

# **UTI**

**Universal Telephone Interface**

**Operation Manual**

**Preliminary**

**© Ohio Scientific**

**March 1980**

UTI  
Universal Telephone Interface

Table of Contents

	Page
Section 1. Introduction	
1.1 General Description	1
Section 2. Installation/Set Up	
2.1 Introduction	3
2.2 Set Up Procedures	3
Section 3. Operating Instructions	
3.1 Introduction	9
3.2 Initializing the UTI	9
3.3 Initializing "Connect" to the Telephone Line	10
3.4 Initializing "Disconnect" to the Telephone Line	10
3.5 Using the FAILSAFE	10
3.6 Tape Recording a Telephone Call	10
3.7 Playing a Taped Message Over the Telephone Line	11
3.8 Monitoring the Telephone Line	12
3.9 Outputting a Touch-Tone to the Telephone Line	12
3.10 Decoding a Touch-Tone	13
3.11 Outputting Audio From an Auxiliary Device to the Telephone	14
3.12 Outputting VOTRAX to the Telephone Line	14
3.13 Detecting the Telephone Ringing on Incoming Calls	14
3.14 Detecting a Busy Signal	15
3.15 Automatic Tone Dialing	15
3.16 Automatic Pulse Dialing	18
3.17 Originating a Call to a Modem	18
3.18 Automatically Answering a Call From a Modem	19
Section 4. Theory of Operation	
4.1 Introduction	21
4.2 Block Diagram Discussion	21
4.3 Connect Control	24
4.4 Output Device Control	24
4.5 Automatic Telephone Dialing	24
4.6 Tape Player Control	24
4.7 Tape Recorder Control	25
4.8 FAILSAFE Control/Monitor	25
4.9 Ring Sense	25
4.10 DTMF Decode (Read Touch Tones)	25
4.11 Dial Tone and Busy Detection	26
4.12 Modem	26
A. Originate Mode Control	26
B. Answer Mode Control	26
C. Mode Sense	26
D. Self Test	26
E. Input Ready Monitor	26
F. Output Ready Monitor	27
4.13 VOTRAX Control	27

	Page
Section 5. Software	
5.1 Introduction	28
5.2 OS-65D V3.2 HC Phone Demo	28
5.3 OS-65U Level 3 Phone Demo	33
Appendix A - Hardware Tables	44
Appendix B - Software Tables	49
Appendix C - Specifications	57
Appendix D - Schematic Diagrams	58
VOTRAX Manual (if applicable)	

## SECTION 1

### INTRODUCTION

#### Ohio Scientific's Universal Telephone Interface

##### 1.1 General Description

Ohio Scientific's Universal Telephone Interface (UTI) is a PC board which plugs into your Ohio Scientific computer. The other end of the board connects to an FCC approved CBT DAA (data coupler) which plugs into a telephone jack on a private telephone line (just like a telephone). The data coupler simply isolates the telephone line from the computer.

##### Automatic Touch-Tone and Rotary Dialing

The UTI allows the computer to "pick up" the telephone line and then dial any telephone number. If the telephone line is a touch-tone line, the computer will dial with tones; otherwise, it will pulse dial just like a rotary dial telephone. If the line is busy, the computer can "hang up" and re-dial, if desired.

##### Built-In Modem

The UTI has a built-in 300 baud answer and originate modem for computer-to-computer communication.

##### Accepts Commands Via Touch-Tone

The UTI also has tone decoding capabilities which allow the computer to accept commands from any touch-tone telephone. This will allow you to "TALK" to your computer from remote locations by simply using a touch-tone telephone or tone generators attached to any telephone mouthpiece.

##### Record Telephone Messages

The UTI can allow you to tape record telephone messages using a cassette tape recorder. The computer will plug into the tape recorder's "MIKE" and "REMOTE" jacks, which will allow the computer to answer the telephone when it rings, turn on the tape recorder, record the message and turn off the tape recorder when the caller hangs up.

##### Play Tape Recorded Messages Over the Telephone Line

The UTI also can play any pre-recorded message on the telephone by using a cassette tape recorder. The computer will plug into the tape recorder's "AUX" and "REMOTE" jacks, which will allow the computer to turn on the tape recorder (in "PLAY" mode) on command, play the recorded message over the telephone line and turn off the tape recorder when the message is over.

## Play Music or Computer Voice Over the Telephone Line

The UTI has an "AUX" input which allows you to output other devices such as a Digital-to-Analog (D/A) Converter which can generate music or voice. If the UTI board is optionally equipped with VOTRAX (computer simulated voice), your computer can "TALK" over the telephone line and/or on a speaker. This can also be accomplished by connecting an existing VOTRAX to the UTI board.

## SECTION 2

### INSTALLATION/SET UP

#### 2.1 Introduction

This section provides information required to properly install a Universal Telephone Interface (UTI), 567 board, in a Challenger 8P computer. The external connections to the Challenger 8P for the UTI are also included in this section. Upon completion of this entire installation procedure, refer to Section 5 of this manual for instructions on the use of the demo software provided, which will enable you to test the functions of your UTI.

The UTI package includes a 567 board, two cables, six RCA phono jacks and, if the VOTRAX option is included, a VOTRAX module and 16-pin ribbon cable.

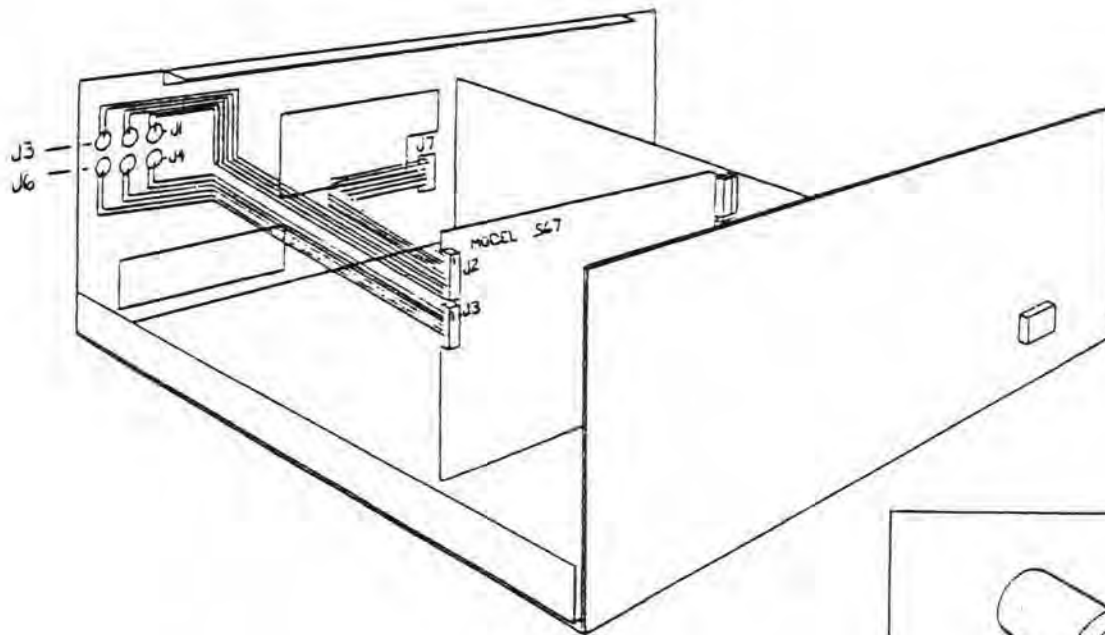
#### 2.2 Set Up Procedures

- A. Install the six RCA phono jacks provided in holes J1 through J6 on the C8P rear panel as shown in Figure 2-1.
- B. Install the DA15S connector (J7) attached to Cable 1 to the C8P rear panel as shown in Figure 2-1.
- C. Install the 567 board in the C8P backplane. Make sure that the slot contains adequate power for the UTI board as described in the specifications listed in Appendix C.
- D. Plug the two Molex connectors into J2 and J3 on the 567 board.

#### W A R N I N G

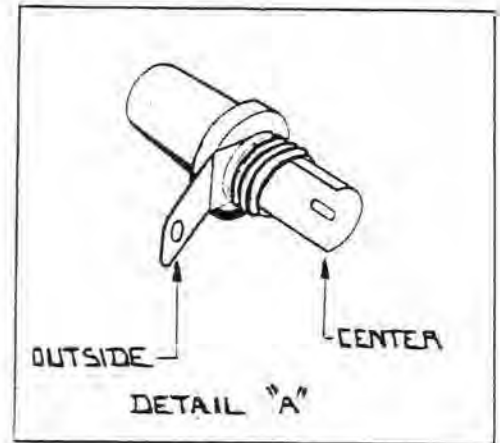
Be sure that these two connectors are plugged in properly. Improper connection of these connectors into J2 and J3 on the 567 board can result in damage to the board. See Figure 2-1 for proper connection of J2 and J3.

- E. Attach the six wires in Cable 2 to the proper terminals of a CBT data coupler as shown in Figure 2-2.
- F. Connect the CBT to the power line and the telephone line as described in the CBT instruction manual.
- G. Recheck all connections before proceeding. Then plug the DA15P connector of Cable 2 into J7 on the C8P rear panel.
- H. Make the appropriate connections to connectors J1 through J6 on the C8P rear panel. See Figure 2-3 to determine the appropriate connections. Note: For initial testing you may find it desirable to connect the input of an amplifier to the Telephone Line Monitor Output connector, J2, on the C8P rear panel.



Wiring List

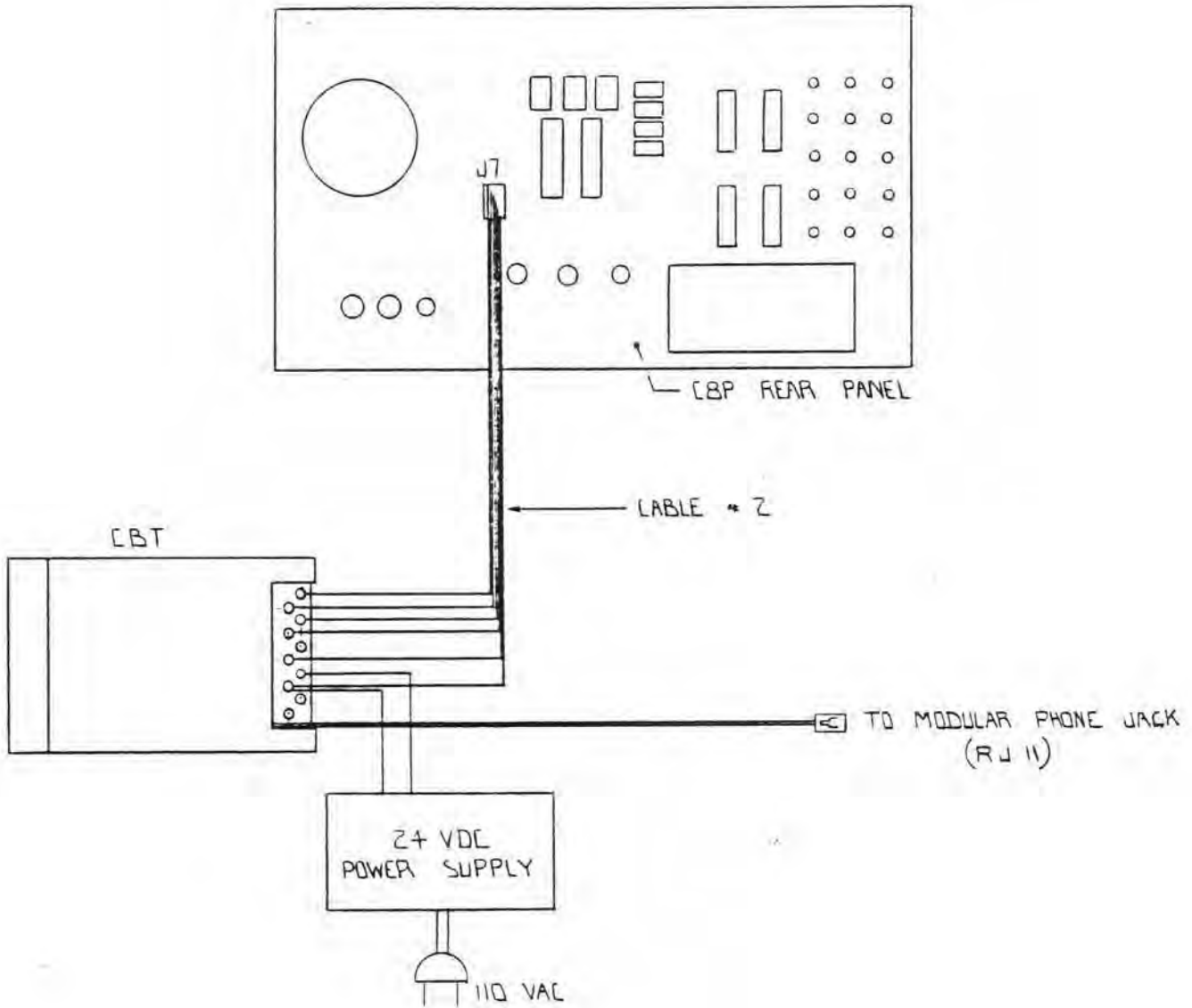
<u>567 Board</u>	<u>C8P Rear Panel</u>
J2 Pin 1	J7 Pin 1
J2 Pin 2	J7 Pin 2
J2 Pin 3	J7 Pin 3
J2 Pin 4	J7 Pin 4
J2 Pin 5	J7 Pin 5
J2 Pin 6	J7 Pin 6
J2 Pin 7	J3 Center
J2 Pin 8	J3 Outside
J2 Pin 9	J2 Center
J2 Pin 10	J2 Outside
J2 Pin 11	J1 Center
J2 Pin 12	J1 Shield
J3 Pin 1	No Connection
J3 Pin 2	J6 Center
J3 Pin 3	J6 Outside
J3 Pin 4	J5 Center
J3 Pin 5	J5 Outside
J3 Pin 6	J4 Center
J3 Pin 7	J4 Outside
J3 Pin 8	No Connection
J3 Pin 9	No Connection
J3 Pin 10	No Connection
J3 Pin 11	No Connection
J3 Pin 12	No Connection



Note: J1 through J6 on the C8P Rear Panel are "RCA" PHONO jacks. Connections to these jacks are shown in detail "A"

Connections between 567 Board and C8P Rear Panel

Figure 2-1



Connector J7 (DA15S)  
On C8P Back Panel

Terminal Block  
Designation

Pin #

CBT

1

DR and SH1 and V-

2

DA

3

V+ (24VDC)

4

OH

5

RI

6

DT

7-15

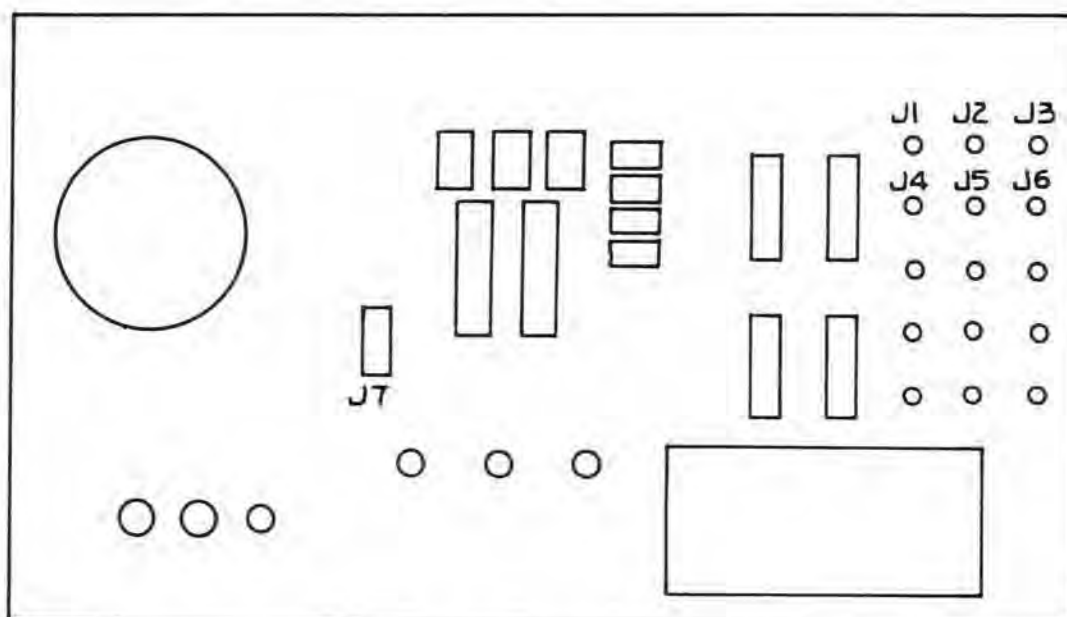
UNUSED

UTI Cable #2 Wire List

Figure 2-2



UNIVERSAL TELEPHONE INTERFACE (U.T.I.)



CBP BACK PANEL  
CONNECTION FOR U.T.I.

- J1 - CASSETTE RECORDER ON/OFF CONTROL.  
CONNECT TO CASSETTE RECORDER "REMOTE JACK."
- J2 - PHONE LINE MONITOR OUTPUT.  
CONNECT TO CASSETTE RECORDER "MIKE JACK."
- J3 - VOTRAX OUTPUT.  
CONNECT TO 8-OHM SPEAKER.
- J4 - CASSETTE PLAYER ON/OFF CONTROL.  
CONNECT TO CASSETTE RECORDER "REMOTE JACK."
- J5 - CASSETTE PLAYER OUTPUT.  
CONNECT TO CASSETTE RECORDER "AUX JACK."
- J6 - AUXILLIARY INPUT TO PHONE LINE.
- J7 - TELEPHONE INPUT/OUTPUT CONNECTOR.  
CONNECT TO CBT DATA COUPLER.

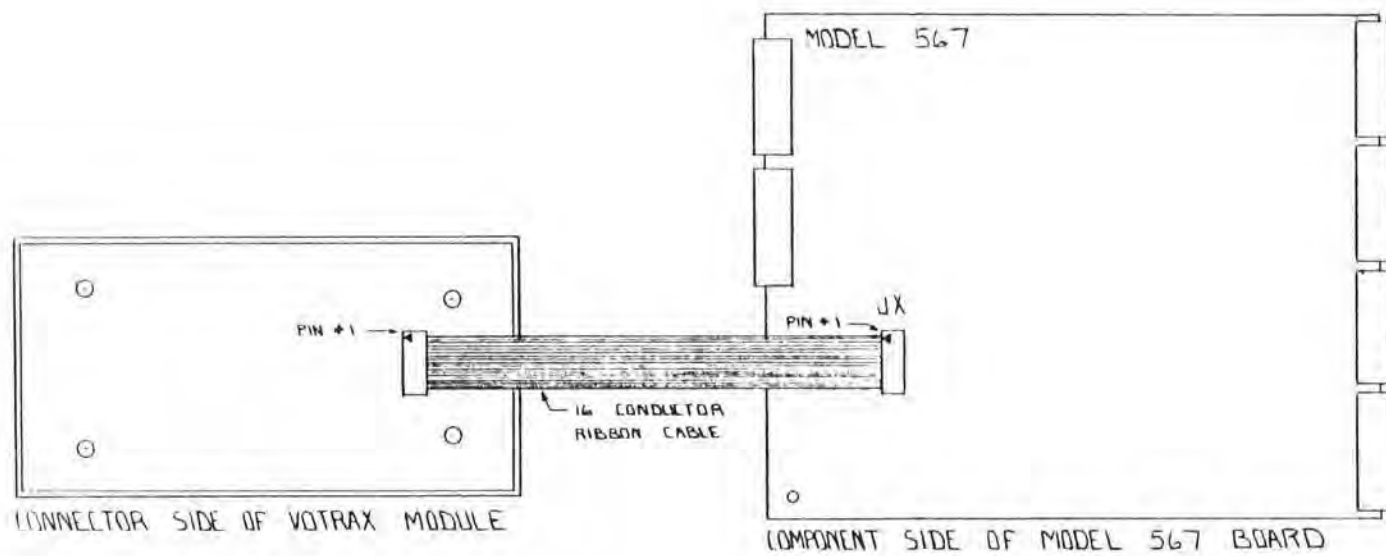
NOTE: J1 AND J2 ARE "RECORD" FUNCTIONS.  
J4 AND J5 ARE "PLAYBACK" FUNCTIONS.

Figure 2-3

- I. If your UTI includes the VOTRAX option, then the VOTRAX module must be electrically connected to the UTI board via a 16-pin ribbon cable with 16-pin DIP connectors on each end of it. When connecting the VOTRAX module to the UTI board make sure that Pin 1 on the VOTRAX module is connected to Pin 1 on 567 (UTI) board as shown in Figure 2-4. The VOTRAX module may be mechanically connected to the UTI board with four No. 6 screws and at least 1/4" long stand-offs. The stand-offs allow enough clearance for the ribbon cable to run between the 567 board and the VOTRAX module.
- J. If your UTI board does not include the VOTRAX, but you have a CA-14 VOTRAX board, then you can use your VOTRAX module with the UTI. This is accomplished by the following procedure:
  1. Remove the CA-14 (565) board from your computer.
  2. Unplug the 16-conductor ribbon cable from the CA-14 board. Leave the other end connected to the VOTRAX module.
  3. Follow the instructions in Section 2.2 J.

Note: If your VOTRAX module is a Model "VSK" it will work with the UTI (567) board, but it still will not have inflection capabilities. Only Model "VSL" VOTRAX modules have inflection capabilities.

4. Connect your speaker to J3 on the rear panel for VOTRAX voice.



Votrax Module to UTI Board Connection

Figure 2-4

## SECTION 3

### OPERATING INSTRUCTIONS

#### 3.1 Introduction

After the UTI has been properly installed, the system may be checked out using the demonstration software provided with the UTI. The documentation for this software should allow you to understand the workings and use of the UTI immediately. You may modify these programs to implement your desired function for the UTI board, or you can build your own program using the description of the operating instructions in this section.

This section contains operating instructions for using all of the functions to the UTI board using the BASIC language. These instructions are provided so that the user can write his own software for the UTI. Each section contains a description of the function and the BASIC command to implement it. The user will have to combine several of these functions in order to implement the desired function.

#### 3.2 Initializing the UTI

The UTI board interfaces to the computer via two peripheral interface adapters (PIA) and one asynchronous communications interface adapter (ACIA). These devices are cleared and initialized on Power Up of the computer. The "BREAK" and/or "RESET" buttons are not connected to the UTI board. This means that the "BREAK" and "RESET" buttons have no effect on the UTI board. The UTI will hang up the telephone when the computer is powered down and the telephone will remain disconnected (hung up) after the computer has been turned on. The UTI board's FAILSAFE feature will hang up the telephone within two minutes after the "BREAK" or "RESET" key has been depressed. The following instructions should be inserted at the beginning of each of your programs that make use of the UTI. Note that these instructions will completely reset the UTI and hang up the telephone. The initialization routine is:

```
100 REM INITIALIZE THE UTI
110 FOR I=0 TO 6 STEP 2: POKE 63489+I,4: NEXT: REM SELECT DATA REGS
120 FOR I=0 TO 6 STEP 2: POKE 63488+I,255: NEXT: REM ALL OUTPUTS HI
130 FOR I=0 TO 6 STEP 2: POKE 63489+I,0: NEXT: REM SEL DIR REGS
140 POKE 63488,7: POKE 63490,47: POKE 63492,224: POKE 63494,255
150 FOR I=0 TO 6 STEP 2: POKE 63489+I,6: NEXT
160 POKE 63496,67: FOR I=1 TO 200: NEXT: POKE 63496,17
170 FOR I=0 TO 6 STEP 2: POKE 63488+I,255: NEXT I
180 POKE 63490,14: REM HANG UP PHONE AND TURN TAPES OFF
190 REM END OF INITIALIZATION
```

### 3.3 Initiating "Connect" to the Telephone Line

The UTI can "connect" the UTI to the telephone line ("pick up" the telephone) by:

1. Strobe the FAILSAFE (see Section 3.5).
2. Enable the connection to the telephone line by running the BASIC command:

```
POKE 63490,13
```

### 3.4 Initiating "Disconnect" to the Telephone Line

The UTI can "disconnect" the telephone line ("hang up") by running the BASIC command:

```
POKE 63490,14
```

This statement will also turn off both tape recorders.

### 3.5 Using the FAILSAFE

The FAILSAFE feature of the UTI must be strobed at least once every 90 seconds while the UTI is accessing the telephone line. Failure to do so by the software will result in the FAILSAFE feature disconnecting the telephone line from the UTI, 108 seconds after the last FAILSAFE STROBE has occurred. The FAILSAFE STROBE may be enabled by the following instructions:

```
X=PEEK(63490): POKE 63490,(X OR 32): POKE 63490,(X AND 223)
```

The above command will make the FAILSAFE STROBE high and then low which enables the UTI board to access the telephone line for the next 108 seconds. Note that a highly experienced user of the UTI may disable this FAILSAFE feature as described in Section 4.8 of this manual.

The status of the FAILSAFE time out feature may be determined by executing the following BASIC command:

```
X=(PEEK(63488) AND 64)
```

If  $X=0$ , then the FAILSAFE has timed out and the telephone is disconnected from the UTI.

If  $X \neq 0$ , then the FAILSAFE is not timed out and the UTI may still access the telephone line (see Table T-2B and Table S-2).

### 3.6 Tape Recording a Telephone Call

The tape recorder must be connected to the UTI via a cable connecting the recorder's "MIKE" jack to the UTI's "TELEPHONE LINE MONITOR OUTPUT" jack (J2 on the C8P rear panel). Note: The FAILSAFE must be strobed at least once every 90 seconds while the UTI is connected to the telephone line (see Section 3.5.).

### Manual Control of the Tape Recorder

You may manually record the telephone conversation anytime the UTI is accessing the telephone by manually turning the tape recorder to "RECORD" mode. The tape recorder's "MONITOR" jack must not have a plug in it during manual recording.

### Computer Control of the Tape Recorder

1. Manually connect the tape recorder's "REMOTE" jack to the UTI's "CASSETTE RECORDER ON/OFF CONTROL" jack (J1 on the C8P rear panel).
2. Manually set the tape recorder to the "RECORD" mode.
3. After the UTI has accessed the telephone line, the computer can turn the tape recorder on by the following BASIC command:

```
X9=PEEK(63490) : POKE 63490,(X9 AND 251)
```

4. The computer can turn off the tape recorder by running the following BASIC command:

```
X9=PEEK(63490) : POKE 63490,(X9 OR 4)
```

### 3.7 Playing a Taped Message Over the Telephone Line

The tape player's "AUX" jack must be connected to the UTI's "CASSETTE PLAYER OUTPUT" jack (J5 on the C8P rear panel). Note: The FAILSAFE must be strobed at least once every 90 seconds while the UTI is connected to the telephone line (see Section 3.5).

### Manual Control of the Tape Player

You may manually play a taped message over the telephone line when the UTI is accessing the telephone line by:

1. Select the tape player output to the telephone line by running the BASIC command:

```
POKE 63488,3
```

2. Manually turn the tape player to the "PLAY" mode. (The tape player's "REMOTE" jack must not have a plug in it during manual operation.)
3. When the message is over, the tape player output to the telephone may be shut off (deselected) by running the BASIC command:

```
POKE 63488,7
```

### Computer Control of the Tape Player

1. Connect the tape player's "REMOTE" jack to the UTI's "CASSETTE PLAYER ON/OFF CONTROL" jack, J4, located on the C8P rear panel.
2. Manually set the tape player to the "PLAY" mode.
3. After the UTI has accessed the telephone line, the computer can select the tape player output to the telephone line and turn the tape player on by running the following BASIC command:

```
POKE 63488,3 : X9=PEEK(63490) : POKE 63490,(X9 AND 247)
```

4. When the message is completed, the computer can shut off (deselect) the tape player output to the telephone and turn the tape player off by running the BASIC command:

```
POKE 63488,7 : X9=PEEK(63490) : POKE 63490,(X9 OR 8)
```

### 3.8 Monitoring the Telephone Line

You can monitor the telephone line while the UTI is accessing it by manually connecting the "TELEPHONE LINE MONITOR OUTPUT" jack (J2 on the C8P rear panel) to the input of an audio amplifier. This is the same connection that is used for the audio to the tape recorder.

