



Exploding The Phone

www.explodingthephone.com

Bibliographic Cover Sheet



Title **Evidence Examination Report: Pennsylvania State Police
Incident No. N6-39474, Bell Telephone Company of Pennsylvania
Case No. 23-50-E77**

Date 1977-00-00

Author(s) Bell Laboratories

Abstract Evidence examination report for Draper's 1977 arrest.

Keywords John Thomas Draper; John T. Draper; John Draper; Captain Crunch;
Charley board

Source Anonymous

The following pages may contain copyrighted material. We believe that our use of this material for non-commercial educational and research purposes constitutes "fair use" under Section 107 of U.S. Copyright Law. If you wish to use this material for purposes that go beyond "fair use," you must obtain permission from the copyright owner, if any. While it will make us slightly sad to do so, we will nonetheless comply with requests from copyright owners who want their material removed from our web site.



Bell Laboratories

EVIDENCE EXAMINATION REPORT

PENNSYLVANIA STATE POLICE
INCIDENT NO. N6-39474

BELL TELEPHONE COMPANY OF PENNSYLVANIA
CASE NO. 23-50-E77

Hand - Oct 22, 1977

EVIDENCE EXAMINATION REPORT

PENNSYLVANIA STATE POLICE
INCIDENT NO. N6-39474

BELL TELEPHONE COMPANY OF PENNSYLVANIA
CASE NO. 23-50-E77

Prepared by

Walter W. Heinze	Dept. 3331
Kenneth D. Hopper	Dept. 3331
Richard A. Previte	Dept. 3323
Paul Rabinowitz	Dept. 3325
Gerald S. Soloway	Dept. 3311

December 14, 1977

Evidence Examination
File 39865-4
Pennsylvania State Police
Incident N6-39474



Bell Laboratories

Holmdel, New Jersey 07733
Phone (201) 949-3000
December 13, 1977

Mr. W. D. Beam
Security Manager
Bell of Pennsylvania
901 Fayette Street
1st Floor
Conshohocken, Pennsylvania 19428

Trooper James R. Harris, Jr.
Pennsylvania State Police
P.O. Box 220
Swiftwater, Pennsylvania 18370

Gentlemen:

This report covers our examination of items of evidence relating to Pennsylvania State Police incident number N6-39474, involving subjects John Thomas Draper and Wortley Andrew Wright, Jr. The items examined were brought to our laboratory by you on October 27, 1977 or turned over to us at the Swiftwater station of the Pennsylvania State Police on November 1, 1977.

Custody and Control of Items of Evidence During Examination

On October 27, 1977, Trooper James R. Harris, badge no. 466, of the Pennsylvania State Police, stationed at Swiftwater, Pennsylvania, visited the Holmdel Laboratory of Bell Telephone Laboratories, Incorporated. Trooper Harris then turned over custody of 25 items of evidence to Mr. Kenneth D. Hopper, Member of Technical Staff, of Bell Telephone Laboratories, Incorporated for laboratory examination. The items received were described on Pennsylvania State Police Property Record inventory no. 1205 (dated 22 Oct. 77) as follows:

ITEM NO.

1. - ONE (1) PANASONIC CCTV MOD. #TR-9001M, SER. #68528067
2. - ONE (1) PANASONIC PORTABLE TAPE RECORDER, MODEL #RQ413AF

ITEM NO.

3. - FOUR (4) CASSETTE TAPES, ONE MARKED DYNAMIC DEBUGGING
4. - ONE (1) WHITE CARDBOARD BOX CONTAINING SIXTEEN (16) CASSETTE TAPES
5. - ONE (1) PACKAGE OF TWO (2) CASSETTE TAPES
6. - ONE (1) SOL TERMINAL COMPUTER, MOD. #20, SER. #213894, MADE BY PROCESSOR TECHNOLOGY
7. - ONE (1) PROCESSOR TECHNOLOGY SOL SYSTEMS MANUAL
8. - THREE BOOKS, "DYNAMIC DEBUGGING", "ALS-8" & 8080 MICRO COMP. SYSTEM USER MANUAL
9. - TWO SHEETS OF LINED PAPER CONTAINING NUMBER CODES
10. - ONE (1) TABLET OF LINED PAPER CONTAINING CODES AND DIAGRAMS
14. - ONE (1) "APPLE 11" MINI MANUAL COMPUTER BOOK
15. - ONE (1) BLUE, TOP FLIGHT, NOTEBOOK
16. - ONE (1) BROWN NOTEBOOK WITH ASSORTED PAPERS
20. - ONE (1) BROWN PAPER BAG CONTAINING SIX (6) COMPONENTS FOR "REDBOXES" AND BATTERIES FOR FOUR (4) OF SAME PLUS TWO PRINTED CIRCUITS AND MISCELLANEOUS COMPONENTS, & NINE (9) CASSETTE TAPES
21. - ONE (1) PRINTED CIRCUIT BOARD FOR "APPLE" COMPUTER
22. - ONE (1) TWO INCH SPEAKER
23. - ONE (1) COMPUTER KEYBOARD FOR "APPLE" COMPUTER
24. - ONE (1) IC BREADBOARD CIRCUIT, MARKED "BLUE BOX"
25. - TWO (2) ADVERSARY RESET SWITCHES
26. - ONE (1) BOXER FAN, MODEL WS2107FL-55
27. - ONE (1) PACKAGE OF SIX (6) CASSETTE TAPES
28. - ONE (1) HEWLETT-PACKARD TRANSFORMER WITH MISCELLANEOUS WIRES
29. - ONE (1) "APPLE" COMPUTER POWER SUPPLY
30. - ONE (1) PANASONIC PORTABLE TAPE RECORDER MODEL #RQ309DS WITH CASSETTE
31. - ONE (1) GENERAL ELECTRIC PORTABLE T.V., NO SERIAL NUMBER

A Pennsylvania State Police form SP-4 "Request for Laboratory Analysis" listing all of the above items was provided by Trooper Harris. A copy was receipted by Mr. Hopper and returned to Trooper Harris.

On November 1, 1977, Mr. Kenneth D. Hopper visited the Swiftwater Station of the Pennsylvania State Police and Trooper Harris turned over to him the following items:

ITEM NO.

11. - ONE (1) SMALL MEMO NOTEBOOK CONTAINING NUMBERS & DATA

ITEM NO.

- 12. - ONE (1) CLIPBOARD CONTAINING MISCELLANEOUS PAPERS WITH NUMBERS AND DIAGRAMS
- 13. - ONE (1) AMPAD BOX CONTAINING PAPER PARAPHENALIA
- 17. - ONE (1) AMSCO, BLUE, THREE RING BINDER, CONTAINING ASSORTED PAPERS
- 18. - ONE (1) BOX CONTAINING MISCELLANEOUS PAPERS

A separate Pennsylvania State Police form SP-4 "Request for Laboratory Analysis" listing the above items was provided by Trooper Harris. A copy was receipted and returned to Trooper Harris.

All items of evidence were in the custody of or under the control of Mr. Kenneth D. Hopper or Mr. Walter W. Heinze throughout the examination procedure. While at the Holmdel Laboratory, all items were kept in Room 3F-609 at all times.

Room 3F-609 consists of an outer office area, secured by a key-locked steel door and an inner laboratory area having a combination-locked vault-type door. The laboratory walls are steel from floor to ceiling. During night hours, all items of evidence except items 1, 2, and 6 were within the laboratory. Items 1, 2, and 6 were within the locked office area. Key possession and combination knowledge was limited only to the following persons:

Messrs. Walter W. Heinze
Alfred C. Bandini
Kenneth D. Hopper

and two members of higher management. No master keys exist. The Holmdel Laboratory building is under 24-hour continuous guard by Wackenhut Security Services, Inc.

On Wednesday, November 30, 1977, all items of evidence were brought to Stroudsburg, Pennsylvania by Messrs. Hopper and Heinze. They were contained in seven sealed boxes. The boxes were placed in the money counting room of the Bell Telephone Company of Pennsylvania business office located at 20 South 7th St., Stroudsburg, Pa. Access to this room is strictly controlled and keys are retained by Mr. Elmer B. Chura, Manager, and Ms. Betty Jane Decker, Supervisor.

On the morning of Thursday, December 1, all items of evidence were secured by Messrs. Hopper and Heinze and taken to the Cresco, Pa. central office where some of the sealed containers were opened and certain items of evidence were removed for testing. Messrs. Hopper and

Heinze maintained custody and control of all items of evidence throughout the testing procedure. At approximately 7:30 p.m., the tests were concluded and the items of evidence were returned to the containers and resealed. They were again taken to the Stroudsburg business office where Mr. Chura unlocked the building, the door to the business office, and the door to the money counting room. The sealed boxes remained in the money counting room until 11:00 a.m., of the following morning, Friday, December 2. At that time, Messrs. Hopper and Heinze transported the evidence to the Swiftwater Station of the Pennsylvania State Police. Trooper James R. Harris, Jr. then verified that all items were present and accepted custody. He acknowledged receipt by endorsements on the two PSP forms SP-4, previously referenced.

Item Identification

All item number references throughout our report relate to Pennsylvania State Police Property Record inventory no. 1205 (dated 22 Oct. 77) item numbers, except for magnetic recording cassettes. Since there were 39 cassettes (and two empty boxes inventoried as cassettes), a "Cassette Number" was marked on each to facilitate identification and maintain control. A cross reference list of "Cassette Numbers" to "Item Numbers" follows:

<u>Cassette Number</u>	<u>Property Record List of 22 Oct. 77 Item Number</u>	<u>Analysis Request List of 27 Oct. 77 Item Number</u>	<u>Analysis Request List of 1 Nov. 77 Item Number</u>
1	3	3	
2	3	3	
3	3	3	
4	3	3	
5	4	4	
6	4	4	
7	4	4	
8	4	4	
9	4	4	
10	4	4	
11	4	4	
12	4	4	
13	4	4	
14	4	4	
15	4	4	
16	4	4	

<u>Cassette Number</u>	<u>Property Record List of 22 Oct. 77 Item Number</u>	<u>Analysis Request List of 27 Oct. 77 Item Number</u>	<u>Analysis Request List of 1 Nov. 77 Item Number</u>
17	4	4	
18	4	4	
19 (empty box)	4	4	
20 (empty box)	4	4	
21	5	5	
22	5	5	
23	20	14	
24	20	14	
25	20	14	
26	20	14	
27	20	14	
28	20	14	
29	20	14	
30	20	14	
31	20	14	
32	27	21	
33	27	21	
34	27	21	
35	27	21	
36	27	21	
37	27	21	
38	30	24	
39	18		6
40	18		6
41	18		6

Report Organization

This report is divided into several parts as follows:

Part 1 - Covers the items of evidence relating to the "Processor Technology Sol Terminal Computer. This includes items 1 through 10 inclusive, and Cassette Numbers 1 through 22 inclusive. This examination was conducted by Mr. Richard A. Previte.

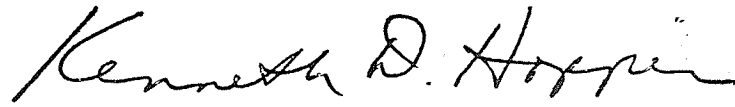
Part 2 - Covers the items of evidence relating to the Apple Computer. This includes items 14, 20 through 31 inclusive, and Cassette Numbers 23 through 41 inclusive. This examination was conducted by Mr. Paul Rabinowitz, Dr. Gerald S. Soloway, and Mr. Walter W. Heinze.

Part 3 - Covers items of evidence relating to "red box" coin telephone fraud device construction plans and

partially assembled devices. This relates to parts of items 18 and 20. This examination was conducted by Messrs. Kenneth D. Hopper and Walter W. Heinze.

Part 4 - Covers documentary items of evidence relating to methods of telephone toll fraud, privacy intrusion, toll evasion, and telephone network subversion. This examination was conducted by Mr. Kenneth D. Hopper.

Part 5 - Covers tests made at Cresco, Pa. central office utilizing items of evidence to (1) find password of customer remote access unit, and (2) place a fraudulent test call (customer authorization secured.) This test procedure was conducted by Messrs. Walter W. Heinze, Kenneth D. Hopper, Richard A. Previte, and Paul Rabinowitz.



Kenneth D. Hopper
Member of Technical Staff

HO-3331-KDH-al

Atts.
Sworn Statement
Part 1
Part 2
Part 3
Part 4
Part 5
(Sworn Statements appended to each)



Bell Laboratories

subject: Evidence Examination
File 39365-4
Pennsylvania State Police
Incident N6-39474

date: December 13, 1977
from: R. A. Previte

PART 1

Examination Report, Pennsylvania State Police Property Record Items 1-3, inclusive, Item 16, and including Cassette Numbers 1-22, inclusive.

