ,e,s
cave	k,e,v	k,e,n
through	-,r,u	-,r,[illegible]
sugar	ʃ,ʊ,g,ɝ	ʃ,ʊ,v,ɝ
office	ɔ,-,I,s	ɔ,-,ʌ,k
voo-doo	v,u,d,u	v,u,m,u
economy	i,k,a,n,ʌ,m,i	i,k,a,m,ʌ,m,i
weather	w,ɛ,th,ɝ	ℓ,ɛ,th,ɝ
hatching	h,ae,tʃ,I,ŋ	h,ae,tʃ,I,b
gawking	g,ɔ,k,I,ŋ	g,ɔ,p,I,ŋ
empathy	ɛ,m,p,ʌ,-,i	ae,m,p,ʌ,-,i
edifice	ɛ,d,I,-,ʌ,s	ɛ,n,I,-,ʌ,s
checker	tʃ,ɛ,k,ɝ	tʃ,ɝ,k,ɝ
audition	ɔ,d,I,ʃ,ʌ,n	ɔ,d,I,ʃ,I,ŋ

Table III-8 (continued)

Spoken Word	Transcription	Response
walking	w,ɔ,k,I,ŋ	w,ɔ,k,I,b
yogurt	y,o,g,ɝ,t	y,o,v,ɝ,t
wasp	w,ɔ,s,p	w,ɔ,s,null
season	s,i,z,ʌ,n	s,i,z,ʌ,ŋ
woman	w,ʊ,m,ʌ,n	w,ʌ,m,ʊ,n
butcher	b,ʊ,tʃ,ɝ	b,ʊ,ʃ,ɝ
hobby	h,a,b,i	k,a,n,i
shopping	ʃ,a,p,I,ŋ	ʃ,a,null,ɛ,b
feather	-,ɛ,th,ɝ	-,ɛ,th,ʊ
that	th,ae,t	b,ae,t
give	g,I,v	g,I,ŋ
gaze	g,e,z	m,e,n

5. Recognition Examples

The computer representation of the phonemes is a set of one or two upper-case alphabetic characters. A phoneme/ symbol translate table is given in Table III-9. These particular characters were chosen for their similarity to IPA phonetic symbols and their relationships to the actual pronunciation of the phonemes. Note that because of the inability to recognize the individual elements of the /b,d,g/ set, all three elements are identified as B. For the same reason, bith /v/ and /th/ are identified as V.

Five recognition examples are shown in Figures III-e through III-i. Some of the words are the same as words in the data base, but not the same recording. Figure III-e illustrates the correct recognition of the phonemes in the word "input". There was no silence preceding the /p/, but it was still correctly recognized. There are two blank segments after the /ʊ/; the first is null and the second is silence. Run time was 537 msec.

The word "reset" is shown in Figure III-f with accurate segmentation and recognition results. This time there was no silence before the burst release of the /t/ with no ill effects on its recognition. Run time for this example was 490 msec. Figure III-g illustrates recognition of the phonemes for the word "robot". Segment boundaries are well placed with regard to what we might "eyeball" as phoneme boundaries. Sufficient data is present in the released segment of the /t/ to allow it to be recognized correctly. Run time for this word was only 454 msec.

The results for the word "listening" in Figure III-h reveal that an extra phoneme has erroneously been placed between the /I/ and /s/. This extra segment was recognized as the /z/ because a small amount of voicing was present during that interval, along with the "s"-like friction. The speech rate for this word was faster than normal in order that it could be contained in the 610 msec data buffer. Therefore, this example serves as a good demonstration of the capabilities of the segmentation and the accuracy of the recognition. This word took 581 msec to process.

Figure III-i shows that the /e/ of display was misidentified as the vowel /ɛ/. The first phoneme is designated as /b/, but indicates the presence of a /b,d,g/. The interval of silence was a good separator of the two unvoiced consonants /s/ and /p/. Highlights of this example are the accurate segmentation and recognition of the short duration phonemes /p/ and /ℓ/. Run time for this example was 496 msec.

Table III-9

Computer representation of phonemes.

Phoneme	Representation	Example	Phoneme	Representation	Example
I	II	bit	v	V	vat
i	EE	beat	z	Z	zoo
u	UU	boot	th	V	that
æ	AE	bat	m	M	met
ɛ	EH	bet	n	N	net
a	AH	bomb	ŋ	NG	sing
ɔ	AW	bought	y	Y	you
ʌ	UH	but	r	R	rent
ʊ	OO	book	w	W	wit
o	OH	boat	p	P	pet
e	AY	bait	t	T	ten
ɝ	ER	bird	k	K	kit
ℓ	L	let	ʃ	SH	shut
b	B	bet	tʃ	CH	church
d	B	debt	s	S	sat
g	B	get	h	H	hat

Null and silence intervals are left blank

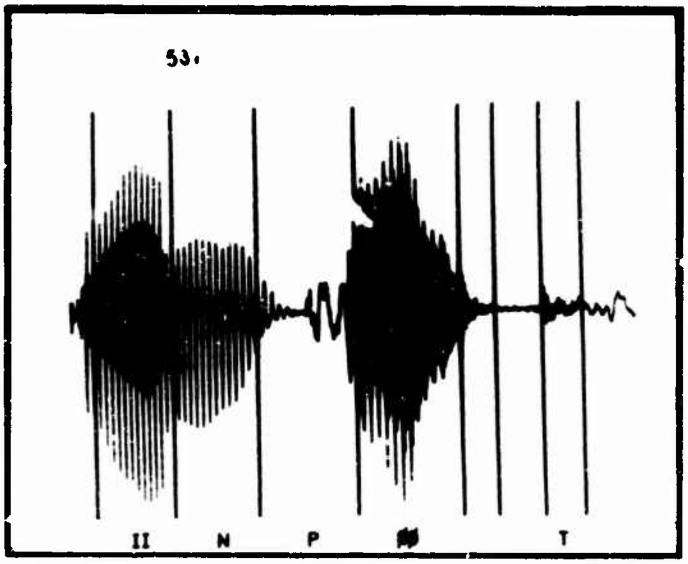


Figure III-e Phoneme recognition for the word "input".

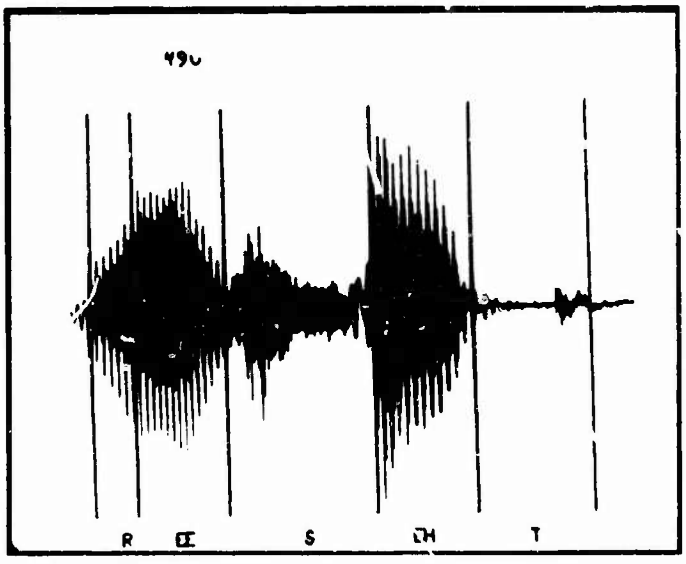


Figure III-f Phoneme recognition results for the word "reset".

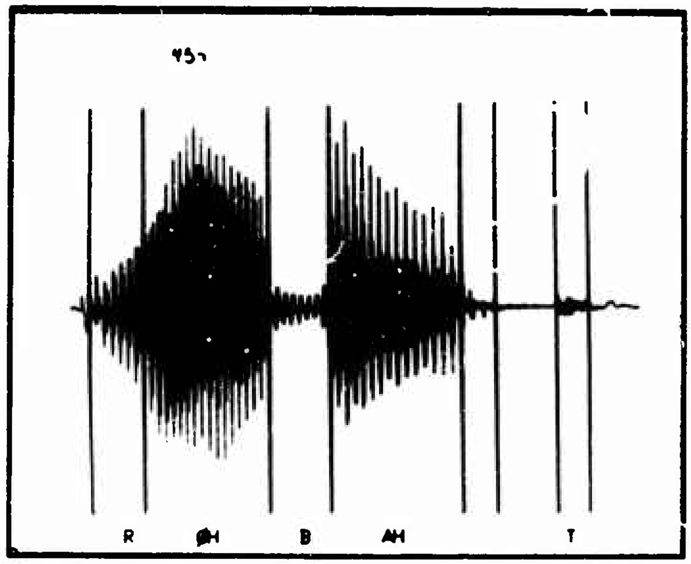


Figure III-g Recognition of the phonemes in the word "robot".

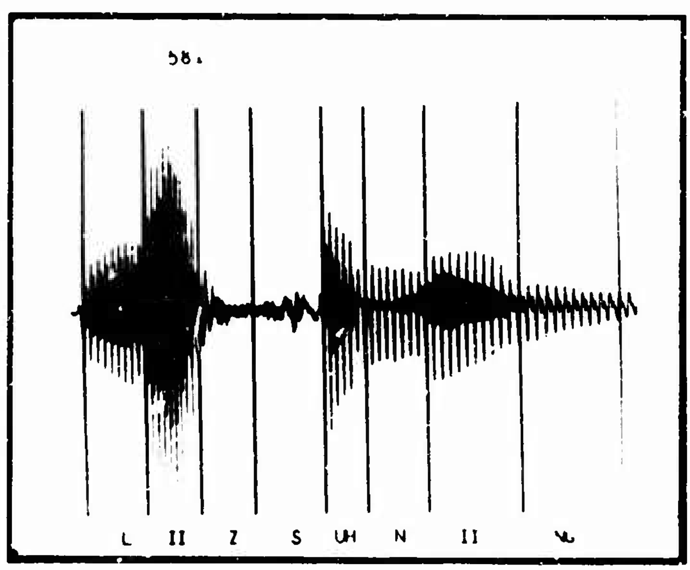


Figure III-h Phoneme segmentation/recognition for the word "listening".

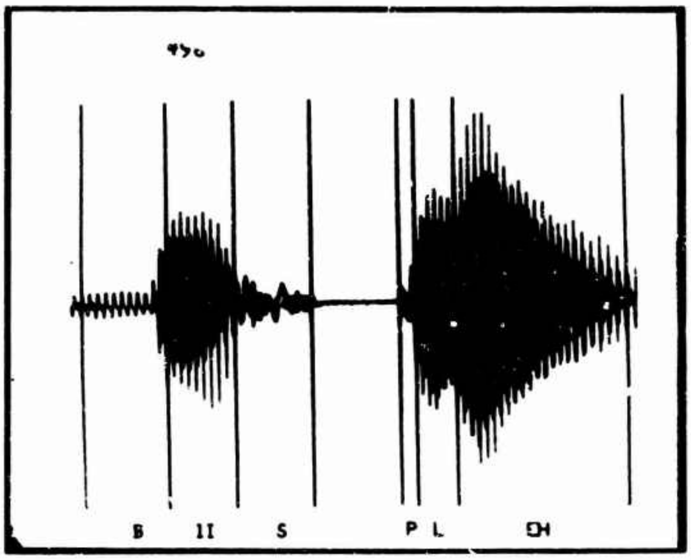


Figure III-i Recognition of the phonemes in the word "display".

6. Conclusions

The techniques employed in this research have enabled us to perform recognition which has either never been done before or is classically difficult. The recognizable vowel set is one of the most complete with thirteen vowel-like sounds. The difficulties in identifying between /a/ and /ɔ/ have been resolved and the vowel-like occurrences of both /ℓ/ and /r/ (which becomes /ɝ/ as in "bird") are permitted. Within the group of thirteen voiced consonants, the elements of the subgroup /m,n,ŋ/ can be distinguished from each other with an accuracy of 86 percent. This is not too far from the error rate experienced by humans themselves in the perception of nasals. Recognition of the subgroup /y,ℓ,r,w/ is correct 98 percent of the time, an excellent recognition score. As with other phoneme recognition systems, the voiced consonants /b,d,g/ continue to be a problem. Recognition of these phonemes is contingent upon features derived from the transition into or out of the adjacent phoneme. Therefore, the frequency/energy features taken from the center of these phonemes carried only interclass information and not intra-class information.

Results for the identification of the unvoiced consonants are very promising. Recognition between /p,t,k/ is fairly reliable with an accuracy of 90 percent. This score was obtained for various occurrences of these phonemes in the initial, medial, and final (released) positions. The burst release segment in the final position has proven to contain distinctive information regarding the identity of a /p,t,k/. The unvoiced fricatives /f,θ/ are not allowed because they usually lack sufficient energy to pass the amplitude threshold of the analyzer, and are therefore detected as silence. This could probably be improved upon by either increasing the upper-frequency bound or using high frequency pre-emphasis on the acoustic wave. Transitional information may also be required as an aid to identification.

The results just presented substantiate our beliefs that wavefunction analysis, providing an accurate time domain representation of filtered speech, is also a good basis on which to perform speech recognition. The simple and easily extracted features of frequency and energy in each filter band are adequate descriptors for most of the phonemes. All processing, including filtering and analysis, was done using single-precision fixed-point arithmetic on a relatively small-scale computer, thus demonstrating that large computer systems are not a requirement for speech recognition or speech research. Because of the reduced complexity of our recognition system, the entire process of converting wavefunction parameter strings to phoneme strings is faster than real time.

In general, the recognition scores represent accuracy and consistency in the method of transformation and all the intermediate processes which lead to the final recognition. Although the results shown are for a single speaker only, it is believed that speaker dependency will eventually be dealt with in the pattern recognition phase. All processes leading to that step have been developed with the speaker-independence in mind.