### 3.9 Outputting a Touch-Tone to the Telephone Line

The UTI can output a touch-tone to the telephone line under computer control. The following example describes one way of outputting one tone onto the telephone line.

1. After the UTI has connected the telephone line, the computer can select the tone generator to the telephone line by the BASIC command:

```
POKE 63433,5
```

2. Output a tone on the telephone line by running the BASIC command:

```
POKE 63494,T
```

where T is the tone code located in Table T-1.

3. The tone is now on the telephone line, wait for the length of time that you wish the tone to be heard, then turn off the tone by turning off the tone generator using the BASIC command:

```
POKE 63494,255
```

4. The tone generator can be deselected from the telephone line by running the BASIC command:

```
POKE 63488,7
```

Example:

Output a #1 Key Tone for one second, then turn the tone off and disable the tone generator.

```
100 N=4600 :REM FOR 2 MHZ SYSTEM
110 POKE 63488,5 : POKE 63494,238
120 FOR I=ITON : NEXT
130 POKE 63494,255 : POKE 63488,7
```

Note: Set N=2300 for 1MHz system or set N=7590 for a "GT" system.

### 3.10 Decoding a Touch-Tone

After the UTI has accessed the telephone line, the computer can decode (receive) touch-tones (DTMF) by using the following procedure:

1. Before requesting a tone, the tone decoder should be cleared (reset) by reading the input. (See statement 100 in the example.)
2. The value of X above will be disregarded. Strobe the FAILSAFE. (See statement 110 in the example.) Now wait for a "VALID TONE". (See statements 120, 130 and 140 in the example.)
3. When a valid tone is present, read the tone. (See statement 150 in the example.)
4. X now contains a tone input code which is described in Table S-5.
5. If no tone is found after 90 seconds, the FAILSAFE timed out and statement 170 in the example will be executed.

Example:

Decode a touch-tone with a FAILSAFE time out check.

```
100 X=PEEK(63492)
110 X=PEEK(63490) : POKE 63490,(X OR 32) : POKE 63490,(X
AND 223)
120 IF (PEEK(63493) AND 128)>0 THEN 150
130 IF (PEEK(63480) AND 128)=0 THEN 170
140 GO TO 120
150 X=(PEEK(63492) AND 15)
160 REM X CONTAINS THE TONE INPUT CODE FOR THE TONE JUST
RECEIVED
170 REM NO TONE WAS FOUND IN THE LAST 90 SECONDS
```



### 3.11 Outputting Audio From An Auxiliary Device to the Telephone

The UTI will allow you to output audio onto the telephone line by connecting an audio source to the "AUXILIARY INPUT" jack (J6 on the C8P rear panel). After the UTI has accessed the telephone line, the audio can be selected by running the following BASIC command:

```
POKE 63488,2
```

The audio may be shut off (deselected) from the telephone line by the following BASIC command:

```
POKE 63488,7
```

Note: The audio volume should be controlled by the outside source. If this is not possible, potentiometer R36 on the 567 board can be adjusted for the correct volume. Turn R36 clockwise to increase the volume.

### 3.12 Outputting VOTRAX to the Telephone Line

The UTI must have the VOTRAX option on your UTI board or a separate VOTRAX must be connected to the UTI board. After the UTI has accessed the telephone line, the VOTRAX speech synthesizer can be selected to output to the telephone line using the BASIC command:

```
POKE 63488,1
```

The VOTRAX can be deselected from the telephone line by the BASIC command:

```
POKE 63488,7
```

Note: The VOTRAX volume over the telephone can be adjusted by potentiometer R37 on the 567 board. Turning R37 clockwise will decrease the volume of the VOTRAX on the telephone line.

### 3.13 Detecting the Telephone Ringing on Incoming Calls

The UTI can detect the telephone ring signal on incoming calls. This is demonstrated by the following BASIC program:

```
100 POKE 63496,67 : FOR I=1 TO 100 : NEXT : POKE 63496,17, N=4600
110 FOR I = 1 TO N : NEXT
120 IF(PEEK(63488) AND 128) = 0 THEN 120
130 REM THE PHONE IS RINGING, INSERT THE
140 REM PROGRAM TO ANSWER IT (CONNECT PROGRAM) HERE
```

Note: Set N=3300 for a 1MHz system, 4600 for a 2MHz system, and 7600 for a "GT" system.

### 3.14 Detecting a Busy Signal

The UTI can detect the "LINE BUSY" and "TRUNK BUSY" signals on calls which the UTI has originated. A busy detect routine located at lines 1700-1790 and 2200-2290 in the Demo #1 program on the disk, has properly detected the busy signal in most cases. However, this program may not properly detect all busy signals in all locations of the country. In addition, the program must decide what to do if neither a busy or a ring signal has been sensed after a period of time.

### 3.15 Automatic Tone Dialing

The following steps are recommended for automatic touch-tone dialing with the UTI. These steps refer to a sample program written in BASIC.

1. Initialize the UTI. This is only to be done once at the beginning of the program. Statements 70 through 280 set up the proper inputs and outputs of the two PIA's on the UTI, and set up the ACIA for 300 baud operation of the modem. The telephone is hung up after this initialization.
2. Input a telephone number to the program as in statement 290. (The telephone number may already be stored in a more advanced program that has a telephone number MENU.) The sample program requires only numbers to be entered (i.e., "-" not allowed).
3. Check the telephone number for syntax and convert each digit of the telephone number to its corresponding dial code that can be output to the tone dialer. These dial codes for each button on a telephone are shown in Table T-4A. This conversion is accomplished in the subroutine at statement 9100 through 9150 in the sample program. If this subroutine detects an illegal character in the telephone number, it will return to the main program at statement 305 with E=1. If no errors exist in the telephone number, E=0, this subroutine is called by the main program at statement 300. Statement 305 checks to see if the subroutine has found an error in the telephone number. If an error has occurred, the program asks for the number again at statement 290.
4. Connect UTI to Telephone Line and Wait For Dial Tone

Statement calls the subroutine at statement 9100 which:

- a. Strokes the FAILSAFE
- b. Connects the UTI to the telephone line
- c. Waits for a dial tone
- d. If no dial tone is detected within 10 seconds, the subroutine returns with E=1

If no dial tone is detected within 10 seconds, the program disconnects the UTI from the telephone line (hangs up) and tries to get a dial tone again (statements 360 through 385).

### 5. Select Tone Dialer

Next the program selects the tone dialer at statement 320.

## 6. Dial Telephone Number

Statement 330 calls the subroutine at statement 9200 which dials the telephone number. Statement 9230 turns on one tone for OT seconds, then turns off the tone. The tones must be separated by a quiet time (OFFTIM) in statement 9220. OT and OFFTIM should each be greater than or equal to 50 milliseconds.

## 7. Wait For Hang Up Request

The telephone number has now been dialed. This program will continue to strobe the FAILSAFE and wait for you to depress the <SPACE> key which will hang up the telephone and wait to enter another telephone number.

## 8. Wait Subroutine

The subroutine at statement 9500 will wait for the time specified by:

WX 10 milliseconds

W is specified before executing this subroutine. This subroutine will give the desired wait time regardless of machine speed if it is used with the OS-65D NMHz operating system. This subroutine uses the speed normalization constant, C, which is initialized in statement 70.

For example, the statement:

W=100 : GOSUB 9500

will take one second to execute (i.e., 100 x 10 ms = 1 second).

## 9. FAILSAFE Subroutine

The subroutine at statement 9600 will strobe the FAILSAFE feature. This must be executed at least once every 90 seconds while the UTI is connected to the telephone line.

### Automatic Tone Dialing Program

```
10 REM DEMO PROGRAM FOR UTI
15 REM DATE 2/13/80
20 REM THIS PROGRAM ASKS FOR A PHONE NUMBER, PICKS UP
30 REM THE PHONE, WAITS FOR A DIAL TONE, DIALS THE NUMBER,
40 REM WAITS FOR USER TO TYPE A CHAR TO HANG UP THE PHONE
70 C=49/PEEK(9851) :REM SPEED NORMALIZATION CONSTANT
72 OT=5
74 OFFTIM=5:REM TONE OFF FOR 50 MS
76 DT=20 :REM DIAL TONE DETECTED FOR 0.6 SEC
100 DIM N(16),N$(16),D(16)
110 DATA 238,222,290,237,221,189,235,219,187,215
120 DATA 231,183,126,125,123,119
130 DATA 1,2,3,4,5,6,7,8,9,0,*,#,A,B,C,D
140 FORI=1TO16:READN(I):NEXTI:REM READ DIALER DATA
150 FORI=1TO16:READN$(I):NEXTI:REM READ INPUT CHAR DATA
```

```

160 REM MAKE ALL UTI PIA OUTPUTS HIGH
170 FORI=1TO4:POKE63486+(I*2),NEXTI
180 REM SWITCH BOTH PIA'S TO DATA DIRECTION MODE
190 FOR I=1TO4:POKE63487+(I*2),0:NEXTI
200 POKE63488,7 :REM SET PA DATA DIRECTION
210 POKE63490,47 :REM SET PB DATA DIRECTION
220 POKE63492,224 :REM SET 1PA DATA DIRECTION
230 POKE6344,255 : REM SET 1PB DATA DIRECTION
240 REM SET PIA'S BACK TO DATA REGISTER
250 FOR I=1TO4: POKE63487+(I*2),6:NEXTI
260 FOR I=1TO4: POKE63486+(I*2),255:NEXTI
270 POKE63496,3:POKE63496,17: REM SETUP ACIA
280 POKE63490,14:REM HANG UP THE PHONE
290 INPUT"NUMBER PLEASE",A$
300 GOSUB9100: REM CHECK PHONE # & READY DIAL CODE
305 IF E=1 THEN PRINT"NUMBERS ONLY !:"GOTO290:REM BAD #,TRY AGAIN
308 PRINT"PICK UP PHONE AND WAIT FOR DIAL TONE"
310 GOSUB9000 :REM PICK UP PHONE & WAIT FOR DIAL TONE
312 IFE=1THEN360: REM IF NO DIAL TONE, THEN TRY AGAIN
315 PRINT"SELECT TONE DIALER"
320 POKE63488,5 :REM SELECT TONE DIALER
325 PRINT"DIAL PHONE #"
330 GOSUB9200 : REM DIAL PHONE #
340 ?"PHONE # DIALED
400 REM CHECK FOR MENU REQUEST, LOOK FOR ANY KEY DOWN
410 PRINT" PRESS ANY KEY FOR HANG UP AND NEW MENU!"
420 GOSUB9700:REM GO CHECK FOR KEY PRESSED
430 IF KEY =0 THEN 420:REM NO KEY PRESSED
440 POKE63490,14 :REM HANG UP THE PHONE
450 GOTO290 :REM GO RQST ANOTHER PHONE #
8999 REM
9000 REM SUBR TO PICK UP PHONE & WAIT FOR DIAL TONE
9003 GOSUB 9600 :REM STROBE FAILSAFE
9005 E=0 :REM RESET ERROR FLAG
9010 POKE63490,13 :REM LIFT HOOK (PICK UP PHONE)
9020 X9=0 :REM SET TIMEOUT TIMER
9030 IF(PEEK(63490)AND 128)=0 THEN 9060
9040 X9=X9+C:IF X9(330 THEN 9030:REM DT WITHIN 10 SEC ?
9050 PRINT"NO DIAL TONE! AUTO REDIAL":E=1:RETURN
9060 X8=0 :REM SET DIAL TONE TIMER
9070 X9=X9+C: IF X9>330 THEN 9050: REM DT WITHIN 10 SEC ?
9080 IF(PEEK(63490)AND128)=0 THEN 9030:REM NOISE
9090 X8=X8+C: IF X8<DT THEN 9070:REM SIGNAL ON PHONE
9095 RETURN :REM DIAL TONE PRESENT ON PHONE LINE
9099 REM
9110 REM SUBR TO CHECK PHONE # & READY DIAL CODE
9105 L=LEN(A$) :REM L = # OF DIGITS IN PHONE #
9100 FOR B=1 TO L :E=1:REM IF E=1, THEN ERROR IN PHONE #
9120 FORI=1TO16:IFMID$(A$,B,1)=N$(I)THEN D(B)=N(I):E=0
9130 NEXTI:IF E=1 THEN 9150:REM ERROR IN PHONE #
9140 NEXTB
9150 RETURN :REM # IS NOW IN D(0)-D(15) $ # OF DIGITS IN L
9199 REM
9200 REM SUBR TO DIAL A PHONE #
9210 FOR I=1 TO L :REM L=# OF DIGITS IN PHONE #
9220 W=OFFTIM:GOSUB9500:REM TONE ON 'ONTIM', OFF'OFFTIM'
9230 POKE63494,D(I):W=OT:GOSUB9500:POKE63494,255
9240 NEXT I

```

```

9250 RETURN :REM PHONE # HAS BEEN DIALED
9299 REM
9300 REM SUBR TO CHECK FOR BUSY OR RING
9310 X9=0:BUSY=0:REM START TIMEOUT TMR & SET NOT BUSY
9320 B=0 :REM START BUSY COUNTER
9325 X9=X9+C: IFC9>990 THEN 9380 :REM TIMEOUT, ASSUME BUSY
9328 IF(PEEK(63490)AND128)>0THEN9325:REM SIGNAL YET ?
9329 B=B+C :REM SIGNAL JUST OCCURRED CONTINUE CHECKING LINE
9330 X9=X9+C: IF X9>990 THEN 9380 :REM IF TIMEOUT THEN BUSY
9340 IF (PEEK(63490)AND128)=0THEN B=B+C: GOTO9330
9350 REM END OF SIGNAL, NOW CHECK IT
9360 IF B<GLITCH THEN 9320: REM JUST A NOISE GLITCH
9370 IF B < BY THEN BUSY = 1:REM SET BUSY FLAG
9380 RETURN
9499 REM
9500 REM WAIT LOOP, W IS SET FOR 10MS INTERVALS
9505 C=49/PEEK(9851)
9510 WW=INT(W*28/C)
9520 FORX2=1TOWW:NEXTX2
9530 RETURN
9599 REM
9600 REM SUBROUTINE TO STROBE FAILSAFE
9610 X6=PEEK(63490)
9620 POKE63490, (X6OR32) :REM MAKE FAILSAFE HIGH
9630 POKE63490, (X6AND223) :REM MAKE FAILSAFE LOW
9640 RETURN
9699 REM
9700 REM SUBROUTINE TO CHECK KEYBOARD FOR ANY KEY PRESSED
9705 KEY=0 :REM KEY NOT PRESSED
9710 POKE2073,96:POKE57088,255:REM DISABLE CTRL-C
9720 IF PEEK(57088)<>1THEN KEY=1:REM IF KEY=1THEN KEY PRESSED
9730 POKE2073,173 :REM ENABLE CTRL-C
9740 RETURN

```

### 3.16 Automatic Pulse Dialing

The preceding description and program for touch-tone dialing can be used for pulse dial telephone lines by changing statement 320 to:

```
320 POKE 63488,6 : REM SELECT PULSE DIALER
```

Then add the following instructions which will delay the printing of "PHONE # DIALED" on the screen until after the pulse dialer is finished dialing. Note that the pulse dialer will take longer to dial a telephone number than the tone dialer.

```

331 W=0
332 FOR B=1 TO L
334 FOR I=1 TO 10 : IF MID$(A$,B,1)=N$(I) THEN W=(I*10+W)
336 NEXT I : NEXT B
338 W=W-100+L*68 : GOSUB 9500

```

### 3.17 Originating a Call to a Modem (Originate Mode)

The UTI can originate a call to a 300 baud modem using the following steps:

1. Reset the modem by executing:

```
POKE 63492,224
POKE 63496,67, FOR I=1 TO 100 : NEXT : POKE 63496,3
FOR I=1 TO : NEXT
```

2. Set the desired modem status by using one of the commands in Table S-8. For example, the command to select a modem data format of 8 data bits, no parity bit and 2 stop bits is:

```
POKE 63496,17
```

3. Dial the telephone number of a 300 baud modem as described in the attached dialing program.
4. After the telephone number has been dialed, place the modem in the Originate Mode by executing:

```
POKE 63492,160
```

5. Enable modem output to the telephone line by the following command:

```
POKE 63488,4
```

6. Now wait for the remote modem to answer the telephone. The remote modem will then send a tone to our UTI modem. The UTI will signal us that it has received the tone from the remote modem that was called. We sense the remote modem by executing the following instruction:

```
100 IF (PEEK(63496)AND 8) = 0 THEN 200
110 IF (PEEK(63488) AND 64) > THEN 100
120 PRINT "NO MODEM WAS DETECTED WITHIN 90 SECONDS"
130 REM ENTER COMMANDS HERE TO HANG UP AND REDIAL
200 REM MODEM HAS BEEN DETECTED
```

7. If line 130 is executed, the remote modem has not been detected within 90 seconds after dialing, you should insert commands at line 130 to hang up and redial the telephone number again.

If line 200 is executed, the remote modem has been detected and you should insert a program at line 200 that will READ and WRITE to the MODEM.

### 3.18 Automatically Answering a Call From a Modem

The UTI can automatically answer a call from a 300 baud asynchronous modem using the following steps:

1. Reset the modem by executing the following commands:

```
POKE 63492,224
POKE 63496,67, FOR I=1 TO 100: NEXT: POKE 63496,3
FOR I=1 TO 5000: NEXT
```

2. Set the desired modem status by using one of the commands in Table S-8. For example, the command to select a modem

data format of 8 data bits, no parity bit and two stop bits, is:

```
POKE 63496,17
```

3. Now wait for the telephone to ring. After it rings, command the UTI to answer the telephone. (See Sections 3.3 and 3.13.)
4. Now, enable the modem output to the telephone line by the following command:

```
POKE 63488,4
```

5. Wait for the remote modem to signal us that it is ready by the following instructions:

```
100 IF 8PEEK(63496) AND 8) = 0 THEN 200
110 IF (PEEK(63488) AND 64) > 0 THEN 100
120 PRINT "NO MODEM WAS DETECTED WITHIN 90 SECONDS"
130 REM ENTER COMMANDS HERE TO HANG UP AND WAIT FOR
    ANOTHER CALL
200 REM MODEM HAS BEEN DETECTED
```

6. If instruction 130 is executed, the remote modem has not been detected within 90 seconds after answering the telephone, you should insert commands at line 130 to hang up and wait for another call.
7. If instruction 200 is executed, the remote modem has been detected, you should insert the program at line 200 that will read and write to the remote modem.

## SECTION 4

### THEORY OF OPERATION

#### 4.1 Introduction

This section contains a description of the major functions performed by the UTI. Figure 4-1 is a simplified block diagram of the UTI which shows the signal flow between the telephone line and the UTI and between external devices and the UTI.

#### 4.2 Block Diagram Discussion (Figure 4-1)

The UTI is a single board that plugs into a Challenger computer and connects to a private telephone line via a CBT data coupler. The CBT data coupler has three signal paths that connect to the UTI. They are: an output to the UTI's Ring Detection circuitry, inputs from the UTI's telephone line Connect Control circuitry to the CBT's relays, and the bidirectional Sound Line to the telephone which connects to the UTI's Sound Duplexer.

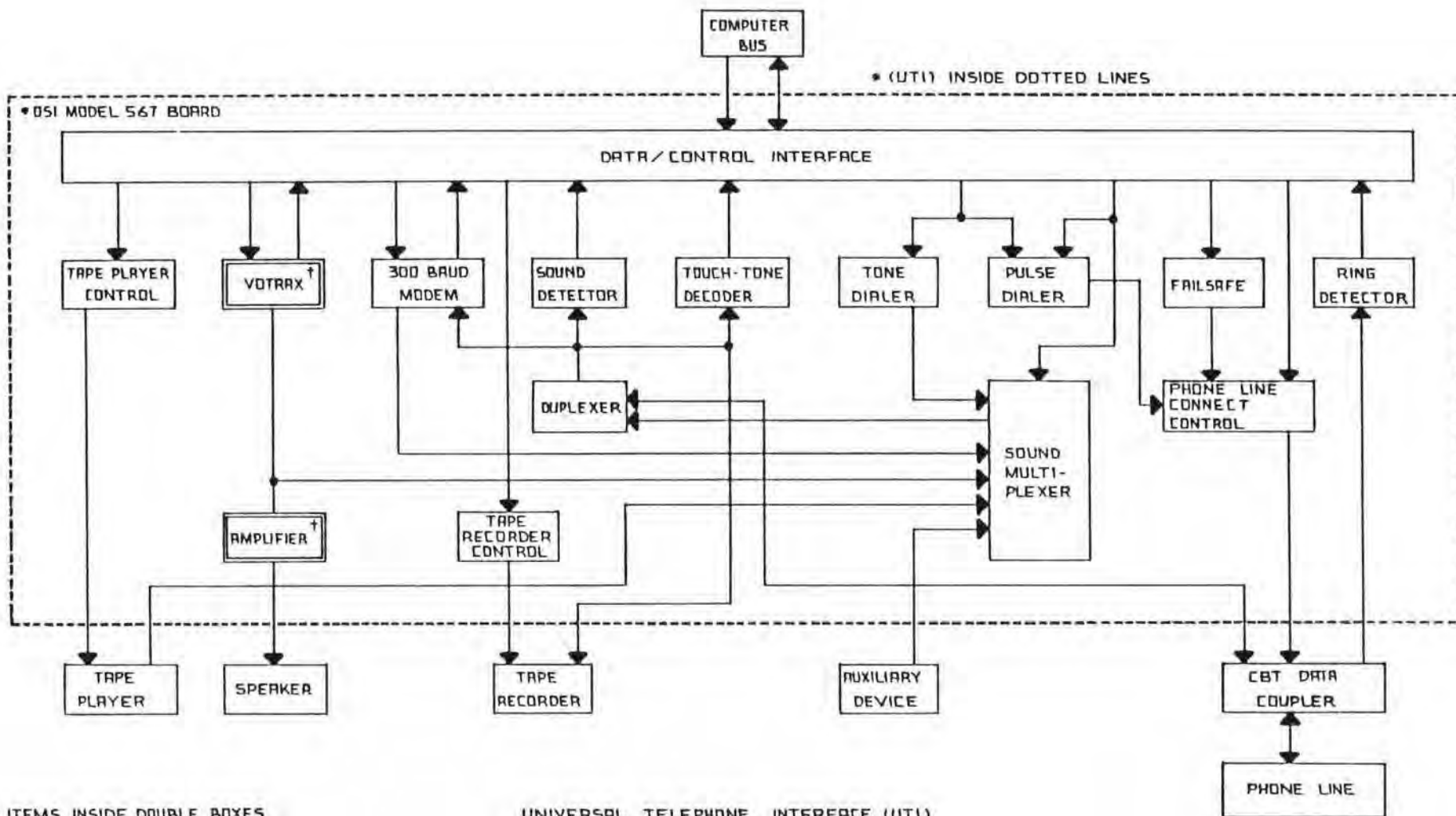
The UTI's Ring Detection circuitry detects the contact closures from the CBT when the telephone is ringing and outputs a Ring Detect signal to the computer via the UTI's Data/Control Interface Circuitry.

The telephone line is accessed (connected) and disconnected (hung up) via the UTI's telephone line Connect Control circuitry which interfaces directly to the relays inside the CBT data coupler. The computer can access the telephone line by outputting to the FAILSAFE circuitry and the telephone line Connect Control circuitry on the UTI. The telephone may be disconnected by the same methods. In addition, the FAILSAFE circuitry provides extra safety by disconnecting the telephone line should the computer fail. The Pulse Dialer dials the telephone by using the same telephone line Connect Control circuitry that controls the line connect relays in the CBT. Pulse Dialing has been implemented so that the software required for pulse dialing is almost identical for that of tone dialing. The software need only to select which type of dialing (pulse or tone) is to be implemented before actually dialing the desired telephone number.

The Sound Duplexer on the UTI interfaces to the CBT's sound path in the telephone line. The function of the Sound Duplexer is to separate the sound output by the UTI from the sound input from the telephone line. This function is also used on a regular telephone so that you hear the party to whom you are speaking louder than you hear yourself in the earpiece of the telephone.

The sound that comes in from the telephone line after passing through the Duplexer is sent to the Touch-Tone Decoder, the Sound Detection Circuitry, the 300 baud modem input and the Telephone Line Monitor output (connector J2 on the C8P rear panel). This means that all four devices simultaneously listen to the telephone line whenever the UTI Connect Control accesses the telephone line.





† ITEMS INSIDE DOUBLE BOXES ARE OPTIONAL

UNIVERSAL TELEPHONE INTERFACE (UT1)  
BLOCK DIAGRAM

Figure 4-1

-22-

The Touch-Tone Decoding circuitry on the UTI is constantly decoding touch-tones from the telephone line whenever the UTI has accessed the telephone. The Touch-Tone Decoder interfaces to the computer through the Data/Control Interface on the UTI board.

The Sound Detection circuitry on the UTI converts the sound heard from the telephone line to a digital representation which can be read by the computer. This circuitry allows the computer to determine when the dial tone is present or when a busy signal is heard.

The Telephone Line Monitor Output is made to be connected to a tape recorder's "MICROPHONE" input to allow you to tape record the telephone conversation or it may be connected to the input of an amplifier, which will allow you to monitor the telephone conversation. This signal is present whenever the UTI has connected the telephone line. The tape recorder used to record the conversation may be remotely controlled by the computer through the Tape Recorder Control signal (J1 on the C8P rear panel). This signal controls a relay which switches the tape recorder's motor on and off.

The sound from the telephone line also comes into a 300 baud modem in the UTI. This 300 baud modem interfaces directly to the computer via the Data/Control Interface on the UTI board. The modem outputs a modem tone to the telephone line via the Sound Multiplexer on the UTI board. This tone is output to the telephone line only when the Sound Multiplexer has been set by the computer to select the modem output.

The computer selects the desired sound output to the telephone line via the output Sound Multiplexer circuitry on the UTI board. This multiplexer can select any one of the five devices to output to the telephone line. Note that only one device at a time can output to the telephone line. These five devices are: the 300 baud modem, the Tone Dialer, the Tape Player, an Auxiliary Device and the VOTRAX (if the VOTRAX option has been included on the UTI board or if you have a CA-14 VOTRAX board in your computer).

The 300 baud modem will output the modem frequency only when the computer has selected its output through the output Sound Multiplexer. The 300 baud modem allows your computer to communicate with another computer that has a 300 baud modem connected to it. The direct connection of this 300 baud modem to the telephone line eliminates many problems encountered with acoustically coupled modems.

Another device which can be output to the telephone line through the Sound Multiplexer circuitry is the Tone Dialer. The Tone Dialer can provide touch-tone dialing to the telephone lines and can be used for computer communications.

The Tape Player which can be selected to output to the telephone line, allows the playing of tape recorded messages over the telephone line. The Tape Player can be controlled by the Tape Player Control circuitry on the UTI board. This control line (J4 on the C8P rear panel) must be connected to the tape player's "REMOTE" jack. This will allow the computer to automatically turn the tape player on and off under program control. The tape player's volume control should be manually adjusted for proper volume over the telephone line.

The UTI has an extra input line that can be selected by the computer which allows different Auxiliary Devices to output sound to the telephone line. One type of Auxiliary Device is a Digital-to-Analog (D/A) Converter which can be programmed to output sound or music to the telephone line.

If the UTI contains the VOTRAX option or if your computer contains a CA-14 VOTRAX board and that board is connected properly to the UTI board, then the VOTRAX voice synthesizer can be output to the telephone line. The VOTRAX option on the UTI board also includes an amplifier for the VOTRAX which can directly drive an 8 ohm speaker through connector J3 on the C8P rear panel.