Computer Hardware Description

The Sol Terminal (Item 6) is a complete micro-processor controlled computer. It utilizes the Intel 8030 Central Processor Unit (CPU) microprocessor. The processor can use a standard BASIC language to run programs. Standard cassette tapes provided by the Processor Technology Corporation and received for evidence investigation include;

- 1) BASIC (cassette 5)
- 2) Data Debugging System (DDS) - (cassettes 1,2, and 3)
- 3) Assembly Language System (ALS) - (cassettes 6,15)

The computer as received includes;

a) The main computer cabinet (serial number 305993 located beneath the memory board within the cabinet) contains the CPU and the means to connect 4 additional card packs with a common connector bus system. This cabinet includes a keyboard and interface connection to both a PDI (Parallel Data Interface) and SDI (Serial Data Interface) in addition to the connection for the video display and cassette recorder.

b) The computer came equipped with, two 3K ram boards manufactured by Industrial Micro Systems serial numbers, 20371 and 20373. Each memory board contains a strap which will memory map the ram in any of the 8K memory blocks from 0 through 7. When received the lower memory board in slide one was memory mapped for block 0 (0000-1999). The upper memory board in slide three was memory mapped for block 7 (E000-FFFF).

c) Panasonic Video Display (item 1), connected to the computer by a coaxial cable.

d) Panasonic Cassette Tape Recorder (item 2), connected to the computer by three cords;

- 1) Monitor
- 2) Record
- 3) Motor Control

Switch Settings on Terminal when received

These switches are located on the main computer board in the cabinet.

1) Baud Switch (S3)

8 switches 1 on, 2-8 off Set for 75 baud

2) Computer Serial Interface (S4) SWITCH

6 Switches	1 off	No effect not set for parity
	2 and 3 off	8 bit word length
	4 off	2 stop bits
	5 off	no parity
	6 off	FDX

3) Sense Switch (S2)

Switches 0 through 7 off Used to manually input data

Switch Change made for our Investigation

Switch S3, 1 moved to off, 4 moved to on

The switch was moved to permit the Serial Peripheral to function with a 300 baud computer system at the laboratories.

General System Operation

To operate the system, the user loads programs stored on the cassette tapes into the SOL Terminal resident ram memory. This is done via commands from the keyboard, located on the computer module. Once the programs are loaded, information may be added to the memory and the programs may be executed by the user when desired. Programs may be written in either assembly, machine or BASIC language. A BASIC interpreter is loaded into the 0 block of the ram memory via a cassette tape.

Cassette Tape Items (items 1 through 22, inclusive)

22 cassette tapes were examined. These items have been cataloged in the attached Appendix 1.

Documents Examined

ALS-8 user manual (part of item 8)

This manual explains the use of ALS (Assembly Language System) with the SOL computer. The manual was printed and distributed by the Processor Technology Corp. The manual looked essentially new and little used. There were no markings in this manual.

SOL Terminal Computer, Sol Systems Manual (item 7)

This manual explains the assembly, test, use and operation of the SOL Processor. The manual contained some markings on the printed text and within the drawing Section X, two added pencilled pages. One pencilled page explains an overvoltage fix. Another pencilled page (Appendix 2) is a schematic of a Digital Analog Converter (DAC), with terminal numbers corresponding to the PDI (Parallel Data Interface) of the SOL computer..

A INTEL 8080 Microcomputer System Users Manual (part of item 8)

A Dynamic Debugging System Manual, by Computer Mart (part of item 8)

A notebook containing handwriting (item 16)

Comments on this item can be found in Part 4 of the report.

Parallel Data Interface (PDI)

A DAC which was constructed according to the pencilled drawing found in item 7, was attached to the PDI to demonstrate its use with the program routines running in the SOL Processor. A brief explanation of the DAC follows:

A DAC is connected to all eight bits of the PDI. The DAC accepts a binary number from the output port as inputs and generates a corresponding DC voltage on the DAC output. The DAC in the schematic is of standard design. The DC voltage output is:

$$V = (I/255) \times 5$$

I is the binary value on the PDI. The output ports can generate any count from 0 to 255 (256) If the output bits are all zero the output voltage would be 0. If the output ports were all ones (255) the output voltage would be the highest possible voltage. In this way any voltage from 0 to 5 in 256 steps could be generated. By applying different binary numbers on the output port a changing voltage swing would, through an attached speaker, generate a specific tone.

User Operation

This BASIC language program, MF KEY, is located on tape 4. As the title indicates the MFKEY program can be used to emulate the MF KEYSET. The MFKEY program is loaded in memory locations 1A09 through 208F, and sets up the video display to access 6 called numbers in sender registers 1 through 6 (Appendix 3). The MFKEY program also contains the CALL routines to the MFGEN routine, loaded in memory locations 3F00 to 3FD1. The MFGEN program contains the SINE TABLE chart and the timing necessary to emulate the actual MF signals. The MFKEY program leaves the cursor in the home location, that is, immediately above the sender register column and all the way to the left of the video display.

With the MFKEY and MFGEN programs properly loaded and running a user enters telephone numbers which he may wish to call in the following manner;

1) Type the number of the sender register to which the telephone number is to be added. The cursor then locates itself in the first printing location, associated with the sender register location specified, to which the telephone number is to be entered.

2) Type the character "K" (key pulse), designating the key pulse tone which is to be the first tone sent, and the start of the called number sequence which is to be added.

3) Type the digits (maximum of 22) representing the telephone number to be accessed. The cursor advances as each number is entered.

4) Type the character "S" (start pulse), designating the closing or stop entering digits in this sender register, and generate the start tone, in the MF KEYSET emulation. The cursor then returns to the home position awaiting a new command.

The user emulates the MF KEYSET in the following manner (assuming that a DAC is connected to the PDI port);

1) At that time when the user identifies he has accessed the trunk facilities he depresses the "I" keyboard character.

2) The computer then generates the idle trunk tone 2500hz (5F), to seize control of the switching network.

- 3) The user types the number of the sender register corresponding to the telephone number which is to be emulated by the computer.
- 4) The cursor relocates itself to that sender register location.
- 5) The user types the character "T" (transmit).
- 6) The computer outputs the MF Keypulses, the KP, digits, and the Start pulse in sequence. As the digits are pulsed, the cursor follows, in order, the new digit numbers to be pulsed. When complete, the cursor returns to the home location.

Programs

The programs which were relevant to telephone system operation were located on tape 4. These programs are listed on appendix 4 and were the only ones examined.

MF KEYSET Emulator (Appendix 4)

With the MF KEYSET emulator and the MFGEN programs properly loaded the program starts by setting up the C (13 locations) frequency array and the R (143 locations) calling number array. By printing specific control characters the program then continues to write the instructions and the 6 register locations, which will accept the called numbers to be specified by the user. All this information is displayed on the video display, as shown on Appendix 3.

After writing the instructions and setting up the arrays the program runs in the MFGEN routine generating dummy frequencies (100-100) which do not affect the DAC and no tone is output. The program continues to run in this routine waiting for the insertion of keyboard input characters.

A number 1 through 6 is now entered by the user to choose a sender register where a telephone number is to be input. If any other character or digit is input via the keyboard, no change takes place and the program remains in the poll mode until a proper digit is input.

The user can now input a specific number relating to a specific sender register 1 through 6. When the computer receives this number the program is arranged to place the cursor in the first location associated with the specific 1 through 6 sender register. The user then inputs the character "K" which resets the register, if a previous number had been specified in that location, or opens the register to permit the user to enter a new called number. The character "K" is placed in that location and the cursor moves to the first digit location to be input. The user then inputs as many as 22 digits into the sender register location. The user closes the sender register by typing the character "S". The cursor now returns to the home position. If the user attempts to enter more than 22 digits the register will return a register overflow message. The user may enter in the same fashion, 5 additional numbers. The digits are stored in the 143 location R array. Each number is allotted 24 locations in the array.

The program continues to poll in the MFGEN routine waiting for new instructions. The next instruction which a user sends is the character "I". When a character "I" is received by the computer, the program is arranged to increase the time of generation to 1 second, by recognizing that only one tone is to be generated. The DAC then generates the 2600 hertz signal for that period of time. The user now selects the sender register location of the called number he wishes to generate by typing the number associated with that register. The cursor advances to that location and awaits the receipt of the character "T" before commencing the tone generation through the speaker.

MFGEN Program (Appendix 4)

This program also appears on cassette 4, under the title of MFGEN. The assembly program is shown. This program contains the SINE TABLE which is accessed by the program to generate the MF tones. This routine also acts as the program to which the peripherals are called. This routine is constantly running and generating the dummy 100/100 tone (no tone). The tone frequency which is to be generated is a function of the SINE TABLE and its spacing provided by the number called in the C array. By judiciously applying the correct number, thereby providing different voltages on the PDI port to the DAC, any number of frequencies may be generated. The single tone (SF) is generated for a 1 second duration in this program and the MF tones for 80ms durations.

Conclusions

The computer was set up in our laboratories at Holmdel, N.J. A DAC, which was built following the pencilled drawing located in the SOL System Manual and shown in Appendix 2, was attached to the PDI port. With the programs loaded and running, tones were generated. Some additional frequencies were seen on our frequency analyzer which interfered with the proper detection of some of the MF digits. The basic 2600 hertz (3F) was accepted by our equipment. A similar field test was set up in the Cresco, Pa, community dial office. With the SOL computer up and running with the programs of cassette tape 4, and associated with a telephone line, it was possible to take control of the switching network. Minor debugging would be necessary to clean up the frequencies for reliable operation.

Touch tone frequency generation is accomplished in much the same manner via the TTGEN program, also listed in appendix 4.

Some of the programs that were observed on other cassette tapes relate to TV cable switching and game routines, and did not appear to contain any telephone fraud related procedures.

PART 1

APPENDIX 1

Pennsylvania State Police
Incident N5-39474

R.A.PREVITE

REPORT

A total of 22 cassette items were received for evidence examination. The following lists these items, in numerical order, in which they appeared on the cassette tapes and the counter location of these programs.

Cassettes 1,2, 3

These three cassette tapes are debugging procedures and were not examined as to their content.

Cassette 4

Side 1

DACEK	A	000-013
blank		013-to end

Side 2

MOD7B	A	000-013
MFGEN	A	013-014
MFGEN	A	014-025
MFGEN	A	025-026
MFGEN	A	026-038
MFGEN	O	038-040
MFGEN	O	040-041
BASIC	B	041-057
MFGEN	B	057-058
MFGEN	B	058-059
TRASH	B	059-062
MFKEY	B	062-065
MFKEY	B	065-069
MFKEY	B	069-074
MFKEY	B	074-079
TTGEN		079-080
TTGEN		080-092
Solid Tone		092-200
Silent		200-272(end)

This cassette tape contained all the programs which were loaded into and ran in the SOL computer and which appeared to contain the programs capable of generating TT and MF tones.

Cassette 5

This tape is a standard tape purchased from Processor Tech Corp, the manufacturer of the SOL computer. The title is "Standard Basic/5".

Title	Type	Tape Counter
BASIC5		000-033
ATCH	B	033-040
LUNAR	B	040-043
END		043-049

This is a standard tape, and included a number of game programs. The rest of the tape was not examined.

Side 2

This side was audibly reviewed and determined to contain data, but was not loaded and run on the SOL computer.

Cassette 6

This tape is another standard tape provided by the Processor Technology Corp. The title of the tape is "ALS"

Side 1

This contained the standard ALS-3 data for Sol computer entry. This side was not reviewed.

Side 2

No data was heard on the first foot of this tape. After that silent period a solid tone was heard to tape count 017. From this point data pulses were audibly heard for the rest of the tape. While data was present on the tape it was not loaded and run on the SOL computer.

Cassettes 7 and 8

Cassette tapes 7 and 8 included music for the duration of the tape.

Cassette 9

This cassette tape appears to be a games tape. The following is a list of the files on this tape.

Title	Type	Tape Count
CBI	C	000-008
TWINT	C	008-011
BEADS	C	011-013
SPLAT	C	013-014
TWINT	C	014-021
TWINT	C	021-027
DUMP	C	027-029
DDS	D	029-040
BSORT	@	040-042
ALS*	@	042-062
ALS3*	@	062-082
Blank tape		082-278

Side 2

MOD7B	A	000-012
DDS	D	012-022
EPR0M	S	022-025
RSORT	A	025-026
RSORT	A	026-028
RSORT	@	028-029
EPR0M	A	029-039
EPR0M	A	039-046
LIFE1	A	046-048
LIFE1	A	048-056
LIFE2	A	056-057
LIFE2	A	057-066
LIFE3	A	066-068
LIFE3	A	068-077
LIFE3		077-078
LIFE3		078-087
		087-end (Blank no data)

Cassette 10

Side 1

Blank no data

Side2

Tone audibly heard for approx first 2 feet of tape.
No tone heard from there to the end of the tape.

Cassette 11

Side 1

This tape included a number of programs associated with a cable TV switching method. It does not appear to contain any data used to complete fraudulent telephone connection through the telephone network.

Title	Type	Tape Counter
CBI	C	000-008
TWIN	C	008-014
TEXT		014-016
NONE		016-019
TEXT2		019-023
TEXT3		023-028
TEXT4		028-033
MODEL		033-036
MESSG		036-044
TEXT5		044-050
MESS8		050-060
TEXT8		060-066
DEMO1		066-076
DEMO2		076-087
DEMO-		087-097
		097-end (Blank)

Side 2

BASIC	B	000-016
PRSUB	B	016-017
BORDR	B	017-019
		019-100 (Tone)
		100-end (Blank)

Cassette1?