D. Summary and Recommendations

Phoneme recognition is a multistep process with each stage acting to simplify and reduce the output of the previous stage. Throughout the course of this research, it became necessary to develop techniques to perform these steps. Thus, in working toward the final objective, the following major results were obtained.

1. High-Speed Digital Filtering

Prefiltering speech is a prerequisite for wavefunction analysis and not a requirement for recognition. While it does not constitute a data reduction, it is a form of simplification. Its purpose is to expand the complex acoustic data into a set of substrings, each of which can be described as a series of coupled wavefunctions. Since filtering accounted for most of the processing time, it was desirable to have it run as fast as possible within the computer. As a result, a method for very fast nonrecursive digital filtering was developed in which over three fourths of the coefficients of the filter impulse response are forced to zero by taking advantage of its symmetry and through the judicious choice of filter center frequency, bandwidth, and window. One-octave frequency bands are obtained using the same set of coefficients for each filter operation provided that either the data is "decimated" or the impulse response is "stretched" prior to each pass. This technique turned out to be faster than FFT convolution for filters having up to 300 coefficients and therefore enabled us to process larger amounts of data in a shorter period of time.

2. Voice Detection

Speech sounds can be divided into two fundamental sets; those which are voiced and those which are unvoiced. An algorithm was devised which permits the detection of voicing through an examination of the A and C parameters of the 150-400 Hz frequency band. It is a one-pass operation with the time of the voiced/unvoiced boundaries being output sequentially. The algorithm has been used primarily on male speech, but it has also worked successfully on female and childrens' speech.

During the process of voice detection the pitch-period itself is being measured and is made available to other pitch-synchronous operations. It is a faster-than-real-time operation, and although it was developed for isolated words, it is just as applicable to continuous speech.

3. Definition of Recognition Features

Based upon an examination of wavefunction parameters for a large number of vowels, it was determined that frequency and energy estimates for each band were distinctive enough to qualify as features for recognition. Frequency/energy descriptions are generated pitch-synchronously in the voiced sounds. If a single formant is bounded by a particular filter band, then the frequency estimate for that band will closely approximate the frequency of the formant peak. An important outcome in this area was the extraction of a special feature from the wavefunction parameters which enables accurate discrimination between the vowels /a/ and /ɔ/.

4. Table-Driven Tree Recognizer

Taking advantage of the repetitive processes in the traversal of a binary tree, a table-driven tree recognizer was implemented for pattern classification. A recognizer of this type is fast, simple, and flexible. For the case of the linear two-dimensional decision element, every seven entries in a table completely define a node of the tree. In this form the recognizer could be thought of as a sequential machine whose operational sequence is defined by the recognizer (state) table. The flexibility of this approach is in the capability to change tables and thus adapt the recognizer to a new speaker.

5. Embedded Vowel Recognizer

A recognizer was implemented for twelve vowels by a single speaker. Inputs were restricted to words containing any sequence of vowels as long as those vowels were separated by unvoiced consonants. Nearly 98 percent accuracy was obtained for the speaker on which the system was trained. This was a significant step for the following reasons:

a) It demonstrated that meaningful recognition features could be extracted from wavefunction parameters.

b) Error rate was very low, thus giving further justification for the use of wavefunction analysis as a speech-processing tool.

c) The allowable input set of 12 vowels was much more complete than most vowel recognizers, which usually accept between five and 10 vowels.

6. Phoneme Segmentation

The primary objective of this research was to determine the applicability of wavefunction parameters to phoneme recognition. In order to extract features from representative phoneme samples it was necessary to develop a method for phoneme segmentation. Segmentation of arbitrary speech is a difficult problem, so a restricted-input phoneme boundary detector was used to define the intervals over which features were to be extracted. For the unvoiced sounds, silence was used as a phoneme separator. For the voiced sounds, transitions between peaks and valleys of a composite amplitude envelope were related to the transitions between vowel/consonant or consonant/vowel phoneme sequences. The number of words which could be input to such a limited segmenter was large enough to permit a meaningful number of phoneme samples to be generated.

In spite of the restriction placed on phoneme sequences within a word, the segmentation techniques and the concepts involved are most useful and important. The accurate detection of silence, though seemingly simple, provides:

a) A form of time separation between certain pairs of unvoiced phonemes.

b) A clue as to what type of unvoiced consonant may occur next.

c) An indication of a potential word boundary.

The composite amplitude envelope is obtained by adding together the individual amplitude envelopes of three filter bands. This envelope function is an excellent indicator of acoustic events, whether they be phonemes or syllables. As a primary segmentation function, the peaks and valleys of the composite envelope are related to vowel-like and consonant-like sounds respectively. Since the detection of peaks and valleys on the envelope is a function of their amplitude differences, this technique has the advantage that phoneme duration has reduced effect on segmentation accuracy. This level of segmentation appears to be applicable to male, female, and child speakers.

7. Phoneme Recognition

All of the previously summarized processes were put together to make up a single-speaker phoneme recognition system capable of identifying one of 32 possible phonemes with an accuracy of 92 percent. This was done by generating a reference set of 1170 phonemes, crossplotting their corresponding features, and building three recognizer tables; one for vowels, one for voiced consonants, and one for unvoiced consonants.

The recognition score for the unvoiced consonants was highest at 96 percent. While the phonemes /f,θ/ were not included in the recognizer, the achievement of this degree of accuracy demonstrates that the phonemes are adequately represented by their descriptive features, and that difficult phonemes, such as /p,t,k/, can be treated satisfactorily with our techniques.

The accuracy in recognizing the vowel-like phonemes was 92 percent. This is quite good considering that the allowable input set consists of twelve vowels as well as the vowel-like occurrences of /ℓ/. The voiced consonant recognition score rested at 86 percent. Separation of the consonants /b,d,g/ was not attained and likewise the phonemes /v,th/ could not be distinguished from each other. However, excellent results were obtained for /y,ℓ,r,w,z/ and the three nasals.

8. Wavefunction Analysis as Recognition Tool

The overall results of this research demonstrate that wavefunction analysis can be successfully applied to the various phases involved in (acoustic) speech recognition. This approach to speech analysis, being a data-dependent process, generates a nearly continuous description of the time and frequency behavior of filtered speech with a finer resolution than repetitive fixed-interval processing. This results in a solid information base on which recognition elements can be built.

In general, with regard to the analysis technique, the results of this study exemplify:

a) The accuracy of wavefunction parameters in representing speech behavior.

b) The ability of each processing step to systematically reduce the parametric description of the acoustic signal to elementary sound elements.

c) That the features derived from wavefunction parameters provide a unique and consistent description of many phonemes of the English language.

d) That recognition results can be achieved from a wavefunction parameter base in faster than real time.

9. Recommendations for Further Research

Successful recognition of the phonemes in isolated words by a single speaker is the first step in the direction of automatic speech recognition. In the acoustic (as opposed to the linguistic) phase of speech recognition the next higher level of interest lies in the identification of the phonemic elements of continuous or conversational speech of multiple speakers. This study, being the first application of wavefunction analysis to phoneme recognition, provides a foundation for the pursuit of the following areas of research.

a. Multiple-Speaker Recognition

A complete set of reference data and crossplots now exists for a single speaker. Studies could be made with regard to other speakers using the system. For a new speaker, is a simple adjustment of the decision lines all that is necessary to recognize his phonemes, or is a completely new set of crossplots required? Research in the area of inter-speaker and intra-speaker variation of the recognition features would be most useful.

b. Continuous Speech

Many of the present techniques are directly applicable to continuous speech. The detection of voicing, segmentation and feature extraction are not limited to isolated words. To demonstrate this, a simulation of continuous speech was made by recording the phrase, "we know two bad boys", spoken at a faster-than-normal rate. Only the first four words fit into our 609 msec speech buffer,but this was sufficient. The results of processing that phrase up through segmentation are shown in Figure III-j. All segments have been correctly preclassified as one of the three phoneme groups, thus illustrating the proper operation of all steps.

More research is necessary to develop a more general segmentation technique for arbitrary phoneme sequences. For the voiced phonemes, this would involve the detection of boundaries between consecutive phonemes of the same class. One suggestion might be to make use of a secondary function such as the pitch synchronous frequency data which is generated as part of the feature extraction. For example, Figure III-k contains the pitch synchronous frequency tracks for Bands 2, 3 and 4 for most of the phrase, "may we all learn". The bottom curve is the corresponding amplitude envelope. In this illustration we can see that the presence of the vowel /i/ in "we" is masked in the composite envelope. However, the rapidly changing frequency data provides a strong indication that another phoneme (/i/) exists between the valley corresponding to the /w/ of "we" and the peak occurring at the /ɔ/ of "all". Further study in this area is a necessity if research in continuous speech is initiated.

c. Phoneme Features

The information content and usefulness of phoneme transitions is another subject which merits study. With particular regard to stops and plosives such as /b,d,g/, it would be extremely valuable if unique features for these phonemes could be obtained (as with the vowels /a/ and /ɔ/) from the detailed information content of the wavefunction parameters. Likewise, further research in the area of distinctive features which can be derived from wavefunction parameters might be worthwhile.

d. Pattern Recognition

Researchers may also wish to pursue the application of different pattern recognition techniques to the identification of phonemes represented by n-dimensional feature lists. This could also include a study of learning algorithms for the purpose of adapting the recognizer to a new speaker. For the case of the recognizer used in this research, this would result in an automatic table generator based upon an appropriate number of phoneme samples from the new speaker.

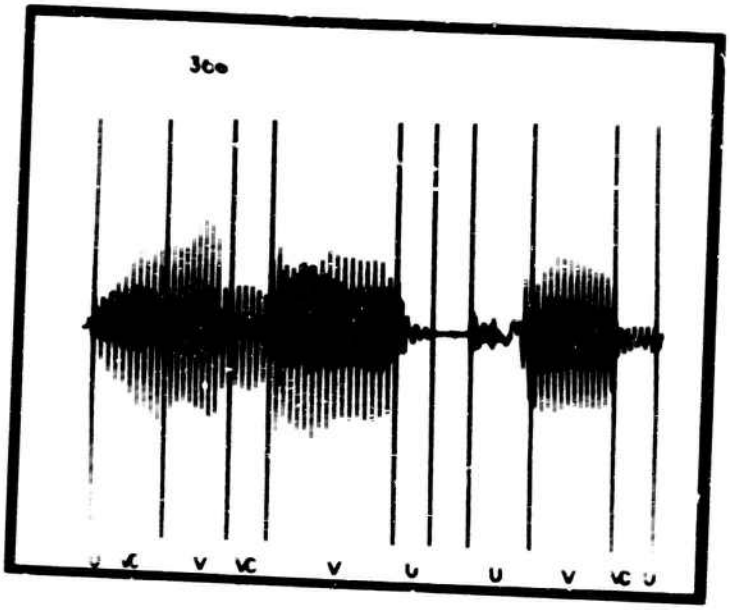


Figure III-j Correct segmentation of the first four words in the phrase, "we know two bad boys".

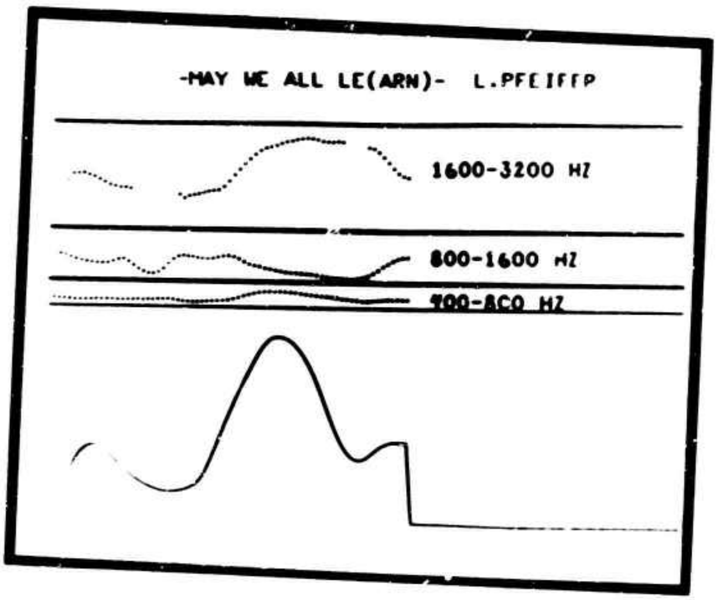


Figure III-k Frequency tracks for Bands 2, 3 and 4 and composite envelope (bottom) for the phrase, "may we all le(arn)".

e. Syntactic Analysis

Phoneme identification is not the last step in the overall speech recognition process. There is room for research in the analysis, manipulation and recognition of phoneme strings. This would include syntactic analysis techniques, error detection and correction, and possibly feedback to the acoustic recognizer. The recognizer discussed in this dissertation could be used as a phoneme generator for linguistic studies.

10. Conclusion

The recognition system presented in this report has met the objectives of this research. The techniques developed herein are original and the results reflect favorably on the wavefunction representation of speech. This work did not solve "the speech problem", but it must be considered a step forward.

IV. INTERACTIVE SIGNAL ANALYSIS SYSTEM

The SEL signal processing system has been developed as a tool for studying various techniques for processing human speech. The system has been used to implement the high speed algorithm for digital filtering of signals into octave bands and wavefunction analysis and synthesis techniques, which were developed at UCSB during previous ARPA contracts.

A. Technical Problems

Research during the report period has aimed at the following goals:

1) Providing a general purpose experimental interactive system for signal analysis.

2) Investigating the techniques of wavefunction analysis for connected speech.

3) A continued study of the interrelationships between a time-domain wave packet representation of a signal and its frequency domain representation.

4) An investigation of the use of the wavefunction representation in the data compression of digital speech.

B. General Methodology

The signal processing system described here is functionally similar to the UCSB On-Line System. Programming of the system is accomplished at an on-line terminal and display is presented to the user on a Tektronix 611 storage display unit. Programs to perform input and output of real-time sampled data and buffering to disk were written early in the report period. To off-set the limited storage capacity of the SEL 810B processor, system programs are swapped from disk or drum.