#### 4.3 Connect Control - PB0, PB1

These are two redundant outputs which determine whether the computer gets connected to the telephone line (i.e., picks up and hangs up the telephone) - see Table T-1 and Table S-3. The redundancy was implemented so that if one of these two signals does not work correctly, then the telephone will not be accidentally connected to the computer (i.e., picked up). The FAILSAFE feature, described later, also controls the telephone connection and provides additional protection against accidental connection to the telephone line.

#### 4.4 Output Device Control - PA0, PA1, PA2

Only one device at a time can be output to the telephone line. The device to be sent onto the telephone line is selected by three PIA outputs, PA0, PA1 and PA2 as determined by Table 2-A and Table S-2. For example, if you want the TAPE PLAYER to be heard on the telephone, then use the command:

```
POKE 63488,3
```

#### 4.5 Automatic Telephone Dialing - 1PB0 - 1PB7

The telephone can be automatically dialed by software commands to the eight PIA output signals 1PB0 through 1PB7. Both tone and pulse dialing share these data lines. The type of dialing (pulse or tone) must be selected before sending the dialing data to the UTI. This selection of dialing method is described in Section 4.4, Output Device Control, and in Table T-2A and Table S-2. If the computer is attached to a touch-tone telephone line, then tone dialing can always be used. If the telephone line does not have tone decoding capabilities, then pulse dialing must be used. The last data sent to the automatic dialer MUST be in the "OFF" state, i.e., POKE 63494,255. (See Table T-1 and Table S-6.)

#### 4.6 Tape Player Control - PB2

The UTI can turn a Cassette Tape Player's motor on and off by using a relay on the UTI board which is controlled by PB2. The tape player's "REMOTE" jack must be connected to the Tape Player Control jack, J4 on the back of the computer, via a patch cord. (See Table T-3 and Table S-3.)

#### 4.7 Tape Recorder Control - PB3

The UTI can turn a Cassette Tape Recorder's motor on and off by using a relay on the UTI board which is controlled by PB3. The tape recorder's "REMOTE" jack must be connected to the Tape Recorder's Control jack, J1 on the back of the computer, via a patch cord. (See Table T-3.)

#### 4.8 FAILSAFE Control/Monitor

The status of the FAILSAFE feature can be read by the computer on PA6, FAILSAFE STATUS (at decimal address 63488). If the FAILSAFE STATUS bit is low (0), then the FAILSAFE feature has timed out and the telephone is hung up. If the FAILSAFE STATUS signal is high (1), then no time out on the FAILSAFE feature has occurred and the telephone lines may still be connected to the UTI. This signal allows the software to determine if the FAILSAFE feature has disconnected the telephone line from the UTI.

You may override the FAILSAFE feature by making a hardware modification to the 567 (UTI) board. To do this, you must cut the land connecting U5C Pin 4 (74LS04) to U3C Pin (74LS20). Then add a jumper connecting the pullup (4.7K ohm resistor) to U3C Pin 5 (see Figure 4-2). This modification eliminates the FAILSAFE feature which times out and hangs up the telephone in the event of a computer failure or when the "BREAK" or "RESET" buttons are depressed. This modification has been documented so that if the advanced user of the UTI wishes to do special programming, he may do so without constantly strobing the FAILSAFE STROBE. However, we suggest that the user who makes this modification disabling the FAILSAFE feature do so with full knowledge of the safety feature which he is disabling. This will allow easier user programming of the UTI, however, the demonstration software provided uses this FAILSAFE feature and, if the FAILSAFE feature is disabled by the user, then the demonstration software may not operate properly.

#### 4.9 Ring Sense

The UTI can determine if the telephone is ringing by the state of PA7. If PA7 is low (0), then the telephone is ringing. If PA7 is high (1), then the telephone is not ringing. (See Table 2-B and Table S-4.)

#### 4.10 DTMF (Touch-Tone) Decode - 1PA0, 1PA1, 1PA2, 1PA3

The UTI can determine the button depressed on a touch-tone telephone located on either end of the telephone line by using 1PA0-1PA3 PIA input signals. The state of input signal 1CA1 determines whether a new tone has been decoded. (See Table T-4A and Table S-5.)

#### 4.11 Dial Tone and Busy Detection

A dial tone and a busy signal can be detected using the input signal Telephone Line Threshold (PB7). This input signal will be a zero (low) if any sound is present on the telephone line. Since a dial tone is a constant signal on the telephone line, it will be represented by a constant zero on PB7. Since the busy signal is alternating on and off tones on the telephone line, it will be represented by alternating ones and zeros on the telephone line threshold signal (PB7). More information is given about this signal in Table T-3 and Table S-3.

The advanced user who wishes to experiment with other methods of dial tone detection and busy detection may wish to use the telephone line hard limited input signal (PB6). This signal is a real time logic representation of the signal present on the telephone line.

#### 4.12 Modem

##### A. Originate Mode Control

To switch the modem to Original Mode, the PIA output signal 1PA6 must be set to zero. This makes the SH signal on the modem LOW. The RI (1PA7) signal must be HIGH when in Originate Mode. (See Table T-4B and Table S-1.)

##### B. Answer Mode Control (RI) 1PA7

To switch the modem to Original Mode, set the PIA output signal 1PA7 to zero. This makes the RI signal on the Modem LOW. The SH (1PA6) signal must be HIGH when in "Answer" Mode. (See Table T-4B and Table S-1.)

##### C. Mode Sense - 1PA4

The modem located on the UTI can function in the Originate or Answer Mode. The modem's mode is determined by the CPU's control of the ST (1PA5), SH (1PA6), and RI (1PA7) output signals. The mode is sensed by reading the 1PA4 PIA input signal. (See Table S-1 and Table T-4B.)

##### D. Self-Test (ST) 1PA5

The modem on the UTI board can be run in the self-test mode. The PIA's output controls the self-test of the modem. (See Table T-4B and Table S-1.)

##### E. Input Ready Monitor

The modem on the UTI board provides a signal which tells the computer when a character has been input to the modem and is ready for the computer to read it. This signal is called "Receive Data Register Full" (RDRF). It is located in Bit 0 at address 63497. If this bit is HIGH (1), then the modem has received new data and this data should be read immediately. The reading of this data by the computer will reset the RDRF flag to zero (0).

If the RDRF flag is a zero, then the modem has not received any new data.

#### F. Output Ready Monitor

The modem on the UTI board provides a signal which tells the software if it is ready to accept a new output byte. This signal is called "Transmit Data Register Empty" (TDRE) flag. It is located in Bit 1 at location 63496. If this bit is a one (TRUE), then the modem is now ready to accept another output character.

If the TDRE flag is a zero (FALSE), then the modem is not ready to accept another output character. The program should continue reading this flag until it is (TRUE) before outputting another character to the modem.

#### 4.13 VOTRAX Control

If the UTI contains the VOTRAX option or if you have a CA-14 board which is properly connected to the UTI board (see Section 2.2 for proper connection), the VOTRAX output may be sent to the telephone line by using the output device control signals PA0, PA1 and PA2, which are described in Section 4.4. By selecting the VOTRAX output to the telephone line, the VOTRAX will be heard on both the telephone and on the external speaker. The VOTRAX on the UTI board is controlled identically to that of a separate VOTRAX on a CA-14 board. Any VOTRAX on the UTI board and the output of the VOTRAX will be heard on the speaker if it is connected properly to the UTI board.

## SECTION 5

### SOFTWARE

#### 5.1 Introduction

The UTI comes with one disk of demo software running under OS-65D. Included are such programs as:

- \*Home monitor demo allows a touch tone equipped caller to interrogate the AC-17P home security system status and to send commands to AC-12P by A.C. remote control.
- \*Automatic dial modem program allows UTI equipped computer to function as a timeshare terminal with hands off modem operation.
- \*Remote computer interrogation: demo software for two UTI equipped computers. One machine automatically interrogates the other, useful as the basis of remote monitoring and process control systems.

The descriptions for this disk and its programs are in the following section.

#### 5.2 OS-65D V3.2 Phone Demo

The first piece of software is called "PHONE DEMO", which is based on OS-65D V3.2 HC system with the modifications shown in SP-1.

A failsafe time out routine is the first bit of code added. This code checks if the failsafe has hung up the phone and, if so, recycles the software by running "PHONE". To enable the section of code: a one must be placed in the time out flag (\$E4 hex. or 228 dec.) by:

```
POKE 228,1
```

or disabled by:

```
POKE 228,0
```

The other main addition to the system is an auto strobe for the failsafe. This is used in modem operation. The code first checks for a clear-to-send (CTS) from a modem. Then as long as the CTS remains on the phone line the failsafe is continuously strobed. But, if CTS is lost for any reason the software will recycle itself by running "PHONE". This enables any user of modem operations to use the modem continuously without having to strobe

the failsafe manually every 108 seconds. To enable this code; a one must be placed in the modem auto strobe flag (\$E3 hex. or 227 dec.) by:

POKE 227,1

or disabled by: POKE 227,0

Some of the DOS commands were removed, as well as 'EM' and 'ASM', which were removed in the original Home Control system. The other commands which were removed were; 'IO', 'IN', and 'EX' (Examin). The commands were removed to create room for these system modifications. Use any OS-65D V3.2 disk if the 'IN' or 'EX' commands are desired by booting up on the regular 65D and switching diskettes. Using the 'IO' command can be accomplished by POKEing 8993 dec. with the input devices and 8994 dec. with the output devices.

Example: If you wish serial input from a terminal and serial output to the terminal and to a parallel printer you would:

POKE 8993, 1 and POKE 8994, 9

\*Note: HC programs cannot be loaded into regular OS-65D.

There is also a VOTRAX and AC driver, a 'DUMB' terminal program and one disk buffer in front of the programs 'PHONE', 'PHONE\*', 'WETINT', 'WETANS', and 'MODSEF'.

The printers supported by this system are:

- 1) Parallel printers on Serial systems
- 2) Serial printers on Polled systems

During all operations the disk should be left in the disk drives since the disk is accessed in different ways by many of the programs.

The disk contains seven main selections. Your selections are:

- 1 CALL ANY NUMBER
- 2 MONITOR PHONE CALLS
- 3 CALL ANY MODEM
- 4 MONITOR FOR MODEM CALLS
- 5 INTERROGATE WEATHER STATION
- 6 WEATHER STATION
- 7 MODEM SELF TEST

There are explanations for each of these selections in the following sections.



The first selection is 'CALL ANY NUMBER'. This will dial any number entered. Enter the number as prompted using numbers, letters, blanks, parentheses, or dashes in typing out the phone number. The number will not be dialed if any other character is entered. The program dials the number and determines if there is a voice, a busy signal, or voice after rings. The program then lets you hang up the phone when desired.

- NOTES:
- 1) If you do not physically pick up the phone before 108 seconds, the failsafe will time-out and hang up the phone automatically.
  - 2) All programs check for a dial tone first, therefore, if a dial tone is not detected, you will again be asked for a phone number.
  - 3) All programs use touch tone dialing, which can be changed by changing all GO SUB 1500 to GO SUB 1600.

The second selection is 'MONITOR FOR PHONE CALLS'. This program requires a tape recorder to be hooked into the system (see hardware hook up instructions). First you have a choice of resetting the time. This is not necessary if time has previously been set to the correct time. Now, enter the password (10 letters or less) which will give you access to your computer by phone using touch tone codes. The steps for calling this monitor are as follows:

<u>STEP NO.</u>	<u>DESCRIPTION</u>
1	Dial the computer on the telephone. (The computer will answer the phone with voice from VOTRAX. It will ask you to leave a message after the tone.)
2	A. Leave any message. (The computer will record the message and after 70 seconds, thank you, hang up and go back to step 1.)

-OR-

- |   |  |
|---|--|
|   | B. Dial the PASSWORD you defined for this monitor.<br>(The computer will give the time of day and allow you to control and monitor the home.) (If you dial the wrong PASSWORD then dial * or #, and re-dial it.) |
| 3 | A. Enter a "1" for lamp and appliance control, then:   |

1. Enter the two-digit lamp/appliance DEVICE NUMBER.  
(Enter "\*" to return you to step 3.)
2. Enter the one-digit OPERATION code:
 

1 = ON	2 = BRIGHT	3 = ALL LIGHTS ON
4 = OFF	5 = DIM	6 = ALL DEVICES OFF

(If operation 2 or 5 is selected, the computer will go to step 3 A3.)
3. Enter a two-digit LEVEL code between 01 and 15.  
(Lamp will brighten or dim and the computer will go back to step 3 A1.)

-OR-

- B. Enter a "2" for a security interrogation.  
(Computer will give a report on the status of your home security system and return to step 3.)

-OR-

- C. Enter a "3" for end of call.  
(Computer will thank you, hang up, and go back to step 1.)

The next selection is 'CALL ANY MODEM'. The first question is about enabling start and stop print commands sent to the intelligent computer during printer operations done locally by the terminal program. Next we check to see if our start print and stop print codes agree with that of the modem being called (CTRL-S is a common stop print code and CTRL-Q or CTRL-W are common start print codes). If you wish to change the start and stop print codes type 'YES' and then the control characters used for each of the commands. The phone number must then be entered in the same manner as in the first selection. After receiving a CTS from the intelligent computer, your terminal program will be enabled (see SP-2). These are your control commands when operating in the 'DUMB' terminal program;

- |        |   |
|--------|---|
| CTRL-B | To hang up phone (i.e., GOOD-BYE)                             |
| CTRL-D | To enable/disable a duplex condition                          |
| CTRL-E | To enable/disable an auto echo condition                      |
| CTRL-Y | To turn on/off a serial printer if you have a Polled system   |
| CTRL-Z | To turn on/off a parallel printer if you have a Serial system |

The program uses your specified prints start and print stop codes when output to a printer is selected. Therefore, sometimes the intelligent computer will miss a start print command and the output from the intelligent terminal will hang. Simply depress your start code and the output from the intelligent terminal will again start. This program can be used to call the weather station (selection number 6).

\*NOTE: Weather station uses CTRL-S and CTRL-Q for stop print and start print codes.

Selection number four is 'MONITOR FOR MODEM CALLS'. This program will monitor for 'DUMB' terminal calls by answering the phone upon ringing and looking for a CTS. If CTS is not detected or if CTS is lost at anytime, the program will hang up and wait for another call. After answering, the program allows the caller to enter OS-65D V3.2 HC system itself and essentially use the computer by modem over the phone.

The fifth selection is 'INTERROGATE WEATHER STATION'. This program will first interrogate the user. First, which weather operations the user wishes to interrogate. Then, the program asks if output to a printer is desired. Now, the program will ask for the time in minutes between calls, enter any fraction or whole number of minutes. This will be the amount of time after hanging up until the program will call the weather station again. Next, you are asked how many times you wish to call the weather station and sample these readings. Enter the phone number and let the program go. As long as the phone line to the weather station is free and you haven't recalled too soon (less than .01 minutes between calls) the program will give you the time for each sample along with each sampling as many times as specified.

Selection number six is a simulated weather station. It will auto answer any call and send simulated weather statistics over the phone to any modem or 'DUMB' terminal. The weather station will hang up the phone after sign-off or any break in CTS from the calling modem.

NOTE: This program is used by the auto interrogate program (selection #5) and can be used by selection #3. System uses CTRL-S and CTRL-Q for stop print and start print codes.

The last selection is 'MODEM SELF TEST'. This program checks the modem for character input and output through the modem. It does a test with five hundred characters and does not pick up the phone in the program. This program should be run anytime you believe the modem seems to be creating problems during monitor or 'DUMB' terminal operations.

These are the descriptions of the programs on this disk. After boot-up you are asked for the time. Enter the time, and if you wish the system unlocked, enter "PASS" in response to "A.M. or P.M.?" then enter A.M. or P.M. When the menu appears on the screen, just depress <RETURN>. You will now be in the system and able to do any modification you desire. The following are listings of the overlays to DOS (SP-1), the "DUMB" terminal program (SP-2) and "PHONE" (SP-3). "PHONE" is used to recycle all monitors and contains many subroutines used by other programs on the disk which may be helpful in your programming efforts.

Software Program 1 (SP-1)

```

10 2E54          *=$2E54
20 2E54 01      . BYTE 1          DISABLE 'IN' COMMAND
30
40 2E58          *=$2E58
50 2E58 01      . BYTE 1          DISABLE 'IO' COMMAND
60
70 325C          *=$325C
80 325C 4C552B  JMP $2B55          SET UP JUMP TO PATCH FROM HC CODE
90
100 2B55         *=$2B55
110
120 22C5=        CEVENT=$22C5
130 00C7=        TXTPTR=$00C7
140 07E0=        GONE=$07E0
150 00E3=        MODEMF=$00E3
160 00E4=        TIOUTF=$00E4
170
180 2B55 A5E4     LDA TIOUTF          GET FAILSAFE TIME OUT FLAG
190 2B57 4A      LSR A              SHOULD WE LOOK AT THIS CODE
200 2B58 9007    BCC MODCHK        NO, SEE IF WE ARE STROBING THE FAILSAFE
210 2B5A AD00F8  LDA $F800         GET BYTE
220 2B5D 2940    AND #64           BIT #6 LOW ?
230 2B5F F01E    BEQ GOODBY       YES, THEN HANG UP PHONE & RETURN
240
250 2B61 A5E3    MODCHK LDA MODEMF        NO, CHECK AUTO STROBE
260 2B63 4A      LSR A              ARE WE IN AUTO STROBE ?
270 2B64 9016    BCC ANRTS        NO, RETURN TO ORIGINAL CODE
280 2B66 AD08F8  LDA $F808        YES, GET BYTE NEEDED
290 2B69 4A      LSR A              SEE
300 2B6A 4A      LSR A              IF
310 2B6B 4A      LSR A              CTS
320 2B6C 4A      LSR A              FOUND IN BIT #3
330 2B6D B010    BCS GOODBY       NO, HANG UP AND RETURN
340 2B6F AD02F8  LDA $F802        YES, GET BYTE NEEDED
350 2B72 0920    ORA #32           AND STROBE THE FAILSAFE
360 2B74 8D02F8  STA $F802
370 2B77 29DF    AND #223
380 2B79 8D02F8  STA $F802
390 2B7C 4CC522  ANRTS JMP CEVENT    ALL DONE HERE RETURN TO ORIGINAL CODE
400
410 2B7F A907    GOODBY LDA #7
420 2B81 8D00F8  STA $F800        TURN OFF OUTPUT TO PHONE
430 2B84 A90E    LDA #14
440 2B86 8D02F8  STA $F802        HANG UP PHONE
450 2B89 A900    LDA #0
460 2B8B 85E3    STA MODEMF       TURN AUTO STROBE OFF
470 2B8D 85E4    STA TIOUTF       TURN OFF FAILSAFE TIME OUT CHECK
480 2B8F A92B    LDA #PHONE/256
490 2B91 85C8    STA TXTPTR+1
500 2B93 A99A    LDA #PHONE
510 2B95 85C7    STA TXTPTR
520 2B97 4CE007  JMP GONE         SET UP AND RUN PROGRAM 'PHONE'
530
540 2B9A 3A      PHONE . BYTE $3A, $89, "PHONE", $0
540 2B9B 89
540 2B9C 22
540 2B9D 50
540 2B9E 48

```

```

540 2B9F 4F
540 2BA0 4E
540 2BA1 45
540 2BA2 22
540 2BA3 00
550
560 2BA4 00      . BYTE 0,0,0   PROGRAM SELECT, DEFAULT DEVICE, FREE
560 2BA5 00
560 2BA6 00
570
580 2E44          **$2E44
590 2E44 01      . BYTE 1      . DISABLE 'EM'  COMMAND
600
610 2E48          **$2E48
620 2E48 01      . BYTE 1      . DISABLE 'EX'  COMMAND (EXAMIN)
630
640 2B2F          **$2B2F
650
660 2B2F 48      OUT550 PHA          SHORT OUTPUT SUB FOR
670 2B30 AD00FC  LDA $FC00      SERIAL TERMINAL WHEN IN
680 2B33 4A      LSR A          USE BY THE MODEM
690 2B34 4A      LSR A          THROUGH
700 2B35 90F9    BCC OUT550+1  MODIFICATIONS
710 2B37 68      PLA          TO DEVICE #1
720 2B38 8D01FC  STA $FC01
730 2B3B 60      RTS
740
750          ; BYTES FOR 10 LETTER PASSWORD IN MONITOR PHONE PROGRAM
760          ;
770 2B3C 00      . BYTE 0,0,0,0,0,0,0,0,0,0
770 2B3D 00
770 2B3E 00
770 2B3F 00
770 2B40 00
770 2B41 00
770 2B42 00
770 2B43 00
770 2B44 00
770 2B45 00

```

Software Program 2 (SP-2)

```

10 4000          *=$4000
20
30 2599=        CRTOUT=$2599
40 2AC5=        DEFAULT=$2AC5
50 3180=        KBDIN=$3180
60 2644=        KBSWAP=$2644
70 267A=        MRKT=$267A
80 2343=        OUTCH=$2343
90 2322=        OUTFLG=$2322
100 00E5=       SFLAG=$00E5
110 24CD=       TTYOUT=$24CD
120
130 4000 AD2223 LDA OUTFLG      GET SELECTED DEVICE
140 4003 204E41 JSR PRINTR      SET UP OUTPUT FOR DEVICES SELECTED
150
160 4006 ADC62A LDA DEFAULT+1   GET DEFAULT DEVICE
170 4009 C902   CMP #2         IS IT VIDEO ?
180 400B F014   BEQ START      YES, CODE IS OKAY START EXECUTION
190
200 400D A940   LDA #KBSERL/256
210 400F 8D5840 STA KEY+2      SET UP I/O FOR A SERIAL TERMINAL
220 4012 A9E3   LDA #KBSERL
230 4014 8D5740 STA KEY+1
240 4017 A924   LDA #TTYOUT/256
250 4019 8D6840 STA KBD1+2
260 401C A9CD   LDA #TTYOUT
270 401E 8D6740 STA KBD1+1
280
290 4021 204426 START JSR KBSWAP      DO A SWAP IN CASE KBD POLLED
300
310 4024 AD08F8 P1   LDA $F808      DO WE HEAR CTS ?
320 4027 2908     AND #$08      CHK BIT #3
330 4029 D077     BNE GOODBY+2  NO, HANG UP AND RETURN TO BASIC
340
350 402B AD02F8 STROBE LDA $F802      YES, ALL IS WELL
360 402E 0920     ORA #32      STROBE THE FAILSAFE
370 4030 8D02F8 STA $F802      SO WE DON'T HANG UP
380 4033 29DF     AND #223     THE PHONE.
390 4035 8D02F8 STA $F802
400
410 4038 AD08F8 MOREAD LDA $F808      READ THE MODEM ?
420 403B 4A      LSR A        IS THERE A BYTE READY ?
430 403C 9015     BCC KEYBRD   NO, CHK KEYBOARD FOR INPUT
440 403E AD09F8 MODMIN LDA $F809     YES, GET THE BYTE
450 4041 EA      . BYTE $EA, $EA   CMP #'B-$40 (MODIFIED)
460 4042 EA      . BYTE $EA, $EA   BEQ GOODBY+2 (MODIFIED)
470 4043 EA
480 4044 EA
490 4045 ACE240   LDY AUTOEC    CHK AUTO ECHO ?
500 4048 F003     BEQ OUTCHR   NO, OUTPUT CHR
510 404A 207040   JSR MOWRIT   YES, ECHO IT TO THE MODEM
520 404D 201141 OUTCHR JSR CHECK     CHK IF STOP PRINT NEEDED OVER MODEM
530 4050 204323   JSR OUTCH    OUTPUT CHR
540
550 4053 20EF40 KEYBRD JSR CHECKS    CHK IF START PRINT NEEDED OVER MODEM
560 4056 208031 KEY     JSR KBDIN    GO GET ANY KEYS DEPRESSED
570 4059 F0C9     BEQ P1       NO KEYS START LOOP OVER
580 405B 207D40   JSR KEYCHK   CHK FOR ANY SPECIAL KEY DEPRESSIONS