Side 1

This cassette tape included games on both sides of the tape.

Title	Type	Tape Counter	
BASIC	B	000-011	
MFCHS	B	011-013	
NUMBER	B	013-015	
BAGLE	B	015-018	
CLOCK	B	018-019	
DSPLY	B	019-020	
LIFE	B	020-025	
BAGLE	B	025-028	
BAGNW	B	028-036	
DSPLY	S	036-040	
HANGO	B	040-044	
HANGO	B	044-046	ERROR
LIFE	B	046-040	
HANGP	B	050-053	
HGMAN	B	058-064	
HANGO	B	064-070	
HANGU		070-076	
HANGU		076-081	
HANGU		081-086	

Examination ended at this point. Appear to be entirely games.

Side 2

LIFE	@	000-002	
LIFE	A	002-003	
LIFE	A	003-003	
LIFE	A	003-022	ERROR
LIFE	A	022-024	ERROR
END	?	024-029	
		029-end (Blank)	

NOTE: An ERROR appears on the screen when a particular program is unsuccessfully loaded into memory.

Cassette 13

Side 1

This side of the tape was blank, no tone or signals observed.

Side 2

This side of the tape contained a number of games programs.

Title	Type	Tape Counter	
DDS	D	000-002	ERROR
ASSM	A	002-012	
DDS	D	013-017	
SLIFE	A	017-023	
SLIFE	A	023-027	
LIFE	A	-27-028	
LIFE	A	028-034	
LIFE	A	034-036	
LIFE	A	036-047	

Music was audibly heard on this tape from location 047 to 103. From 108 to the end sounds like an open mike. No discernable sounds could be understood. It was very likely done in the erase procedure.

Cassette 14

No data or tone heard on either side of this tape. Quite likely, a new tape.

Cassette 15

Side 1

Title	Type	Tape Counter	
HGMAN	G	000-003	ERROR
BASIC5"		003-026	

From this point until the last two feet of tape the BASIC5 program is written and re-written. The last two feet to the end of the tape is blank.

Side 2

This side of the tape contains a number of games programs.

Title	Type	Tape Counter	
HGMAN	G	000-019	ERROR
BASIC B		019-040	
NUMBER	B	040-044	
LUNAR	B	044-050	ERROR
BAGLE	B	050-055	
BAGLE	B	056-061	
CLOCK	B	061-062	
DSPLY	B	062-064	
DSPLY	B	064-065	
LIFE	B	065-073	
HTEMP	B	073-077	
ATT		077-086	
HAMRB	B	086-095	
HAMRB	B	095-104	
		104-end (Tone)	

Cassette 15

Side 1

Standard ALS tape.

Title	Type	Tape Counter	
ALS.1	@	000-003	
ALS.1	@	003-031	
ALS.2	@	031-046	
ALS.2	@	046-052	
ALS.3*	@	052-057	ERROR
ALS.3*	@	057-095	
		095-end (Blank)	

Side 2

No data or tone on this side of the tape

Cassette 17

Side 1

Title	Type	Tape Counter
HGMAN	G	000-025
BASIC5"		025-026

From here until tape count 75 all BASIC5" programs are written and re-written. From 75 to the end no data or tones heard.

Side 2

No tone or data heard on this side of the tape.

Cassette 13

Side 1

Title	Type	Tape Counter
BASIC	B	000-012
BAGEL	B	012-017
CLOCK	B	017-019

From this point until tape count 39, The program CLOCK of type B is written and re-written 8 times. Each time it is received with an error. Interestingly this portion of the tape includes music. The data is written over the music. From tape count 39 to the end no tone, music, or data is found.

Side 2

This tape is plank, no tone, data or music for its entire length until the last two feet. At this point data is audible, but was not accessible to the SOL computer.

Cassettes 19 and 20

These items were found to be empty boxes and contained no cassette tapes.

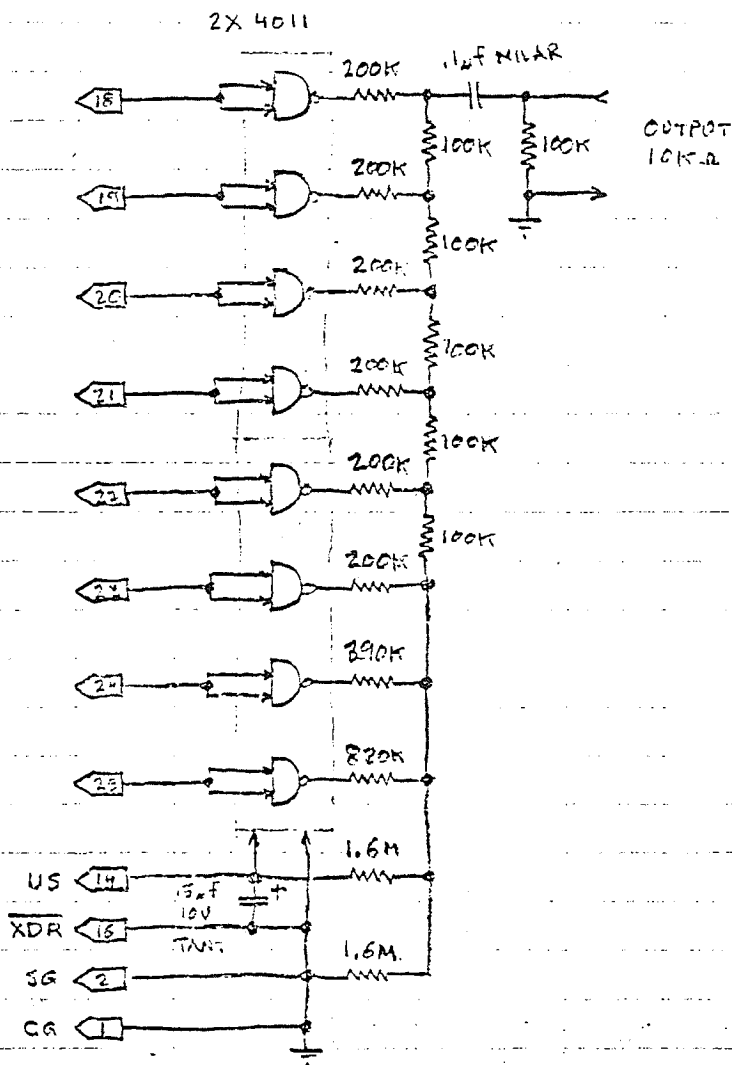
Cassettes 21 and 22

These two cassette tapes included only music .

Pennsylvania State
Police Incident N6-39474

R. A. PREVITE

SELF POWERED 8-BIT DAC



EVIDENCE
 Part of PSP item 7
 Case N6-39474
 100K

23

PART 1

APPENDIX 3

Pennsylvania State Police
Incident N6-39474

R.A.PREVITE

The following illustration is displayed on the video monitor when the MF programs, of cassette tape 4, are run on the computer. The user must add the telephone numbers he wishes to pulse after each register.

MF Keyset Software Emulator

Vocabulary: { K (KP), 0,1,2,3,4,5,6,7,8,9, S (ST), I (Tk Idle) }

Immediate Commands: I => Idle Trunk

nK => Reset & open Sender Register n

S => Close current Sender Register

nF => Transmit Sender Register n

Sender Register 1:

Sender Register 2:

Sender Register 3:

Sender Register 4:

Sender Register 5:

Sender Register 6:

PART 1

APPENDIX 4

Pennsylvania State Police
Incident N6-39474

R.A.PREVITE

This appendix lists several programs found on cassette tape 4. Tape headings have been provided for identification, otherwise the listing are printed exactly as found without changes or editing of any kind. The headings have been underlined. Many of these programs are referenced in the text.

The following programs of cassette tape 4 are listed herein:

mfgen 040-041
mfgex 057-058
mfgex 058-059
trash 059-062
mfkey 062-065
mfkey 069-074
ttgen 030--92

Cassette 4 includes other programs, which have not been listed.

MFGEN tape 4 tape count 010-041

LIST

12337 00 * MFGEN is a multi-frequency tone signal
 12337 10 * generator for use with 1 or 2 out of N
 12337 20 * signaling codes. It drives an 8 bit,
 12337 30 * inverted MSB, unlocked DAC on port
 12337 40 * FD. A system clock frequency of 2.045
 12337 50 * MHz and no wait states are assumed.
 12337 50 * Tonal frequency in 100's Hz is pass-
 12337 70 * sed in reg C and tone2 frequency or
 12337 80 * 0 in reg B. A 30ms 2 tone burst or 1 S
 12337 90 * 1 tone burst is generated. If a key
 12503 00 * was struck during tone generation,
 12503 10 * its ASCII value, else 0, is returned
 12503 20 * in rj 3C. All other reg.s are saved.
 12503 30 *
 12503 40 *
 12503 40 *
 12338 00 * Sine function table. MFGEN must be
 12338 10 * assembled so that SINTB starts at a
 12338 20 * 256 byte page boundary.
 12333 30 SINTB DB 0
 12333 40 DB 3
 12333 50 DB 6
 12333 50 DB 9
 12333 70 DB 12
 12333 80 DB 15
 12333 90 DB 18
 12504 00 DB 21

APP4-3

12594 10 DB 24
12594 20 DB 27
12594 30 DB 30
12594 40 DB 32
12594 50 DB 35
12594 60 DB 38
12594 70 DB 40
12594 80 DB 42
12594 90 DB 45
12850 00 DB 47
12850 10 DB 49
12850 20 DB 51
12850 30 DB 52
12850 40 DB 54
12850 50 DB 56
12850 60 DB 57
12850 70 DB 58
12850 80 DB 59
12850 90 DB 60
13106 00 DB 61
13106 10 DB 62
13106 20 DB 62
13106 30 DB 63
13106 40 DB 63
13106 50 DB 63
13106 60 DB 63
13106 70 DB 63

13106	30	DB	62
13106	90	DB	62
13362	00	DB	61
13362	10	DB	60
13362	20	DB	59
13362	30	DB	58
13362	40	DB	57
13362	50	DB	56
13362	60	DB	54
13362	70	DB	52
13362	80	DB	51
13362	90	DB	49
13618	00	DB	47
13618	10	DB	45
13618	20	DB	42
13618	30	DB	40
13618	40	DB	38
13618	50	DB	35
13618	60	DB	32
13618	70	DB	30
13618	80	DB	27
13618	90	DB	24
13874	00	DB	21
13874	10	DB	18
13874	20	DB	15
13874	30	DB	12
13874	40	DB	9
13874	50	DB	5

13874 60 DB 3
13874 70 DB 0
13874 80 DB -3
13874 90 DB -6
14130 00 DB -9
14130 10 DB -12
14130 20 DB -15
14130 30 DB -18
14130 40 DB -21
14130 50 DB -24
14130 60 DB -27
14130 70 DB -30
14130 80 DB -32
14130 90 DB -35
14386 00 DB -38
14386 10 DB -40
14386 20 DB -42
14386 30 DB -45
14386 40 DB -47
14386 50 DB -49
14386 60 DB -51
14386 70 DB -52
14386 80 DB -54
14386 90 DB -56
14642 00 DB -57
14642 10 DB -58
14642 20 DB -59
14642 30 DB -60

14642	40	DB	-61
14642	50	DB	-62
14642	60	DB	-62
14642	70	DB	-63
14642	80	DB	-63
14642	90	DB	-63
12339	00	DB	-63
12339	10	DB	-63
12339	20	DB	-62
12339	30	DB	-62
12339	40	DB	-61
12339	50	DB	-60
12339	60	DB	-59
12339	70	DB	-58
12339	80	DB	-57
12339	90	DB	-56
12595	00	DB	-54
12595	10	DB	-52
12595	20	DB	-51
12595	30	DB	-49
12595	40	DB	-47
12595	50	DB	-45
12595	60	DB	-42
12595	70	DB	-40
12595	80	DB	-38
12595	90	DB	-35
12851	00	DB	-32
12851	10	DB	-30

APP4-7

```

12851 20  DB -27
12851 30  DB -24
12851 40  DB -21
12851 50  DB -18
12851 60  DB -15
12851 70  DB -12
12851 80  DB -9
12851 90  DB -6
13107 00  DB -3  END OF 128 SAMPLE SINE TABLE
13107 10  *
13107 20  *
12340 00  *  MFGEN - Main routine entry point
12340 10  MFGEN  PUSH  PSW  SAVE MACHINE REG.S .
12340 20          PUSH  D
12340 30          PUSH  H
12340 35          DI          DISABLE INTERRUPTS
12340 40          LXI  H,0500H  SET 80 MS TIMEOUT
12340 50          INR  B          IF SINGLE TONE
12340 60          DCR  B
12340 70          JNZ  MULTI
12340 80          LXI  H,3300H  SET 1 SEC TIMEOUT
12340 90  MULTI  PUSH  H          PHANTOM TIME COUNT
12596 00          LXI  D,SINTB  SINTB BASE ADDR
12596 10          LXI  H,SINTB
12596 20  LOOP  MOV  A,E  ADV TONE1 PHASE
12596 30          ADD  C          TONE1 FREQUENCY
12596 40          ANI  7FH  MASK TO 7 BITS
12596 50          MOV  E,A