The techniques of digital filtering and wavefunction analysis have been implemented and modified to allow representation of a continuous stream of speech. A new technique for generating a frequency domain representation of connected speech from a set of wavefunction parameters has been developed to allow evaluation of the frequency spectra over any frequency interval. A program for displaying orthographic projections of the spectra generated from wavefunctions has been developed.

A new technique for the data compression of speech for digital transmission has been implemented through sorting and quantizing wavefunction parameters.

C. Results

Development of the system described here has been completed and is being applied to the study of connected speech. The present digital filtering scheme runs in twice real time. Wavefunction analysis runs in slightly more than real time. The process of creating frequency spectra from the wave packets runs in slightly more than real time.

Quantization of the wavefunction parameters in performing data compression of speech has produced intelligible speech at a 4.6 kilobit rate.

D. Important Equipment Developed

Equipment developed for the signal analysis system includes interfaces to two IBM 1311 disk drives, a 96K word drum, an A/D converter and a D/A converter. The SEL system has also been connected to the IBM 1800 system in order to make use of the 1800 peripherals and to the 360/75 system to gain access to the ARPANET.

Computer Systems
Laboratory-UCSB
Roland F. Bryan
30 June 1972

APPENDIX II-A

UCSB COMPUTER SYSTEM RESOURCES

The UCSB Computer Center has evolved under the mandate to provide reliable and friendly service to its user community. Both local and remote users are served. Over the past several years the UCSB user community has grown to include the ARPANET and the UCSB Computer Center has been designated as a "Server" site on this nation-wide computer telecommunications network.

The UCSB system includes specialized hardware and software as shown in Figures II-a and II-b. The key to ARPANET operation is the Network Control Program (NCP) which now links users on the network with resources at the Center. Since the NCP is bi-directional, all of the resources of the ARPANET are now available to the UCSB user community as well.

The following paragraphs describe the major hardware elements shown in Figure II-a.

<u>IBM 360/75 with Standard Peripherals and Mass Storage</u> - A standard IBM System/360 with 512 K bytes of high-speed core, 2 M bytes of bulk core, and 400 M bytes of 2314 disk storage.

<u>Store and Forward Buffer</u> - This controller provides dynamically allocated core storage which is used for selector channel data transfer and control of the various attachments shown in the diagram. It deals with the 360 on a block transfer basis and requests service by way of external interrupts.

<u>IMP Interface</u> - A hardware interface that provides the means for communication between the 360 and the Interface Message Processor (IMP) of the ARPANET. This controller operates on the Multiplexor Channel and assumes separate device addresses for Read and Write operations.

<u>Interface Message Processor (IMP)</u> - The basic communications processor for the ARPANET. The IMP provides access to the ARPANET for UCSB. Other Hosts to be attached in the future will be the PDP11 at SCRL as shown in Figure II-a.

High Speed Data Link - This interface controller provides a 50 K bits/sec attachment between a selector subchannel of the 360 and an SEL 810B computer in Electrical Engineering. Core-to-Core data transfers take place over this connection.

SEL 810B Processor - This processor, equipped with disks, drums and an on-line console is used principally for speech research. A user sitting at the console of this computer may gain access to the 360 for time shared or batch processing as can any other user of the system. The SEL 810B is located in Electrical Engineering.

IBM 1800 - This processor is equipped with disks, readers, printers, and consoles plus a direct transfer path to the SEL 810B. The 1800 is used principally for speech processing. It has additional equipment for video and graphic input processing as well.

Multi-Line Controller (MLC) - This unit is attached to the multiplexor channel of the 360 and operates with multiple device addresses. To this controller are attached a number of synchronous, asynchronous, parallel, serial, and variable data-rate devices. All attachments operate independently under program control just as standard IBM control units. Attachments include mini-computers in Chemistry, Poli-Sci, and Physics and many display consoles such as Teleputers, Tektronix 4002, and IMLAC. The small processors and consoles have access to the ARPANET by way of the Network Control Program.

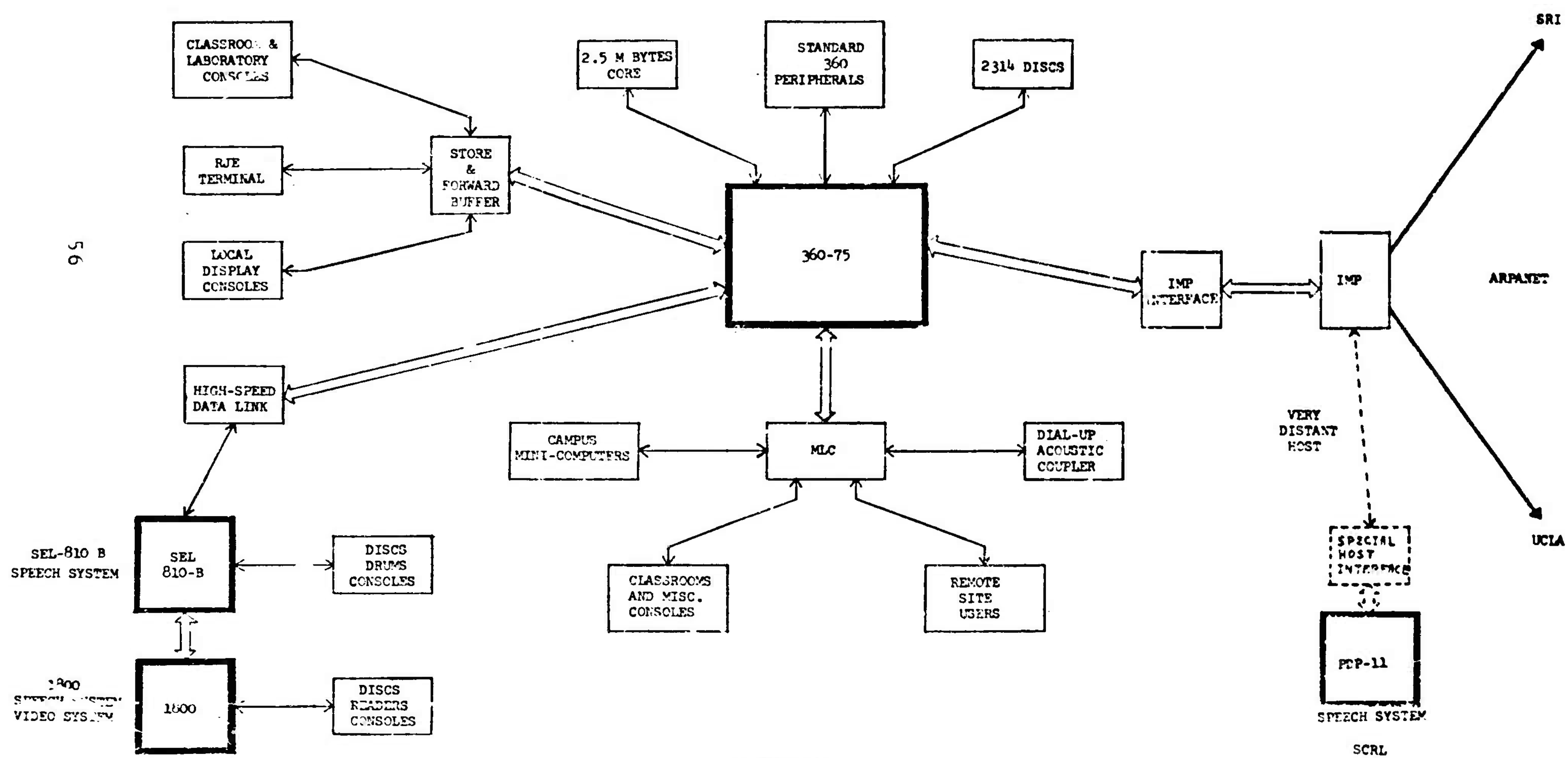
The major software elements are shown in Figure II-b and are described below.

OS Batch - The standard Operating System batch processing software that is included with the 360 installation. Interconnection is maintained between OS Batch, HASP, and the OS files.

HASP (RJE) - HASP provides scheduling of system operation for batch processing and remote job entry. Local batch input/output under operator control is handled by way of HASP. Submission of batch jobs by way of the Card Oriented Language (COL) and by way of Remote Job Service (RJS) are linked to OS Batch and scheduled by HASP.

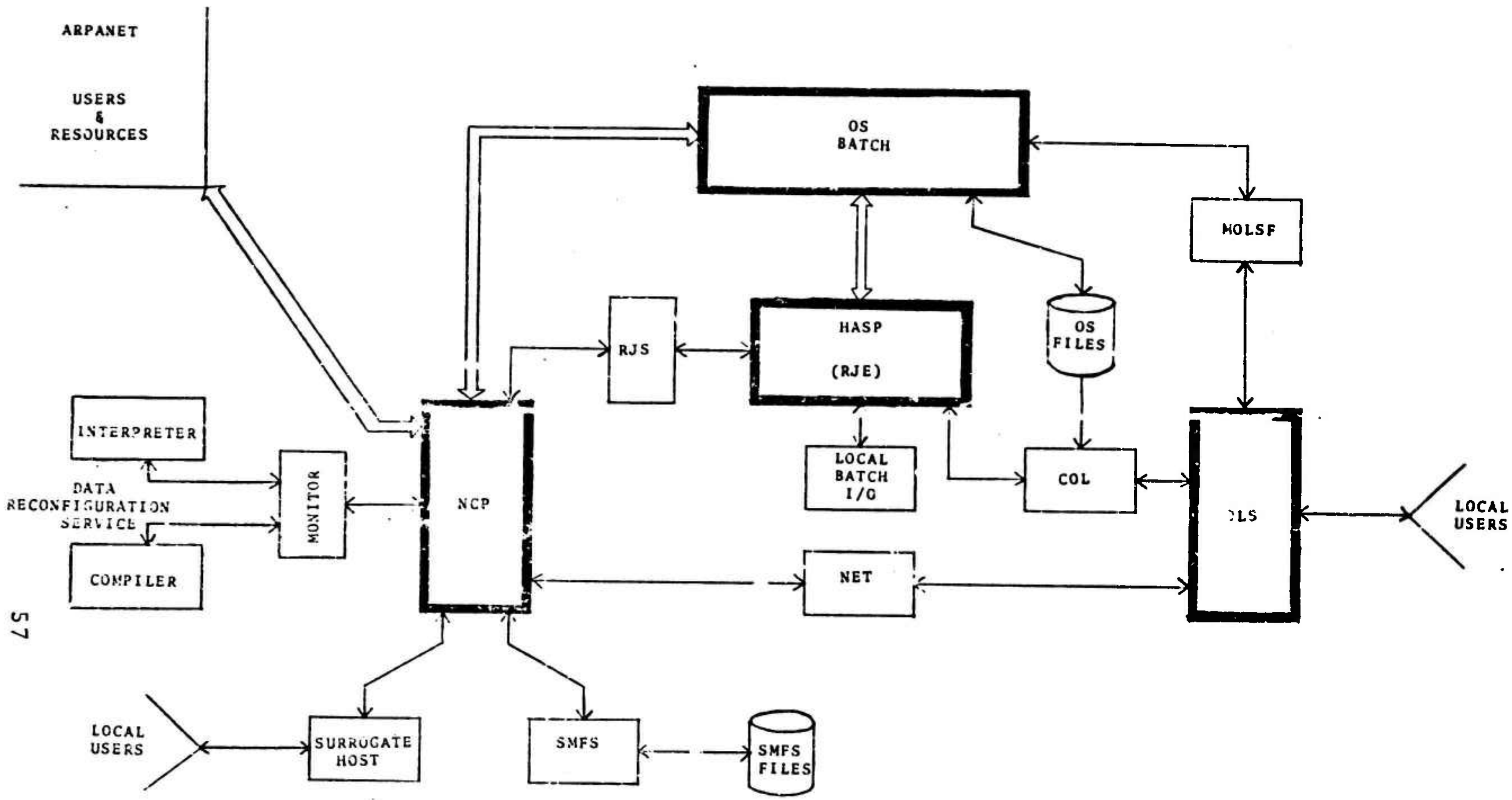
On-Line System (OLS) - The UCSB On-Line System for interactive graphics problem solving (designated the Culler-Fried System). This is the primary support for graphic display terminals using the UCSB complex. The basic programs are the Mathematical On-Line System (MOLSF) and the Card Oriented Language (COL). Links to and from the ARPANET are provided by the NET program package.

Network Control Program (NCP) - This control program provides the interface between the UCSB computer complex and the ARPANET. All external users as well as local users with mini-processors and consoles are linked to resources by way of the NCP. Network support programs such as Data Reconfiguration Service (DRS), Surrogate Host, and the Simple Minded File System (SMFS) are ajuncts to the NCP.