```

570	405E	F0C4		BEQ P1	YES, THERE WAS ONE EXECUTED
580	4060	ACE140	KBD	LDY DUPLEX	DUPLEX FLAG SET
590	4063	F005		BEQ KBD2	NO, WRITE CHR TO MODEM
600	4065	48		PHA	YES
610	4066	209925	KBD1	JSR CRTOUT	OUTPUT CHR AT LOCAL TERMINAL FIRST
620	4069	68		PLA	RESTORE A
630	406A	207040	KBD2	JSR MOWRIT	WRITE KEY DEPRESSION TO MODEM
640	406D	4C2440		JMP P1	START LOOP OVER
650					
660	4070	48	MOWRIT	PHA	SAVE A
670	4071	AD08F8		LDA \$F808	WAIT FOR READY TO SEND
680	4074	4A		LSR A	
690	4075	4A		LSR A	READY ?
700	4076	90F9		BCC MOWRIT+1	NO, CHK AGAIN
710	4078	68		PLA	YES, RESTORE A
720	4079	8D09F8		STA \$F809	SEND BYTE
730	407C	60	ANRTS	RTS	ALL DONE
740					
750	407D	C902	KEYCHK	CMP #'B-\$40	CTRL-B ? (HANG UP PHONE)
760	407F	F01F		BEQ GOODBY	YES, HANG UP AND RETURN
770	4081	C904		CMP #'D-\$40	CTRL-D ? (DUPLEX ENABLED/DISABLED)
780	4083	F051		BEQ FLPDUP	YES, CHANGE FLAG
790	4085	C905		CMP #'E-\$40	CTRL-E ? (AUTO ECHO ENABLE/DISABLE)
800	4087	F042		BEQ FLPECO	YES, CHANGE FLAG
810	4089	C919		CMP #'Y-\$40	CTRL-Y ? (SERIAL PRINTER E/D)
820	408B	F02F		BEQ SERIAL	YES, MAKE CHANGES
830	408D	C91A		CMP #'Z-\$40	CTRL-Z ? (PARALLEL PRINTER E/D)
840	408F	D0EB		BNE ANRTS	NO, RETURN
850					
860	4091	AD2223	PARALL	LDA OUTFLG	YES, SEE WHAT DEVICES WE HAVE
870	4094	C902		CMP #2	VIDEO ?
880	4096	F0E4		BEQ ANRTS	YES, DON'T TURN ON PARALLEL PRINTER
890	4098	4908		EOR #8	NO, CHANGE OUTPUT FLAG
900	409A	204E41		JSR PRINTR	GO CHANGE CODE TO SUPPORT CONDITIONS
910	409D	A900		LDA #0	CLEAR A
920	409F	60		RTS	ALL DONE
930					
940	40A0	68	GOODBY	PLA	RETURN TO THE RIGHT PLACE
950	40A1	68		PLA	
960	40A2	ADC62A		LDA DEFAULT+1	SET BACK TO DEVICES WE ENTERED WITH
970	40A5	8D2223		STA OUTFLG	
980	40A8	AD0A41		LDA CTRLQ+1	SEND START PRINT JUST IN CASE
990	40AB	207040		JSR MOWRIT	
1000	40AE	A90E		LDA #14	HANG UP THE PHONE
1010	40B0	8D02F8		STA \$F802	
1020	40B3	A907		LDA #7	TURN OFF OUTPUT TO PHONE
1030	40B5	8D00F8		STA \$F800	
1040	40B8	204426		JSR KBSWAP	RETURN POLLED KEYBOARD
1050	40BB	60		RTS	RETURN TO TO CALLER
1060					
1070	40BC	AD2223	SERIAL	LDA OUTFLG	SEE WHAT DEVICES WE HAVE
1080	40BF	C901		CMP #1	SERIAL ?
1090	40C1	F0B9		BEQ ANRTS	YES, DON'T TURN ON SERIAL PRINTER
1100	40C3	4901		EOR #1	NO, CHANGE OUTPUT FLAG
1110	40C5	204E41		JSR PRINTR	GO CHANGE CODE TO SUPPORT CONDITIONS
1120	40C9	A900		LDA #0	CLEAR A
1130	40CA	60		RTS	ALL DONE
1140					
1150	40CB	ADE240	FLPECO	LDA AUTOEC	CHANGE AUTO ECHO FLAG E/D
1160	40CE	4901		EOR #1	



```

1170 40D0 8DE240      STA AUTOEC
1180 40D3 A900        LDA #0
1190 40D5 60          RTS
1200
1210 40D6 ADE140 FLPDUP LDA DUPLEX      CHANGE DUPLEX FLAG E/D
1220 40D9 4901        EOR #1
1230 40DB 8DE140      STA DUPLEX
1240 40DE A900        LDA #0
1250 40E0 60          RTS
1260
1270 40E1 00          DUPLEX . BYTE 0      DUPLEX FLAG
1280 40E2 00          AUTOEC . BYTE 0    AUTO ECHO FLAG
1290
1300 40E3 AD00FC KBSERL LDA $FC00      SERIAL KEYBOARD INPUT ROUTINE
1310 40E6 4A          LSR A
1320 40E7 A900        LDA #0
1330 40E9 9003        BCC KBSER1
1340 40EB AD01FC      LDA $FC01
1350 40EE 60          KBSER1 RTS
1360
1370 40EF A5E5 CHECKS LDA SFLAG      LOAD STOP PRINT FLAG
1380 40F1 CD1B41      CMP CTRLS+2        HAS A STOP PRINT BEEN SENT
1390 40F4 D01A        BNE OUT2           NO, RETURN
1400 40F6 AE7141      LDX COUNT          YES, IS THERE ANY CHR IN BUFFER
1410 40F9 F00E        BEQ CTRLQ          NO, SEND START PRINT CODE
1420 40FB A000        LDY #0            YES, GET CHR OUT OF BUFFER
1430 40FD B97241 LOOPER LDA BUFFER, Y
1440 4100 204323      JSR OUTCH          SHOW WHAT WE HAVE
1450 4103 C9          INY
460 4104 CC7141      CPY COUNT          IS THAT ALL
1470 4107 D0F4        BNE LOOPER        NO, GET NEXT ONE
1480 4109 A911 CTRLQ  LDA #'Q-$40        YES, SEND START PRINT COMMAND
1490 410B 85E5        STA SFLAG          RESET FLAG
1500 410D 207040      JSR MOWRIT
1510 4110 60          OUT2  RTS          ALL DONE
1520
1530 4111 C90D CHECK  CMP #$0D          IS CHR A CRT OR LF
1540 4113 F004        BEQ CTRLS          IF YES SEND STOP PRINT
1550 4115 C90A        CMP #$0A          IF NO RETURN
1560 4117 D02E        BNE OUT
1570 4119 48 CTRLS  PHA          SAVE A
1580 411A A913        LDA #'S-$40
1590 411C 85E5        STA SFLAG          SET FLAG
1600 411E 207040      JSR MOWRIT          SEND STOP PRINT CODE
1610 4121 A200        LDX #0
1620 4123 8E7141      STX COUNT          SET # IN BUFFER TO 0
1630 4126 AC7B26 LOOPY  LDY MRKT+1        WAIT 4 CHR TIME FOR ANY CHR STILL COMING
1640 4129 AD08F8 BEGIN  LDA $F808
1650 412C 4A          LSR A              IS THERE A CHR WAITING
1660 412D 900B        BCC LOOP49        NO, CONT. WAIT
1670 412F AD09F8      LDA $F809        YES, SAVE CHR
1680 4132 EA          . BYTE $EA, $EA   CMP #'B-$40 (MODIFIED)
1680 4133 EA
1690 4134 EA          . BYTE $EA, $EA   BEQ OUT3 (MODIFIED)
1690 4135 EA
1700 4136 9D7241      STA BUFFER, X
1710 4139 E8          INX              NEXT ?
1720 413A 18 LOOP49 CLC
1730 413B 88          DEY
1740 413C D0EB        BNE BEGIN        LOOP AGAIN

```

```

1750 413E EE7141      INC COUNT      DONE ?
1760 4141 D0E3        BNE LOOPY      NO. TRY AGAIN
1770 4143 8E7141      STX COUNT      SAVE # OF CHRS IN BUFFER
1780 4146 68          PLA           RESTORE A
1790 4147 60          OUT          . BYTE $60, $02      CMP #'B-$40 (RTS, #'B-$40)
1790 4148 02
1800 4149 D0C5        BNE OUT2      (MODIFIED)
1810 414B 4CA040      JMP GOODBY
1820
1830 414E 8D2223 PRINTR STA OUTFLG     SAVE DEVICES SELECTED
1840 4151 C909        CMP #9        IS EITHER PRINTER TURNED ON ?
1850 4153 F00D        BEQ PRTEANA   YES, ENABLE PRINT START & STOP
1860 4155 C903        CMP #3
1870 4157 F009        BEQ PRTEANA   YES, ENABLE PRINT START & STOP
1880 4159 A960 PRTDIS LDA ##60      NO, DISABLE PRINT START & STOP
1890 415B 8DEF40      STA CHECKS
1900 415E 8D1141      STA CHECK
1910 4161 60          RTS           RETURN
1920 4162 A9A5 PRTEANA LDA ##A5
1930 4164 8DEF40      STA CHECKS
1940 4167 A9C9        LDA ##C9
1950 4169 8D1141      STA CHECK
1960 416C 60          RTS           ALL DONE
1970
1980 416D 68          OUT3        PLA           SET TO RIGHT RETURN (MODIFIED)
1990 416E 4CA040      JMP GOODBY    HANG UP AND RETURN
2000
2010 4171 00          COUNT      . BYTE 0      COUNTER BYTE
2020
2030 4172=          BUFFER==    BUFFER FOR INCOMING CHRS

```

Software Program 3 (SP-3)

```

1 REM ** VARIABLE/PIA INITIALIZATION
3 POKE56832, 1: POKE227, 0: POKE9425, 74: POKE9426, 74: POKE9427, 144
4 JX=0: I=0
5 V1=1: V2=300: V3=128: V4=13696: V5=7: GL=9: R=0: X=0: J=0: E=0: BU=0
10 C=49/PEEK(9851): OT=4: DT=40: RT=500
15 IF C<1. 2 THEN JX=0: IF C<. 7 THEN JX=14: IF C<. 4 THEN JX=23
20 FOR I=0 TO 9: F8(I)=63488+I: NEXT
30 DIM TN$(15): FOR I=0 TO 15: READ TN$(I): NEXT
40 FOR I=0 TO 9: READ OT(I): NEXT
50 POKE249, 1: POKE548, 127: POKE549, 66: POKE8985, 125: POKE8986, 62
51 POKE17149, 32: POKE17150, 30: POKE17151, 50
55 FOR I=0 TO 6 STEP 2: POKE F8(I)+I, 6: NEXT
60 FOR I=0 TO 6 STEP 2: POKE F8(I)+I, 255: NEXT
65 FOR I=0 TO 6 STEP 2: POKE F8(I)+I, 0: NEXT
70 POKE F8(0), 7: POKE F8(2), 47: POKE F8(4), 224: POKE F8(6), 255
75 FOR I=0 TO 6 STEP 2: POKE F8(I)+I, 6: NEXT
80 FOR I=0 TO 6 STEP 2: POKE F8(I)+I, 255: NEXT
85 POKE F8(8), 67: W=10: GOSUB 1000: POKE F8(8), 17: POKE F8(2), 14
86 W=300: GOSUB 1000
90 DATA 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, *, #, A, B, C
91 DATA 215, 238, 222, 190, 237, 221, 189, 235, 219, 187
95 GOSUB 1200
100 REM ** SOURCE GOES HERE; LINES 100-999
103 IF PEEK(11172)=6 THEN ZX=6: GOTO 400
104 IF PEEK(11068)<>0 THEN RUN "PHONE*"
105 IF PEEK(9423)=8 AND PEEK(9424)=248 THEN GOSUB 3500
106 IF PEEK(11172)<>0 THEN ZX=PEEK(11172): GOTO 400
107 FOR I=1 TO 30: PRINT: NEXT
110 PRINT " PHONE DEMO ROUTINES"
120 PRINT: PRINT
130 PRINT " 1> CALL ANY NUMBER"
140 PRINT " 2> MONITOR PHONE CALLS"
150 PRINT " 3> CALL ANY MODEM"
160 PRINT " 4> MONITOR FOR MODEM CALLS"
170 PRINT " 5> INTERROGATE WEATHER STATION"
180 PRINT " 6> WEATHER STATION"
190 PRINT " 7> MODEM SELF TEST"
390 PRINT: PRINT: INPUT "SELECTION"; ZX
400 IF ZX=1 THEN GOSUB 4000: GOTO 100
410 IF ZX=2 THEN POKE 11172, 2: RUN "PHONEM"
420 IF ZX=3 THEN GOSUB 3000: GOTO 100
430 IF ZX=4 THEN GOSUB 3500: GOTO 100
440 IF ZX=5 THEN RUN "WETINT"
450 IF ZX=6 THEN POKE 11172, 6: RUN "WETANS"
460 IF ZX=7 THEN RUN "MODSEF"
900 GOTO 100
999 REM
1000 REM ** WAIT W * 10 MS
1010 WW=INT(W*7. 6/C): FOR X1=1 TO WW: NEXT: RETURN
1100 REM ** WAIT FOR RING
1110 GOSUB 1200: POKE F8(8), 67: W=4: GOSUB 1000: POKE F8(8), 17
1120 W=350: GOSUB 1000
1130 IF (PEEK(F8(0)) AND 128)=0 THEN 1130
1140 RETURN
1200 REM ** HANG UP PHONE
1210 POKE F8(2), 14: IF (PEEK(F8(2)) AND 14)<>14 THEN W=100: GOSUB 1000: GOTO 1210
1220 RETURN
1300 REM ** ANSWER PHONE
1310 GOSUB 1800: POKE F8(2), 13: RETURN

```

```

1400 REM ** DECODE A TONE
1410 REM     N=NUMBER OF TONES ; A$(X2)=NUMBERS DIALED
1420 GOSUB1800: FORX2=1TON: TE=PEEK(F8(4))
1430 GOSUB1850: IFPEEK(F8(5))<128THEN1430
1440 A$(X2)=TN$(PEEK(F8(4))AND15)
1450 NEXT: RETURN
1500 REM ** TONE DIALER
1510 REM     A#=NUMBER TO DIAL ERROR FREE
1515 POKEF8(0), 5
1520 FORX2=1TOLEN(A$): TE=OT(VAL(MID$(A$, X2, 1)))
1530 W=OT: GOSUB1000: POKEF8(6), TE: GOSUB1000: POKEF8(6), 255
1540 NEXT: RETURN
1600 REM ** PULSE DIALER
1610 REM     A#=NUMBER TO DIAL ERROR FREE
1620 POKEF8(0), 6: GOSUB1520: W=LEN(A$)*140: GOSUB1000: RETURN
1700 REM ** BUSY/RING DETECT
1703 FORI=4TO7: J(I)=0: NEXT
1705 R=0: GOSUB2200: GOSUB1800
1710 FORX=V1TOV2: FORI=V1TOJC: NEXT: E=(PEEK(F8(2))ANDV3): POKEV4+X, E: NEXT
1720 E=0
1730 J=V2: I=V1: FORX=V1TOV2: JX=PEEK(V4+X): IFJX<>JTHEN1750
1740 I=I+V1
1745 NEXT: GOTO1770
1750 IFI<GLTHENI=V1: J=JX: GOTO1745
1755 E=E+V1: IFE<OTTHENI=V1: J=JX: GOTO1745
1760 IFE>V5THEN1745
1765 J(E)=I: I=V1: J=JX: GOTO1745
1770 TE=J(4)+J(5): T1=J(6)+J(7): BU=0
1775 IFTE>55ANDTE<70THENIFT1>55ANDT1<70THENBU=V1
1780 IFTE>27ANDTE<38THENIFT1>27ANDT1<38THENBU=V1
1783 GOSUB1850
1785 IFJ(6)=0ANDJ(7)=0THENR=1: GOTO1710
1790 RETURN
1800 REM ** FAILSAFE STROBE
1810 X=PEEK(F8(2)): POKEF8(2), (X OR 32): POKEF8(2), (X AND 223)
1820 RETURN
1850 REM ** CHECK FOR FAILSAFE DISCONNECT
1860 IF(PEEK(F8(0))AND64)<>0THENRETURN
1870 RUN
1900 REM ** TAPE RECORD ON
1910 TE=PEEK(F8(2)): POKEF8(2), (TE AND 247): RETURN
1950 REM ** TAPE RECORD OFF
1960 TE=PEEK(F8(2)): POKEF8(2), (TE OR 8): RETURN
2000 REM ** TAPE PLAY ON
2005 POKEF8(0), 3
2010 TE=PEEK(F8(2)): POKEF8(2), (TE AND 251): RETURN
2050 REM ** TAPE PLAY OFF
2055 POKEF8(0), 7
2060 TE=PEEK(F8(2)): POKEF8(2), (TE OR 4): RETURN
2100 REM ** DIAL TONE DETECT
2105 REM     E=0 DIAL TONE ; E=1 NO DIAL TONE
2110 X9=0: E=0: GOSUB1800
2120 IF(PEEK(F8(2))AND128)=0THEN2150
2130 X9=X9+C: IFX9<RTTHEN2120
2140 E=1: RETURN
2150 X8=0: GOSUB1800
2160 X9=X9+C: IFX9>RTTHEN2140
2170 IF(PEEK(F8(2))AND128)>0THEN2130
2180 X8=X8+C: IFX8<DTTHEN2160
2190 RETURN

```

```

2200 REM ** SCAN OFF GLITCHES
2210 X9=0
2220 B=0
2230 X9=X9+C: IFX9>660THEN2290
2240 IF<PEEK<F8<2>>AND128>>0THEN2230
2250 B=B+C
2260 X9=X9+C: IFX9>660THEN2290
2270 IF<PEEK<F8<2>>AND128>=0THENB=B+C: GOTO2260
2280 IFB<GLTHEN2220
2290 RETURN
2300 REM ** ENABLE VOTRAX TO PHONE
2310 POKEF8<0>, 1: RETURN
2400 REM ** ENABLE AUX. DEVICE TO PHONE
2410 POKEF8<0>, 2: RETURN
2500 REM ** ENABLE MODEM OUTPUT
2510 POKEF8<0>, 4: RETURN
2600 REM ** DISABLE OUTPUT TO PHONE
2610 POKEF8<0>, 7: RETURN
3000 REM ** MODEM CALLER
3001 GOSUB6000
3002 POKEF8<4>, 224: PRINT
3003 IFL$<>" "THENPRINT"TYPE * OR # TO DIAL "L$" AGAIN. "
3004 PRINT
3005 POKEF8<8>, 67: W=10: GOSUB1000: POKEF8<8>, 3: W=300: GOSUB1000
3010 GOSUB1200: GOSUB5000: GOSUB1300: GOSUB2100
3020 IFE=1THENGOSUB1200: W=200: GOSUB1000: GOTO3010
3030 GOSUB1500: POKEF8<8>, 17: POKEF8<4>, 160: POKEF8<0>, 4
3040 IF<PEEK<F8<8>> AND 8>=0THEN3060
3050 GOSUB1850: GOTO3040
3060 POKEF8<4>, 224: POKE9689, 96: DISK!"GO 4000"
3065 POKE16721, 201: POKE9689, 169
3070 X=PEEK<10950>: POKE8993, X: POKE8994, X: RETURN
3500 REM ** MODEM AUTO ANSWER
3503 GOSUB3700: POKEF8<4>, 224
3505 POKEF8<8>, 67: W=10: GOSUB1000: POKEF8<8>, 3: W=300: GOSUB1000
3507 POKEF8<8>, 17: W=400: GOSUB1000
3508 FORI=1TO20: PRINT: NEXT: PRINT"WAITING FOR PHONE TO RING. "
3509 FORI=1TO10: PRINT: NEXT
3510 GOSUB1100: GOSUB1300: POKEF8<0>, 4
3520 IF<PEEK<F8<8>> AND 8>=0 THEN3540
3530 GOSUB1850: GOTO3520
3540 PRINT" ** COMPUTER NOW IN USE BY MODEM **"
3545 FORI=1TO15: PRINT: NEXT
3546 POKE9425, 41: POKE9426, 10: POKE9427, 240
3550 POKE9423, 8: POKE9424, 248
3555 POKE9431, 9: POKE9432, 248
3560 POKE9435, 8: POKE9436, 248
3565 POKE9466, 8: POKE9467, 248
3570 POKE9472, 9: POKE9473, 248
3575 POKE8993, 1: POKE8994, 1
3576 POKE10950, 1: POKE227, 1
3577 POKE8991, 46: POKE8992, 43
3578 IFDF=1THENPOKE8979, 46: POKE8980, 43
3580 W=100: GOSUB1000: FORI=1TO30: PRINT: NEXT
3590 PRINT"You are now running OS-65D V3.2 HC ."
3600 PRINT"The computer is now under your control. "
3610 PRINT"Typing 'PRINT#2,' , followed by any string will"
3615 PRINT"echo the string to the intelligent terminal. "
3620 FORI=1TO10: PRINT: NEXT
3630 END

```

```

3700 DF=PEEK(11173): POKE10950, DF: POKE8993, DF: POKE8994, DF
3705 POKE9425, 74: POKE9426, 74: POKE9427, 144
3710 POKE9423, 0: POKE9424, 252
3720 POKE9431, 1: POKE9432, 252
3730 POKE9435, 0: POKE9436, 252
3740 POKE9466, 0: POKE9467, 252
3750 POKE9472, 1: POKE9473, 252
3755 POKE8979, 152: POKE8980, 37
3760 POKE8991, 188: POKE8992, 36: RETURN
4000 REM ** CALL ANYONE
4010 REM RECORD FULL CALL
4030 REM HANG UP IF BUSY
4045 PRINT
4050 IFL$<>" THENPRINT"TYPE * OR # TO DIAL "L$" AGAIN. "
4060 PRINT
4100 GOSUB1200: GOSUB5000: GOSUB1300: GOSUB2100
4110 IFE=1THENGOSUB1200: W=200: GOSUB1000: GOT04100
4120 GOSUB1500: GOSUB1900: GOSUB1700
4125 PRINT
4130 IFBU=0ANDR=1THENPRINT"VOICE/RINGS"
4140 IFBU=0ANDR=0THENPRINT"VOICE"
4150 IFBU=1THENPRINT"BUSY"
4160 PRINT
4170 PRINT"DEPRESS <RETURN> TO HANG UP THE PHONE"
4175 GOSUB5100
4180 GOSUB1200: RETURN
5000 REM ** GET NUMBER TO BE DIALED <ERR, FREE> IN A$
5005 E=0: INPUT"PHONE NUMBER"; A$: IFA$="*"ORA$="#" THEN5080
5010 L$="": B$="ABCDEFGHIJKLMNPRSTUVWXY": FORI=1TOLEN(A$)
5020 TE$=MID$(A$, I, 1): IFTE$>="0"ANDTE$<="9" THENL$=L$+TE$: GOT05070
5025 IFTE$="-"ORTE$="("ORTE$=")"ORTE$=" " THEN5070
5030 IFTE$<"A"ORTE$>"Y"ORTE$="Q" THENE=1: GOT05070
5040 Z=1: FORJ=1TO22STEP3: Z=Z+1: FORK=0TO2
5050 IFTE$=MID$(B$, J+K, 1) THENL$=L$+RIGHT$(STR$(Z), 1)
5060 NEXTK: J
5070 NEXTI: IFE=1THEN5005
5080 A$=L$: IFA$="" THEN5005
5090 RETURN
5100 REM ** WAIT FOR A <RETURN> DEPRESSION
5110 POKE2888, 0: POKE8722, 0
5120 INPUTQX$
5130 POKE2888, 27: POKE8722, 27: RETURN
6000 FORI=1TO30: PRINT: NEXT
6001 PRINT"DO YOU WISH STOP PRINT AND START PRINT CODES SENT AFTER"
6002 INPUT"EACH LINE DURING PRINTER OPERATIONS"; TE$
6003 IFLEFT$(TE$, 1)="N" THENPOKE16623, 96: POKE16657, 96: POKE16721, 96: RETURN
6004 PRINT: PRINT
6005 PRINT"THE STOP PRINT AND START PRINT CODES ARE NOW: "
6010 PRINT: PRINT" CTRL-"CHR$(PEEK(16667)+64)" FOR THE STOP PRINT CODE
6020 PRINT: PRINT" CTRL-"CHR$(PEEK(16650)+64)" FOR THE START PRINT CODE
6030 PRINT: PRINT"DO THESE CODES AGREE WITH THE CODES USED BY"
6040 INPUT"THE MODEM YOU ARE CALLING"; TE$: PRINT
6050 IFLEFT$(TE$, 1)="Y" THEN RETURN
6055 GOSUB6080: GOSUB6060: RETURN
6060 INPUT"WHAT IS YOUR START PRINT CODE: CTRL-"; TE$
6070 TE=ASC(LEFT$(TE$, 1))-64: POKE16650, TE
6075 RETURN
6080 INPUT"WHAT IS YOUR STOP PRINT CODE: CTRL-"; TE$
6090 TE=ASC(LEFT$(TE$, 1))-64: POKE16667, TE
7000 PRINT: RETURN

```

APPENDIX A  
HARDWARE TABLES

TABLE T-1

Column and Row	C1	C3	C2	C1	R4	R3	R2	R1	PIA Data Value (Hex)	PIA Data Value (Dec.)	Key
PIA Bit:	1PB7	1PB6	1PB5	1PB4	1PB3	1PB2	1PB1	1PB0			
	1	1	1	0	1	1	1	0	EE	238	1
	1	1	1	0	1	1	0	1	ED	237	4
	1	1	1	0	1	0	1	1	EB	235	7
	1	1	1	0	0	1	1	1	E7	231	*(1)
	1	1	0	1	1	1	1	0	DE	222	2
	1	1	0	1	1	1	0	1	DD	221	5
	1	1	0	1	1	0	1	1	DB	219	8
	1	1	0	1	0	1	1	1	D7	215	0
	1	0	1	1	1	1	1	0	BE	190	3
	1	0	1	1	1	1	0	1	BD	189	6
	1	0	1	1	1	0	1	1	BB	187	9
	1	0	1	1	0	1	1	1	B7	183	*(1)
	0	1	1	1	1	1	1	0	7E	126	A <sup>(1)</sup>
	0	1	1	1	1	1	0	1	7D	125	B <sup>(1)</sup>
	0	1	1	1	1	0	1	1	7B	123	C <sup>(1)</sup>
	0	1	1	1	0	1	1	1	77	119	D <sup>(1)</sup>
	1	1	1	1	1	1	1	1	FF	255	"OFF" <sup>(2)</sup>

(1) These keys exist on Touch-Tone<sup>(C)</sup> sets only.

(2) "OFF" is used for space between tones. It does not have a corresponding key. A space of  $\geq 50$  ms<sup>(3)</sup> must exist between tones. Tone duration must be  $\geq 50$  ms<sup>(3)</sup>.

(3) These timing values are telephone industry minimum values. Some telephone exchanges may require longer times for reliable operation.

We use the PIA practice of locating the control register one location higher than the port it controls. For example, the PIA tone generator port =  $F806_{16}$ , and the tone generator control port =  $F807_{16}$ .

Each of the other PIA ports serves multiple functions, with each bit serving to choose or exclude a particular function. In order of locations these PIA's and their functions are described in Table T-2A thru T-6.

Table T-2A

Address Decimal	Hex	Bit			Value of			Function, which line is selected. All lines are output (from CPU).
		PA2	PA1	PA0	PA2	PA1	PA0	
63488	F800	0	0	0	0	0	0	No sound out to phone
		0	0	1	1	0	0	Votrax Module
		0	1	0	2	0	0	Auxiliary Device (Digital to Analog Converter, DAC)
		0	1	1	3	0	0	Tape Recorder Play
		1	0	0	4	0	0	Modem
		1	0	1	5	0	0	Tone Dialer Generator
		1	1	0	6	0	0	Pulse Dialer
		1	1	1	7	0	0	No sound out to phone

Note: Select #7 when you are not outputting to the phone.



Table T-2B

Address

Decimal	Hex	Bit	Use
63488	F800		
		PA3	Spare
		PA4	Spare
		PA5	Spare
		PA6	Failsafe Status 0=Timeout 1=No Timeout
		PA7	Ring sense (1=Ring, 0=No Ring) (input to CPU)
		CA1	Spare
		CA2	Spare
63489	F801		Control register for 63488 decimal.

\*\*\*\*\*

Table T-3

Address

Decimal	Hex	Bit	Use
63490	F802		
		PB0	Connect control (1=true)(output from CPU)
		PB1	<u>Connect control</u> (0=true)(output from CPU)
		PB2	Tape input control (0=tape on)(output from CPU)
		PB3	Tape output control (0=tape on) (output from CPU)
		PB4	Spare
		PB5	Failsafe strobe (1 to 0) (output from CPU)
		PB6	Telephone line hard limited (0=true) (input to CPU)
		PB7	Telephone line threshold (0=true) (input to CPU)
		CB1	Spare
		CB2	Spare
63491	F803		Control register for 63490

Table T-4A

Address		Bits (See Note 1)				Hex Value				Tone Input From Phone (input to CPU)
Decimal	Hex	1PA3	1PA2	1PA1	1PA0	1PA3	1PA2	1PA1	1PA0	
63492	F804	0	0	0	0	0	0	0	0	D
		0	0	0	1				1	1
		0	0	1	0				2	2
		0	0	1	1				3	3
		0	1	0	0				4	4
		0	1	0	1				5	5
		0	1	1	0				6	6
		0	1	1	1				7	7
		1	0	0	0				8	8
		1	0	0	1				9	9
		1	0	1	0				A	0
		1	0	1	1				B	*
		1	1	0	0				C	#
		1	1	0	1				D	A
		1	1	1	0				E	B
		1	1	1	1				F	C

1CA1 - 1 = valid tone decode

0 = no tone

- Note:
- 1) PA0 - PA3 must be read within 25 ms of 1CA1 going high (i.e., IRQA1=1).
  - 2) 1CA1 must be continuously read at least every 33 ms (IRQA1) when listening for tones.
  - 3) Software should program 1CA1 control for low to high transitions of 1CA1.
  - 4) A "1" preceding a PIA signal (e.g. 1CA1) indicates that the PIA signal is located on the PIA that resides from hex address F804 thru F807 (decimal address 63492 thru 63495).

Table T-4B

Address

Decimal	Hex	Bits	Function
63492	F804	1PA4	- Modem mode sense (1=originate) (input to CPU)
		1PA5	- Modem self-test ( $\overline{ST}$ ) (CPU output)
		1PA6	- Modem originate mode ( $\overline{SH}$ ) (CPU output)
		1PA7	- Modem answer mode ( $\overline{RI}$ ) (CPU output)
		1CA1	- DTMF decodes strobe (input to CPU) This line is polled to determine whether data on 1PA0 to 1PA3 is valid.
		1CA2	- Spare
63493	F805	Control register for 63492	

\*\*\*\*\*

Table T-5

Address

Decimal	Hex	Bits	Function
63494	F806	1PB0	- 1PB7 Dialer data
		1CB1	- Spare
		1CB2	- Spare
63495	F807	Control register for 63494	

\*\*\*\*\*

Table T-6

Address

Decimal	Hex	Bits	Function
63496	F808	Control register for 63499 (ACIA)	
63497	F809	ACIA	

APPENDIX B  
SOFTWARE TABLES

TABLE S-1

1PA5	1PA6	1PA7	1PA4
$\overline{ST}$	$\overline{SH}$	$\overline{RI}$	MODE
H	L	H	H
H	H	L	L
L	L	H	L
L	H	L	H

ORIGINATE MODE = H(1)

ANSWER MODE = L(0)

TABLE S-2

Select Output to the Phone Line

Select NO output to the phone	POKE 63488,0
Select VOTRAX output to the phone	POKE 63488,1
Select AUXILIARY DEVICE output to the phone	POKE 63488,2
Select TAPE PLAYER output to the phone	POKE 63488,3
Select MODEM output to the phone	POKE 63488,4
Select TONE DIALER GENERATOR output to the phone	POKE 63488,5
Select PULSE DIALER output to the phone	POKE 63488,6
Select NO output to the phone	POKE 63488,7

Note: Select #7 when you don't want to output anything to the phone or when the phone line is disconnected.