```

```

12596 60      MOV      A,L      ADV TONE2 PHASE
12596 70      ADD      B
12596 80      ANI      7FH
12596 90      MOV      L,A
12852 00      LDAX   D      TONE1 SAMPLE
12852 10      ADD      M      TONE2 SAMPLE
12852 20      XRI      80H     DAC INVERTS MSB
12852 30      OUT     DAC     OUTPUT SAMPLE
12852 40      XTHL             GET TIME COUNT
12852 50      DCX    H      DECREMENT
12852 60      INR    H      TEST FOR TIMEOUT
12852 70      DCR    H
12852 80      XTHL             SAVE TIME COUNT
12852 90      PUSH   H      DELAY UNTIL TOTAL
13108 00      POP    H      LOOP LENGTH=160
13108 10      NOP                    MACHINE CYCLES
13108 20 *    Note: Imperical tuning=> 159 cycles
13108 30      JNZ    LOOP     IF NOT TIMED OUT
13108 40      LXI    B,0     CLEAR RP BC
13108 50      IN     AUXST   GET KEYBD STATUS
13108 60      ANI    KSBIT
13108 70      JNZ    NOKEY   IF KEY STRUCK
13108 80      IN     KEYBD   INPUT CHARACTER
13108 90 NOKEY POP    H      CHUCK TIME COUNT
13364 00      POP    H      RESTORE MACHINE
13364 10      POP    D      REGISTERS
13364 20      POP    PSW
13364 30      RET                    RETURN
13364 40 *    End of program

```

```

13364 50 *
13364 60 *
12341 00 DAC      EQU      0FDH   DAC = PORT FD
12341 10 AUXST   EQU      0FAH   AUX STATUS=PORT FA
12341 20 KSBIT   EQU      001H   KEYBD STATUS BIT=0
12341 30 KEYBD   EQU      0FCH   KEYBOARD = PORT FC
12341 40 *      End of file .....
12341 50 *
12341 60 *
12342 00 *      Test programs
12342 10 *1) Single 600 Hz tone
12342 20          LXI      B,6
12342 30          CALL    MFGEN
12342 40          RET
12342 50 *2) Double 600/1000 Hz tone
12342 60          LXI      B,0A06H
12342 70          CALL    MFGEN
12342 80          RET
12342 90 *      End of test

```

mfgex on tape 4 tape count 057-053

LIST

```

100 D=ARG(6): IF CALL(16256)<128 THEN 100
200 FOR I=0 TO 26
210     FOR J=0 TO 18
220         D=ARG(256*J+I)
230         PRINT CALL(16256);
240     NEXT J

```

250 PRINT

260 NEXT I

mfgex on tape 4 tape counter 053-059 LIST

100 D=ARG(6): IF CALL(16256)<128 THEN 100

200 FOR I=0 TO 26

210 FOR J=0 TO 18

220 D=ARG(256*J+I)

230 PRINT CALL(16256);

240 NEXT J

250 PRINT

260 NEXT I

trash on tape 4 tape counter 059-052

LIST

190 DIM C(13)

200 C(0)=256*13+15

210 C(1)=256*7+9

220 C(2)=256*7+11

230 C(3)=256*9+11

240 C(4)=256*7+13

250 C(5)=256*9+13

260 C(6)=256*11+13

270 C(7)=256*7+15

280 C(8)=256*9+15

290 C(9)=256*11+15

310 C(10)=256*11+17

320 C(11)=256*15+17

APP4-11

```

330 C(12)=26
340 C(13)=256*1+1
380 FOR I=0 TO 199: C=CALL(0*ARG(C(13))+16256): NEXT
400 FOR I=0 TO 20: READ D: C=CALL(0*ARG(C(D))+16256)
410     IF D=11 THEN I=20
420 NEXT
500 DATA 12,10,4,1,5,9,6,4,9,0,4,1,11

```

mfkey on tape 4 tape counter 062-065

LIST

```

10  REM MFKEY    MF KEYSET SOFTWARE EMULATOR
20  REM
30  REM
40  SET S=0: DIM C(13): DIM R(144)
50  PRINT "&K&Z                MF Keyset Software
Emulator"
60  PRINT "&ZVocabulary:{K(KP), 0, 1, 2, 3, 4, 5, 6,
7, 8, ";
70  PRINT " S(ST), I(Tk Idle)}";
80  PRINT "Immediate Commands:  I => Idle Trunk"
90  PRINT "                    nK => Reset  && open
Sender ";
100 PRINT "Register n"
110 PRINT "                    S => Close current
Sender ";
120 PRINT "Register"
130 PRINT "                    nT => Transmit  Sender
";
140 PRINT "Register n"
150 PRINT : FOR I=1 TO 6
160     PRINT "    Sender Register";I;":"

```



```
200 C(0)=256*13+15
210 C(1)=256*7+9
220 C(2)=256*7+11
230 C(3)=256*9+11
240 C(4)=256*7+13
250 C(5)=256*9+13
260 C(6)=256*11+13
270 C(7)=256*7+15
280 C(8)=256*9+15
290 C(9)=256*11+15
310 C(10)=256*11+17
320 C(11)=256*15+17
330 C(12)=26
340 C(13)=256*1+1
350 SET S=5: PRINT "&[&B&C": STOP
380 FOR I=0 TO 199: C=CALL(0*ARG(C(13))+16256): NEXT
400 FOR I=0 TO 20: READ D: C=CALL(0*ARG(C(D))+16256)
410     IF D=11 THEN I=20
420 NEXT
```

mfkey on tape 4 tape counter 069-074

LIST

```
10 REM MFKEY MF KEYSET SOFTWARE EMULATOR
20 REM
30 REM
40 SET S=0: DIM C(13): DIM R(143)
50 PRINT "&K&Z MF Keyset Software
Emulator"
```

```
60      PRINT      "&ZVocabulary:{      K      (KP),
0,1,2,3,4,5,6,7,8,9,";

70  PRINT " S (ST), I (Tk Idle) }"

80  PRINT "Immediate Commands:  I => Idle Trunk"

90  PRINT "      nK => Reset  &&  open
Sender ";

100 PRINT "Register n"

110 PRINT "      S => Close current
Sender ";

120 PRINT "Register"

130 PRINT "      nT => Transmit  Sender
";

140 PRINT "Register n"

150 PRINT : FOR I=1 TO 6

160     PRINT "      Sender Register";I;":"

170 NEXT

190 FOR I=0 TO 5: R(24*I)=11: NEXT

200 C(0)=256*13+15

210 C(1)=256*7+9

220 C(2)=256*7+11

230 C(3)=256*9+11

240 C(4)=256*7+13

250 C(5)=256*9+13

260 C(6)=256*11+13

270 C(7)=256*7+15

280 C(8)=256*9+15

290 C(9)=256*11+15

310 C(10)=256*11+17

320 C(11)=256*15+17

330 C(12)=26

340 C(13)=256*1+1

350 REM COMMAND LEVEL INPUT ROUTINE ENTRY
```

```
&[&B&H";
365 GOSUB 970: IF C<49 THEN 365
370 IF C>54 THEN 365
380 REM REGISTER COMMAND INTERPRETER
390 N=24*(C-49): PRINT "&[&A&X&[&B&H";
400 FOR I=1 TO C-48: PRINT "&Z";: NEXT
410 M=0: PRINT "&[&A&X";
415 GOSUB 970: IF C=84 THEN 500
420 IF C<>75 THEN 415
430 IF C<128 THEN D=CALL(0*ARG(256*C)+49177)
440 IF C=75 THEN R(N+M)=10: M=1: PRINT
"&[&A&Y&M&[&A&Y";
450 IF C=83 THEN R(N+M)=11: GOTO 360
460 IF C>47 THEN IF C<58 THEN R(N+M)=C-48: M=M+1
470 IF M<24 THEN GOSUB 970: GOTO 430
480 M=0: R(N)=11: PRINT "&[&A&X- reg. overflow -&M";:
GOTO 410
500 REM TRANSMIT ROUTINE
510 FOR I=0 TO 23: PRINT "&S";
520 C=CALL(0*ARG(C(R(N+I)))+16256)
530 IF C=73 THEN C=CALL(0*ARG(C(12))+16256): I=23
540 IF R(N+I)=11 THEN I=23
550 NEXT : GOTO 360
960 REM GET CHARACTER ROUTINE
970 C=CALL(0*ARG(C(13))+16256)
980 IF C=73 THEN C=CALL(0*ARG(C(12))+16256): GOTO 980
990 IF C<>128 THEN RETURN
999 PRINT "&K";: SET S=5: END
```

ttgen on tape 4 tape counter 080-092

LIST

12337 00 * TTGEN is a multi-frequency tone signal
12337 10 * generator for use with 2 out of N
12337 20 * signaling codes. It drives an 8 bit,
12337 30 * inverted MSB, unclocked DAC on port
12337 40 * FD. A system clock frequency of 2.045
12337 50 * MHz and no wait states are assumed.
12337 60 * Tone1 frequency in 75/16's Hz is
12337 70 * passed in reg C and tone2 in 75/8's
12337 80 * Hz in reg B. A 80mS tone burst is
12337 90 * generated. If a key was struck dur-
12593 00 * ing tone generation, its ASCII value,
12593 10 * else 0, is returned in rp BC. All
12593 20 * other reg.s are saved.
12593 30 *
12593 40 *
12338 00 * Sine function table. TTGEN must be
12338 10 * assembled so that SINTB starts at a
12338 20 * 256 byte page boundary.
12338 30 SINTB DB 0
12338 40 DB 3
12338 50 DB 6
12338 60 DB 9
12338 70 DB 12
12338 80 DB 15

12338	90	DB	18
12594	00	DB	21
12594	10	DB	24
12594	20	DB	27
12594	30	DB	30
12594	40	DB	32
12594	50	DB	35
12594	60	DB	38
12594	70	DB	40
12594	80	DB	42
12594	90	DB	45
12850	00	DB	47
12850	10	DB	49
12850	20	DB	51
12850	30	DB	52
12850	40	DB	54
12850	50	DB	56
12850	60	DB	57
12850	70	DB	58
12850	80	DB	59
12850	90	DB	60
13106	00	DB	61
13106	10	DB	62
13106	20	DB	62
13106	30	DB	63
13106	40	DB	63
13106	50	DB	63
13106	60	DB	63

13106	70	DB	63
13106	80	DB	62
13106	90	DB	62
13362	00	DB	61
13362	10	DB	60
13362	20	DB	59
13362	30	DB	58
13362	40	DB	57
13362	50	DB	56
13362	60	DB	54
13362	70	DB	52
13362	80	DB	51
13362	90	DB	49
13618	00	DB	47
13618	10	DB	45
13618	20	DB	42
13618	30	DB	40
13618	40	DB	38
13618	50	DB	35
13618	60	DB	32
13618	70	DB	30
13618	80	DB	27
13618	90	DB	24
13874	00	DB	21
13874	10	DB	18
13874	20	DB	15
13874	30	DB	12
13374	40	DB	9

13874	50	DB 6
13874	60	DB 3
13874	70	DB 0
13874	80	DB -3
13874	90	DB -6
14130	00	DB -9
14130	10	DB -12
14130	20	DB -15
14130	30	DB -18
14130	40	DB -21
14130	50	DB -24
14130	60	DB -27
14130	70	DB -30
14130	80	DB -32
14130	90	DB -35
14386	00	DB -38
14386	10	DB -40
14386	20	DB -42
14386	30	DB -45
14386	40	DB -47
14386	50	DB -49
14386	60	DB -51
14386	70	DB -52
14386	80	DB -54
14386	90	DB -56
14642	00	DB -57
14642	10	DB -58
14642	20	DB -59