UC SANTA BARBARA COMPLEX

Figure II-a



UCSB SOFTWARE SYSTEM

Figure II-b

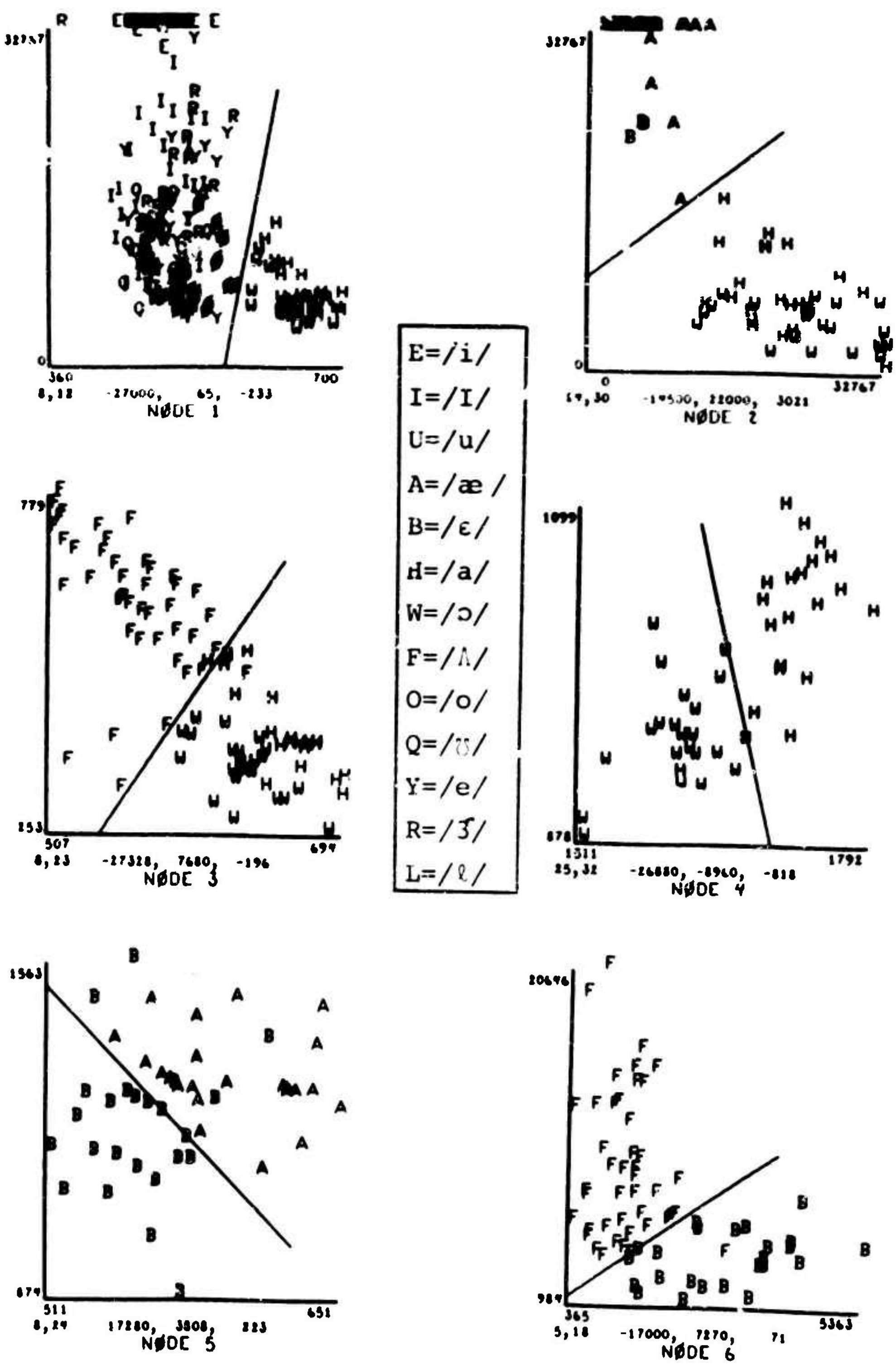
APPENDIX III-A

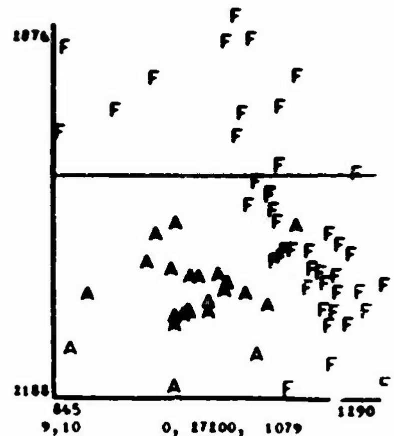
The following section contains supplementary information regarding the phoneme recognizer and the phoneme data base. The complete set of crossplots illustrates the features, decision line and data distribution of each node of a recognition tree. Nodes 1 through 30 correspond to the vowel recognition tree of Figure III-b. Nodes 100 through 139 correspond to the voiced consonant recognition tree of Figure III-c. Nodes 200 through 237 correspond to the unvoiced consonant recognition tree of Figure III-d. An explanation of a sample crossplot is given in Section B-3.

Following the crossplots are the tables which define the three recognition trees. The labels XCON, YCON and THRES correspond to the constants A, B and C, respectively, in the decision function

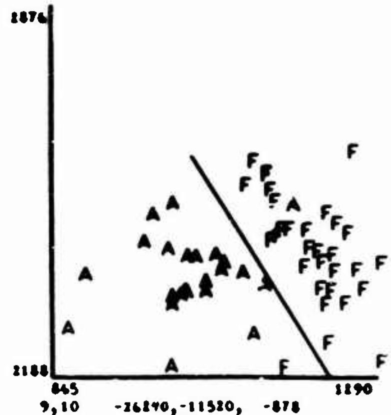
$$Ax+By = C .$$

XITEM and YITEM specify the feature number in the expanded feature list of Table III-3, these features corresponding to x and y respectively. Therefore, if $Ax+By \geq C$ the next node in the sequence is the one under the heading PASS, if $Ax+By<C$ the next node in the sequence is the one under the heading FAIL. The tree (table) search is terminated when a phoneme code is encountered as the next node.

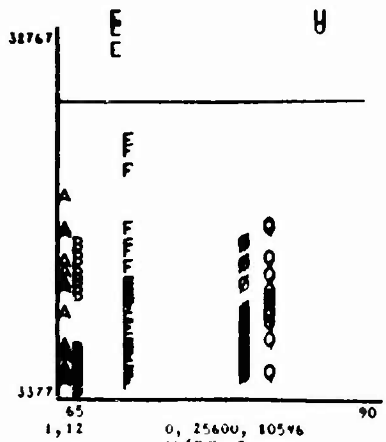

NØDE 1
NØDE 2
E=/i/
I=/I/
U=/u/
A=/æ/
B=/ε/
H=/a/
W=/ɔ/
F=/ʌ/
O=/o/
Q=/ʊ/
Y=/e/
R=/ɝ/
L=/ℓ/
NØDE 3
NØDE 4
NØDE 5
NØDE 6