Read FAILSAFE Status

X=(PEEK(63488) AND 64)

If X=0 then the FAILSAFE has timed out.

If X≠0 then the FAILSAFE has not timed out.

Check for Phone Ringing

X=(PEEK(63488) AND 128)

IF X=0 then the phone is not ringing.

If X≠0 then the phone is ringing.

TABLE S-3

	(PB0, PB1) Phone Connected	(PB1) Tape Player On	(PB3) Tape Recorder On	(PB5) Failsafe Strobe Low (1)
POKE 63490,0*	N*	Y	Y	Y
POKE 63490,1	Y	Y	Y	Y
POKE 63490,2	N	Y	Y	Y
POKE 63490,3*	N*	Y	Y	Y
POKE 63490,4	N*	N	Y	Y
POKE 63490,5	Y	N	Y	Y
POKE 63490,6	N	N	Y	Y
POKE 63490,7*	N*	N	Y	Y
POKE 63490,8*	N*	Y	N	Y
POKE 63490,9	Y	Y	N	Y
POKE 63490,10	N	Y	N	Y
POKE 63490,11*	N*	Y	N	Y
POKE 63490,12*	N*	N	N	Y
POKE 63490,13	Y	N	N	Y
POKE 63490,14	N	N	N	Y
POKE 63490,15*	N*	N	N	Y
POKE 63490,32*	N*	Y	Y	N
POKE 63490,33	Y	Y	Y	N
POKE 63490,34	N	Y	Y	N
POKE 63490,35*	N*	Y	Y	N
POKE 63490,36*	N*	N	Y	N
POKE 63490,37	Y	N	Y	N
POKE 63490,38	N	N	Y	N
POKE 63490,39*	N*	N	Y	N
POKE 63490,40*	N*	Y	N	N
POKE 63490,41	Y	Y	N	N
POKE 63490,42	N	Y	N	N
POKE 63490,43*	N*	Y	N	N
POKE 63490,44*	N*	N	N	N
POKE 63490,45	Y	N	N	N
POKE 63490,46	N	N	N	N
POKE 63490,47*	N*	N	N	N

\*This command is not recommended because both phone "Connect Controls" are not off. It does not use the redundant control of the phone "Connect Controls".

(1) The purpose and use of the FAILSAFE STROBE (PB5) is described in the "UTI BOARD FAILSAFE ENABLE" section.

### Phone Connection Control

Connect the UTI to the phone line:

```
X=(PEEK(63490) OR 1): POKE 63490,(X AND 253)
```

Disconnect the UTI from the phone line:

```
X=(PEEK(63490) OR 2): POKE 63490,(X AND 254)
```

### Tape Recorder On/Off Control

Turn the tape recorder on:

```
X=PEEK(63490) : POKE 63490,(X AND 251)
```

Turn the tape recorder off:

```
X=PEEK(63490) : POKE 63490,(X OR 4)
```

### Tape Player On/Off Control

Turn the tape player on:

```
X=PEEK(63490) : POKE 63490,(X AND 247)
```

Turn the tape player off:

```
X=PEEK(63490) : POKE 63490, (X OR 8)
```

### Strobe the UTI FAILSAFE

```
X=PEEK(63490) : POKE 63490,(X OR 32) : POKE 63490,(X AND 223)
```

TABLE S-4

Read Telephone Line HARD Limited

If X8=0, then some sound is on the phone line and the signal is in the positive voltage state. X8=(PEEK(63490) AND 64)

If X8≠0, then either there is no sound on the phone line or the signal is in the negative voltage state.

Read Phone Line Threshold

If X9=0, then some sound is on the phone line. X9=(PEEK(63490) AND 124)

If X9≠0, then no sound is on the phone line.

Note: Any frequency greater than 50 Hz will appear as a continuous LOW (0) on the line threshold (X9). This means, for example, that a "DIAL TONE" will appear as a continuous LOW (0) on the line threshold (X9=0 above) as long as the dial tone is present. The volume of the signal may be so low in amplitude that a faint ring signal heard when you call someone will not trigger this threshold detect line.

TABLE S-5

Read Touch-Tone (DTMF) From Phone Line X=(PEEK(63492) AND 15)

<u>Value of X</u>	<u>Button Depressed</u>
0	D
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	0
11	*
12	#
13	A
14	B
15	C

Note: Buttons A,B,C and D are available on SPECIAL Touch-Tone phones.



TABLE S-6A

Read "MODEM MODE SENSE"

If  $X=0$ , then the modem in the computer is in the "ANSWER" mode.

$X=(PEEK(63492) AND 16)$

If  $X \neq 0$ , then the modem in the computer is in the "ORIGINATE" mode.

Tone Ready Status

If  $X=0$ , then no new Touch-Tone has been received yet.

$X=(PEEK(63493) AND 128)$

If  $X \neq 0$ , then a new Touch-Tone has been received. Now the new tone must be read within 25 milliseconds to guarantee that this tone will be valid. This 25 millisecond specification must be strictly adhered to only if another machine is sending codes as fast as allowed.

Normally, if the above command results in  $X=0$ , then just follow that BASIC instruction by the instruction  $X8=(PEEK(63492) AND 15)$ . Now  $X8$  will have a number corresponding to the Touch-Tone button depressed. See Table T-4A for the Tone Code.

TABLE S-6B

Output to Tone or Pulse Dialer

POKE 63494,X

See Table T-1 for values of  $X$  corresponding to a phone digit or button.

Note: If neither the TONE GENERATOR nor the PULSE DIALER is being used, always make sure that the last command given to address 63494 is:

POKE 63494,255

TABLE S-7

Check Modem Receive Data Status

If  $X=0$ , then the modem has not received new data. (The ACIA's Receive Data Register Full (RDRF) is FALSE.)  $X=(PEEK(63496) \text{ AND } 1)$

If  $X \neq 0$ , then the modem has received new data. (The ACIA's Receive Data Register Full (RDRF) is TRUE.) This new data should be read immediately by using the instruction:  $X8=(PEEK(63497))$

Check Modem Transmit Data Status

If  $X=0$ , then the modem has not transmitted the last data character that the computer gave it. (The ACIA's Transmit Data Register Empty (TDRE) is FALSE.)  $X=(PEEK(63496) \text{ AND } 2)$

If  $X \neq 0$ , then the modem is ready to send more data. The next character may be sent by the instruction: POKE 63496, X8 (where X8 is the next character to be sent). (The ACIA's Transmit Data Register Empty (TDRE) is TRUE.)

Check for Modem Detected

If  $X=0$ , then the modem has detected a remote modem. (The ACIA's CLEAR-TO-SEND (CTS) is TRUE.)  $X=(PEEK(63496) \text{ AND } 8)$

If  $X \neq 0$ , then the modem has not detected a remote modem. (The ACIA's CLEAR-TO-SEND (CTS) is FALSE.)

TABLE S-8

<u>Modem Data Format</u>			
<u>Command</u>	<u># Data Bits</u>	<u>Parity</u>	<u># Stop Bits</u>
POKE 63496,1	7	Even	2
POKE 63496,5	7	Odd	2
POKE 63496,9	7	Even	1
POKE 63496,13	7	Odd	1
POKE 63496,17	8	No	2
POKE 63496,21	8	No	1
POKE 63496,25	8	Even	1
POKE 63496,28	8	Odd	1
POKE 63496,3		MASTER RESET	

Output Data to Modem

Output one byte (X) to the Modem:

POKE 63497,X

Input Data From Modem

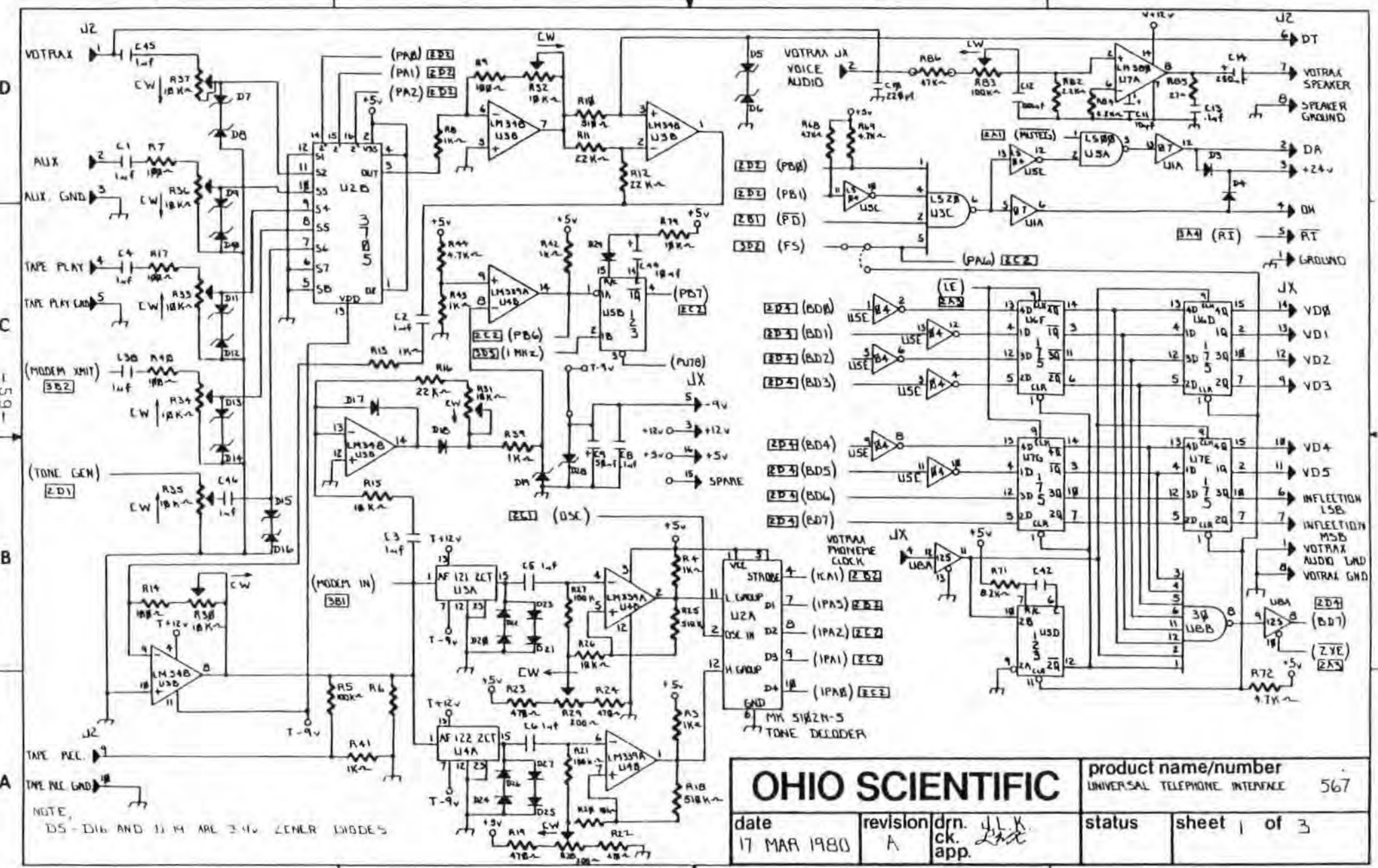
Input one byte (X) from the Modem:

X=PEEK(63497)

APPENDIX C  
SPECIFICATIONS

<u>Power</u>	<u>Typical Current for</u>	
<u>Voltage</u>	<u>CA-15 (Without VOTRAX)</u>	<u>CA-15V (With VOTRAX)</u>
+5 VDC	0.9 amps	1.0 amp
+12 VDC	65 ma	130 ma
-9 VDC	55 ma	100 ma

APPENDIX D  
SCHEMATIC DIAGRAMS

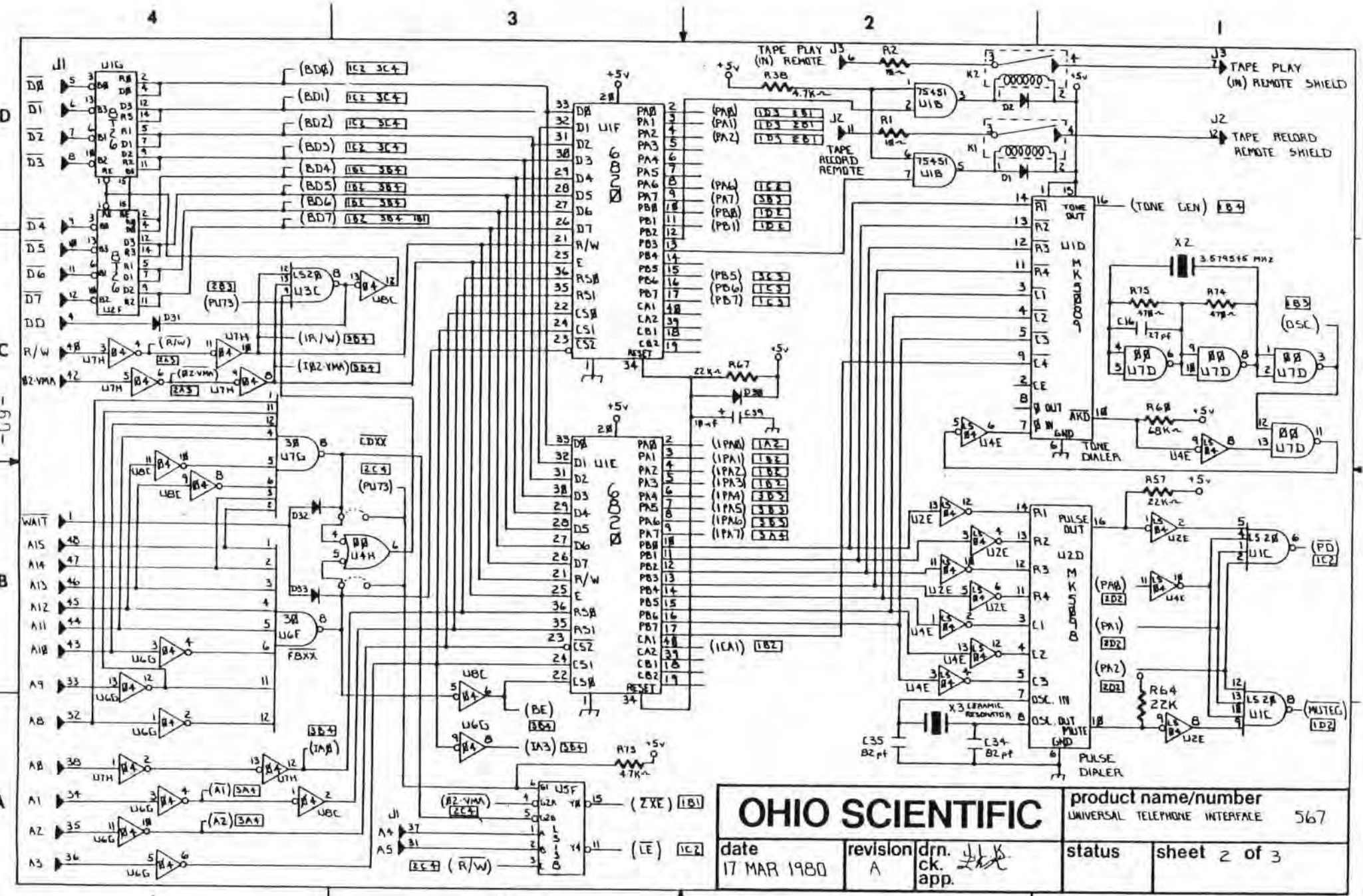


# OHIO SCIENTIFIC

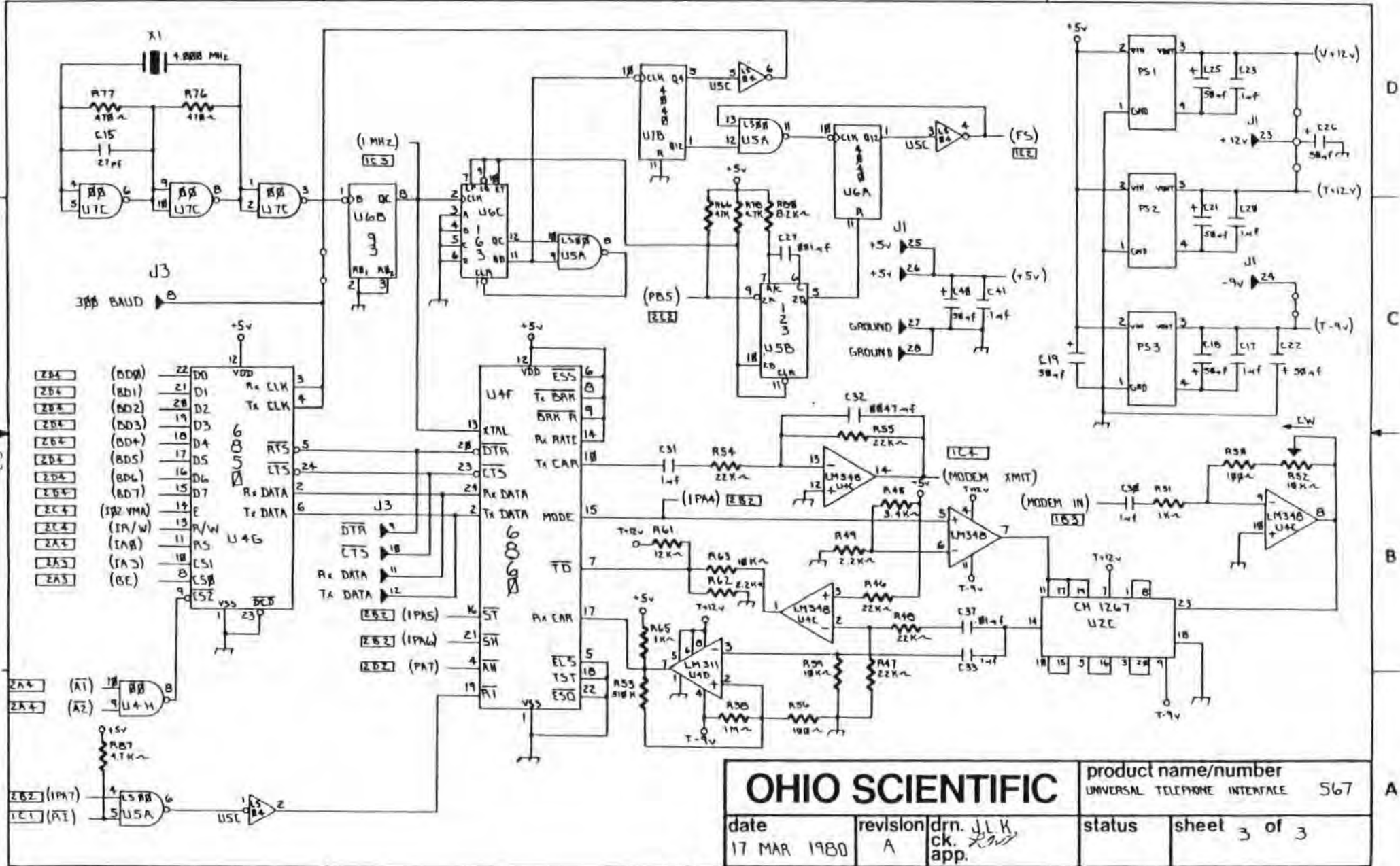
product name/number  
UNIVERSAL TELEPHONE INTERFACE 567

date 17 MAR 1980	revision A	drn. J.L.K. ck. app. [signature]	status	sheet 1 of 3
---------------------	---------------	---	--------	--------------

NOTE,  
D5-D16 AND J1-H ARE 3-40 ZENER CODES



<b>OHIO SCIENTIFIC</b>			product name/number UNIVERSAL TELEPHONE INTERFACE 567	
date 17 MAR 1980	revision A	drn. c.k. app. <i>[Signature]</i>	status	sheet 2 of 3



**OHIO SCIENTIFIC**

product name/number  
UNIVERSAL TELEPHONE INTERFACE 567

date  
17 MAR 1980

revision  
A

drn. J.L.K.  
ck. *[Signature]*  
app.

status

sheet 3 of 3

D  
C  
B  
A

4

3

2

1





DELETED COMPONENTS LIST

The following components are not populated on the 567 Board  
but are listed on the schematic diagrams:

C9	D13-D16
C13	D19
C18	D28
C19	P51
C21	P52
C25	P53
C42	R70
C43	R81
	R85

VOTRAX Preliminary Programming Manual

Part Number 942

## VS-6 Programming Rules

The sixty-three (63) phoneme commands in concert with the four (4) choices of inflection provide the user with 252 unique synthesizer inputs which may program the synthesizer in any desired sequential order. There are, for instance,  $(252)^4$  or over 4 billion possible utterances containing four (4) phonemes! The phonetic rules are based on the actual acoustic content of human speech. These rules do not necessarily correlate with dictionary phonetics nor do they necessarily correlate with phonetics as taught in the public schools. Thus, the following rules may appear alien to the uninitiated, but once the user becomes acquainted with them, determining the optimum programming for any word or phrase is quite easy and considerably easier than in previous synthesizer models.

The VS-6 has been optimized for the Mid-Western or standard American English dialect which is the dialect used almost exclusively by the nationwide media. It is also the native dialect of the Central United States, West Coast and Mid-Eastern states excluding the East Coast. This dialect is spoken by the largest majority of U.S. citizens and to a large extent by central and western Canadians. It is important that the following programming rules be adhered to closely. The VS-6 electronics as well as the phonetic alphabet itself have been designed around these phonetic rules. With very few exceptions, the VS-6 performance as judged by the majority of U.S. and Canadian residents will be degraded if these rules are not followed.

Such caution is necessary because ultimately the synthesizer is

subjectively judged by a human listener during manual optimization of a word or phrase. The user may ask the question, "What if breaking the rules produces an apparent improvement in naturalness?". This will occasionally happen, but the user is strongly cautioned in this area. The problem is phonetic substitutions. What is acceptable to one listener, is not necessarily acceptable to the next, particularly one who speaks another dialect. A dialect trains our ears as well as our mouths.

For instance, in the Great Lakes area, many people consider OTTO an acceptable pronunciation of the word AUTO. People from other areas will perceive this alternative pronunciation as an error. Another example is the New York City area's pronunciation of the NG phoneme. In this substitution, the word THING becomes THING-G with a hard G at the end of the word. A New Yorker could perceive this alternate substitution as perfectly acceptable, but the rest of the North American Continent would not.

A Southerner would accept the prefix NON - as an acceptable pronunciation of the number NINE, but people in other areas would not.

Thus, it is recommended that an individual given the responsibility of determining the optimum programming for many words have a central dialect, particularly if the synthesizer output is to be heard on a nation-wide hook-up such as a computer audio response system.

The phonetic structure of any language is much more complicated than most of us realize. There are many pronunciation habits which we acquire as we grow up of which we are not aware, but we use them anyway. Proper

synthesized speech must take into account all subconscious or automatic articulatory habits as well as the rules of which we are consciously aware.

Perhaps the most common of these unlearned habits of English is gliding of vowels, particularly the long vowels. This gliding is called diphthongization. It means that a vowel ends as a different sound than it was when it started. The "long" vowels A, E, I, O, U have glides which are sounded independently of preceding and following phonemes, even if these sounds are spoken alone. These vowels are programmed as follows on the VS-6:

<u>Vowel</u>	<u>Typical Usage</u>	<u>VS-6 Equivalent</u>
A	name	A, AY
E	tree	E or I3 El
I	high	AH, El
O	note	UH1, Ol
U	two	IU, U

The "short" vowels are as follows:

<u>Vowel</u>	<u>Typical Usage</u>
AW	awful
EH	ten
ER	her (R, usually a consonant, can be a vowel in some cases)
UH	but
AE	hat
Y	Mary
I	kit
AH	hot
OO	book

These vowels also glide from beginning to end, but their glides are normally determined by the preceding and following phonemes. These glides are generated automatically by the VS-6. However, the programmer may find occasional use of coupling vowels I3, EH3, or UH3 in conjunction with a short vowel is an aid to naturalness or intelligibility.

The numbered vowels such as UH1, UH2, and UH3 are of increasingly shorter duration. The higher the number, the shorter the duration. As a syllable is stressed less and less, a higher and higher number must be selected. The numbered vowels are also used in the "long" vowel phoneme pairs. For instance, A, AY is the vowel in MAIN; A1, AY or A2, AY would be the vowel in the same syllable in the word MAINTAIN. This rule applies to all numbered vowels except Y1.

Y1 is Y used as a consonant such as in the word YES. It is also found at the beginning of a syllable such as the second syllable in the word UNUSUAL. Y1 is a different kind of sound than Y as is evidenced when these phonemes are sounded continuously. However, all other numbered vowels are exactly like their unnumbered, fully stressed counterparts if sounded continuously. As the number on a vowel increases, its time interval shortens, and its dynamic interaction between the preceding and following phoneme commands changes. Thus, two short duration phonemes will not have quite the same sound as one longer one of equal time duration. These differences are not evident if the phoneme is sounded continuously.

In addition to the syllabic stress factor affecting vowel length, the

following phoneme is also a factor. If it is a sustained fricative such as S, the preceding vowel may be shortened. If it is a fricative stop such as T, the vowel may be shortened still further. Of course, subjective judgment should still be the final determiner of vowel stress.

There are three classes of diphthongs, or distinctively two - part vowels:

<u>Diphthong</u>	<u>Typical Usage</u>	<u>VS-6 Equivalents</u>
I	high	AH, EI
I	height	AH2, EI
OW, OU	cow	AH1, OI *
		AH1, UI
OI, OY	noise	OI, EH3, EI *
		OI, I3, EI
		O, EI
		* preferred

The consonants are as follows:

<u>Consonant</u>	<u>Typical Usage</u>	<u>Notes</u>
TH	three	
THV	then (the voiced TH)	
W	won	
R	area	
T	tea	
P	pot	
YI	yes	
S	sec	
D	day	



<u>Consonant</u>	<u>Typical Usage</u>	<u>Notes</u>
F	fire	
G	get (not the G in George)	
H	hay, ahead	
J	jet, George (J is D, J or DT, J)	
K	key, sick, car	
	queer (Q is KW)	
L	lie, well	
NG	bring	
Z	zero	
SH	shy	
CH	match, chair (CH is T, CH or DT, CH)	
V	seven	
B	bob	
N	nine	
M	my	
ZH	azure (the Z)	
	measure (the S)	
DT	butter (DT is merely a tongue flap in this word)	

There are two affricates in English. An affricate is a two-consonant combination with a fricative stop.

The voiced affricate J is programmed D, J or DT, J. The unvoiced affricate CH is programmed T, CH or DT, CH.

All phonemes in English fall into seven (7) categories as follows:

1. voiced

A. All vowels and diphthongs of vowels.

A, E, I, O, U, AW, EH, ER, UH, AE, Y, I, AH, OO, and  
OI, OY, OU, OW diphthongs.

B. All liquid consonants

R, L, W, Yl

2. voiced stops B, D, G

3. nasal closures M, N, NG

4. unvoiced

A. fricatives S, SH, F, TH

B. Aspirant H

5. voiced fricatives Z, ZH, V, THV

6. fricative stops T, P, K

7. affricates

A. voiced J

B. unvoiced CH

There are several particular cases of interest which should be mentioned. The letter combination NG doesn't necessarily call for the NG phoneme. This is true if the N forms the end of one syllable and the G forms the beginning of the following syllable. In the word ENGAGE, the N and G are sounded separately. Thus, in this case, NG is programmed N, G.