14642	30	DB	-60
14642	40	DB	-61
14642	50	DB	-62
14642	60	DB	-62
14642	70	DB	-63
14642	80	DB	-63
14642	90	DB	-63
12339	00	DB	-63
12339	10	DB	-63
12339	20	DB	-62
12339	30	DB	-62
12339	40	DB	-61
12339	50	DB	-60
12339	60	DB	-59
12339	70	DB	-58
12339	80	DB	-57
12339	90	DB	-56
12595	00	DB	-54
12595	10	DB	-52
12595	20	DB	-51
12595	30	DB	-49
12595	40	DB	-47
12595	50	DB	-45
12595	60	DB	-42
12595	70	DB	-40
12595	80	DB	-38
12595	90	DB	-35
12851	00	DB	-32
12851	10	DB	-30
12851	20	DB	-27


```
12851 30 DB -24
12851 40 DB -21
12851 50 DB -18
12851 60 DB -15
12851 70 DB -12
12851 80 DB -9
12851 90 DB -6
13107 00 DB -3 END OF 128 SAMPLE SINE TABLE
13107 10 *
13107 20 *
12340 00 * TTGEN - Main routine entry point
12340 10 TTGEN PUSH PSW SAVE MACHINE REG.S
12340 20 PUSH D
12340 30 PUSH H
12340 40 DI DISABLE INTERRUPTS
12340 50 MOV E,C SCALE TONE1 FREQ
12340 55 MVI D,0
12340 60 MOV L,E
12340 65 MOV H,D
12340 70 DAD H MULTIPLY BY 3/64
12340 75 DAD D
12340 80 DAD H
12340 85 DAD H
12340 90 SHLD FREQ1+1 TONE1 INCRE
12340 95 MOV E,B SCALE TONE2 FREQ
12596 00 MVI D,0
```

```

12596 15      MOV      L,E
12596 20      MOV      H,D
12596 25      DAD      H          MULTIPLY BY 3/128
12596 30      DAD      D
12596 35      DAD      H
12596 40      DAD      H
12596 45      DAD      H
12596 50      SHLD    FREQ2+1  TONE2 INCRE
12596 55      LXI     B,0500H  SET 80 MS TIMEOUT
12596 60      LXI     H,0      INITIALIZE PHASE
12596 65      PUSH    H
12596 70      LOOP
12596 75      FREQ1  LXI     D,0      ADV TONE1 PHASE
12596 80      DAD      D
12596 85      XTHL
12852 00      LDAX   D          TONE1 SAMPLE
12852 10      ADD     M          TONE2 SAMPLE
12852 20      XRI     80H      DAC INVERTS MSB
12852 30      OUT     DAC        OUTPUT SAMPLE
12852 50      DCX    B          DECR TIME COUNT
12852 60      INR    B          TEST FOR TIMEOUT
12852 70      DCR    B
13108 20      *      Note: Imperical tuning=> 159 cycles
13108 30      JNZ    LOOP    IF NOT TIMED OUT
13108 50      IN     AUXST  GET KEYBD STATUS
13108 60      ANI    KSBIT
13108 70      JNZ    NOKEY   IF KEY STRUCK
13108 80      IN     KEYBD   INPUT CHARACTER

```

APP4-22

```
13108 85          MOV    C,A
13108 90 NOKEY  POP    H      CHUCK TIME COUNT
13364 00          POP    H      RESTORE MACHINE
13364 10          POP    D      REGISTERS
13364 20          POP    PSW
13364 30          RET              RETURN
13364 40 *      End of program
13364 50 *
13364 60 *
12341 00 DAC      EQU    0FDH   DAC = PORT FD
12341 10 AUXST   EQU    0FAH   AUX STATUS=PORT FA
12341 20 KSBIT   EQU    001H   KEYBD STATUS BIT=0
12341 30 KEYBD   EQU    0FCH   KEYBOARD = PORT FC
12341 40 *      End of file .....
```



Bell Laboratories

subject: Evidence Examination
File 39865-4
Pennsylvania State Police
Incident N6-39474

date: December 9, 1977

from: P. Rabinowitz
G. S. Soloway
W. W. Heinze

PART 2

Examination Report, Pennsylvania State Police Property
Record Items 14, 20 - 31, inclusive, and including
Cassette Numbers 23 - 41, inclusive.

APPLE COMPUTER SYSTEM DESCRIPTION

Hardware Description

The Apple Computer system used for driving the telephone line circuit is comprised of the following hardware:

Item 21: Single Board Apple Computer Containing:

- a. 6502 Microprocessor
- b. 32K Bytes of Random Access Memory (RAM)
- c. 8K Bytes of Read Only Memory (ROM)
- d. 2K Bytes of Programmable Read Only Memory
- e. ASCII Keyboard Interface
- f. Television Set Display Interface
- g. Cassette Tape Recorder Interface
- h. Eight Connectors for General Purpose Input-Output Control

Item 29: Apple Computer Power Supply (+12, +5V and -5v)

Item 23: Apple Computer ASCII Keyboard

Item 31: General Electric T.V. Set

Item 30: Panasonic Tape Recorder

The above items provide a complete and self-contained small computer system. Functionally, items 21 and 29 provide system control capability, item 23 provides system input and command capability, item 31 provides system display and output capability, and item 30 provides off-line storage capability.

Operational Overview

In order to operate the system, the user loads programs stored on the cassette tapes into the Apple Computer memory. This is done via command from the keyboard into the resident ROM monitor provided with the computer. Once the programs are loaded, they can be executed and/or modified as desired by the user. Subsequently, the modified version can be written back onto the cassette tapes for later use.

Programs can be written in either assembly language or BASIC language. Assembler functions are provided by the system monitor and BASIC interpreter functions are provided by a resident 6K program provided with the Apple Computer. Among the many functions provided by this version of BASIC, is the ability to CALL assembly language programs.

A detailed functional description of the hardware and software system capabilities can be found in the "Apple II" manual (item 14 of the evidence).

TELEPHONE LINE INTERFACE CIRCUITOverview

A handwired printed circuit board (item 24) plugged into peripheral connector J4 of the Apple computer at the time of the seizure was also connected to the telephone line. This circuit board, measuring 3 x 7 inches contains several integrated circuits, a transformer, a relay, a transistor, many resistors, and capacitors. A detailed analysis of this board has revealed it to be a sophisticated telephone line interface circuit for the Apple computer.

Circuit Function

This circuit board was found to perform several telephony-related functions. Switchhook and dial pulsing are performed by having the computer control a relay in series with the telephone set. Generation of tones is accomplished with the use of a digital-to-analog converter integrated circuit. The presence or absence of various tones, (dial tone, trouble alerting, etc.,) can be detected by the use of a phase-locked-loop integrated circuit. Up to 256 different frequencies may be detected under program control.

Circuit Description

A signal flow diagram for the circuitry contained on this handwired board is shown in Figure 1. A detailed parts list is included as Table 1. A description of the important circuit components and their operation follows.

Transformer X1 is used to terminate the telephone line and isolate it from the remainder of the circuitry. Relay K1 is used for switchhook control and dial pulsing. Its contact is in series with the telephone line and the primary of transformer X1. The relay is driven by transistor Q1, which in turn, is turned on and off from pin 11 of IC-C4. This integrated circuit is known as an addressable latch. When the computer processes a read instruction for memory location C09D (Apple BASIC address: -16227) the relay closes, making the telephone line appear off-hook. (C09X is the address of peripheral connector J4; D activates IC-C4.) A read instruction for memory location C09C (Apple BASIC address: -16228) causes the relay to open (telephone line on-hook or dial pulse).

Tone generation is accomplished using part of IC-D3, and ICs C2, D2, D1, and C1. IC-D3 decodes the address of

latches C2 and D2 (C09A). These latches store the information taken from the computer's data bus and presents it to the digital-to-analog (D/A) converter (IC-D1). The D/A converter uses the 8-bit binary input signal from the latches and converts it to the appropriate voltages at its outputs. The two complimentary outputs may assume any one of 256 different voltages, depending on the state of the eight input leads. The two outputs of the D/A converter are then used as differential inputs to IC-C1, which is an operational amplifier. The operational amplifier and its associated circuitry serve to filter and amplify the outputs of the D/A converter. The output of the amplifier goes to the secondary of the line coupling transformer X1 for transmission onto the telephone line.

Signals received on the telephone line are coupled by transformer X1 to dual amplifiers contained in IC-A1. The first stage, whose inputs are pins 2 and 3 and whose output is pin 1, contains both manual and automatic gain control (AGC). Its output goes to an energy detecting circuit composed of diode D1, resistor R8 and capacitor C4. The output of the energy detector is used for AGC, and is also input to the second amplifier in IC-A1 which acts as a comparator. The output of the comparator is thus an indication of whether any signals are present on the telephone line or not.

The output of the first amplifier in IC-A1 is also used as an input to IC-B4, which is a phase-locked loop signal detector.

This device is capable of detecting tones of frequencies which fall in a range determined by the value of resistors and capacitors connected to it. In this circuit, a fixed capacitance was connected, however, the resistance is selected under computer control to be any of 256 different values. This is accomplished by having eight resistors (R22-R29) of different values each connected in series with an analog switch. Four analog switches are contained in both IC-A3 and IC-B3. The correct value of total resistance is determined by the parallel combination of the resistances in series with the closed analog switches. The analog switches are controlled from ICs A2 and B2, which are each quad latches. They will latch data appearing on the data bus when the appropriate address, C099 (Apple BASIC -16231) is written.

The presence or absence of telephone line energy, and the presence or absence of the selected frequency range can

be read by the computer with the aid of IC-C3. This device is an eight-input multiplexer. Two of its inputs are the above described signals. The computer can determine the state of these signals by read instructions to appropriate memory addresses, C092 (Apple BASIC -16238) for the tone detector and C091 (Apple BASIC -16239) for the energy detector.

Operational Experience

This circuit board was connected to the Apple computer and the telephone line in the same way that it was when seized. It has been used extensively at Bell Laboratories, Holmdel, N.J., while running programs described elsewhere in this document. Its operation has been predictable and it has been found to operate as previously described. The programs executed in the laboratory have used this interface circuit to perform switchhook control, dial pulsing, TOUCH-TONE® dialing, generation of single frequency and multifrequency tones, and for tone detection.

METHOD OF TONE GENERATIONOverview

The telephone line interface circuit, already described, is capable of generating almost any complex waveform under the control of the computer. Programs found on many of the cassettes are capable of producing either a single frequency sine wave or the sum of two different sine waves over a wide frequency range. These programs, along with the special circuit, have been demonstrated in the laboratory to produce TOUCH-TONES and single and multifrequency tones.

Description of Method of Tone Generation

Tone generation is predominantly performed by a subroutine written in assembly language and stored beginning at memory location 0A00 (Apple BASIC 2560) and ending at 0A47. This subroutine makes use of three tables. The first table starts at 0B00 and ends at 0BFF. This entire table of 256 entries make up one complete cycle of a sine function. It is plotted in Figure 2.

The second table starts at address 0C00 and ends at 0C1F. It has been found that its entries, taken two at a time, represent TOUCH-TONE digits. The third table starts at 0C20 and ends at 0C3F. It has been found that its entries, also taken two at a time, represent multifrequency digits.

The subroutine requires four bytes of information be passed to it. These are stored in memory locations 0-4. The first byte is frequency information for one tone. The second byte is frequency information for the second tone. The next two bytes give the duration of the tone. The subroutine operates in the following way. Two samples of the sine wave are fetched during each pass through the main loop of the subroutine (from 0A12 to 0A37). The samples to be fetched during a given pass are determined by incrementing the position of the each sample fetched during the previous pass by the frequency parameters given to the subroutine. Thus, the parameters in memory locations 0 and 1 can be thought of as the rate at which the program advances through the sine wave table. The higher the frequency parameter, the higher the frequency generated. The two sine wave values are then added together, divided by two, and output to the A/D converter. The tone duration parameters are then decremented and a test is made to determine if the tone

should be terminated. If not, the next samples are determined, etc.

To produce a single frequency (SF) tone, one of the frequency parameters is set to zero. To produce two tones, such as constitute TOUCH-TONES or multifrequency keypulsing (MF) signals, two non-zero frequency parameters are used. The second and third tables store the frequency parameters for TOUCH-TONES and multifrequencies, respectively. The tables are used by BASIC language programs (for example, program 29a. 7-15, line 8230) to determine what frequency parameters correspond to a particular digit.

Operational Experience

The tone generating programs and circuitry were exercised at Bell Laboratories in Holmdel, N.J. The programs were found to generate TOUCH-TONE, multifrequency keypulsing (MF), and single frequency (SF) signals according to Bell System standards.

"DIAL DEMO" PROGRAM

A search of cassettes 23 to 41 revealed six instances of BASIC language programs that are called "Dial Demo" or "Dial Demonstration" program. Comparison of the six versions revealed only minor differences between them, so the following description essentially describes all of them. The particular program described below was found on Cassette 33 (part of Item 27).

Operation of this program uses a feature of business telephone service private branch exchanges (PBX) known as "Remote Access". This feature allows calls from any telephone into the PBX, to be switched through the business telephone subscriber's facilities to various destinations. This service permits authorized users (e.g., executives and salesmen) to make use of their business telephone facilities from their residence telephone or from any telephone.

Typical uses are as follows:

1. The authorized user, from his home telephone dials a special telephone number to reach his company's "Remote Access" facility. This could be an "800" INWATS number.
2. The call advances to the Remote Access line and is automatically answered. At this point, PBX dial tone is heard. The authorized user must then TOUCH-TONE dial a secret 4-digit authorization code (password).
3. If the authorization code was sent correctly, PBX dial tone will again be heard by the user and he may dial a code to gain access to a business telephone line. This could be an OUTWATS line, foreign exchange line, corporate network, or an overseas private wire.
4. Dial tone is heard from the selected business line and the user dials the number of the desired distant party.

The "Dial Demonstration" program automates all of these procedures and has the ability to use any one of five different PBXs, all having 800 INWATS lines to gain admittance to their remote access feature.

Additionally, the program includes a listing of over 100 names and telephone numbers to be reached through the remote access lines, once they have been seized.