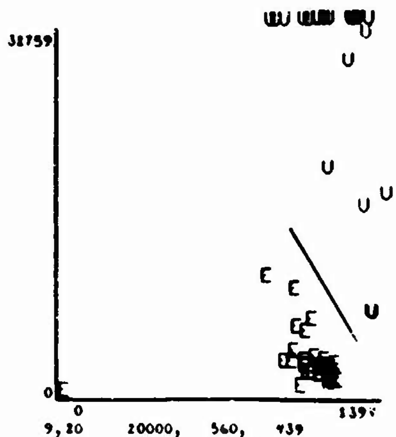
NØDE 7



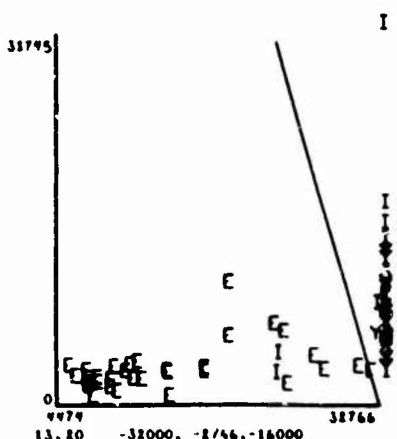
NØDE 8



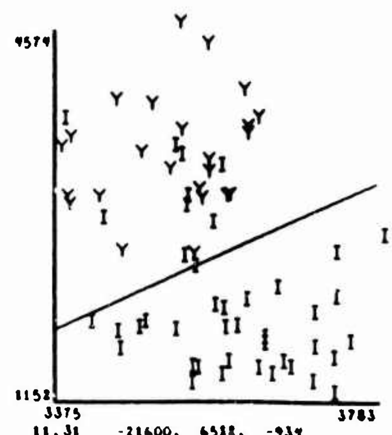
NØDE 9



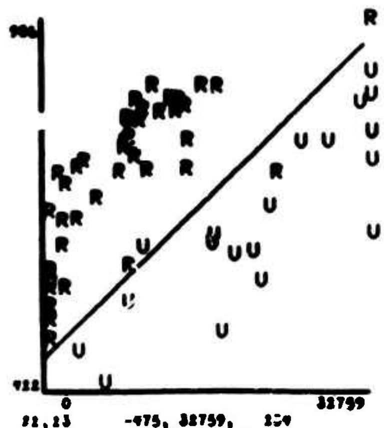
NØDE 10



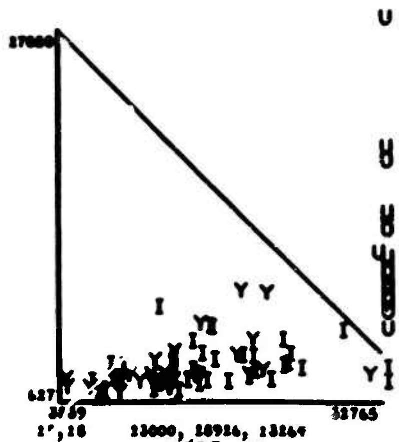
NØDE 11



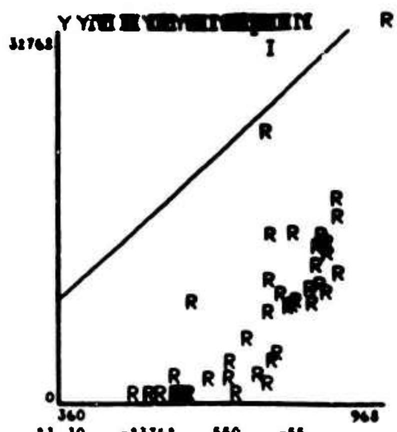
NØDE 12



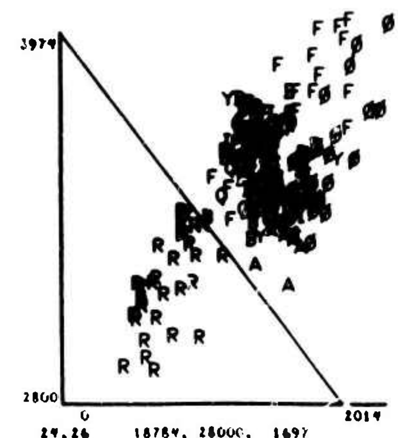
NØDE 13



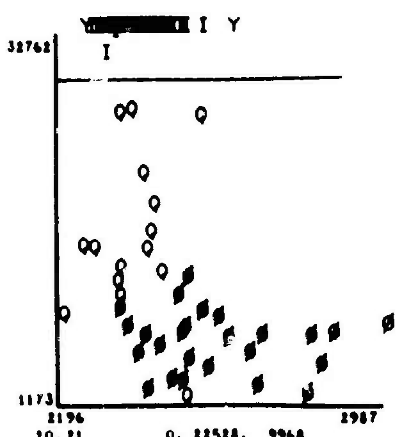
NØDE 14



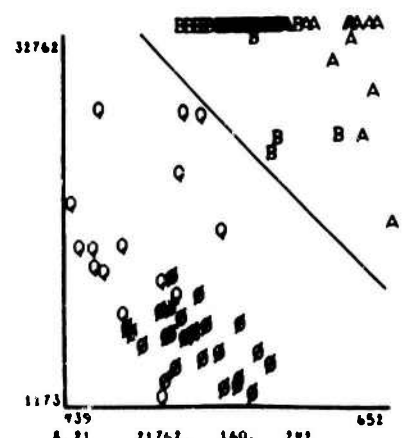
NØDE 15



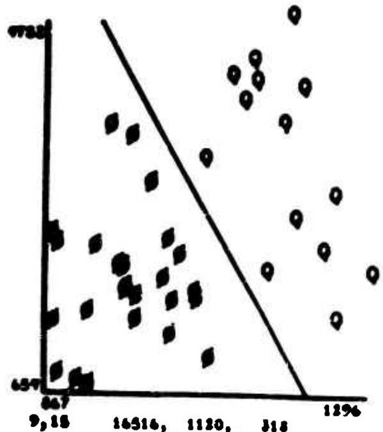
NØDE 16



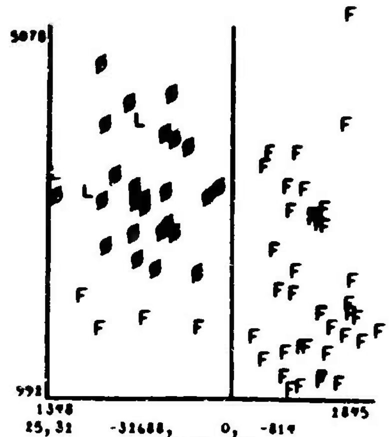
NØDE 17



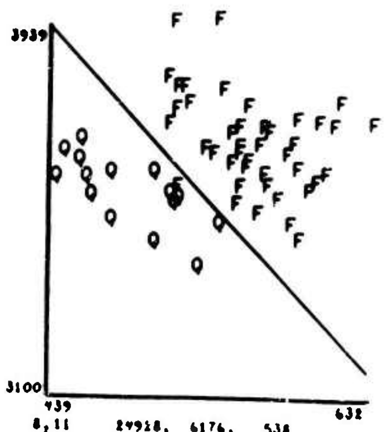
NØDE 18



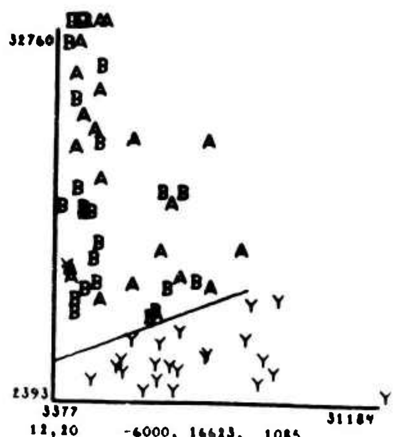
NØDE 19



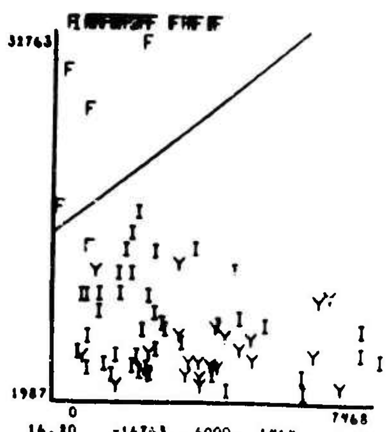
NØDE 20



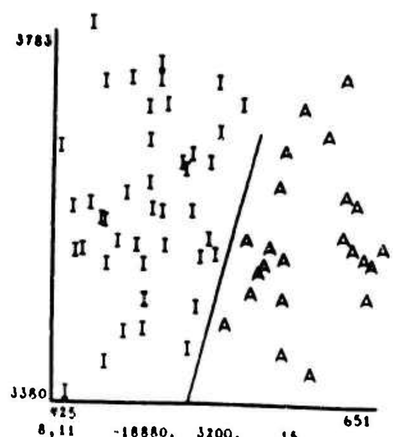
NØDE 21



NØDE 22

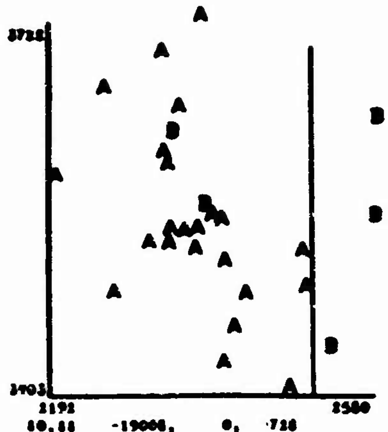


NØDE 23

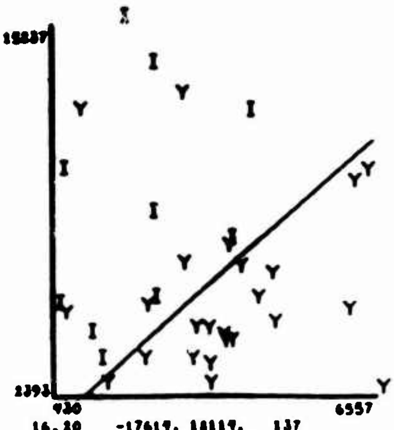


NØDE 24

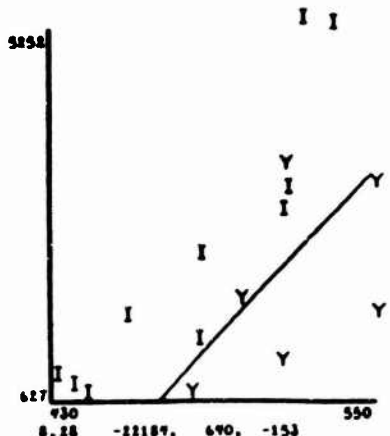
NØDE 25



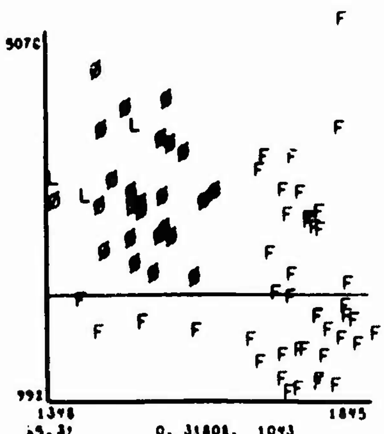
NØDE 26



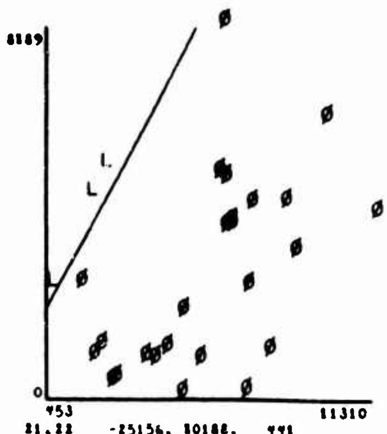
NØDE 27



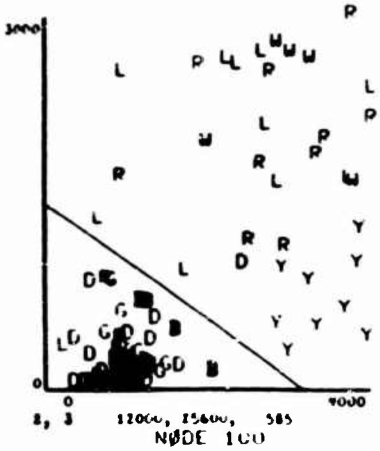
NØDE 28



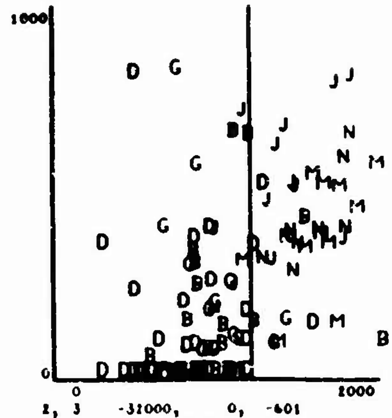
NØDE 29