In the word FINGER, the NG can be programmed as N,G, or as NG,G.

In the word THANK, the N can be programmed as N or NG. The A in thank is half way between A and AE. It is best programmed as AEI, I3. This is a classical example of an aliphone.

An aliphone is a variation of a basic phoneme. All phonemes in speech context are modified by their phonetic "environment". That is, whatever is going on before and after the phoneme affects its characteristics - its duration, amplitude, frequency components, etc. This effect is what we call the dynamic continuum of which speech is made. A phoneme is merely an operator on the human acoustic output which, in turn, gets operated upon.

To produce intelligible synthetic speech, the proper aliphone must be generated. This is done automatically in the VS-6. Another example of aliphones (perhaps the most extreme) is the numerous forms of K: the K in KEY, the K in LOOK, and the K sound in Q which is programmed K,W such as in the word QUIT. At times, the VS-6 needs some help from the programmer to produce the proper aliphone.

A particular class of phonemes, the liquids - R, L, W, and Yl, largely depend on the transitions into and out of these sounds for their recognition. These transitions, important as they are, depend a great deal on dialect. It is therefore necessary for the programmer to insert transitional phonemes around these

sounds in some cases to maximize naturalness and intelligibility.

Transitions around W and Yl are dependent enough on surrounding phonemes for the VS-6 to generate them automatically.

The pre-vocalic R (R at the beginning of an utterance) normally doesn't require a transitional phoneme after it. However, the post-vocalic R (R at the end of an utterance) may require it, depending on the preceding vowel such as in the word AIR. This is programmed A,I3,R. Note similarity to spelling. EH3 in place of I3 is also acceptable here.

The pre-vocalic L is enhanced by following it with UH3, EH3, or I3, depending on the following vowel. For instance, the word LAY is programmed L, EH3, A1, AY. The post-vocalic L is likewise enhanced by preceding it with UH3, EH3, or I3 such as in the word AIL. This is programmed as A, I3, L. Again, note similarity to spelling. EH3 instead of I3 is also acceptable here.

The above suggests a possible need for more phonemes. Surely increasing the phoneme vocabulary size would enhance naturalness, but at the expense of increased hardware costs and greatly increased programming effort.

The phonetic keyboard layout was chosen to be as consistent as possible with the ubiquitous typewriter keyboard. At the same time, a phonetic labeling system had to be chosen which used alphanumerics so that regular typewriters and computer interface equipment could be used in describing programming. For those who are interested in the equivalency with the I. P. A.

or international phonetic alphabet, consult the I. P. A. phoneme equivalents table.

Inflection is extremely important for VS-6 speech to avoid a machine-like quality. Although ideal inflection is a subjective matter, a few simple rules will make this job easy. First determine the relative stress on various syllables in a word and on various words in a statement. Stronger stresses get longer phonemes and higher inflection pitches. Unlike previous synthesizer models, the VS-6 inflection is completely independent from phoneme timing, but it still affects loudness as well as pitch. The VS-6 is designed so that the nominal pitch is half way between inflection levels 2 and 3. The nominal pitch has a frequency of 125 to 130 Hertz which is equivalent to the average pitch of an adult Caucasian male.

Thus, the VS-6 is designed for most frequent use of inflections 2 and 3. The user may choose to depart from this rule at his discretion. However, unnatural pitch contours may be produced. Higher inflections and longer vowels should be used on stressed syllables and words; lower inflections and shorter vowels on unstressed syllables.

The pitch from any inflection command appears in the VS-6 output  $\frac{1}{2}$  a phoneme to a full phoneme later. Thus, if it is desired to stress a given phoneme, that phoneme should be a long one (if a vowel) and the highest inflection might need to be put on the preceding phoneme.

The VS-6 has the capability of having its inflection under computer

or software control. This allows the user to assemble a sentence out of a pre-stored vocabulary of words and have these words inflected according to sentence grammar. This is a complex technique and it is often sufficient to store a vocabulary including nominal inflection commands for these words. When these words are assembled into sentences, the resulting speech may not be inflected perfectly but, more importantly, it will be intelligible and the pitch will vary smoothly from word to word. Thus, there will be no pitch discontinuities at word boundaries, an inevitable shortcoming of systems utilizing pre-stored human speech.

Whether sentence programming is determined by software or manually by a programmer, the inflection level during any pause commands as well as the binary zero position should be inflection 1. Use of successive pause commands to produce long pauses may result in a slight background noise during the protracted pause interval - this effect is normal.

To produce prepausal lengthening of the last phoneme in an utterance (lengthening of the time duration of the last phoneme), the PA1 or pause one must be placed at the end of the utterance. The PAØ or short pause may be used instead but the last phoneme in the utterance will be a little shorter. The PAØ is normally used between words in a sentence where very short pauses are desired. The PA1 is normally used at the end of an entire utterance, even if there is only a single word in the utterance. A word must not be followed by a binary zero (Null) command. If nulls are part of a phonetic data stream driving the synthesizer such as in the RS232 interface, a PAØ or PA1 must precede a null to prevent chopping off part of the last phoneme before the null.

The synthesizer is designed to generate continuous speech which means that the words in a sentence have little or no gap in between. This is consistent with conversational English habits. In conversation it is usual for words within a phrase or clause to be strung together while phrases and clauses are separated by a gap. There is no doubt that intelligibility in either human or synthesized speech is enhanced by short gaps between words. This is accomplished merely by placing a PAØ after each word. The resulting speech is acceptable but somewhat choppy, though highly intelligible. Use of such gaps will slow the output rate of the synthesizer accordingly. Of course, the speech rate may be increased to compensate for this but the resulting speech segments will be articulated faster than normal.

In continuous speech the synthesizer's built-in speech rules correspond to those in human speech. For instance, if two stop phonemes are adjacent, only the second phoneme is sounded. In the word FACT the K sound of the letter C and T are both fricative stops and thus only the second one in the pair is sounded. In this case the only acoustic output attributable solely to these phonemes is the voiceless release of the T. The same rule applies if two adjacent stop phonemes are each in a different word. Examples of this are TACK-BOARD and CREDIT-BALANCE. If a pause command is programmed in between the two words in each pair, then both stop phonemes are sounded - the K and B in the first example and the T and B in the second set. Note that in these examples, the first phoneme of the pair in question is voiceless while the second phoneme is voiced. However, this is irrelevant - the rule still applies. It is optional whether or not to sound both stop phonemes. It is

certainly natural in human speech not to sound the first stop phoneme. Although it is more natural, it is less intelligible whether the speech is human or synthetic. This example is a classic case of naturalness being opposed to intelligibility. Thus, the option is left to the user.

We are thus given several choices of programming at word boundaries:

1. PAI for separation of phrases and clauses and sentences.
2. PAØ for optional separation of words.
3. PAØ for enhancement of pronunciation of closure phoneme pairs.
4. PAØ or PAI followed by a null as a word boundary indicator.



## VOTRAX VS-K PHONETIC CODES

Bulletin #  
MTB-8MΛ01-MB

<u>PHONEME</u>	<u>HEX</u>	<u>OCTAL</u>	<u>PHONEME</u>	<u>HEX</u>	<u>OCTAL</u>
PAØ	Ø3	ØØ3	G	1C	Ø34
PA1	3E	Ø76	H	1B	Ø33
A	2Ø	Ø4Ø	I	27	Ø47
A1	Ø6	ØØ6	II	ØB	Ø13
A2	Ø5	ØØ5	I2	ØA	Ø12
AE	2E	Ø56	I3	Ø9	Ø11
AE1	2F	Ø57	IU	36	Ø66
AH	24	Ø44	J	1A	Ø32
AH1	15	Ø25	K	19	Ø31
AH2	Ø8	Ø1Ø	L	18	Ø3Ø
AW	3D	Ø75	M	ØC	Ø14
AW1	13	Ø23	N	ØD	Ø15
AW2	3Ø	Ø6Ø	NG	14	Ø24
AY	21	Ø41	O	26	Ø46
B	ØE	Ø16	O1	35	Ø65
CH	1Ø	Ø2Ø	O2	34	Ø64
D	1E	Ø36	OO	17	Ø27
DT	Ø4	ØØ4	OO1	16	Ø26
E	2C	Ø54	P	25	Ø45
E1	3C	Ø74	R	2B	Ø53
EH	3B	Ø73	S	1F	Ø37
EH1	Ø2	ØØ2	SH	11	Ø21
EH2	Ø1	ØØ1	T	2A	Ø52
EH3	ØØ	ØØØ	TH	39	Ø71
ER	3A	Ø72	THV	38	Ø7Ø
F	1D	Ø35	U	28	Ø5Ø

<u>PHONEME</u>	<u>HEX</u>	<u>OCTAL</u>	<u>PHONEME</u>	<u>HEX</u>	<u>OCTAL</u>
U1	37	067	W	2D	055
UH	33	063	Y	29	051
UH1	32	062	Y1	22	042
UH2	31	061	Z	12	022
UH3	23	043	ZH	07	007
V	0F	017	* NULL	FF	377

\* Note: Null should be continuously output to VS-K when power is on and speech is not being produced to eliminate noise.

### VS-K PROGRAMMING EXAMPLES

Zero: PA1, Z, AY, I3, R, UH3, 02, U1, PA1

One: PA1, W, UH3, UH2, UH2, N, N, PA1

Two: PA1, T, IU, IU, U1, U1, PA1

Three: PA1, TH, TH, R, E, Y, PA1

Four: PA1, F, 01, 01, R, PA1

Five: PA1, F, AH2, AH1, I3, Y, V, V, PA1

Six: PA1, S, I, I3, K, PA0, S, PA1

Seven: PA1, S, EH3, EH2, V, EH2, N, N, PA1

Eight: PA1, A2, A2, AY, Y, T, PA0 \*\*

Nine: PA1, N, AH1, EH2, Y, N, PA1

Cancel: PA1, K, AE1, EH3, N, S, UH3, L, PA1

Add: PA1, AE1, AE, EH3, D, PA0 \*\*

Subtract: PA1, S, UH2, B, PA0, T, R, AE1, AE1, K, PA0, T, PA0 \*\*

Multiply: PA1, M, UH2, L, T, UH3, P, L, AH1, Y, PA1

Divide: PA1, D, I2, V, AH2, AH1, EH3, I3, AY, D, PA0 \*\*

Equals: PA1, E1, AY, K, W, 001, L, Z, PA1

\* Note: All stop plosives (P, T, K, DT, B, D, G) preceding a pause (PA1) must be followed by a PA0.

VOTRAX Voice Synthesizer (VS-6) Word Vocabulary

Index

1. Pages A thru Z alphabetical listing of VOTRAX phonetic programming.
2. Alphabetical listing of verbs, adverbs, and adjectives found with some frequency in spoken English.

Able	2/PA1, 2/A2, 1/A2, 1/Y, 1/B, 1/L
Account	1/PA1, 1/UH2, 2/K, 1/AH1, 1/U1, 1/N, 1/T
Active	2/PA1, 2/AE1, 2/EH3, 1/K, 1/PA Ø, 1/T, 1/I2, 1/V
Add	2/PA1, 1/AE, 1/EH3, 1/D
Address	1/PA1, 1/UH2, 2/D, 2/R, 1/EH1, 1/S, 1/S
Adjust	1/PA1, 1/UH2, 1/D, 2/J, 2/UH3, 1/UH1, 1/S, 1/T
Advance	1/PA1, 1/EH2, 1/D, 2/V, 1/AE1, 1/EH3, 1/N, 1/S
After	2/PA1, 1/AE, 1/F, 1/T, 1/ER
Afternoon	1/PA1, 1/AE1, 1/F, 2/T, 1/R, 2/N, 1/IU, 1/U, 1/N
Again	1/PA1, 1/UH1, 2/G, 1/EH, 1/N
Air	2/PA1, 1/A2, 1/EH2, 1/ER
All	2/PA1, 1/AW, 1/UH3, 1/L
Am	2/PA1, 1/AE1, 1/EH3, 1/UH3, 1/M
A.M.	1/PA1, 1/A2, 2/AY, 2/Y, 2/PAØ, 2/PAØ, 2/EH3, 1/EH1, 1/UH3, 1/M
America	1/PA1, 1/UH2, 3/M, 2/EH2, 1/EH2, 1/R, 1/EH3, 1/K, 1/UH1
Amount	1/PA1, 1/UH2, 2/M, 2/UH3, 1/AH1, 1/U1, 1/N, 1/T
An	2/PA1, 1/AE1, 1/EH3, 1/I3, 1/N
And	2/PA1, 1/AE1, 1/EH3, 1/I3, 1/N, 1/D
Answer	2/PA1, 2/AE1, 1/I3, 2/N, 1/S, 1/R, 1/R
Application	1/PA1, 1/AE1, 2/P, 1/L, 1/UH3, 3/K, 1/A1, 1/Y1, 1/SH, 1/UH3, 1/N

Approve	1/PA1, 1/UH2, 3/P, 2/R, 1/U, 1/V
April	1/PA1, 1/A1, 2/AY, 1/P, 1/R, 1/UH3, 2/L
Are	2/PA1, 1/AH1, 1/UH3, 1/ER
Assign	1/PA1, 1/UH2, 3/S, 1/AH1, 1/E1, 1/N
Assist	1/PA1, 1/UH2, 3/S, 1/I, 1/S, 1/T
At	2/PA1, 1/AE1, 1/EH3, 1/T
Atlantic	1/PA1, 1/EH2, 1/T, 2/L, 1/AE1, 1/N, 1/T, 1/I3, 1/K
August	2/PA1, 1/AW, 2/G, 1/EH2, 1/S, 1/T
Available	2/PA1, 1/UH2, 2/V, 2/A1, 2/AY, 1/L, 1/UH3, 1/B, 1/UH3, 1/L
Average	2/PA1, 1/AE1, 1/I3, 1/V, 1/R, 1/I2, 1/D, 1/J
Away	1/PA1, 1/UH2, 2/W, 1/A1, 1/AY, 1/AY

Back 1/PA1, 3/B, 2/AE1, 1/EH3, 1/K  
Be 1/PA1, 2/B, 1/E1, 1/E1  
Because 1/PA1, 1/B, 1/Y1, 2/K, 1/UH1, 2/UH2, 1/Z  
Been 1/PA1, 3/B, 1/EH3, 1/EH1, 1/N  
Before 1/PA1, 1/B, 1/E1, 3/F, 1/O, 1/R  
Begin 1/PA1, 1/B, 1/E1, 3/G, 1/I3, 1/I1, 1/N  
Benefit 1/PA1, 3/B, 1/EH1, 1/N, 1/EH3, 2/F, 1/I3, 1/T  
Between 1/PA1, 1/B, 1/I2, 3/T, 1/W, 1/E1, 1/E1, 1/N  
But 1/PA1, 1/B, 1/UH1, 2/UH3, 2/T  
By 1/PA1, 2/B, 1/AH1, 1/E1

Call	2/PA1, 2/K, 1/AW, 1/L
Can	2/PA1, 2/K, 1/AE, 1/EH3, 1/EH3, 1/N
Car	2/PA1, 3/K, 1/AH, 1/R
Care	2/PA1, 2/K, 1/A2, 1/EH2, 1/R
Careful	2/PA1, 2/K, 1/A2, 2/EH2, 2/R, 1/F, 1/UH3, 1/L
Case	2/PA1, 2/K, 1/A1, 1/AY, 1/Y1, 1/S
Cash	2/PA1, 2/K, 1/AE, 1/EH3, 1/SH
Caution	2/PA1, 2/K, 2/AW, 1/SH, 1/UH3, 1/N
Cents	2/PA1, 2/S, 2/EH3, 1/EH1, 1/N, 1/S
Change	2/PA1, 2/T, 2/CH, 2/EH3, 1/A1, 1/Y, 1/N, 1/D, 1/J
Charge	2/PA1, 2/T, 2/CH, 1/AH, 1/R, 1/D, 1/J
Check	2/PA1, 2/T, 2/CH, 1/EH, 1/K
City	2/PA1, 3/S, 1/I1, 1/DT, 1/E1
Class	2/PA1, 2/K, 1/L, 1/AE, 1/UH3, 1/S
Classification	2/PA1, 2/K, 2/L, 1/AE1, 2/S, 1/I3, 2/F, 1/I3, 2/K, 2/A1, 1/Y1, 1/SH, 1/UH3, 1/N
Clear	2/PA1, 2/K, 2/L, 2/I3, 1/I1, 1/R
Clearance	2/PA1, 2/K, 2/L, 2/I3, 1/I2, 1/R, 1/EH3, 1/N, 1/S
Clock	2/PA1, 2/K, 2/L, 1/AH1, 1/UH3, 1/K
Close *	2/PA1, 2/K, 2/L, 1/UH3, 1/O1, 1/U1, 1/Z
Close *	2/PA1, 2/K, 2/L, 1/UH3, 1/O1, 1/U1, 1/S

Code	2/PA1, 2/K, 1/O2, 1/O1, 1/U1, 1/D
Collect	1/PA1, 1/K, 1/UH2, 3/L, 2/UH3, 1/EH2, 1/EH2, 1/K, 1/PAØ, 1/T
Commit	1/PA1, 1/K, 1/UH2, 3/M, 1/I1, 1/T
Complete	1/PA1, 1/K, 1/UH3, 1/M, 3/P, 1/L, 1/E, 1/T
Condition	1/PA1, 1/K, 1/UH3, 1/N, 2/D, 1/I2, 1/SH, 1/UH3, 1/N
Confirm	1/PA1, 1/K, 1/UH3, 1/N, 3/F, 1/R, 1/R, 1/M
Contact	2/PA1, 2/K, 1/AH1, 1/N, 1/T, 1/AE1, 1/EH3, 1/K, 1/PAØ, 1/T
Correct	2/PA1, 1/K, 1/R, 2/R, 1/EH1, 1/K, 1/PAØ, 1/T
Cost	2/PA1, 2/K, 1/AW, 1/S, 1/T
Customer	2/PA1, 2/K, 2/UH2, 1/S, 1/T, 1/UH3, 1/M, 1/R



Daily	2/PA1, 2/D, 2/A1, 1/Y, 1/L, 1/Y
Data	2/PA1, 3/D, 1/A1, 1/AY, 1/DT, 1/UH2
Date	2/PA1, 3/D, 1/A1, 1/AY, 1/Y1, 1/T
Day	2/PA1, 3/D, 1/EH3, 1/A1, 1/AY
December	2/PA1, 1/D, 2/Y1, 3/S, 2/EH2, 2/EH2, 1/M, 1/B, 1/R
Deduct	2/PA1, 1/D, 1/I3, 3/D, 1/UH2, 1/UH2, 1/K, 1/PAØ, 1/T
Defer	2/PA1, 1/D, 1/Y1, 3/F, 1/ER, 1/R
Deliver	2/PA1, 1/D, 1/UH3, 2/L, 1/I2, 1/V, 1/ER
Demand	2/PA1, 1/D, 1/EH3, 2/M, 1/AE1, 1/AE1, 1/N, 1/N, 1/D
Departure	2/PA1, 1/D, 1/I3, 2/P, 2/AH1, 1/R, 1/T, 1/CH, 1/R
Deposit	2/PA1, 1/D, 1/I3, 3/P, 1/AH1, 1/UH3, 1/Z, 1/I2, 1/T
Determine	2/PA1, 1/D, 1/I2, 3/T, 2/R, 1/R, 1/M, 1/I3, 1/N
Did	2/PA1, 2/D, 1/I, 1/D
Difference	2/PA1, 2/D, 1/I2, 1/F, 1/R, 1/EH3, 1/N, 1/S
Direct	1/PA1, 1/D, 1/ER, 2/EH1, 2/K, 2/PAØ, 1/T
Direction	2/PA1, 2/D, 2/ER, 1/EH1, 1/K, 1/SH, 1/UH3, 1/N
Distance	2/PA1, 2/D, 2/I1, 1/S, 1/T, 1/EH2, 1/N, 1/S
Divide	1/PA1, 1/D, 1/I3, 3/V, 1/AH1, 1/UH3, 1/E1, 1/D
Do	2/PA1, 3/D, 1/IU, 1/U
Dollar	2/PA1, 3/D, 2/UH3, 1/AH1, 1/L, 1/UH3, 1/R
Down	2/PA1, 2/D, 1/UH3, 1/AH1, 1/U1, 1/N

East	2/PA1, 1/E, 1/Y1, 1/S, 1/T
Economic	1/PA1, 1/EH2, 1/K, 1/UH3, 2/N, 1/AH1, 1/M, 1/I2, 1/K
Effective	1/PA1, 1/UH2, 3/F, 1/EH1, 1/K, 1/PAØ, 1/T, 1/I3, 1/V
Eight	1/PA1, 1/A1, 1/AY, 2/Y1, 2/T
Eighteen	2/PA1, 1/A1, 1/AY, 2/Y1, 2/T, 2/PAØ, 1/E1, 1/E1, 1/N
Eighty	2/PA1, 2/A1, 1/AY, 1/Y1, 1/D, 1/Y
Electric	1/PA1, 1/UH3, 3/L, 1/EH1, 1/K, 1/PAØ, 1/T, 1/R, 1/I3, 1/K
Eleven	1/PA1, 1/UH3, 3/L, 1/EH1, 1/V, 1/EH3, 1/N
Employee	1/PA1, 1/EH2, 1/M, 3/P, 2/L, 2/O1, 1/I3, 1/E1, 1/E1
Enter	2/PA1, 1/EH1, 1/N, 1/T, 1/ER
Even	2/PA1, 1/E, 1/V, 1/EH2, 1/N
Ever	2/PA1, 2/EH1, 1/V, 1/R, 1/R
Except	1/PA1, 1/EH2, 1/K, 2/S, 1/EH1, 1/P, 1/PAØ, 1/T
Exception	1/PA1, 1/EH2, 1/K, 3/S, 1/EH1, 2/P, 1/SH, 1/UH3, 1/N
Exchange	1/PA1, 1/EH3, 1/K, 1/S, 2/T, 3/CH, 1/A1, 1/AY, 1/Y1, 1/N, 1/D, 1/J
Excuse *	1/PA1, 1/EH2, 1/K, 1/S, 2/K, 1/Y1, 1/IU, 1/UI, 1/Z
Expand	1/PA1, 1/EH3, 1/K, 3/S, 2/P, 1/AE1, 1/EH3, 1/N, 1/D

Expense	1/PA1, 1/EH3, 1/K, 3/S, 2/P, 1/EH, 1/N, 1/S
Experience	1/PA1, 1/EH3, 1/K, 3/S, 3/P, 1/II, 1/R, 1/Y1, 1/UH3, 1/N, 1/S
Express	1/PA1, 1/EH2, 1/K, 3/S, 2/P, 1/R, 1/EH1, 1/S
Extend	1/PA1, 1/EH2, 1/K, 3/S, 2/T, 1/EH1, 1/I3, 1/N, 1/D
Extensive	1/PA1, 1/EH2, 1/K, 2/S, 2/T, 1/EH1, 1/N, 1/S 1/I2, 1/V

Face	2/PA1, 3/F, 1/A1, 1/AY, 1/Y1, 1/S
Far	2/PA1, 3/F, 1/AH1, 1/UH3, 1/R
Fast	2/PA1, 2/F, 1/AE, 1/EH3, 1/S, 1/T
February	2/PA1, 1/F, 2/EH1, 1/B, 2/Y1, 1/U1, 1/EH3, 1/I3, 1/R, 2/E1
Feet	2/PA1, 2/F, 1/E, 1/T
Few	2/PA1, 3/F, 1/Y, 1/IU, 1/U1
Fifteen	1/PA1, 1/F, 1/I1, 2/F, 2/T, 1/E1, 1/E1, 1/N
Fifty	2/PA1, 2/F, 2/I1, 1/F, 1/T, 1/Y
Fight	2/PA1, 2/F, 1/UH3, 1/AH2, 1/E1, 1/T
File	2/PA1, 3/F, 1/AH1, 1/E1, 1/UH3, 1/L
Fill	2/PA1, 3/F, 1/I1, 1/UH3, 1/L
Find	2/PA1, 3/F, 1/AH1, 1/I3, 1/E1, 1/N, 1/D
Five	2/PA1, 3/F, 1/UH3, 1/AH1, 1/E1, 1/V
For	2/PA1, 2/F, 1/O, 1/R
Force	2/PA1, 2/F, 1/O, 1/R, 1/S
Form	2/PA1, 2/F, 1/O, 1/R, 1/M
Forty	2/PA1, 2/F, 1/O1, 1/R, 1/DT, 1/Y
Forward	2/PA1, 3/F, 1/O1, 1/R, 1/W, 1/ER, 1/D
Fourteen	1/PA1, 1/F, 1/O, 2/R, 2/T, 2/PAØ, 1/E1, 1/E1, 1/N
Fourth	2/PA1, 3/F, 1/O, 1/R, 1/TH
Friday	2/PA1, 1/F, 2/R, 1/AH1, 1/E1, 1/D, 1/A1, 1/AY
From	2/PA1, 3/F, 1/R, 1/UH1, 1/M

Front            2/PA1, 3/F, 1/R, 1/UH1, 1/N, 1/T  
Future           2/PA1, 3/F, 2/Y1, 1/U1, 2/T, 1/CH, 1/ER

Gasoline	2/PA1, 1/G, 1/AE1, 1/S, 2/UH3, 3/L, 1/E, 1/N
Glide	2/PA1, 2/G, 1/L, 1/AH1, 1/E1, 1/D
Government	2/PA1, 2/G, 2/UH1, 1/V, 1/R, 1/M, 1/EH3, 1/N, 1/T
Grade	2/PA1, 3/G, 1/R, 1/A1, 1/AY, 1/Y1, 1/D
Great	2/PA1, 2/G, 1/R, 1/A1, 1/E1, 1/T
Ground	2/PA1, 3/G, 1/R, 1/AH1, 1/U1, 1/N, 1/D
Guarantee	1/PA1, 1/G, 1/EH3, 1/R, 1/UH3, 2/N, 3/T, 1/E, 1/E1
Guard	2/PA1, 3/G, 1/AH1, 1/UH3, 1/R, 1/D
Guess	2/PA1, 3/G, 1/EH, 1/S
Guide	2/PA1, 2/G, 1/AH1, 1/E1, 1/D

Had	2/PA1, 2/H, 1/AE, 1/EH3, 1/D
Hair	2/PA1, 2/H, 1/A2, 1/EH2, 1/R
Hand	2/PA1, 3/H, 1/AE, 1/EH3, 1/N, 1/D
Handle	2/PA1, 2/H, 2/AE1, 1/EH3, 1/N, 1/D, 1/UH3, 1/L
Happen	2/PA1, 3/H, 2/AE1, 1/EH3, 1/P, 1/I3, 1/N
Hardware	2/PA1, 3/H, 1/AH1, 1/R, 1/D, 1/W, 1/EH2, 1/R
Has	2/PA1, 2/H, 1/AE, 1/I3, 1/Z
Have	2/PA1, 2/H, 1/AE, 1/UH3, 1/V
He	2/PA1, 2/H, 1/E
Headlight	2/PA1, 2/H, 1/EH1, 2/D, 1/L, 1/UH3, 1/AH2, 1/Y1, 1/T
Hear	2/PA1, 2/H, 1/E, 1/R
Heard	2/PA1, 2/H, 1/ER, 1/R, 1/D
Heart	2/PA1, 2/H, 1/AH1, 1/R, 1/T
Hello	2/PA1, 3/H, 1/EH1, 1/UH3, 1/L, 1/UH3, 1/O2, 2/UI
Help	2/PA1, 3/H, 1/EH1, 1/UH3, 1/L, 1/P
Her	2/PA1, 2/H, 1/ER, 1/R
High	2/PA1, 2/H, 1/AH1, 1/E1
Him	2/PA1, 2/H, 1/I1, 1/M
Hire	2/PA1, 2/H, 1/AH1, 1/Y1, 1/ER
His	2/PA1, 2/H, 1/I, 1/Z
How	2/PA1, 3/H, 1/AH1, 1/O2, 1/UI