Program Operation

It is necessary to first load the BASIC language "Dial Demo" program from cassette into the Apple Computer memory. Additionally, four assembly language routines must also be loaded in a similar manner. After this, the user types the command "RUN" to begin the program. The program then responds with the question, "PRINT NUMBERS?". If the user types in "YES", the program requests a authorization code. We have determined this to be "ASDFGHJKL". If it is furnished, the computer then asks for the "NAME" of the person called. If after "PRINT NUMBERS", the answer is "NO", then the program skips the authorization code requirement and asks immediately for the "NAME" of the party to be called.

After entering the name, the program checks the memory looking for a match, and if it finds one, the name and telephone number are displayed and the question is asked, "WANNA USE A WATS EXTENDER?" Answering YES to this query causes this presentation:

- "WHICH ONE?
1. NEW JERSEY #1
 2. NEW JERSEY #2
 3. ATLANTA
 4. FLORIDA
 5. MINNIAPOLIS" (sic)

At this point, the user selects a single-digit answer "1" through "5" corresponding to the above list. This causes selection of an INWATS (800) telephone number of a PBX having Remote Access. The program then requests the user to "PICK UP THE PHONE". When dial tone occurs, the computer under control of the program proceeds to dial the INWATS number in rotary dial pulsing format. When the Remote Access line is reached, the proper authorization code is sent, tone is recognized, the OUTWATS code is sent, and the phone number of the selected NAME is outpulsed over the OUTWATS line. That line is rung and a conversation can take place. There is no billing to the user and the Remote Access facility, 800-line, and OUTWATS service are all unavailable for the business customer's authorized people.

Other Features

The program also provides for adding new names and telephone numbers, displaying the names and numbers on the screen in various formats, and direct calling of stored numbers. The program provides for checking the progress of a call through a Remote Access facility and reporting on status. Failure usually calls for a request to retry.

Laboratory Tests

The Apple Computer System, consisting of items 21, 22, 23, 24, 25, 26, 28, 29, 30, and 31 was set up in our laboratory at Holmdel, New Jersey. The line interface, (item 24) was connected to telephone line 201-946-8911. The "Dial Demo" program was loaded from the cassette. The part of BASIC instruction 630 which reads POKE -16231,4 was then changed in the computer memory to POKE -16231,6 by means of the keyboard. This changes the frequency to be recognized by the tone detector (IC-B4) in the telephone line circuit to accommodate the dial tone in Holmdel. The contents of the tape cassette remained unchanged.

A name and a WATS line from the computer memory were selected and the computer proceeded to dial automatically in the manner described above. This procedure was accomplished by P. Rabinowitz and G. S. Soloway and was observed by W. W. Heinze and K. D. Hopper.

"AUTO DIAL" PROGRAMOverview

Three copies of a program entitled, "Auto Dial Ver 1.0 10/77" were found on the second side of cassette 29. The cassette itself contains the markings "Dialpulse Scanner". These three programs, written in the BASIC programming language and utilizing the assembly language program already described for generating tones, have been designated as 29a. 7-15, 29a. 16-24, and 29a. 25-33. The first and third copies are identical, the second contains some parameters changed which affect the type of tones detected.

Program Function

This program can be used to dial any telephone number entered by the user using a combination of dialpulse and multifrequency tones. The telephone number is first automatically dialpulsed out by the computer. While the long distance network is in the process of completing the call, or while ringing the called party, the computer, at the user's instigation, will generate the 2600 Hertz single frequency tone. Immediately after, the computer will automatically redial the input telephone number using multifrequency tones. It is believed that the SF and MF tones are used for all but the initial entry into the network to prevent the local telephone operator from coming onto the line after the first call.

This program may also be used to search through a range of 4-digit dialing codes to find one that results in second dial-tone. The program can systematically search through all the possible codes of a DIMENSION® PBX to find the one that yields the Remote Access feature. Once this code is known, calls can be placed at no charge by calling the DIMENSION via its 800 INWATS number, gaining remote access, and then, on an out trunk, dialing out to the called party. The program referred to as "Dial Demo" is specifically designed to do this.

Program Description and Operation

The BASIC language program is loaded from the cassette into the Apple computer by its manufacturer's normal procedures. The machine language program which generates tones is also loaded by normal procedures. The program is initiated by typing RUN.

The program starts by initializing several variables and some parts of memory. It then requests input by displaying on the screen, "AC, PRE, NUMB". The user enters via the keyboard the requested information. The program processes and stores this number. The program then displays on the screen "START, END CODE." The user enters two 4-digit numbers.

The computer will then automatically seize the telephone line and start looking for the presence of dial tone. When dial tone is detected, the telephone number previously entered by the user is automatically dialpulsed out. The program will then display "WANNA REDIAL?". If the response is "Y", the program will hang-up and attempt to dial again. If the user's response begins with anything but "Y", the computer immediately generates a burst of SF, followed by the multifrequency tone KP, followed by the multifrequency representation of the previously entered telephone number, followed by the multifrequency ST.

The program then starts to search for dial tone. If dial tone is not found in a short time, the program attempts to redial using the SF and MF tones. If dial tone is found, the first four digit code previously entered by the user is dialed using TOUCH-TONES. The code dialed is displayed on the screen.

The computer then tries to detect remote access circuit intercept tone. If this is found, it returns to redialing using the SF and MF tones, and then tries the next code. If the intercept tone is not found, the computer tries to identify a second dial tone. If it does not find it, a redial using SF and MF tones is made. If the dial tone is found, a count is incremented and the same code is tried again. If five attempts of the same code all yield second dial tone, the computer displays "WHOOPIE!!!! CODE =" followed by the successful code number. This terminates the program. The program may also terminate if all codes in the prescribed range are tried without success.

Operational Experience

The Apple Computer was set up at Bell Laboratories in Holmdel, N.J., as well as the Cresco, Pa. central office. With minor adjustments for local dial tone and currently valid authorization codes, the program operated as described. (The tests at Cresco, Pa. are described in detail in Part 5.)

"SCANNER" PROGRAM

A search of cassettes 23 to 41 reveals seven instances of BASIC language programs that are called "Scanner". Comparison of the seven versions reveals only minor differences between them, so the following description essentially describes all of them. The particular program described below was found on cassette 24 (part of Item 27).

Operation of this program allows the user to dial a number (an INWATS PBX) and then store in the computer memory certain observations about that number. It would be of use in an attempt to find a telephone number that is the entry to a remote access feature of a PBX.

Program Operation

It is necessary to first load the BASIC language "Scanner" program and two assembly language routines into the Apple computer memory. After this is accomplished, the user types the command, "RUN" to start the program. After initialization, the program asks the user to enter an "AREA CODE", a 3-digit "PREFIX", and a 3-digit "BANK". The program subsequently appends the digit zero to the bank to form a complete telephone number. Then this number is dialed in TOUCH-TONE and after call completion the program scans the keyboard for further direction. These various options are as follows:

<u>Key Entered</u>	<u>Function</u>
E	The "Bank" number is stored in the Apple computer memory.
B	The "Bank" number is decremented by one and then used for dialing.
D	The current "Bank" number is used for dialing.
I	The "Bank" number is incremented by one and then is used for dialing.
L	All of the "Bank" numbers previously stored in memory are displayed.
S	A new "bank" is selected by the user for subsequent dialing.
A	The "bank" number is incremented by one.

<u>Key Entered</u>	<u>Function</u>
--------------------	-----------------

Z	The "bank" number is decremented by one.
F	The current bank is used as a base for detailed investigation of the ten numbers that are in its range.

Once the F option is selected, the user has further options as follows:

<u>Key Entered</u>	<u>Function</u>
--------------------	-----------------

D	Dial the current number.
I	Increment the current number and dial.
C	Display the current number and the word DATA.
T	Display the current number and the word TONE.
R	Enter new 4-digit numbers to be used as last four digits of current telephone number.
U	Decrement the current numbers and dial.

This program allows the user to scan a range of INWATS numbers to see if any of them have special functions. The "C" and "T" options described above allow the user to document that he has found a dataset or a second dial tone associated with that particular telephone number. Detection of the special function is accomplished by listening with the telephone handset to determine what kind of line has been reached.

"DIMENSION" PROGRAM

A search of cassettes, 23 to 41 yielded five instances of BASIC language programs that are called "DIMENSION". Comparison of the five versions reveals only minor differences between them, so the following description essentially describes all of them. The particular program described below was on Cassette 34 (part of Item 27).

Operation of this program allows the user to repetitively dial a PBX in an attempt to discover the authorization code for entry into its Remote Access feature. It achieves this by successively TOUCH-TONE signaling 4-digit codes until it finds one that is successful.

Program Operation

It is necessary to first load the BASIC language "DIMENSION" program from cassette tape into the Apple Computer memory. Additionally, two assembly language program segments must be loaded in a similar manner. After this, the user types the command "RUN" to start the program. The program responds with a request to enter a telephone number in the form "AREA", "PRE", "NUMB". After the user enters the number, the program asks the user to enter two 4-digit numbers to serve as the range of numbers to try as potential codes to a remote access service. After this, the user lifts the telephone handset and the program proceeds to automatically dial the PBX and search for a successful code. A success is indicated by getting a dial tone after TOUCH-TONE signaling the 4-digit code. If there is a successful attempt, the program verifies it by trying four more times. Five successful tries results in the program displaying, "WHOOPIE!!!! CODE =" followed by the successful code number. This terminates the program. Receipt of intercept instead of dial tone after entry of the code causes the program to increment the code by one and then retry the entire procedure. This will continue until the entire range of potential codes is tested.

Attachment

Attached is a computer printout, consisting of 29 listings of computer programs and contents of memory locations.

State of New Jersey)
) SS
County of Monmouth)

Walter W. Heinze, Gerald S. Soloway, and Paul Rabinowitz
being duly sworn according to law upon their oaths depose
and say:

1. We are Members of the Technical Staff of Bell Telephone Laboratories, Incorporated, a Corporation of the State of New York.
2. We have examined the items of evidence referenced herein and our analysis is true in every detail according to our best knowledge, information, and belief.

Walter W. Heinze
Walter W. Heinze

Gerald S. Soloway
Gerald S. Soloway

Paul Rabinowitz
Paul Rabinowitz

Sworn and subscribed
before me this 10th
day of December, 1977

Anne Marie Steeneck
Anne Marie Steeneck
Notary Public of New Jersey

My commission expires May 9, 1982

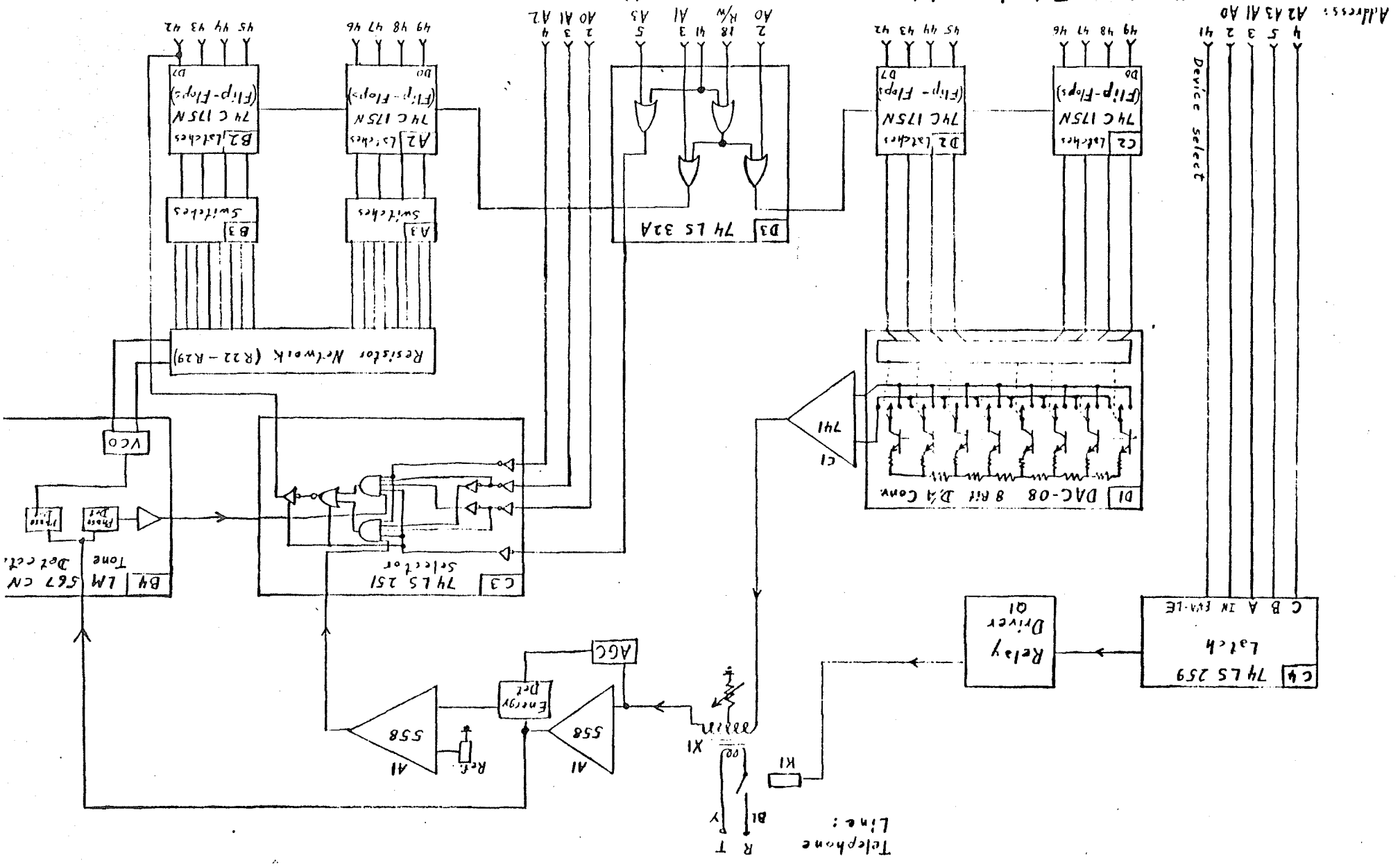
64

TABLE 1

DEVICE NUMBER	FUNCTION	CIRCUIT BOARD LOCATIONS
74 LS 32A	Quad or gates	D3
74 C 175N	Quad Latches	A2, B2, C2, D2
CD4066AE	Quad Bilateral Switches	A3, B3
LM 567 CN	Tone Decoder	B4
74LS251	Data Selector/Multiplexer	C3
PMI DAC 03	Digital to Analog Converter	D1
741	Operational Amplifier	C1
4558	Dual Operational Amplifier	A1
74LS259	Addressable Latch	C4
2N3568	Relay Driver	Q1
LG101-1-5	Standex Relay	B6