NØDE 30



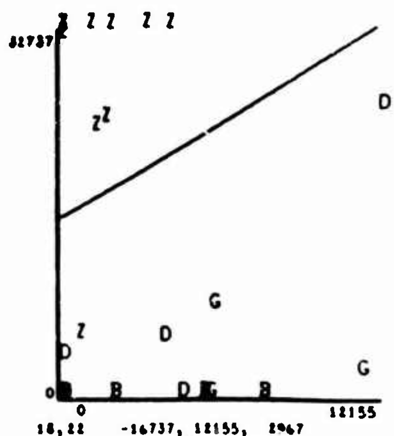
NØDE 100



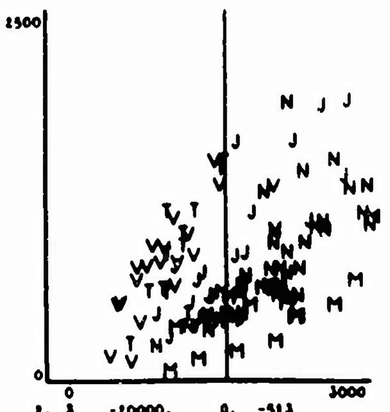
NØDE 101

B=/b/
D=/d/
G=/g/
Y=/y/
L=/ℓ/
R=/r/
W=/w/
M=/m/
N=/n/
J=/ŋ/
V=/v/
T=/th/
Z=/ʒ/

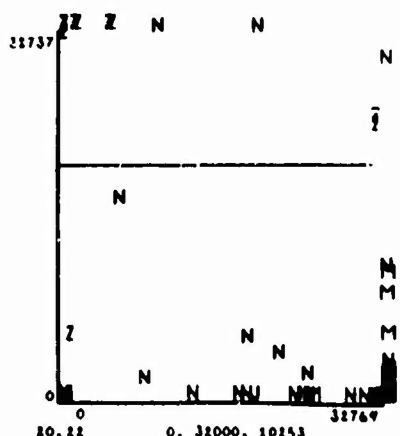
NØDE 102



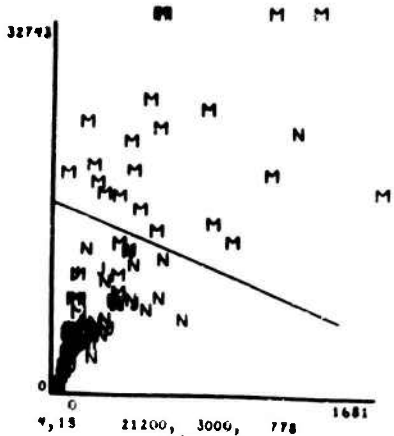
NØDE 103



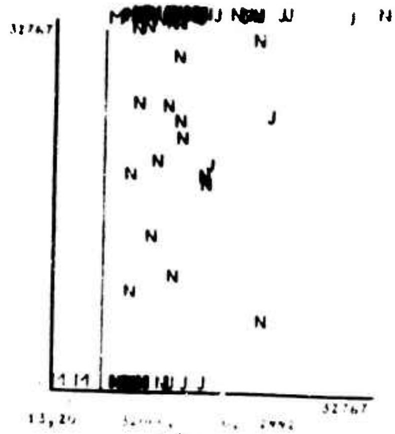
NØDE 104



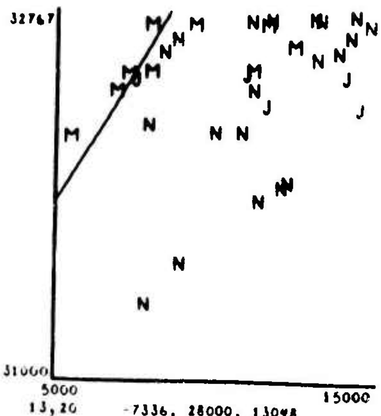
NØDE 105



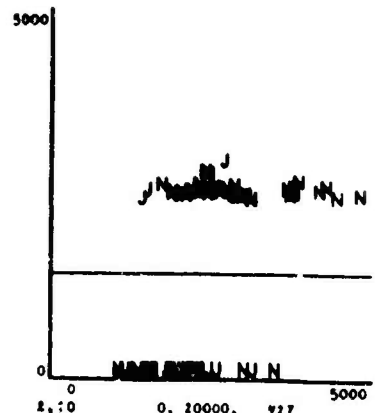
NØDE 106



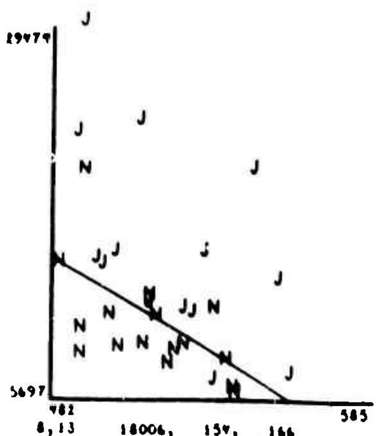
NØDE 107



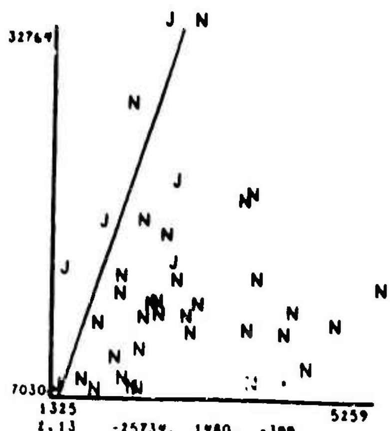
NØDE 108



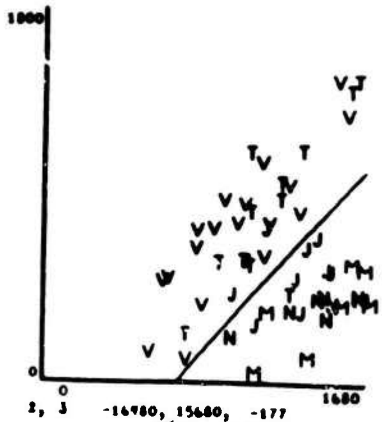
NØDE 109



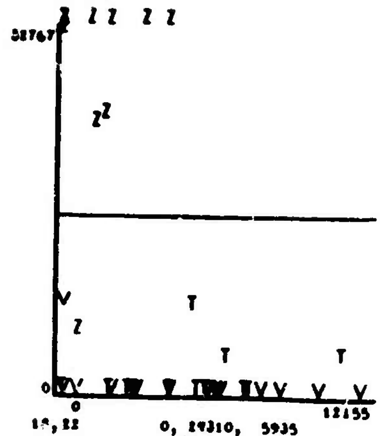
NØDE 110



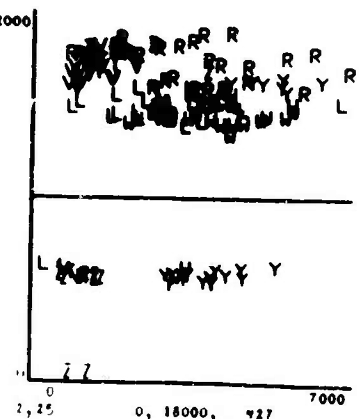
NØDE 111



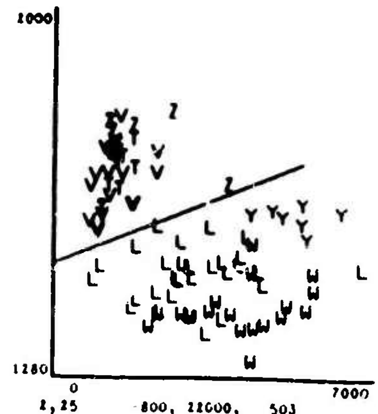
NØDE 112



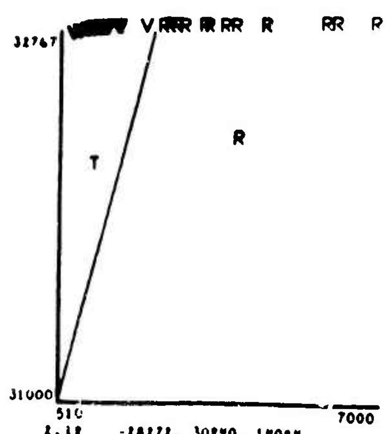
NØDE 113



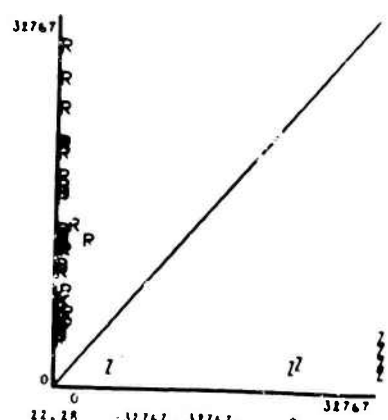
NØDE 114



NØDE 115

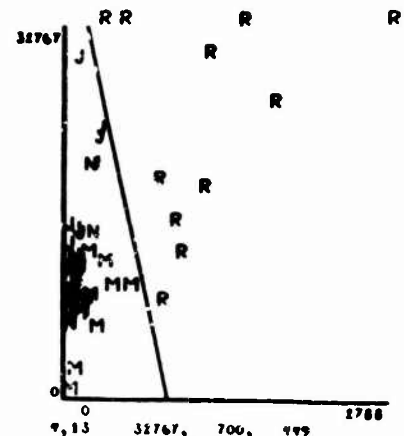


NØDE 116

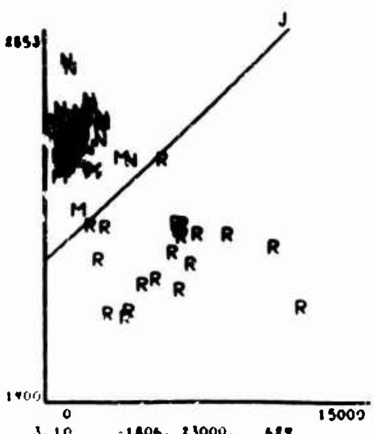


NØDE 117

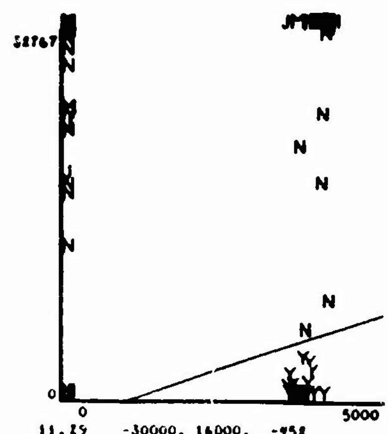
NØDE 118



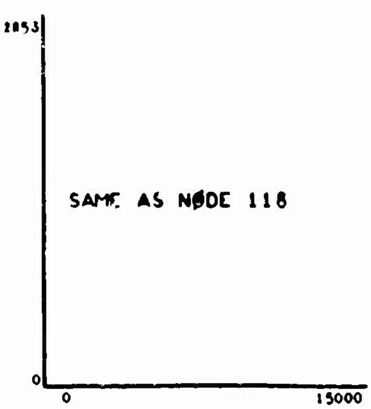
NØDE 119



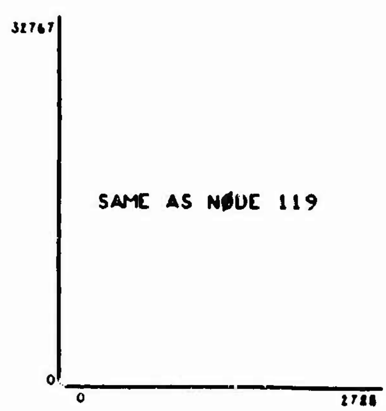
NØDE 120