However	2/PA1, 2/H, 1/AH1, 1/UI, 1/EH1, 1/V, 1/R, 2/R
Hundred	2/PA1, 3/H, 1/UH1, 1/N, 1/D, 1/R, 1/EH2, 1/D
Hurry	2/PA1, 2/H, 1/ER, 1/R, 1/Y



I 2/PA1, 1/AH1, 1/EH3, 1/E1

If 2/PA1, 1/I, 1/F

Impatient 1/PA1, 1/I1, 2/M, 3/P, 2/A1, 1/AY, 1/SH, 1/EH3, 1/N, 1/T

Imperative 1/PA1, 1/I2, 2/M, 2/P, 2/EH1, 1/R, 1/EH3, 1/T  
1/I3, 1/V

Imply 1/PA1, 1/I1, 2/M, 2/P, 2/L, 1/AH1, 1/Y

Impossible 1/PA1, 1/I2, 2/M, 3/P, 1/AH1, 1/S, 1/UH3, 1/B, 1/L

Impress 1/PA1, 1/I2, 1/M, 2/P, 2/R, 1/EH1, 1/S

Improvement 1/PA1, 1/I3, 1/M, 3/P, 1/R, 2/U1, 1/V, 1/M, 1/EH3, 1/N, 1/T

In 2/PA1, 1/I, 1/N

Inch 2/PA1, 1/I1, 1/N, 1/T, 1/CH

Include 1/PA1, 1/I3, 1/N, 3/K, 1/L, 1/IU, 1/U1, 1/D

Inconsistent 2/PA1, 2/I1, 1/N, 1/K, 1/UH3, 2/N, 3/S, 1/I2, 1/S, 1/T, 1/EH3, 1/N, 1/T

Inconvenience-2/PA1, 2/I1, 1/N, 1/K, 1/UH3, 2/N, 3/V, 1/E1, 1/N, 1/Y1, 1/UH3, 1/N, 1/S

Increase 1/PA1, 1/I2, 1/N, 3/K, 1/R, 1/E, 1/S

Independent- 2/PA1, 1/I2, 1/N, 1/D, 1/Y1, 3/P, 1/EH1, 1/N, 1/D, 1/I3, 1/N, 1/T

Indicate 2/PA1, 2/I2, 1/N, 1/D, 1/I3, 1/K, 1/A1, 1/AY, 1/T

Inferior 1/PA1, 1/I2, 1/N, 3/F, 2/E1, 1/R, 1/Y, 1/ER

Influence 2/PA1, 2/I1, 1/N, 2/F, 1/L, 1/U1, 1/I3, 1/N, 1/S

Inform	1/PA1, 1/I2, 1/N, 3/F, 1/O, 1/R, 1/M
Information	2/PA1, 1/I2, 1/N, 1/F, 2/R, 3/M, 1/A1, 1/AY, 1/SH, 1/UH3, 1/N
Initial	1/PA1, 1/I2, 2/N, 2/I1, 1/SH, 1/UH3, 1/L
Inquire	1/PA1, 1/I3, 2/N, 3/K, 1/W, 1/AH2, 1/E1, 1/R
Insert	1/PA1, 1/I2, 2/N, 2/S, 1/ER, 1/R, 1/T
Inside	2/PA1, 1/I3, 1/N, 2/S, 1/AH1, 1/E1, 1/D
Instead	1/PA1, 1/I2, 2/N, 2/S, 1/T, 1/EH, 1/D
Instruct	1/PA1, 1/I2, 1/N, 3/S, 2/T, 1/R, 1/UH1, 1/K, 1/T
Insufficient	2/PA1, 1/I2, 2/N, 1/S, 1/UH3, 2/F, 1/I2, 1/SH, 1/EH3, 1/N, 1/T
Insurance	1/PA1, 1/I1, 1/N, 3/SH, 1/R, 1/R, 1/EH3, 1/N, 1/S
Intelligibility	1/PA1, 1/I2, 2/N, 1/T, 1/EH3, 2/L, 1/UH3, 2/D, 2/J, 1/UH3, 2/B, 1/I2, 1/L, 1/I3, 1/T, 2/Y
Intend	1/PA1, 1/I2, 2/N, 3/T, 1/EH1, 1/I3, 1/N, 1/D
Interest	2/PA1, 2/I2, 2/N, 1/T, 1/R, 1/EH2, 1/S, 1/T
Interrupt	1/PA1, 1/I2, 2/N, 1/T, 1/ER, 2/UH1, 2/P, 2/T
Into	1/PA1, 1/I1, 1/N, 2/T, 1/U
Introduced	1/PA1, 1/I2, 1/N, 1/T, 2/R, 1/UH3, 2/D, 1/IU, 1/U1, 1/S, 1/T
Investigate	1/PA1, 1/I2, 2/N, 2/V, 2/EH1, 1/S, 1/T, 1/I3, 1/G, 1/A2, 2/Y1, 2/T
Is	1/PA1, 1/I, 2/Z
It	1/PA1, 1/I1, 1/T

Jacket	2/PA1, 2/D, 2/J, 1/AE1, 1/EH3, 1/K, 1/EH3, 1/T
January	2/PA1, 2/D, 1/J, 2/AE1, 1/I3, 1/N, 2/Y1, 1/U1, 1/EH3, 1/I3, 1/R, 2/E1
Joint	2/PA1, 2/D, 2/J, 1/O1, 1/I3, 1/AY, 2/N, 1/T
July	1/PA1, 1/D, 1/J, 1/IU, 3/L, 1/AH1, 1/E1
June	1/PA1, 2/D, 3/J, 1/IU, 1/U, 1/N
Justice	2/PA1, 2/D, 2/J, 1/UH1, 1/S, 1/T, 1/I3, 1/S

Keep	2/PA1, 2/K, 1/E1, 1/Y, 1/P
Kind	2/PA1, 2/K, 1/AH1, 1/E1, 1/N, 1/D
Knew	2/PA1, 1/N, 1/IU, 1/U
Know	2/PA1, 1/N, 1/UH3, 1/O1, 1/UI
Knowledge	2/PA1, 2/N, 1/AH1, 1/L, 1/EH3, 1/D, 1/J

Lead *	2/PA1, 2/L, 1/EH1, 1/I3, 1/D
Least	1/PA1, 2/L, 1/E1, 1/Y, 1/S, 1/T
Leave	1/PA1, 2/L, 1/E1, 1/Y1, 1/V
Leeway	2/PA1, 2/L, 2/E1, 1/W, 1/A1, 1/AY
Left	2/PA1, 2/L, 1/EH2, 1/UH3, 1/F, 1/T
Life	2/PA1, 2/L, 1/AH1, 1/E1, 1/F
Light	2/PA1, 2/L, 1/UH3, 1/AH2, 1/Y1, 1/T
Limit	1/PA1, 2/L, 1/I2, 1/M, 1/I3, 1/T
List	2/PA1, 1/L, 1/I1, 1/S, 1/T
Live	2/PA1, 2/L, 1/I1, 1/V
Location	2/PA1, 1/L, 1/UH3, 1/O2, 2/K, 1/A2, 1/Y1, 1/SH, 1/UH3, 1/N

Made	1/PA1, 1/M, 2/A1, 1/AY, 1/Y1, 1/D
Manage	2/PA1, 2/M, 1/AE1, 1/EH3, 1/N, 1/I3, 1/D, 1/J
March	2/PA1, 2/M, 1/AH1, 1/UH3, 1/R, 1/T, 1/CH
Mark	2/PA1, 2/M, 1/AH1, 1/UH3, 1/R, 1/K
Material	2/PA1, 1/M, 1/EH3, 3/T, 2/I1, 1/R, 1/E1, 1/UH3, 1/L
Maximum	2/PA1, 2/M, 1/AE1, 2/K, 1/S, 1/EH3, 1/M, 1/UH3, 1/M
May	2/PA1, 1/M, 2/EH3, 1/A1, 1/AY
Measurement	2/PA1, 2/M, 2/EH1, 1/ZH, 1/R, 1/M, 1/EH3, 1/N, 1/T
Mechanic	1/PA1, 1/M, 1/EH2, 3/K, 1/AE1, 1/EH3, 1/N, 1/I3, 1/K
Medium	2/PA1, 2/M, 2/E1, 1/D, 1/E1, 1/UH2, 1/M
Member	1/PA1, 2/M, 2/EH1, 1/M, 1/B, 1/ER
Men	2/PA1, 2/M, 1/EH1, 1/I3, 1/N
Merge	2/PA1, 2/M, 1/ER, 1/R, 1/D, 1/J
Mile	1/PA1, 2/M, 1/AH1, 1/I3, 1/UH3, 1/L
Million	2/PA1, 1/M, 2/I1, 1/L, 1/Y1, 1/UH3, 1/N
Mind	2/PA1, 2/M, 1/AH1, 1/E1, 1/N, 1/D
Minutes	2/PA1, 2/M, 1/I1, 1/N, 1/I3, 1/T, 1/S
Mistake	2/PA1, 1/M, 1/I3, 2/S, 2/T, 1/A1, 1/AY, 1/Y1, 1/K
Monday	2/PA1, 2/M, 1/UH1, 1/N, 1/D, 1/A1, 1/AY

Month.	2/PA1, 2/M, 1/UH1, 2/N, 1/TH
Mr.	2/PA1, 2/M, 2/I2, 1/S, 1/T, 1/ER
Mrs.	1/PA1, 1/M, 2/I1, 1/S, 1/I2, 1/Z
Much	1/PA1, 2/M, 1/UH1, 1/T, 1/CH
Must	1/PA1, 2/M, 1/UH1, 1/S, 1/T

Nation	2/PA1, 1/N, 2/A1, 1/AY, 1/SH, 1/EH2, 1/N
Necessitate	2/PA1, 1/N, 1/EH3, 2/S, 2/EH1, 1/S, 1/I3, 1/T 1/A1, 1/AY, 1/T
Never	2/PA1, 1/N, 2/EH1, 1/V, 1/ER
Newspaper	2/PA1, 1/N, 2/IU, 1/U1, 2/Z, 1/P, 1/A1, 1/AY, 1/P, 1/R
Nine	2/PA1, 2/N, 1/AH1, 1/I3, 1/Y, 1/N
Nobody	2/PA1, 1/N, 2/O1, 1/B, 1/UH1, 1/D, 1/Y
North	2/PA1, 2/N, 1/O, 1/R, 1/TH
Not	2/PA1, 1/N, 2/AH1, 1/T
November	1/PA1, 1/N, 1/O1, 3/V, 2/EH2, 1/EH3, 1/M, 1/B, 1/ER
Now	2/PA1, 2/N, 2/UH3, 1/AH1, 1/O2, 1/U1
Number	2/PA1, 1/N, 2/UH1, 1/M, 1/B, 1/ER



Objected	1/PA1, 1/UH2, 1/B, 2/D, 2/J, 1/EH1, 1/K, 1/PAØ, 1/T, 1/I3, 1/D
Observe	1/PA1, 1/UH2, 2/B, 1/Z, 1/ER, 2/R, 1/V
O'Clock	1/PA1, 1/O1, 3/K, 1/L, 1/AH1, 1/K
October	1/PA1, 1/AH1, 1/K, 2/PAØ, 3/T, 1/O2, 1/U1, 1/B, 1/R
On	2/PA1, 1/AH, 1/UH3, 1/N, 1/N
Once	2/PA1, 2/W, 1/UH1, 1/UH3, 1/N, 1/T, 1/S
One	2/PA1, 2/W, 1/UH, 1/N
Operate	2/PA1, 2/AH1, 1/P, 1/ER, 1/A1, 1/AY, 1/T
Our(s)	2/PA1, 1/AH1, 1/U1, 1/ER, (1/Z)
Over	2/PA1, 2/O1, 1/U1, 1/V, 1/ER

Pacific	1/PA1, 1/P, 2/EH2, 3/S, 1/II, 1/F, 1/I3, 1/K
Page	2/PA1, 1/P, 2/A1, 1/AY, 1/AY, 1/D1, 1/J
Paper	2/PA1, 1/P, 2/A1, 1/AY, 1/P, 1/ER
Parallel	2/PA1, 2/P, 2/EH1, 1/R, 2/UH3, 1/L, 1/EH3, 1/UH3, 1/L
Part(s)	2/PA1, 2/P, 1/AH1, 1/R, 1/T, (1/S)
Pass	2/PA1, 2/P, 1/AE, 1/EH3, 1/S
Pattern	2/PA1, 2/P, 1/AE1, 1/EH3, 1/DT, 1/ER, 1/N
Pavement	2/PA1, 1/P, 2/A1, 1/AY, 1/V, 1/M, 1/EH3, 1/N, 1/T
Pay	2/PA1, 2/P, 1/EH3, 1/A1, 1/AY
Percent	1/PA1, 1/P, 1/R, 3/S, 1/EH1, 1/N, 1/T
Percentage	1/PA1, 1/P, 1/R, 3/S, 1/EH1, 1/N, 1/T, 1/I3, 1/D, 1/J
Performance	1/PA1, 1/P, 1/R, 3/F, 1/O2, 1/R, 1/M, 1/EH3, 1/N, 1/S
Period	2/PA1, 2/P, 2/I2, 1/R, 1/E1, 1/UH3, 1/D
Pertain	1/PA1, 1/P, 1/ER, 3/T, 1/A1, 1/AY, 1/Y1, 1/N
Phone	2/PA1, 2/F, 1/O, 1/U1, 1/N
Phrase	2/PA1, 2/F, 1/R, 1/A1, 1/AY, 1/Y1, 1/Z
Place	2/PA1, 2/P, 1/L, 1/A1, 1/AY, 1/Y1, 1/S
Placement	2/PA1, 2/P, 2/L, 1/A2, 1/AY, 1/S, 1/M, 1/EH3, 1/N, 1/T
Plan	2/PA1, 2/P, 2/L, 1/AE1, 1/EH3, 1/N

Please	2/PA1, 2/P, 2/L, 1/E1, 1/E1, 1/Z
P.M.	1/PA1, 1/P, 1/E1, 2/E1, 2/PAØ, 2/EH1, 1/EH3, 1/UH3, 1/M
Policy	2/PA1, 2/P, 1/AH1, 1/L, 1/I3, 1/S, 1/Y
Position	1/PA1, 1/P, 2/UH3, 2/Z, 1/I1, 1/SH, 1/UH3, 1/N
Possible	2/PA1, 2/P, 1/AH1, 1/S, 1/UH3, 1/B, 1/L
Postage	2/PA1, 2/P, 1/O1, 1/S, 1/T, 1/I3, 1/D, 1/J
Potential	1/PA1, 1/P, 2/O2, 2/T, 2/EH1, 1/N, 1/SH, 1/UH3, 1/L
Practice	2/PA1, 1/P, 2/R, 1/AE1, 2/K, 1/PAØ, 1/T, 1/I3, 1/S
Preference	2/PA1, 2/P, 2/R, 1/EH2, 1/V, 1/R, 1/EH3, 1/N, 1/S
Preparation	2/PA1, 1/P, 1/R, 1/EH3, 2/P, 1/ER, 2/A1, 1/AY, 1/SH, 1/UH3, 1/N
Prepare	2/PA1, 2/P, 1/R, 1/EH3, 3/P, 1/EH3, 1/A1, 1/R
Present	1/PA1, 1/P, 1/R, 2/I3, 2/Z, 1/EH2, 1/EH2, 1/N, 1/T
Pressure	2/PA1, 1/P, 2/R, 1/EH1, 1/SH, 1/ER
Price	2/PA1, 2/P, 1/R, 1/UH3, 1/AH2, 1/E1, 1/S
Print	2/PA1, 2/P, 1/R, 1/I1, 1/N, 1/T
Problem(s)	2/PA1, 2/P, 2/R, 1/AH1, 1/B, 1/L, 1/EH3, 1/M, (1/Z)
Produce	2/PA1, 1/P, 1/R, 2/UH3, 1/D, 1/IU, 1/U1, 1/S
Program	2/PA1, 1/P, 2/R, 1/O1, 1/G, 1/R, 1/AE1, 1/EH3, 1/M
Progress	2/PA1, 2/P, 2/R, 2/AH1, 1/G, 1/R, 1/EH1, 1/S
Public	2/PA1, 1/P, 2/UH1, 1/B, 1/L, 1/I3, 1/K
Pupil	2/PA1, 1/P, 2/Y1, 2/U1, 1/P, 1/UH3, 1/L

Quantity	2/PA1, 2/K, 1/W, 2/AH2, 1/EH3, 2/N, 1/DT, 1/I2, 1/DT, 1/Y
Quart	2/PA1, 2/K, 2/W, 1/O2, 1/R, 1/T
Question	1/PA1, 2/K, 2/W, 1/EH1, 1/S, 2/T, 2/CH, 2/UH3, 1/N
Quick	2/PA1, 2/K, 1/W, 1/I1, 1/K
Quiet	2/PA1, 2/K, 1/W, 1/AH1, 1/Y, 1/EH3, 1/T
Quite	2/PA1, 2/K, 1/W, 1/UH1, 1/Y, 1/T
Quotation	2/PA1, 1/K, 1/W, 1/O1, 3/T, 2/A2, 1/Y, 1/SH, 1/UH3, 1/N
Quote	2/PA1, 2/K, 1/W, 1/O1, 1/U1, 1/T

Range	1/PA1, 1/R, 2/A1, 1/AY, 1/I3, 1/N, 1/D, 1/J
Rapid	2/PA1, 2/R, 1/AE1, 1/P, 1/I3, 1/D
Reaction	1/PA1, 1/R, 2/E1, 1/AE1, 1/EH3, 1/K, 1/SH, 1/UH3, 1/N
Read *	2/PA1, 2/R, 1/EH3, 1/I3, 1/D
Read *	2/PA1, 2/R, 1/E1, 1/E1, 1/D
Receive	1/PA1, 1/R, 1/E1, 3/S, 1/E, 1/E1, 1/V
Recognition	2/PA1, 1/R, 1/EH3, 2/K, 1/I2, 2/G, 3/N, 1/I2, 1/SH, 1/UH3, 1/N
Recommend	2/PA1, 1/R, 1/EH3, 2/K, 1/UH3, 3/M, 1/EH, 1/N, 1/D
Recommendation	2/PA1, 2/R, 1/EH3, 2/K, 1/EH2, 1/M, 1/EH2, 1/N, 3/D, 1/A1, 1/AY, 1/SH, 1/UH3, 1/N
Refund	2/PA1, 2/R, 1/E1, 1/F, 1/UH3, 1/UH1, 1/N, 1/D
Register	2/PA1, 2/R, 1/EH2, 1/D, 1/J, 1/I3, 1/S, 1/T, 1/R
Release(d)	1/PA1, 1/R, 1/E1, 2/L, 1/E1, 1/Y1, 1/S, (1/T)
Remark	2/PA1, 1/R, 1/E1, 3/M, 1/AH1, 1/R, 1/K
Remember	2/PA1, 1/R, 1/E1, 3/M, 2/EH3, 2/UH3, 1/M, 1/B, 1/R
Remind	2/PA1, 1/R, 1/E1, 3/M, 1/AH1, 1/E1, 1/N, 1/D
Reorganize	2/PA1, 1/R, 1/E1, 3/O1, 1/R, 2/G, 1/UH3, 1/N, 1/AH1, 1/Y1, 1/Z
Repeat	1/PA1, 1/R, 1/E1, 3/P, 1/E1, 1/Y1, 1/T
Replace	1/PA1, 1/R, 1/E1, 3/P, 1/L, 1/A1, 1/AY, 1/S
Replacement	1/PA1, 1/R, 1/E1, 3/P, 1/L, 2/A1, 1/Y1, 1/S, 1/M, 1/EH3, 1/N, 1/T

Report	1/PA1, 1/R, 1/E1, 3/P, 1/O1, 1/R, 1/T
Represent	2/PA1, 2/R, 1/EH3, 1/P, 1/R, 1/I3, 3/Z, 1/EH1, 1/N, 1/T
Request	1/PA1, 1/R, 1/I3, 3/K, 1/W, 1/EH1, 1/S, 1/T
Return	1/PA1, 1/R, 1/E1, 3/T, 1/ER, 1/R, 1/N
Right	2/PA1, 2/R, 1/AH2, 1/UH3, 1/E1, 1/T
Rotation	1/PA1, 1/R, 1/O2, 3/T, 1/A1, 1/AY, 1/SH, 1/UH3, 1/N
Runway	2/PA1, 2/R, 1/UH1, 1/N, 1/W, 1/A1, 1/AY

Said	2/PA1, 3/S, 1/EH1, 1/I3, 1/D
Same	2/PA1, 2/S, 1/EH3, 1/A1, 1/AY, 1/Y1, 1/M
Saturday	2/PA1, 2/S, 1/AE1, 2/EH3, 1/DT, 1/R, 1/D, 1/A1, 1/AY
Save	2/PA1, 2/S, 1/EH3, 1/A1, 1/AY, 1/Y1, 1/V
Say	2/PA1, 2/S, 1/EH3, 1/A1, 1/AY
School	2/PA1, 2/S, 2/K, 1/U1, 1/U1, 1/L
Screen	2/PA1, 2/S, 2/K, 2/R, 1/E1, 1/E1, 1/N
Search	2/PA1, 2/S, 1/ER, 1/R, 1/T, 1/CH
Season	2/PA1, 3/S, 1/E, 1/Z, 1/EH2, 1/N
Second(s)	2/PA1, 1/S, 2/EH1, 1/K, 1/UH2, 1/N, 1/T, (1/S)
Section	2/PA1, 1/S, 2/EH1, 2/K, 1/SH, 1/UH3, 1/N
Secure	1/PA1, 1/S, 1/I3, 2/K, 2/Y1, 1/ER, 1/R
See	2/PA1, 2/S, 1/E1, 1/E1
Seen	2/PA1, 2/S, 1/E1, 1/E1, 1/N
Select	1/PA1, 1/S, 2/UH3, 2/L, 1/UH3, 1/EH1, 1/K, 1/T
Self	2/PA1, 2/S, 1/EH1, 1/UH3, 1/L, 1/F
Sell	2/PA1, 2/S, 1/EH1, 1/UH3, 1/L
Send	2/PA1, 2/S, 1/EH1, 1/I3, 1/N, 1/D
Sent	2/PA1, 2/S, 2/EH3, 1/EH1, 1/N, 1/T
Separate	2/PA1, 2/S, 1/EH1, 1/P, 1/R, 1/EH3, 1/T

Separation	2/PA1, 1/S, 1/EH1, 2/P, 1/ER, 2/A1, 1/AY, 1/SH, 1/UH3, 1/N
September	1/PA1, 1/S, 1/EH2, 2/P, 3/T, 1/EH1, 1/M, 1/B, 1/R
Serve	2/PA1, 1/S, 2/ER, 1/R, 1/V
Seven	2/PA1, 2/S, 1/EH1, 1/V, 1/EH3, 1/N
Seventeen	2/PA1, 2/S, 1/EH1, 1/V, 1/EH3, 2/N, 2/T, 1/E1, 1/E1, 1/N
Seventy	2/PA1, 2/S, 1/EH1, 1/V, 1/EH3, 1/N, 1/D, 1/Y
Several	2/PA1, 1/S, 2/EH1, 1/V, 1/R, 1/UH3, 1/L
She	2/PA1, 2/SH, 1/E1, 1/E1
Shipment	2/PA1, 2/SH, 2/I2, 1/P, 1/M, 1/EH3, 1/N, 1/T
Short	2/PA1, 2/SH, 1/O1, 1/R, 1/T
Shortage	2/PA1, 2/SH, 1/O2, 1/R, 1/T, 1/I2, 1/D, 1/J
Should	2/PA1, 2/SH, 1/IU, 1/IU, 1/IU, 1/D
Side	2/PA1, 2/S, 1/AH1, 1/E1, 1/D
Sign	2/PA1, 2/S, 1/AH1, 1/I3, 1/E1, 1/N
Signal	2/PA1, 2/S, 2/I1, 1/G, 1/N, 1/UH3, 1/L
Signature	2/PA1, 3/S, 2/I1, 1/G, 1/N, 1/I3, 1/T, 1/CH, 1/ER
Significant	1/PA1, 1/S, 1/I2, 2/G, 3/N, 1/I2, 1/F, 1/I3, 1/K, 1/EH3, 1/N, 1/T
Since	2/PA1, 2/S, 1/I, 1/N, 1/S
Single	2/PA1, 2/S, 2/I1, 1/NG, 1/G, 1/L
Situation	2/PA1, 2/S, 1/I2, 1/T, 1/CH, 1/IU, 2/U1, 1/A2, 1/AY, 1/SH, 1/UH3, 1/N



Six	2/PA1, 2/S, 1/II, 1/K, 1/S
Sixteen	2/PA1, 1/S, 1/II, 2/K, 2/S, 2/T, 1/E1, 1/E1, 1/N
Sixty	2/PA1, 2/S, 1/II, 1/K, 1/S, 1/T, 1/Y
Size	2/PA1, 2/S, 1/AH1, 1/I3, 1/E1, 1/Z
Sketch	2/PA1, 2/S, 2/K, 1/EH1, 1/I3, 1/T, 1/CH
Slow	2/PA1, 2/S, 2/L, 1/UH3, 1/O1, 1/U1
Sort	2/PA1, 2/S, 1/O, 1/R, 1/T
Sound	2/PA1, 2/S, 1/AH1, 1/UH3, 1/U1, 1/N, 1/D
South	2/PA1, 2/S, 1/AH1, 1/O2, 1/U1, 1/TH
Speaker	2/PA1, 2/S, 2/P, 1/E, 1/K, 1/ER
Specific	1/PA1, 1/S, 1/EH2, 2/S, 1/II, 1/F, 1/I3, 1/K
Speed	2/PA1, 2/S, 2/P, 1/E1, 1/E1, 1/D
Spend	2/PA1, 2/S, 2/P, 1/EH1, 1/I3, 1/N, 1/D
Spring	2/PA1, 2/S, 2/P, 1/R, 1/II, 1/NG
Square	1/PA1, 1/S, 2/K, 2/W, 1/EH3, 1/I3, 1/R
Stamp	2/PA1, 2/S, 2/T, 1/AE1, 1/EH3, 1/M, 1/P
Standard	2/PA1, 2/S, 3/T, 1/AE1, 1/EH3, 1/N, 1/D, 1/ER, 1/D
State(s)	2/PA1, 2/S, 2/T, 1/A1, 1/AY, 1/Y1, 1/T, (1/S)
Statement	2/PA1, 2/S, 3/T, 1/A1, 1/AY, 2/T, 1/M, 1/EH3, 1/N, 1/T
Station	2/PA1, 2/S, 1/T, 2/A1, 1/AY, 1/SH, 1/UH3, 1/N
Statistics	2/PA1, 1/S, 1/T, 1/EH2, 3/T, 2/II, 1/S, 1/T, 1/I3, 1/K, 1/S

Stock	2/PA1, 2/S, 2/T, 1/AH1, 1/AW2, 1/K
Stop	2/PA1, 2/S, 2/T, 1/AH, 1/UH3, 1/P
Store	2/PA1, 2/S, 2/T, 1/O, 1/R
Subject	2/PA1, 2/S, 2/UH1, 1/B, 1/D, 1/J, 1/EH1, 1/K, 1/T
Subjective	2/PA1, 1/S, 1/UH2, 2/B, 3/D, 2/J, 2/EH1, 1/K, 1/T, 1/I3, 1/V
Substitute	2/PA1, 2/S, 1/UH1, 1/B, 2/S, 1/T, 1/EH3, 1/T, 1/IU, 1/U1, 1/T
Success	2/PA1, 1/S, 1/UH2, 2/K, 2/S, 1/EH1, 1/S
Such	2/PA1, 2/S, 1/UH, 1/T, 1/CH
Sufficient	1/PA1, 1/S, 1/UH2, 3/F, 1/I1, 1/SH, 1/EH3, 2/N, 1/T
Suggest	1/PA1, 1/S, 1/UH2, 2/G, 2/D, 2/J, 1/EH, 1/S, 1/T
Sunday	2/PA1, 2/S, 2/UH1, 1/N, 2/N, 1/D, 1/A2, 1/AY
Supply	2/PA1, 1/S, 1/UH2, 2/P, 2/L, 1/AH1, 1/E1,
Support	2/PA1, 1/S, 1/UH2, 2/P, 1/O, 1/R, 1/T
Surplus	2/PA1, 2/S, 2/ER, 1/P, 1/L, 1/UH2, 1/S