BTL-HO WHH
11-29-77

Fig. 1: Telephone Line Interface Circuit - Item 24
Signal Flow Diagram



Address: A2 A3 A1 A0
4 5 3 2
41
Device select

C4 74LS259 Latch
C B A IN EN-A-E
Driver Q1
Relay

D1 DAC-08 8 Bit D/A Conv
C1 741

D3 74LS32A

A2 Latches
74C175N
A3 Switches

B2 Latches
74C175N
B3 Switches

B4 LM567 CN
Tone Det. ckt.
Phase Det.
VCO

C3 74LS251 Selector

Resistor Network (R22-R29)

A1 558
Energy Pkt
AGC

Telephone Lines:
R T B Y

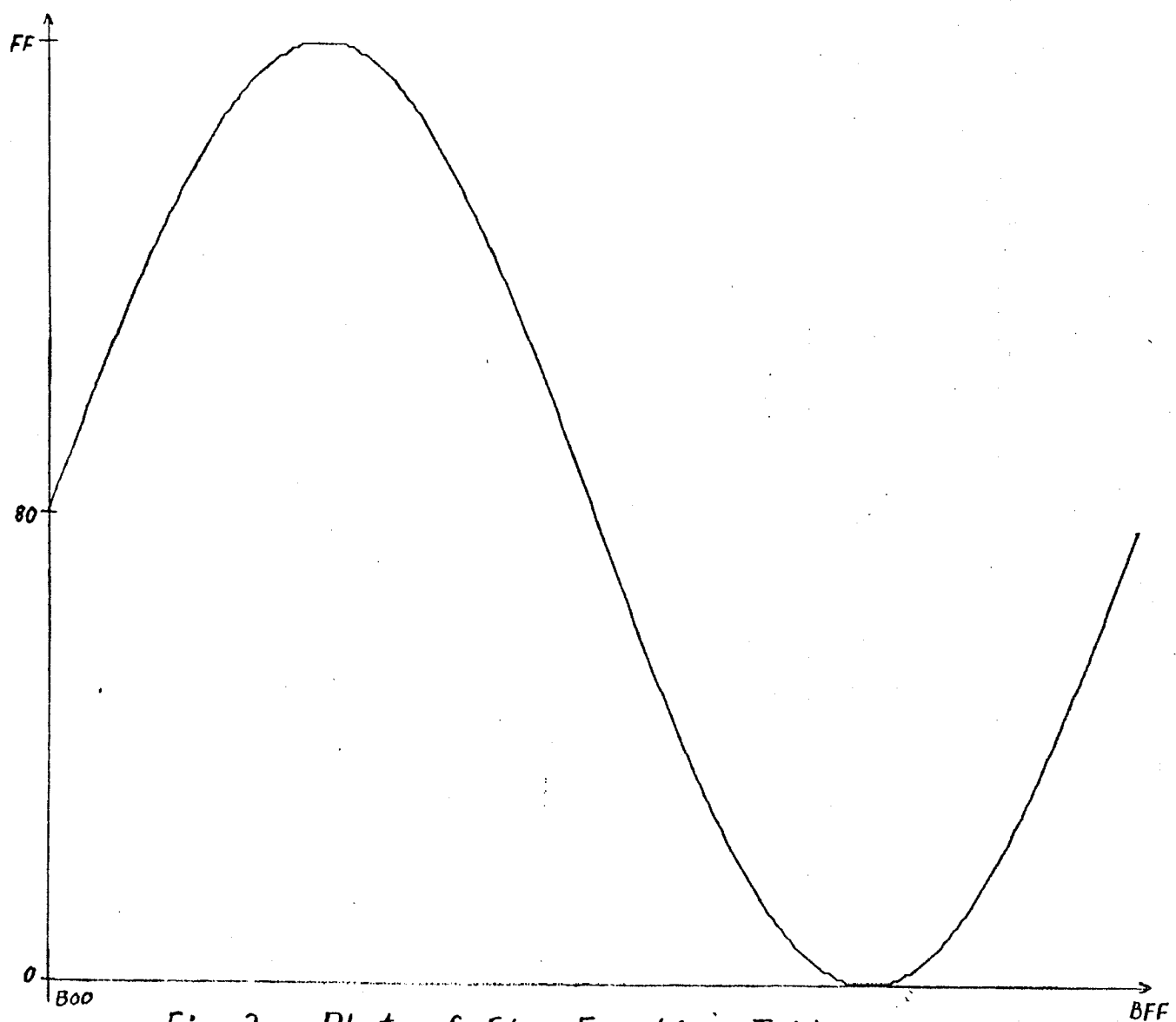


Fig. 2 Plot of Sine Function Table
(From cassette no. 30, p/o item 20)

PART 2

APPENDIX 1

```

24.2-12      1 REM      SCANNER VER 2.8  9/12/      77
26.3-13      1 REM      SCANNER VER 2.7  8/30/
29a.16-24    1 REM      AUTO DIAL   VER 1.0 10/77
29a.25-33    1 REM      AUTO DIAL   VER 1.0 10/77
29a.7-15     1 REM      AUTO DIAL   VER 1.0 10/77
30a.2-10    900L
32.2-9       1 REM      DIMENSION VER 1.4 8/12/7
33.1-8       10 DIM A$(128),L(256),B$(40) DIAL INWATS
33.17-25     10 DIM A$(128),L(256),B$(40) DIAL INWATS
33.33-40     10 DIM A$(128),L(256),B$(40) DIAL INWATS
33.9-16      10 DIM A$(128),L(256),B$(40) DIAL INWATS
33a.1-2     0900-    18          CLC
33a.3-7      100 REM  -----INIT-----
34.16-22     1 REM      SCANNER VER 2.5  8/4/77
34.187-193   1 REM      SCANNER VER 2.6  8/4/77
34.207-214   1 REM      SCANNER VER 2.7  8/12/7
34.228-234   1 REM      DIMENSION VER 1.4 8/12/7
34.246-252   1 REM      DIMENSION VER 1.4 8/12/7
34.286-293   1 REM      DIMENSION VER 1.4 8/12/7
34.3-8       1 REM      SCANNER VER 2.6  8/4/77
34.31-34     1 REM      DIMENSION VER 1.3 8/4/77
34.351-360   1 REM      SCANNER VER 2.7  8/12/7
34.42-47     1 REM      LOOKER VER 5.6  8/3/77
34.75-81     1 REM      DIALER VER 1.6 7/31/77
38.3-15      10 DIM A$(128),L(256),B$(40) DIAL INWATS
38a.27-41    10 DIM A$(128),L(256),B$(40) DIAL INWATS
39.1-4       1 REM      EDITOR VER 1
39.10-14     1 REM      EDITOR VER 1
39.5-9       1 REM      EDITOR VER 1
    
```

69


```

1 REM          SCANNER VER 2.8  9/12/
77
100 POKE 2,100: POKE 3,2: DIM A$(
(3)
110 DT=5:BY=18:RG=16:MEM=4096
120 AD=768: FOR A=0 TO 255: POKE
AD+A,255: NEXT A
130 CALL -936: INPUT "CLEAR AREA"
,A$: IF A$(1,1)="Y" THEN GOSUB
4000
140 INPUT "AC,PRE",AC,PRE
150 INPUT "STARTING BANK:",BANK
160 CALL -936
170 VTAB 3: PRINT "STATUS:"
180 VTAB 4: PRINT "NUMB=";AC;"-"
;PRE;"-";BANK*10
190 NUMB=BANK*10: VTAB 5: PRINT
"
"
300 PH= PEEK (-16228):DLY=350: GOSUB
6000
500 PH= PEEK (-16227)
550 T1=4:DL1=10: GOSUB 7000: IF
T1<4 THEN 300
1000 GOSUB 5500
1040 AD=768: GOSUB 8210
2000 KEY= PEEK (-16384): POKE -16368
,0
2010 IF KEY=197 THEN 3000: REM  ENTER
NUMBER
2020 IF KEY=194 THEN 3500: REM  BACK
UP
2025 IF KEY=196 THEN 300: REM  DIAL
NUM
2030 IF KEY=201 THEN 3600: REM  INCR
& DIAL
2040 IF KEY=204 THEN 4100: REM  LOOK
AT BANKS
2045 IF KEY=211 THEN 4500: REM  STORE
NEW BANKS
2050 IF KEY=193 THEN 4200: REM  ADVAN
CE
2055 IF KEY=218 THEN 4210: REM  (Z) B
ACK-UP
2060 IF KEY=198 THEN 3100: REM  FIND
#'S
2100 GOTO 2000
3000 REM  ENTER BANK IN MEM
3005 PRINT " ";
3010 POKE MEM,(BANK) MOD 256: POKE
MEM+1,(BANK)/256:MEM=MEM+2:
VTAB 5: GOTO 4100
3100 CALL -936: VTAB 3: PRINT "DIALIN
G NUMBERS":MEM=4096
3103 REM  FIND #'S IN BANKS
3105 KEY= PEEK (-16384): POKE -16368
,0