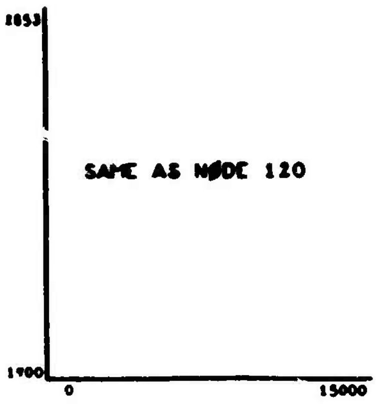
NØDE 121



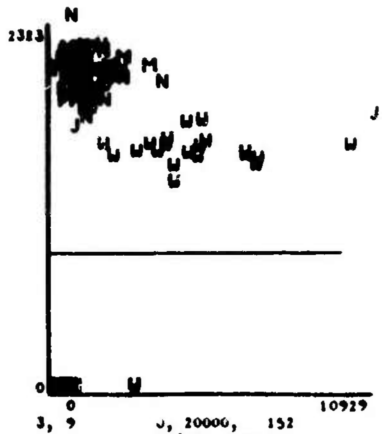
NØDE 122



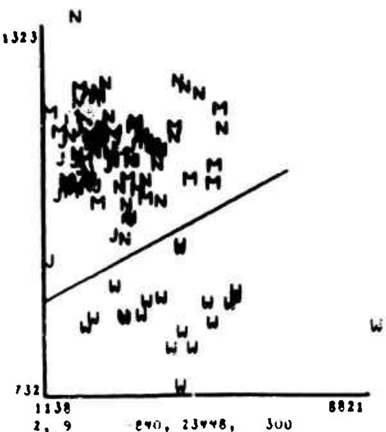
NØDE 123



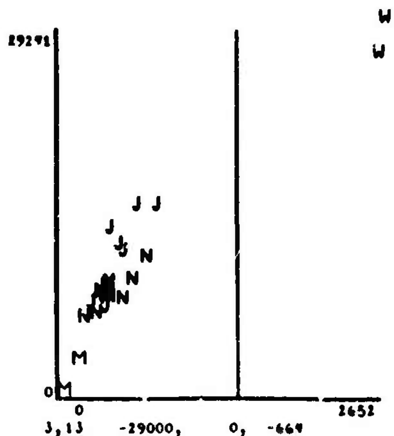
NØDE 124



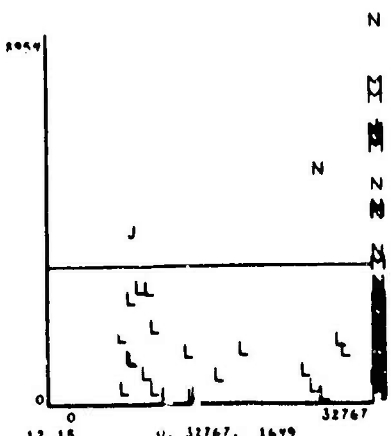
NØDE 125



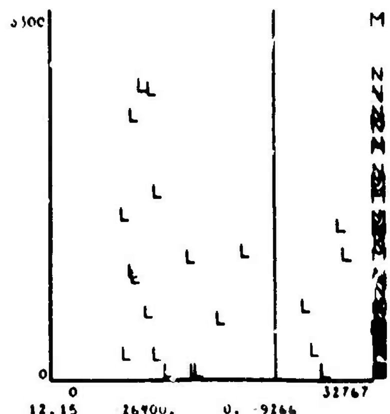
NØDE 126



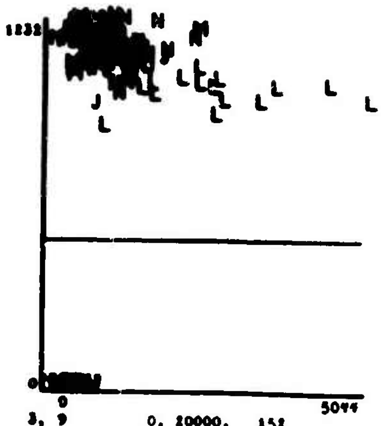
NØDE 127



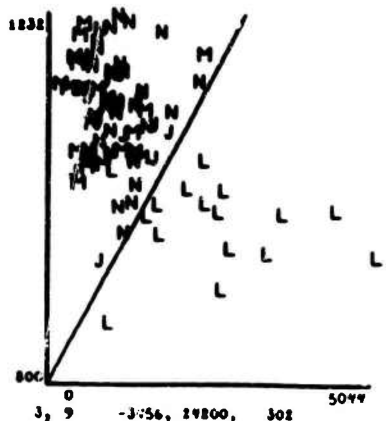
NØDE 128



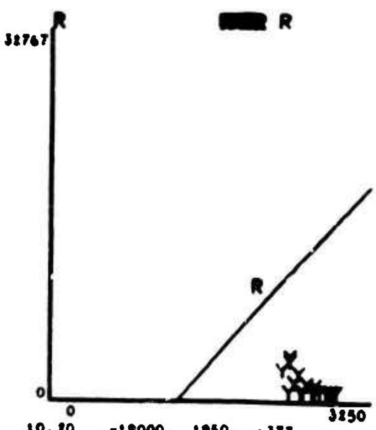
NØDE 129



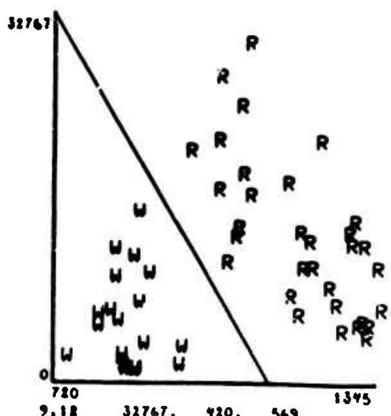
NØDE 130



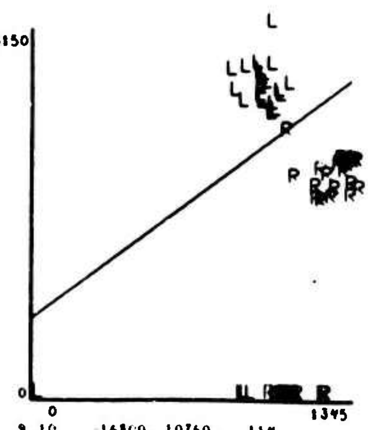
NØDE 131



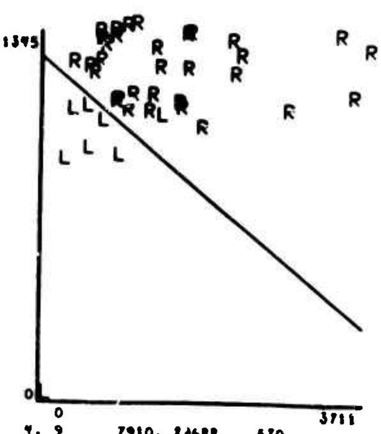
NØDE 132



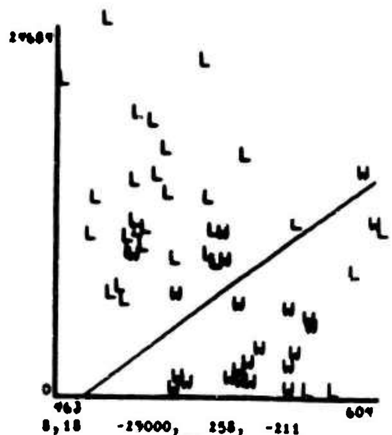
NØDE 133



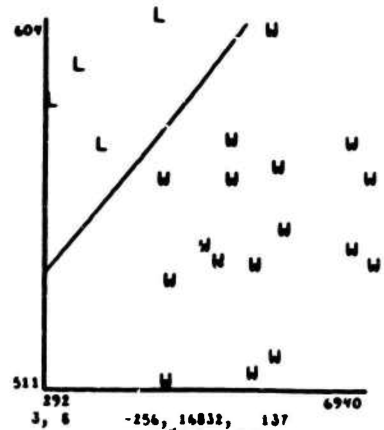
NØDE 134



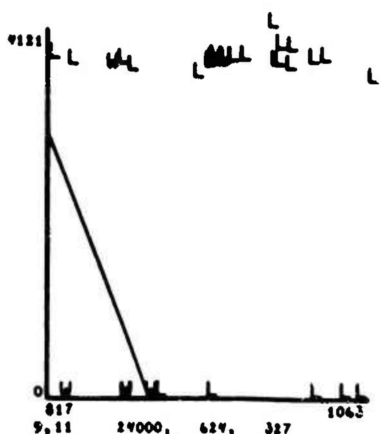
NØDE 135



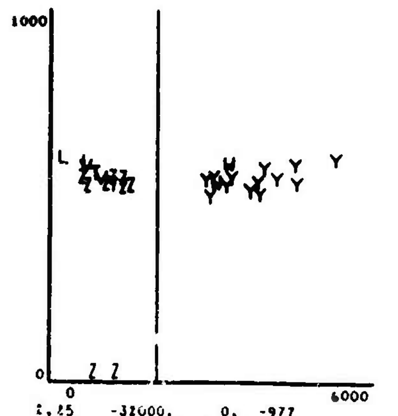
NØDE 136



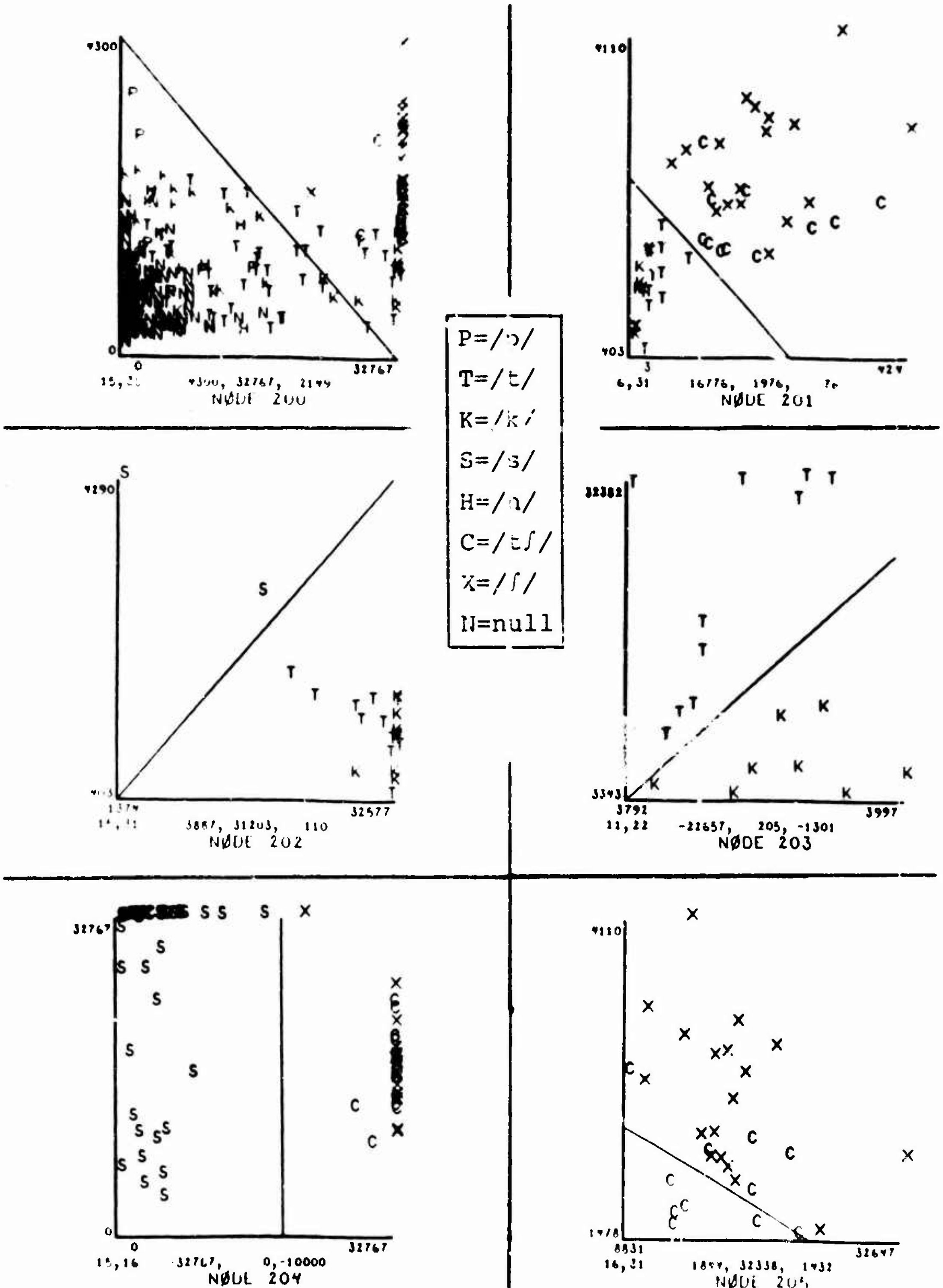
NØDE 137

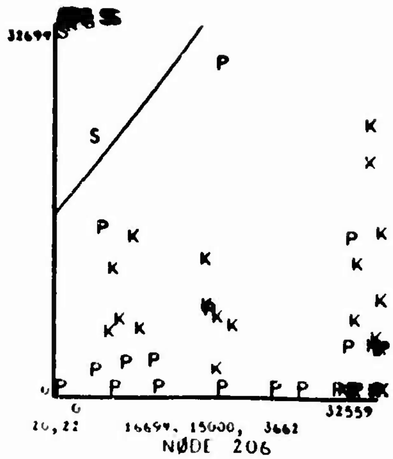


NØDE 138

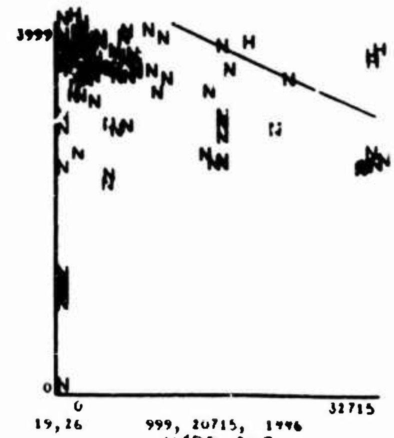


NØDE 139

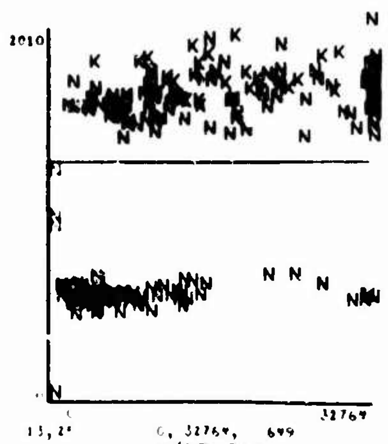

NØDE 200
NØDE 201
NØDE 202
NØDE 203
NØDE 204
NØDE 205
P=/p/
T=/t/
K=/k/
S=/s/
H=/h/
C=/tʃ/
X=/ʃ/
N=null