Taught	1/PA1, 3/T, 1/AW, 1/T
Teacher	2/PA1, 2/T, 1/E1, 1/T, 1/CH, 1/ER
Technical	2/PA1, 1/T, 2/EH1, 1/K, 1/N, 1/I3, 1/K, 1/UH3, 1/L
Technique	1/PA1, 1/T, 1/EH1, 2/K, 1/N, 1/E, 1/K
Telephone	2/PA1, 2/T, 2/EH1, 1/L, 2/UH3, 1/F, 1/O1, 1/U1, 1/N
Television	2/PA1, 2/T, 2/EH1, 1/L, 2/UH3, 1/V, 1/I2, 1/ZH, 1/UH3, 1/N
Ten	2/PA1, 2/T, 2/EH3, 1/EH1, 1/N
Terminate	2/PA1, 2/T, 2/ER, 1/M, 1/I3, 1/N, 1/A2, 1/Y1, 1/T
Test	2/PA1, 2/T, 1/EH, 1/S, 1/T
That	2/PA1, 2/THV, 1/AE1, 1/EH3, 1/T
The	2/PA1, 1/THV, 1/UH3, 2/UH3, 3/UH3
Them	2/PA1, 2/THV, 1/EH1, 1/EH3, 1/M
Then	2/PA1, 2/THV, 1/EH1, 1/EH3, 1/N
There	2/PA1, 2/THV, 1/EH2, 1/A2, 1/R
These	2/PA1, 2/THV, 1/E1, 1/E1, 1/Z
Third	2/PA1, 2/TH, 1/ER, 1/R, 1/D
Thirteen	2/PA1, 3/TH, 1/R, 2/ER, 2/T, 2/PAØ, 1/E1, 1/E1, 1/N
This	2/PA1, 1/THV, 2/I, 1/S
Three	2/PA1, 2/TH, 2/R, 1/E1, 1/Y
Thursday	2/PA1, 2/TH, 2/TH, 2/ER, 1/R, 2/Z, 1/D, 1/A1, 1/AY

Time	2/PA1, 2/T, 1/AH1, 1/E1, 1/M
To	2/PA1, 2/T, 1/IU, 1/U
Topic	2/PA1, 2/T, 1/AH1, 1/P, 1/I3, 1/K
Touch	2/PA1, 2/T, 1/UH, 1/T, 1/CH
Traffic	2/PA1, 2/T, 2/R, 1/AE1, 1/EH3, 1/F, 1/I3, 1/K
Transaction	2/PA1, 1/T, 1/R, 1/AE1, 2/N, 3/Z, 1/AE1, 2/EH3, 1/K, 1/SH, 1/UH3, 1/N
Transfer	2/PA1, 2/T, 2/R, 1/AE1, 2/I3, 1/N, 2/S, 1/F, 1/ER
Tuesday	2/PA1, 2/T, 2/IU, 2/U1, 1/U1, 2/Z, 1/D, 1/A1, 1/AY

Unconscious	1/PA1, 1/UH2, 1/N, 2/K, 2/AH1, 1/N, 1/SH, 1/UH3, 1/S
Understand	1/PA1, 1/UH2, 2/N, 1/D, 1/R, 2/S, 2/T, 1/AE1, 1/EH3, 1/N, 1/D
Understood	1/PA1, 1/UH2, 2/N, 1/D, 1/R, 2/S, 2/T, 1/IU, 1/IU, 1/D
Uniform	2/PA1, 2/Y, 2/IU, 1/U1, 1/N, 2/UH3, 1/F, 1/O1, 1/R, 1/M
Unite	1/PA1, 1/Y1, 1/IU, 2/U1, 2/N, 1/AH1, 1/Y1, 1/T
Unknown	1/PA1, 1/UH2, 1/N, 3/N, 1/UH3, 1/O1, 1/U1, 1/N
Unlimited	1/PA1, 1/UH2, 2/N, 2/L, 2/I2, 1/M, 1/I3, 1/T, 1/I3, 1/D
Until	1/PA1, 1/UH2, 1/N, 2/T, 2/I1, 1/UH3, 1/L
Unusual	1/PA1, 1/UH2, 1/N, 2/Y1, 1/IU, 2/U1, 1/ZH, 1/IU, 1/U1, 1/UH3, 1/L
Upon	1/PA1, 1/UH1, 3/P, 1/AH1, 1/UH3, 1/N
Urgent	2/PA1, 2/R, 1/R, 1/D, 1/J, 1/EH2, 1/N, 1/T
Use *	2/PA1, 2/Y1, 1/IU, 1/U, 1/S
Use *	2/PA1, 2/Y1, 1/IU, 1/U, 1/Z

Vacancy	2/PA1, 2/V, 2/A1, 1/AY, 1/K, 1/EH3, 1/N, 1/S, 1/Y
Valid	1/PA1, 2/V, 1/AE1, 1/UH3, 1/L, 1/UH3, 1/D
Verify	2/PA1, 2/V, 1/EH1, 1/R, 1/EH3, 1/F, 1/AH1, 1/E1
Very	2/PA1, 2/V, 1/EH, 1/R, 1/Y
Voice	2/PA1, 2/V, 1/O1, 1/UH3, 1/E1, 1/S
Volume	2/PA1, 2/V, 1/AH1, 1/L, 1/Y1, 1/U1, 1/M
VOTRAX	2/PA1, 2/V, 1/O, 1/T, 1/R, 1/AE1, 1/EH3, 1/K, 1/S

Walk	2/PA1, 2/W, 1/AW, 1/K
Want	2/PA1, 2/W, 1/AH1, 1/UH3, 1/N, 1/T
Warehouse	2/PA1, 1/W, 2/EH1, 1/R, 1/H, 1/UH3, 1/AH2, 1/U1, 1/S
Watch	2/PA1, 2/W, 1/AW2, 1/AH1, 1/T, 1/CH
Wednesday	2/PA1, 2/W, 1/EH1, 2/N, 1/Z, 1/D, 1/A1, 1/AY
Weigh(t)	2/PA1, 2/W, 1/EH3, 1/A2, 1/AY, 1/Y1, (1/T)
Welcome	2/PA1, 2/W, 1/EH1, 2/L, 1/K, 1/UH1, 1/M
Well	2/PA1, 2/W, 1/EH1, 1/UH3, 1/L
Went	2/PA1, 2/W, 1/EH1, 1/N, 1/T
Were	2/PA1, 2/W, 1/ER, 1/R
West	2/PA1, 1/W, 2/EH1, 1/S, 1/T
What	1/PA1, 1/W, 2/UH1, 1/T
When	1/PA1, 1/W, 2/EH1, 1/N
Where	2/PA1, 1/W, 2/EH2, 1/A2, 1/R
Which	1/PA1, 2/W, 2/I1, 1/T, 1/CH
White	2/PA1, 1/W, 2/UH3, 1/AH2, 1/E1, 1/T
Why	1/PA1, 2/W, 2/UH3, 1/AH1, 1/E1
Will	2/PA1, 2/W, 1/I2, 1/UH3, 1/L
Wind *	2/PA1, 2/W, 1/AH1, 1/E1, 1/N, 1/D
Wind *	2/PA1, 2/W, 1/I1, 1/N, 1/D
Within	1/PA1, 2/W, 1/I2, 2/TH, 3/I2, 1/I3, 1/N

Without	1/PA1, 2/W, 1/I2, 2/TH, 3/UH3, 1/AH2, 1/U1, 1/T
Work	1/PA1, 1/W, 1/ER, 2/R, 2/K
Would	1/PA1, 2/W, 1/U1, 1/IU, 1/IU, 1/D
Wound *	1/PA1, 2/W, 2/UH3, 1/AH1, 1/U1, 1/N, 1/D
Wrap	1/PA1, 2/R, 1/AE1, 1/EH3, 1/P



X-ray

2/PA1, 2/EH3, 2/K, 1/S, 1/R, 1/AY, 1/AY

Yes	1/PA1, 2/Y, 1/EH1, 1/EH3, 1/S
Yesterday	1/PA1, 2/Y, 2/EH1, 2/S, 1/T, 1/R, 1/D, 1/A1, 1/AY
You	2/PA1, 2/Y, 1/IU, 1/U

Zero 2/PA1, 2/Z, 1/II, 1/R, 1/UH3, 1/O1

Zone 2/PA1, 2/Z, 2/UH3, 1/O1, 1/U1, 1/N

NOTES - -

1. This list consists primarily of verbs, adverbs and adjectives found with some frequency in spoken English. Those nouns included were selected due to their frequency of occurrence, but may not be adequate to accommodate the vocabulary needs of any one specific environment. Words with specific application, categorized by area (factory, office, home, classroom, etc.), subject (business, education, medicine, etc.), or reason (announcements, instructions, information, general, etc.) for use, will be programmed on request.
2. In general, single syllable words--those words containing only one vowel sound with or without consonant sounds--are to be considered of low intelligibility when presented in isolation. Words like CODE, EIGHT, LIFE, SELF, GUESS are included in this category. Even though they are recognizable out of context, their intelligibility is enhanced when used in context, as it would be with a human speaker. Words like BE, A, THE, OF, IS (open-ended single syllable words) are, by the nature of spoken English, short in duration and take on characteristics to enhance the word(s) they modify. Often, this may require a change in their pronunciation to accommodate the rhythm and accent of the contextual flow. Thus, these words are low in intelligibility when presented in isolation. When words are selected for VS 6 use, it is recommended that single syllable words be given the advantage of a contextual environment for maximizing their intelligibility, whenever possible.
3. The Asterisk (\*) following certain words indicate that there is more than one pronunciation for that spelling or more than one meaning for that spelling.
4. The following list is made up of words not found on the 500 list, but whose programs are. (Followed by their location word)

accept	(except)	"C"	(see)
ate	(eight)	cell	(sell)
awl	(all)	cent	(sent)
"B"	(be)	dew	(do)
bee	(be)	due	(do)
buy	(by)	eye	("I")
bye	(by)	fore	(for)

forth	(fourth)	scene	(seen)
four	(for)	sea	(see)
grate	(grcat)	sense	(cents)
hart	(heart)	sine	(sign)
herd	(heard)	their	(there)
here	(hear)	they're	(there)
hi	(high)	too	(to)
higher	(hire)	two	(to)
hour(s)	(our(s))	"U"	(you)
Inn	(in)	wait	(weigh(t))
knot	(not)	way	(weigh(t))
leased	(lcast)	wear	(where)
maid	(made)	witch	(which)
no	(know)	wood	(would)
new	(knew)	wright	(right)
"R"	(are)	write	(right)
rap	(wrap)	won	(one)
red	(read (*))	"Y"	(why)
reed	(read (*))		
rite	(right)		

UTI Connector List

<u>Conn</u>	<u>Pin No.</u>	<u>Signal Name</u>
J1	1 - 48	OSI BUS
J2	1	GND TO CBT
12 Pin Molex	2	DA TO CBT
	3	+24VDC FROM CBT
	4	OH TO CBT
	5	$\overline{RI}$ FROM CBT
	6	DT TO CBT
	7	VOTRAX SPEAKER
	8	VOTRAX SPEAKER GND
	9	TAPE RECORDER (MIC)
	10	TAPE RECORDER GND
	11	TAPE RECORDER CONTROL (REMOTE)
	12	TAPE RECORDER CONTROL (REMOTE) SHIELD
J3	1	VOTRAX TO PHONE (FROM CA-14 BOARD)
12 Pin Molex	2	AUX TO PHONE
	3	AUX GND
	4	TAPE PLAYER (MONITOR)
	5	TAPE PLAYER (MONITOR) GND
	6	TAPE PLAYER CONTROL (REMOTE)
	7	TAPE PLAYER CONTROL (REMOTE) SHIELD
	8	300 BAUD
	9	$\overline{DTR}$ FROM MODEM
	10	$\overline{CTS}$ TO MODEM
	11	RXDATA FROM MODEM
	12	TXDATA TO MODEM
JX	1	VOTRAX AUDIO GND
16 Pin DIP	2	VOTRAX AUDIO OUTPUT
	3	V+12V
	4	PHONEME CLOCK
	5	V-9V
	6	INFLECTION BIT LSB
	7	INFLECTION BIT MSB
	8	GND
	9	VOTRAX DATA 3
	10	VOTRAX DATA 4
	11	VOTRAX DATA 5
	12	VOTRAX DATA 2
	13	VOTRAX DATA 1
	14	VOTRAX DATA 0
	15	SPARE
	16	+5V

VOTRAX  
VOICE OUTPUT  
PRELIMINARY MANUAL  
SEPTEMBER 1978  
OHIO SCIENTIFIC INC.

## Installation

The CA-14 Voice I/O module is composed of a Model 567 PC board and the Votrax VSK Voice Output Module which is connected to the unit by a 16 pin ribbon cable. The 567 board can be plugged in directly to one slot of a standard OSI 48 line backplane providing that the system

has +12, +5 and -9 power supplies present. The Votrax Module can be mounted on the 567 board via stand-offs and two screws, however, it is recommended that the unit simply be placed in the rear open area of the standard 8 or 16-slot Challenger chassis. The RCA phono jack at the output of the 565 board can be connected directly to an 8 ohm impedance speaker or through the auxiliary input of an audio amplifier. For initial testing, direct connection to a speaker is advisable. Once the system is installed, normal computer operation should be verified with a standard diskette. Two voice output diskettes accompany this system. These diskettes are identical except that one is for serial interface computers and ASCII keyboard based 440 and 540 video display systems. The other diskettes are for 540 based video systems which utilize the new polled keyboard. The polled keyboard can be spotted by the presence of a shift lock key. Utilize the diskette which is appropriate for your system. The diskette contains a stripped down version of OS-65D Version 3.0. The diskette contains three sample programs which should be present in menu form when the system is booted up. The most interesting of these programs is a talking calculator which either will generate random math problems or allow the user to key in specific math problems. In either case, the computer system displays the problem and the answer on the screen in conjunction with explicitly stating the problem and the answer. The second two programs are simple examples of using the talking 9 digit BASIC. These programs can be run and then later investigated in conjunction with learning how to program in talking BASIC. The fourth program is a SEED program which must be used as a basis of building specific programs in talking BASIC. This SEED program properly initializes the work space for talking BASIC and brings in the talking BASIC voice drivers via a technique which is totally transparent to the user.

## Programming In Talking BASIC

In order for you to successfully program in talking BASIC, it will be necessary to learn about the Votrax Module's operation. The following section of this manual is the Votrax Manual. You should thoroughly investigate this section before returning to the following paragraph.

As stated in the Votrax Reference Manual, the Votrax Module accepts 6 bit binary numbers which it converts to phonetics. The module signals when it is ready for another byte by its ready line. In Ohio Scientific systems, the user has two alternatives in programming the Votrax's output. The first alternative is demonstrated in the talking calculator program. In this particular program, the actual binary numbers which correspond to the phonetics are present in the program DATA statements and are simply pumped out to the Votrax Module via a POKE statement in conjunction with a WAIT statement. This programming technique can be utilized in any form of BASIC in any operating system since it is totally independent upon other software in the system for operation.



The second approach to programming is to utilize a machine code subroutine preferably imbedded in the BASIC language. This subroutine is activated by a selective PRINT statement. In OS-65D Version 3.0, this statement would be PRINT to device 5 which is indicated by a pound sign 5 (#5) after each PRINT to the Votrax Module. To further simplify programming, the system utilizes mnemonic representations of phonetics instead of actual binary numbers. So that the user simply must place the phonetic spelling for the words he wishes to output in a PRINT to device 5 PRINT statement. Examples of this programming are shown in the second and third menu selections on your demonstration disks. The machine subroutine which is utilized in conjunction with the PRINT to device 5 is listed in the back of the manual.

To investigate or modify the programs that are in the menu and to write your own programs, first answer PASS to the question "YOUR SELECTION". Then select the program you want. This will unlock the system and will allow listing and modification of programs. To write your own programs in talking BASIC after answering PASS, select the fourth selection SEED. Enter your program based on SEED. When you are ready to run it, first save it away because running your program may return you to the menu, hence, erase it in memory. To SAVE a program under 65D Version 3.0, first type EXIT which returns you to the operating system. (All responses must be followed by a return.) Then type PUT XX, where X is the track number. The first available track number in this system is 14. To return to BASIC, type RET BA. If your new program is to appear in the menu, add it to the menu program by emulating or mimicking the technique used for the other programs in the menu and SAVE this program on disk by typing EXIT followed by PUT 9. Your programs can be run by simply selecting them from the menu if installed or by typing EXIT LOAD XX where XX refers to the starting track of your program followed by RET BA which will return you to BASIC with your program in the work space. This diskette is fully compatible with the full implementation of OS-65D Version 3.0 which is capable of named file operation in conjunction with convenient-to-use disk data file I/O. The 65 D Version 3.0 file CREATE CHANGE and DATA-FILE-DUMP utilities can all be utilized with this system.

Please note that the Votrax phonetic spelling dictionary included in this manual is for a higher priced version of the Votrax system. It includes an inflection number and slash before the phonetic spelling. For use of this system, please omit the inflection number and slash before the phoentic.

#### Advanced Topics With Voice Output

For best sound quality, the Votrax Module should be fed to the auxiliary input of a high fidelity audio amplifier and out to a good quality speaker. By adjusting the base and treble controls, some improvements in sound output can be achieved. The Votrax output can further be improved by adding inflection or volume changes to phonetics as they are output. This is the basic difference between the low cost VSK Votrax unit and the \$5,000 to \$10,000 professional Votrax Modules that are sold to banks and telephone companies. This inflection capability can be added by an electronics experimenter.

VOTRAX

MODEL VSL SYNTHESIZER INTERFACE SPECIFICATIONS

(Part #1963)

MANUAL PART NUMBER 1964

AUGUST 29, 1977

## TABLE OF CONTENTS

- I. GENERAL DESCRIPTION
- II. SIGNAL FUNCTION DESCRIPTION
- III. HARDWARE DESCRIPTION

## I. GENERAL DESCRIPTION

The synthesizer requires 6 parallel data bits on its input pins in order to operate. These bits are used for phoneme selection.

The synthesizer provides a clock output which must be used to time the input data. Data should only be presented or changed on the positive transition of this clock.

All signals are TTL signal levels, except audio output.

### PIN DESCRIPTION (INPUT CONNECTOR)

<u>Pin No.</u>	<u>Function</u>	<u>Description</u>
1	Audio Ground	Audio Return
2	Audio Output	10 V P-P Maximum into Minimum 30 K Ohm
3	+12 VDC $\pm$ 10%	70 MA
4	Phoneme Clock	Drive 1 TTL Load Neg. Pulse, 350 microseconds wide
5	-9 VDC $\pm$ 10%	70 MA
6	LSB (Inflection Bit)	
7	MSB (Inflection Bit)	
8	Ground	Signal Return
9	Data In. Phoneme, $2^3$	1 TTL Load
10	Data In. Phoneme, $2^4$	1 TTL Load
11	Data In. Phoneme, $2^5$	1 TTL Load
12	Data In. Phoneme, $2^2$	1 TTL Load
13	Data In. Phoneme, $2^1$	1 TTL Load
14	Data In. Phoneme, $2^0$	1 TTL Load

Pos.  
True

<u>Pin No.</u>	<u>Function</u>	<u>Description</u>
15	Spare	---
16	+5 VDC $\pm 5\%$	300 MA

## II. SIGNAL FUNCTION DESCRIPTION

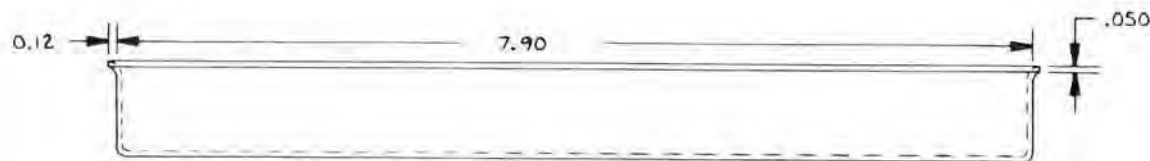
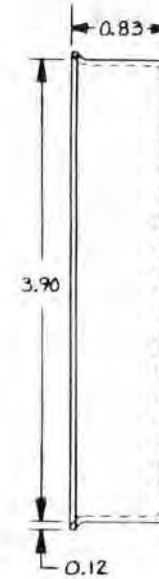
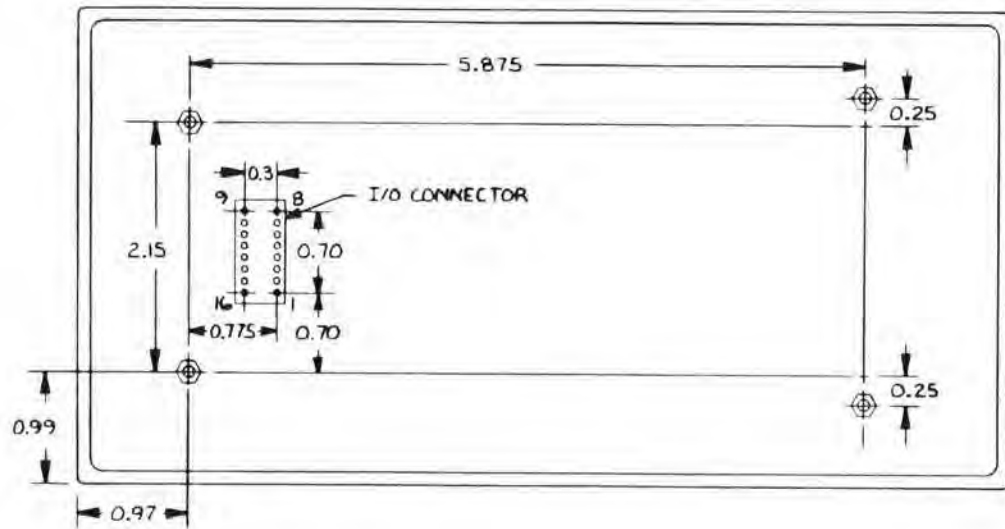
- A. Data In: (Pins 9-14) Six bit phoneme code inputs. All phoneme inputs high are Null code causing no audio output.
- B. Phoneme Clock Output: (Pin 4) Provides internal timing of synthesizer and must be used to control data input. Data should only be presented or changed on the positive edge of this signal. When all phoneme data inputs are high, this clock runs with a period of 40 milliseconds. When data input is present, the period lengthens, depending on the particular phoneme.

## III. HARDWARE DESCRIPTION

The Model VSK Voice Synthesizer consists of a single potted module 3.90" W x 7.95" L x 0.80" H. Mounting is accomplished by four screws engaging the four mounting standoffs. The electrical connections are made via one 16-pin connector on the same side as the standoffs.

The mating connector plug, which is to be mounted on the mother board to receive VSK module connector, is a 16-pin plug, Robinson-Nugent WPB-163, or equivalent.

The Drawing P/N 1963 presents the locations of the mounting standoffs and the module connector.



MATERIAL									
FINISH									
GENERAL TOLERANCE UNLESS SPECIFIED ± .010			CHG NO	ECR NO	DESCRIPTION	DATE	DR	CK	
THREE PLACE DECIMALS ± .005									
TWO PLACE DECIMALS ± .010									
ANGULAR ± 1°									
Votrax™					Vocal Interface Division Federal Screw Works Troy, Michigan 48084				
SCALE FULL	DATE	TITLE							
DR N. SIMPSON	8/30/77	VSK MODULE MOUNTING DIMENSIONS							
CK	6/2/77	FINAL ASS'Y						DRAWING NO	
REL	8/1/77	NEXT ASS'Y						B1963AD	

VOTRAX VS-K PHONETIC CODES

Bulletin #  
MTB-70C11-MB

<u>PHONEME</u>	<u>HEX</u>	<u>OCTAL</u>	<u>PHONEME</u>	<u>HEX</u>	<u>OCTAL</u>
PAØ	Ø3	ØØ3	G	1C	Ø34
PA1	3E	Ø76	H	1B	Ø33
A	2Ø	Ø4Ø	I	27	Ø47
A1	Ø6	ØØ6	I1	ØB	Ø13
A2	Ø5	ØØ5	I2	ØA	Ø12
AE	2E	Ø56	I3	Ø9	Ø11
AE1	2F	Ø57	IU	36	Ø66
AH	24	Ø44	J	1A	Ø32
AH1	15	Ø25	K	19	Ø31
AH2	Ø8	Ø1Ø	L	18	Ø3Ø
AW	3D	Ø75	M	ØC	Ø14
AW1	13	Ø23	N	ØD	Ø15
AW2	3Ø	Ø6Ø	NG	14	Ø24
AY	21	Ø41	O	26	Ø46
B	ØE	Ø16	O1	35	Ø65
CH	1Ø	Ø2Ø	O2	34	Ø64
D	1E	Ø36	OO	17	Ø27
DT	Ø4	ØØ4	OO1	16	Ø26
E	2C	Ø54	P	25	Ø45
E1	3C	Ø74	R	2B	Ø53
EH	3B	Ø73	S	1F	Ø37
EH1	Ø2	ØØ2	SH	11	Ø21
EH2	Ø1	ØØ1	T	2A	Ø52
EH3	ØØ	ØØØ	TH	39	Ø71
ER	3A	Ø72	THV	38	Ø7Ø
F	1D	Ø35	U	28	Ø5Ø

<u>PHONEME</u>	<u>HEX</u>	<u>OCTAL</u>	<u>PHONEME</u>	<u>HEX</u>	<u>OCTAL</u>
U1	37	067	W	2D	055
UH	33	063	Y	29	051
UH1	32	062	Y1	22	042
UH2	31	061	Z	12	022
UH3	23	043	ZH	07	007
V	0F	017			

### VS-K PROGRAMMING EXAMPLES

Zero: PA1, Z, AY, I3, R, UH3, 02, U1, PA1

One: PA1, W, UH3, UH2, UH2, N, N, PA1

Two: PA1, T, IU, IU, U1, U1, PA1

Three: PA1, TH, TH, R, E, Y, PA1

Four: PA1, F, 01, 01, R, PA1

Five: PA1, F, AH2, AH1, I3, Y, V, V, PA1

Six: PA1, S, I, I3, K, PA0, S, PA1

Seven: PA1, S, EH3, EH2, V, EH2, N, N, PA1

Eight: PA1, A2, A2, AY, Y, T, PA1

Nine: PA1, N, AH1, EH2, Y, N, PA1

Cancel: PA1, K, AE1, EH3, N, S, UH3, L, PA1

Add: PA1, AE1, AE, EH3, D, PA1

Subtract: PA1, S, UH2, B, PA0, T, R, AE1, AE1, K, PA0, T, PA1

Multiply: PA1, M, UH2, L, T, UH3, P, L, AH1, Y, PA1

Divide: PA1, D, I2, V, AH2, AH1, EH3, I3, AY, D, PA1

Equals: PA1, E1, AY, K, W, 001, L, Z, PA1



### INFLECTION LEVELS

To add inflection levels, add the selected level below to the phoneme value above.

Hex Octal  
(i.e., 1AH1 = D5 325 )

<u>LEVEL</u>	<u>HEX</u>	<u>OCTAL</u>
4	C0	300 (highest)
3	80	200
2	40	100
1	00	000 (lowest)