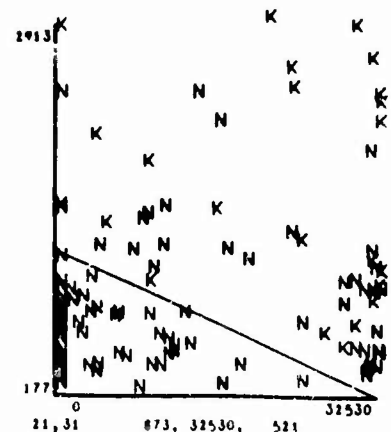
NØDE 206



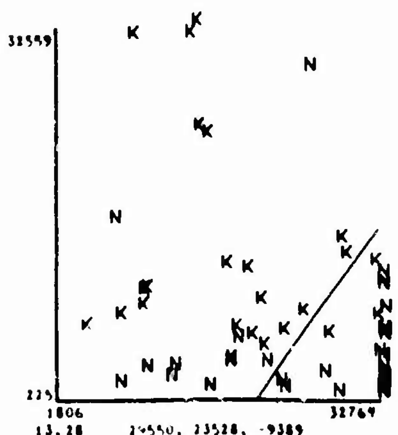
NØDE 207



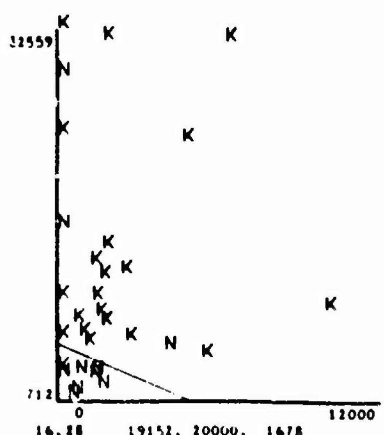
NØDE 208



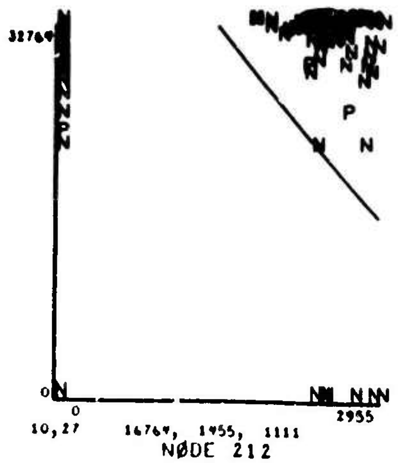
NØDE 209



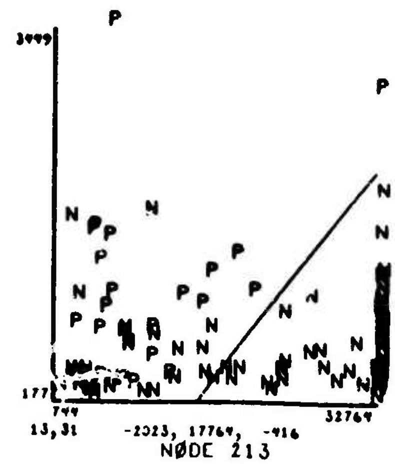
NØDE 210



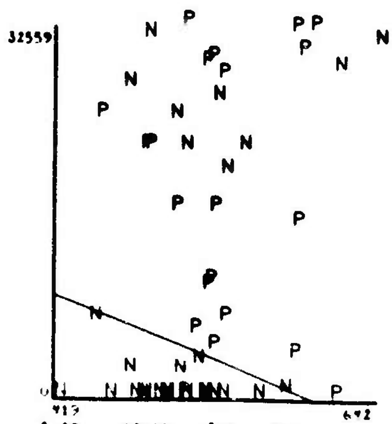
NØDE 211



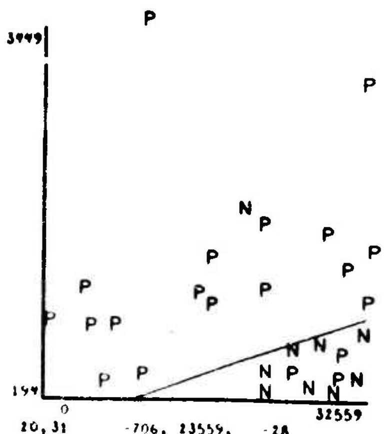
NØDE 212



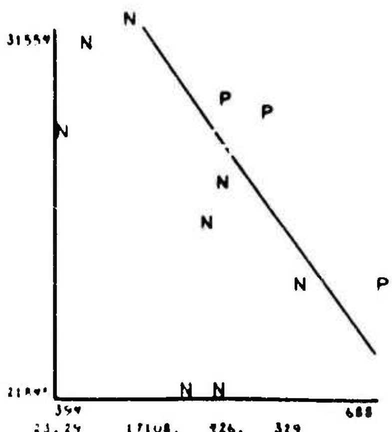
NØDE 213



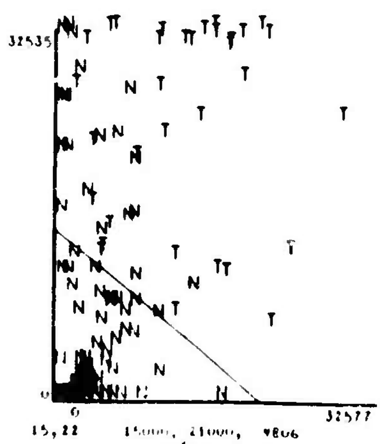
NØDE 214



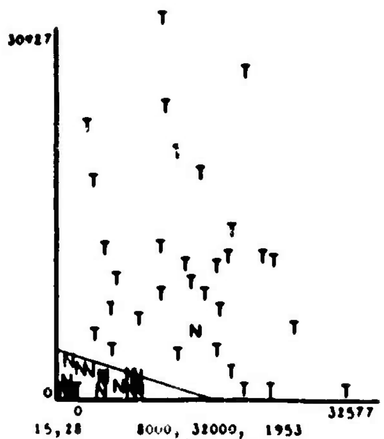
NØDE 215



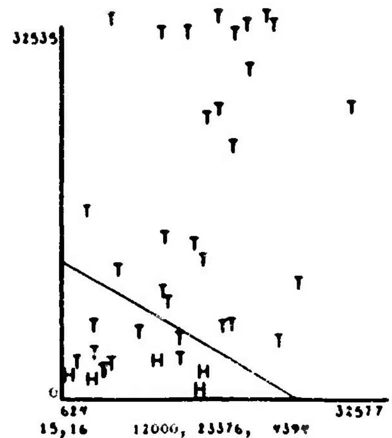
NØDE 216



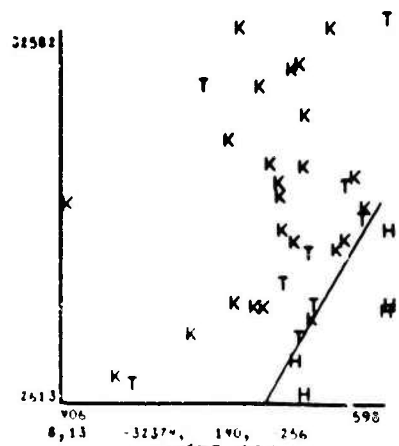
NØDE 217



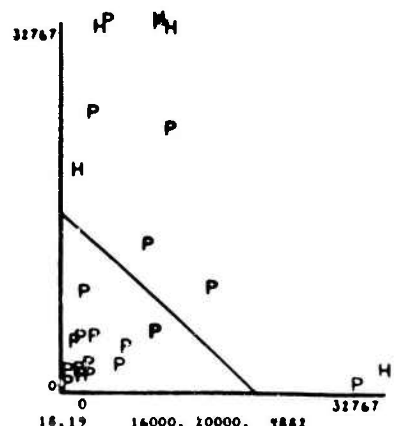
NØDE 218



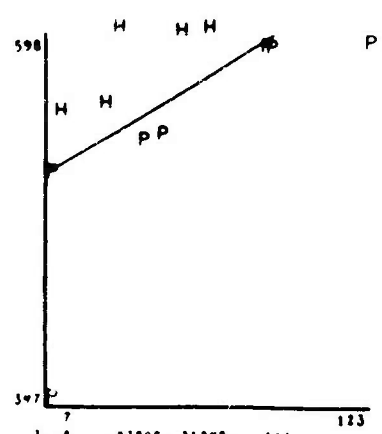
NØDE 219



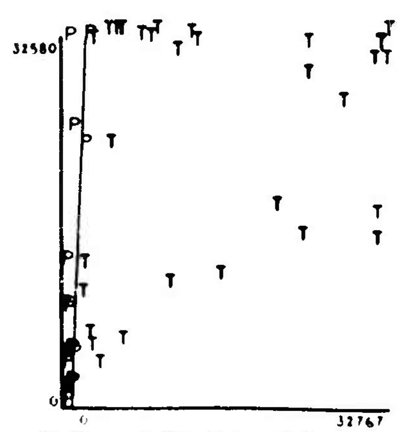
NØDE 220



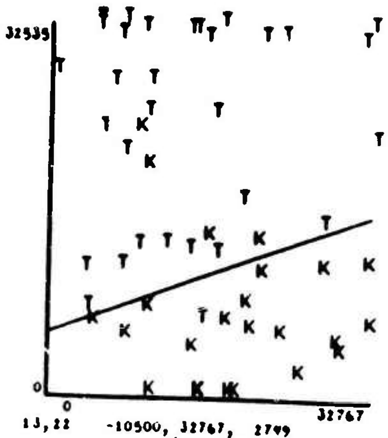
NØDE 221



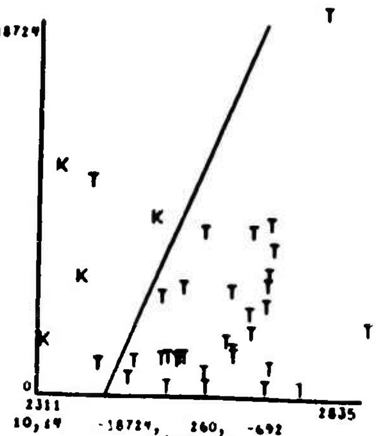
NØDE 222



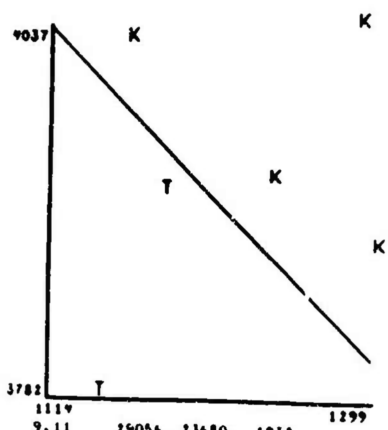
NØDE 223



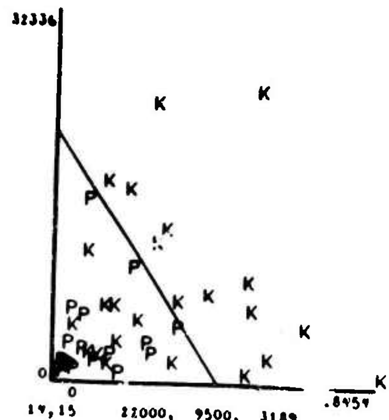
NØDE 224



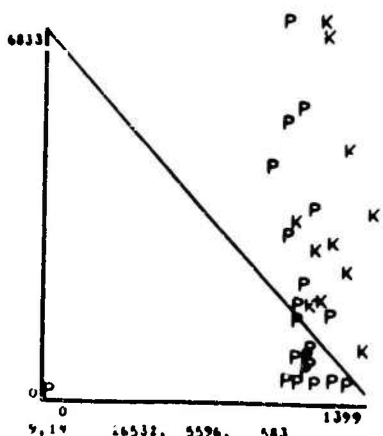
NØDE 225



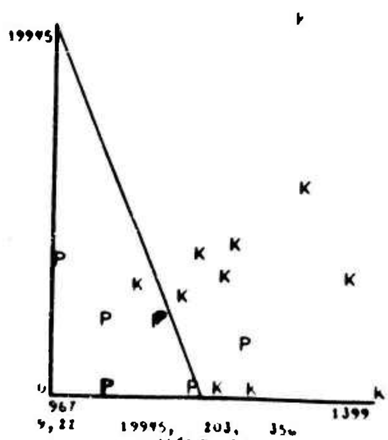
NØDE 226



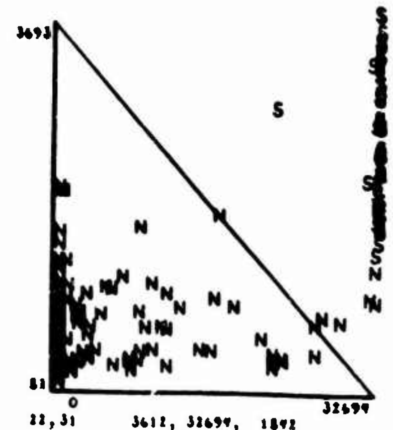
NØDE 227



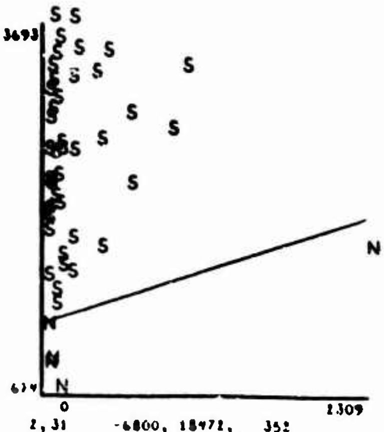
NØDE 228



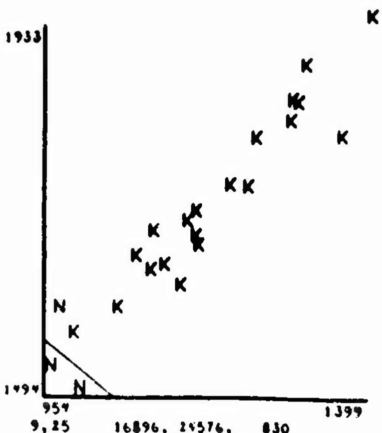
NØDE 229



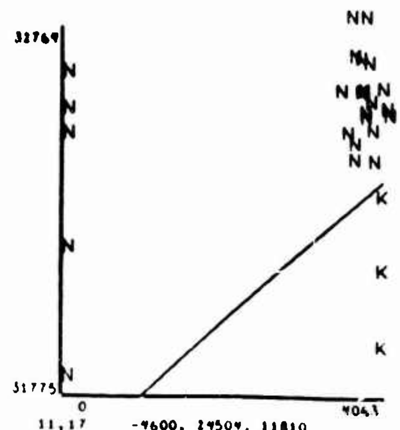
NØDE 230



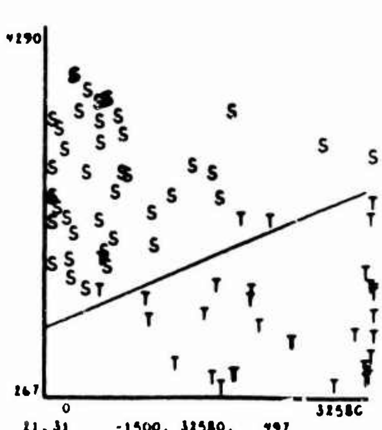
NØDE 231



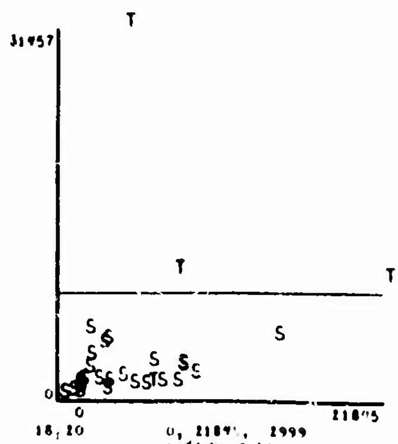
NØDE 232



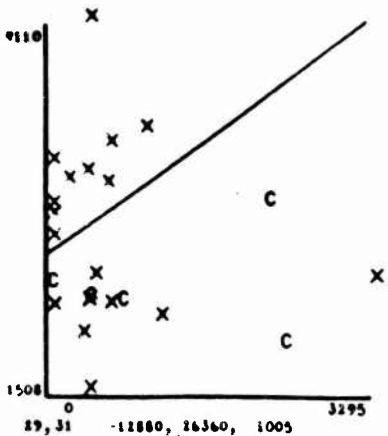
NØDE 233



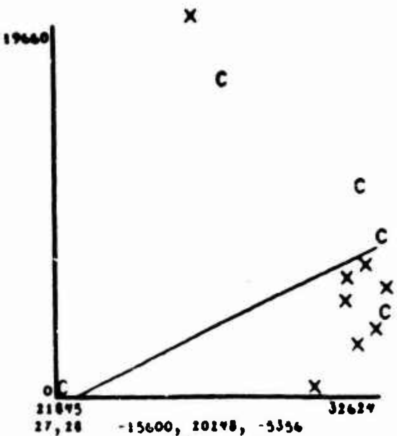
NØDE 234



NØDE 235



NØDE 236



NØDE 237

Table A.1

VOWEL RECOGNIZER TABLE

NODE	XITEM	XCON	YITEM	YCON	THRES	PASS	FAIL
N1	8	-27000	12	65	-233	N9	N2
N2	14	-14500	30	22000	3021	N5	N3
N3	8	-27348	23	7680	-196	ʌ	N4
N4	25	-26880	32	-8960	-818	ɔ	a
N5	8	17280	24	3808	223	N26	N6
N6	5	-17000	18	7270	71	ʌ	ɛ
N7	9	0	10	27200	1079	ʌ	N8
N8	9	-26240	10	-11520	-878	æ	ʌ
N9	1	0	12	25600	10546	N10	N16
N10	9	20000	20	560	439	N13	N11
N11	13	-32000	20	-8766	-16000	i	N12
N12	11	-21600	31	6528	-934	N27	I
N13	21	-475	23	32759	234	N15	N14
N14	12	23000	18	28926	13264	u	N12
N15	23	-32762	30	550	-55	N12	ɝ
N16	24	18784	26	28000	1697	N17	ɝ
N17	10	0	21	22582	9968	N22	N18
N18	8	21762	21	160	242	N5	N19
N19	9	16516	15	1120	318	N21	N20
N20	25	-32688	31	0	-8114	N29	ʌ
N21	8	24928	11	6176	538	ʌ	ʊ
N22	12	-6000	20	16623	1085	N24	N23
N23	16	-16763	20	6000	1464	ʌ	N12
N24	8	-18880	11	3200	15	N25	N5
N25	8	-20623	12	85	-152	N23	N6
N26	10	-19000	11	0	-728	N7	N6
N27	16	-17614	20	11114	137	N28	e
N28	8	-22184	28	640	-153	I	e
N29	25	0	31	31808	1043	N30	ʌ
N30	21	-25156	22	20188	442	ℓ	0

Table A.2

VOICED CONSONANT RECOGNITION TABLE

NODE	XITEM	SCON	YITEM	YCON	THRES	PASS	FAIL
N100	2	12000	3	25600	585	N114	N101
N101	2	-32000	3	0	-601	N103	N104
N102	8	-22400	9	20480	140	v	b
N103	18	-16737	22	12155	2967	z	N102
N104	2	-20000	3	0	-513	N112	N105
N105	20	0	22	32000	10253	z	N106
N106	4	21200	18	3000	778	m	N107
N107	13	-32000	20	0	-2442	m	N108
N108	13	-7336	20	28000	13048	m	N109
N109	2	0	10	20000	427	N111	N110
N110	8	18006	13	154	166	ŋ	n
N111	2	-25734	13	1480	-399	ŋ	n
N112	2	-16480	3	15680	-177	N113	N105
N113	18	0	22	24310	5935	z	v
N114	2	0	25	28000	427	N115	N139
N115	2	-800	25	22000	503	N116	N121
N116	2	-28272	12	30240	14084	N104	N117
N117	22	-32767	28	32767	0	N118	N105
N118	3	0	10	30000	640	N120	N119
N119	4	32767	13	700	449	r	N106
N120	3	-1806	10	23000	684	N106	r
N121	11	-30000	29	16000	-458	N122	N132
N122	3	0	10	30000	640	N124	N123
N123	4	32767	13	700	449	N133	N125
N124	3	-1806	10	23000	684	N125	N133
N125	3	0	9	20000	152	N126	N127
N126	2	-840	9	23448	300	N128	N136
N127	3	-29000	13	0	-664	N128	N136

Table A.2 (Cont'd.)

NODE	XITEM	XCON	YITEM	YCON	THRES	PASS	FAIL
N128	12	0	15	32767	1649	N106	N129
N129	12	-26400	15	0	-9266	ℓ	N130
N130	3	0	9	20000	152	N131	N106
N131	3	-3456	9	24800	302	N106	ℓ
N132	10	-19000	20	1950	-377	N133	y
N133	9	32767	18	420	569	N134	N136
N134	9	-16800	10	10720	114	ℓ	N135
N135	4	7920	9	29680	561	r	ℓ
N136	8	-29000	18	258	-211	N138	N137
N137	3	-256	8	16832	137	ℓ	w
N138	9	24000	11	624	327	ℓ	w
N139	2	-32000	25	0	-977	N104	N121

Table A.3

UNVOICED CONSONANT RECOGNITION TABLE

NODE	XITEM	XCON	YITEM	YCON	THRES	PASS	FAIL
N200	15	4300	31	32767	2149	N20	N206
N201	6	16776	31	1976	76	N204	N202
N202	15	-3887	31	31203	110	s	N203
N203	11	-22657	22	205	-1301	t	k
N204	15	-32767	16	0	-10000	s	N205
N205	16	1844	31	32338	1432	N236	tʃ
N206	20	-16694	22	15000	3662	N230	N207
N207	19	999	26	20715	1446	N219	N208
N208	13	0	25	32764	649	N209	N212
N209	21	873	31	32530	521	N210	N212
N210	13	-29550	28	23528	-9389	N211	N233
N211	16	19152	28	20000	1678	N232	N212
N212	10	16764	27	1455	1111	N213	N217
N213	13	-2023	31	17764	-416	N214	N217
N214	8	18000	20	362	164	N215	N217
N215	20	-706	31	23559	-28	N223	N216
N216	23	17108	29	426	329	N223	N217
N217	15	15000	22	21000	4806	N218	null
N218	15	8000	28	32000	1953	t	null
N219	15	12000	16	23376	4394	N223	N220
N220	8	-32374	13	140	-256	N223	N221
N221	18	16000	19	20000	4882	N222	p
N222	3	-23808	8	21248	161	h	p
N223	16	-32580	19	1500	-498	N227	N224
N224	13	-10500	22	32767	2749	N225	k
N225	10	-18724	14	260	-692	N226	t
N226	9	29056	11	23680	1952	k	t

Table A.3 (Cont'd.)

NODE	XITEM	XCON	YITEM	YCON	THRES	PASS	FAIL
N227	14	22000	15	9500	3189	k	N228
N228	9	26532	14	5596	538	N229	p
N229	9	19945	22	203	356	k	p
N230	22	3612	31	32694	1842	N231	N217
N231	2	-6800	31	18472	352	N234	N217
N232	9	16896	25	24576	830	N223	N212
N233	11	-4600	17	24504	11810	N212	N223
N234	21	-1500	31	32580	497	N235	t
N235	18	0	20	21845	2999	t	s
N236	29	-12880	31	26360	1005	ʃ	N237
N237	27	-15600	28	20248	-5356	tʃ	ʃ