```

```
3110 IF KEY=196 THEN 3230: REM   DIA
      L
      #
3115 IF KEY=201 THEN 3300: REM   INC A
      ND DIAL
3120 IF KEY=195 THEN 3310: REM   (C) D
      ATA
3125 IF KEY=212 THEN 3320: REM   (T) T
      ONE
3130 IF KEY=210 THEN 3330: REM   (R) R
      E-ENTER NUMB
3135 IF KEY=213 THEN 3305: REM   (U) DE
      C AND DIAL
3199 GOTO 3105
3200 REM   GET BANK
3205 INC=0
3210 BANK= PEEK (MEM)+ PEEK (MEM+
      1)*256:MEM=MEM+2
3220 NUMB=BANK*10+INC
3230 GOSUB 5500
3235 VTAB 2: PRINT "NUM=";AC;"-"
      ;PRE;"-";NUMB
3240 AD=768: GOSUB 8210: GOTO 3105

3300 INC=INC+1: IF INC=10 THEN 3200
      : GOTO 3220
3305 INC=INC-1: IF INC<0 THEN 3330
      : GOTO 3220
3310 VTAB 4: PRINT AC;"-";PRE;"-"
      ;NUMB-1;" DATA": GOTO 3105

3320 VTAB 5: PRINT AC;"-";PRE;"-"
      ;NUMB-1;" TONE": GOTO 3105

3330 VTAB 6: INPUT "NUMB=",NUMB:
      BANK=NUMB/10:INC=NUMB-BANK*
      10: GOTO 3105
3500 REM   ----BACK UP
3510 BANK=BANK-1: GOTO 170
3600 REM   INCREMENT ----
3610 BANK=BANK+1: GOTO 170
4000 REM   CLEAR AREA
4010 FOR M=4096 TO 5120: POKE M,
      255: NEXT M: RETURN
4100 CALL -936:ME=4096: VTAB 6
4110 IF ME=MEM THEN 4130
4120 NU= PEEK (ME)+ PEEK (ME+1)*
      256: PRINT NU;" "":ME=ME+2
      : GOTO 4110
4130 PRINT : GOTO 2000
4200 BANK=BANK+1: VTAB 4: TAB 14
      : PRINT BANK*10: GOTO 2000
4210 BANK=BANK-1: VTAB 4: TAB 14
      : PRINT BANK*10: GOTO 2000
4500 REM   ----STORE NEW BANKS----
4510 VTAB 8: PRINT "INPUT BANK,0 IF
      DONE"
```

```
4520 INPUT BA: IF BA=0 THEN 2000
      : POKE MEM,BA MOD 256: POKE
      MEM+1,BA/256:MEM=MEM+2: GOTO
      4520
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
      AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
5500 AD=768: IF AC=0 THEN 5520
5510 NUM=AC: DIG=4: GOSUB 5000
5520 NUM=PRE: DIG=3: GOSUB 5000
5530 NUM=NUMB: DIG=4: GOSUB 5000
5540 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1: TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7005 REM SUBR THAT RETURNS LENGTH O
      F TONE
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH: Y=
      ( PEEK (-16238)>127): VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE="; Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (3072+(2* PEEK (
      AD+DGT))): TONB= PEEK (3072+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9000 KQ= PEEK (-16384): IF KQ>127
      THEN 9010: POKE -16368,0: GOTO
      9000
9010 PRINT KQ: GOTO 9000
9999 END
```

```

1 REM          SCANNER VER 2.7  8/30/
7              7
100 POKE 2,100: POKE 3,2: DIM A$
    (3)
110 DT=5:BY=18:RG=16:MEM=4096
120 AD=768: FOR A=0 TO 255: POKE
    AD+A,255: NEXT A
130 CALL -936: INPUT "CLEAR AREA"
    ,A$: IF A$(1,1)="Y" THEN GOSUB
    4000
140 INPUT "AC,PRE",AC,PRE
150 INPUT "STARTING BANK:",BANK
160 CALL -936
170 VTAB 3: PRINT "STATUS:"
180 VTAB 4: PRINT "NUMB=";AC;"-"
    ;PRE;"-";BANK*10
190 NUMB=BANK*10: VTAB 5: PRINT
    "
300 PH= PEEK (-16228):DLY=350: GOSUB
    6000
500 PH= PEEK (-16227)
520 POKE -16231,6
550 T1=20:DL1=150: GOSUB 7000: IF
    T1<20 THEN 300
1000 GOSUB 5500
1040 AD=768: GOSUB 8210
2000 KEY= PEEK (-16384): POKE -16368
    ,0
2010 IF KEY=197 THEN 3000: REM  ENTER
    NUMBER
2020 IF KEY=194 THEN 3500: REM  BACK
    UP
2025 IF KEY=196 THEN 300: REM   DIAL
    NUM
2030 IF KEY=201 THEN 3600: REM  INCR
    & DIAL
2040 IF KEY=204 THEN 4100: REM  LOOK
    AT BANKS
2045 IF KEY=211 THEN 4500: REM  STORE
    NEW BANKS
2050 IF KEY=193 THEN 4200: REM  ADVAN
    CE
2055 IF KEY=218 THEN 4210: REM  (Z) B
    ACK-UP
2060 IF KEY=198 THEN 3100: REM  FIND
    #'S
2100 GOTO 2000
3000 REM  ENTER BANK IN MEM
3005 PRINT " ";
3010 POKE MEM,(BANK) MOD 256: POKE
    MEM+1,(BANK)/256:MEM=MEM+2:
    VTAB 5: GOTO 4100
3100 CALL -936: VTAB 3: PRINT "DIALIN
    G NUMBERS":MEM=4096
3103 REM  FIND #'S IN BANKS
3105 KEY= PEEK (-16384): POKE -16368

```

```

,0
3110 IF KEY=196 THEN 3230: REM   DIA
      L                               #
3115 IF KEY=201 THEN 3300: REM   INC A
      ND DIAL
3120 IF KEY=195 THEN 3310: REM   (C) D
      ATA
3125 IF KEY=212 THEN 3320: REM   (T) T
      ONE
3130 IF KEY=210 THEN 3330: REM   (R) R
      E-ENTER NUMB
3135 IF KEY=213 THEN 3305: REM   (U) DE
      C AND DIAL
3199 GOTO 3105
3200 REM   GET BANK
3205 INC=0
3210 BANK= PEEK (MEM)+ PEEK (MEM+
      1)*256:MEM=MEM+2
3220 NUMB=BANK*10+INC
3230 GOSUB 5500
3235 VTAB 2: PRINT "NUM=";AC;"-"
      ;PRE;"-";NUMB
3240 AD=768: GOSUB 8210: GOTO 3105

3300 INC=INC+1: IF INC=10 THEN 3200
      : GOTO 3220
3305 INC=INC-1: IF INC<0 THEN 3330
      : GOTO 3220
3310 VTAB 4: PRINT AC;"-";PRE;"-"
      ;NUMB-1;" DATA": GOTO 3105

3320 VTAB 5: PRINT AC;"-";PRE;"-"
      ;NUMB-1;" TONE": GOTO 3105

3330 VTAB 6: INPUT "NUMB=",NUMB:
      BANK=NUMB/10:INC=NUMB-BANK*
      10: GOTO 3105
3500 REM   ----BACK UP
3510 BANK=BANK-1: GOTO 170
3600 REM   INCREMENT ----
3610 BANK=BANK+1: GOTO 170
4000 REM   CLEAR AREA
4010 FOR M=4096 TO 5120: POKE M,
      255: NEXT M: RETURN
4100 CALL -936:ME=4096: VTAB 6
4110 IF ME=MEM THEN 4130
4120 NU= PEEK (ME)+ PEEK (ME+1)*
      256: PRINT NU;" "":ME=ME+2
      : GOTO 4110
4130 PRINT : GOTO 2000
4200 BANK=BANK+1: VTAB 4: TAB 14
      : PRINT BANK*10: GOTO 2000
4210 BANK=BANK-1: VTAB 4: TAB 14
      : PRINT BANK*10: GOTO 2000
4500 REM   ----STORE NEW BANKS----
4510 VTAB 8: PRINT "INPUT BANK,0 IF

```

```

DONE"
4520 INPUT BA: IF BA=0 THEN 2000
      : POKE MEM,BA MOD 256: POKE
      MEM+1,BA/256:MEM=MEM+2: GOTO
      4520
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
      AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
5500 AD=768: IF AC=0 THEN 5520
5510 NUM=AC:DIG=4: GOSUB 5000
5520 NUM=PRE:DIG=3: GOSUB 5000
5530 NUM=NUMB:DIG=4: GOSUB 5000
5540 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7005 REM SUBR THAT RETURNS LENGTH O
      F TONE
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127): VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (3072+(2* PEEK (
      AD+DGT))):TONB= PEEK (3072+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9000 KQ= PEEK (-16384): IF KQ>127
      THEN 9010: POKE -16368,0: GOTO
      9000
9010 PRINT KQ: GOTO 9000
9999 END

```

```

1 REM      AUTO DIAL   VER 1.0 10/77

10 DIM A$(10)
20 CON=8192:AQ=4:AW=8704
30 DIALPULSE=2500:ONTIME=4:OFFTIME=
  4
40 INTERDIG=75:BEEP=2800
50 TTN=3072
55 MF=3104
60 KP=2600:ST=2700
100 POKE 2,100: POKE 3,2
110 AD=768: FOR A=0 TO 255: POKE
  AD+A,255: NEXT A
120 CALL -936
200 VTAB 4: INPUT "AREA,PRE,NUMB"
  ,AC,PRE,NUMB
210 IF AC=0 THEN 310
300 NUM=AC:DIG=4: GOSUB 5000
310 NUM=PRE:DIG=3: GOSUB 5000
320 NUM=NUMB:DIG=4: GOSUB 5000
325 ADR=AD+1
400 INPUT "START,END CODE",CODE,
  ECOD
1000 PH= PEEK (-16228):DLY=350: GOSUB
  6000
1100 PH= PEEK (-16227)
1110 DL1=100: POKE -16231,4:T1=10
  : GOSUB 7000: IF T<10 THEN
  1000
1112 DLY=350: GOSUB 6000
1115 GOSUB DIALPULSE: PRINT
1118 INPUT "WANNA REDIAL?",A$: IF
  A$(1,1)="Y" THEN 1000
1120 GOSUB BEEP:TBL=MF:DLY=350: GOSUB
  6000: GOSUB KP:AD=769: GOSUB
  8210: GOSUB ST
1130 TBL=TTN
1140 DL1=800:T1=4: POKE -16231,7
  : GOSUB 7000
1145 IF T<T1 THEN 1120
1150 AD=ADR:NUM=CODE:DIG=4: GOSUB
  5000:AD=ADR: GOSUB 8210
1155 PRINT "CODE=";CODE
1160 POKE -16231,21:DL1=100:T1=10
  : GOSUB 7000: IF T=T1 THEN
  1500
1168 PRINT "";
1170 DL1=100:T1=20: POKE -16231,
  6: GOSUB 7000: IF T<T1 THEN
  1120:TRY=TRY+1: PRINT "TRYING "
  ;TRY: IF TRY<5 THEN 1120
1180 FOR A=1 TO 50: PRINT "";: NEXT
  A: PRINT "WHOOPIE!!!! CODE="
  ;CODE:PH= PEEK (-16228): END

1500 TRY=0: IF CODE=ECOD THEN 2000

```

23a. 10-24 Page 2

```
:CODE=CODE+1: GOTO 1120
2000 PH= PEEK (-16228): END
2500 AD=768:DGT=0
2510 IF PEEK (AD+DGT)=255 THEN RETURN

2520 COUNT= PEEK (AD+DGT)
2525 PRINT COUNT;
2527 IF COUNT=0 THEN COUNT=10
2530 FOR PULSE=1 TO COUNT
2540 POKE -16228,0:DLY=ONTIME: GOSUB
      6000
2550 POKE -16227,0:DLY=OFFTIME: GOSUB
      6000
2560 NEXT PULSE
2570 DLY=INTERDIG: GOSUB 6000:DGT=
      DGT+1: GOTO 2510
2600 POKE 2,200: POKE 3,3: POKE
      0,47: POKE 1,73: CALL 2560
2610 POKE 2,100: POKE 3,2: RETURN

2700 POKE 0,65: POKE 1,73: CALL
      2560
2710 RETURN
2800 POKE 2,0: POKE 3,7
2810 POKE 0,0: POKE 1,112: CALL
      2560
2815 POKE 2,100: POKE 3,2
2820 RETURN
2999 CALL -936
3000 POKE -16167, PDL (0)/3:T=( PEEK
      (-16238)>127): VTAB 10: PRINT
      "TONE=";T: PRINT "VAL="; PDL
      (0)/3: GOTO 3000
4500 POKE -16231,9: GOSUB 7000: PRINT
      "T=";T: GOTO 4500
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
      AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
```



```
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127): VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (TBL+(2* PEEK (AD+
      DGT))):TONB= PEEK (TBL+(2* PEEK
      (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END
```

```

1 REM      AUTO DIAL  VER 1.0 10/77

10 DIM A$(10)
20 CON=8192:AQ=4:AW=8704
30 DIALPULSE=2500:ONTIME=4:OFFTIME=
  4
40 INTERDIG=75:BEEP=2800
50 TTN=3072
55 MF=3104
60 KP=2600:ST=2700
100 POKE 2,100: POKE 3,2
110 AD=768: FOR A=0 TO 255: POKE
  AD+A,255: NEXT A
120 CALL -936
200 VTAB 4: INPUT "AREA,PRE,NUMB"
  ,AC,PRE,NUMB
210 IF AC=0 THEN 310
300 NUM=AC:DIG=4: GOSUB 5000
310 NUM=PRE:DIG=3: GOSUB 5000
320 NUM=NUMB:DIG=4: GOSUB 5000
325 ADR=AD+1
400 INPUT "START,END CODE",CODE,
  ECOD
1000 PH= PEEK (-16228):DLY=350: GOSUB
  6000
1100 PH= PEEK (-16227)
1110 DL1=100: POKE -16231,4:T1=10
  : GOSUB 7000: IF T<10 THEN
  1000
1112 DLY=350: GOSUB 6000
1115 GOSUB DIALPULSE: PRINT
1118 INPUT "WANNA REDIAL?",A$: IF
  A$(1,1)="Y" THEN 1000
1120 GOSUB BEEP:TBL=MF:DLY=350: GOSUB
  6000: GOSUB KP:AD=769: GOSUB
  8210: GOSUB ST
1130 TBL=TTN
1140 DL1=800:T1=8: POKE -16231,5
  : GOSUB 7000
1145 IF T<T1 THEN 1120
1150 AD=ADR:NUM=CODE:DIG=4: GOSUB
  5000:AD=ADR: GOSUB 8210
1155 PRINT "CODE=";CODE
1160 POKE -16231,19:DL1=100:T1=10
  : GOSUB 7000: IF T=T1 THEN
  1500
1168 PRINT "";
1170 DL1=100:T1=20: POKE -16231,
  6: GOSUB 7000: IF T<T1 THEN
  1120:TRY=TRY+1: PRINT "TRYING "
  ;TRY: IF TRY<5 THEN 1120
1180 FOR A=1 TO 50: PRINT "":; NEXT
  A: PRINT "WHOOPIE!!!! CODE="
  ;CODE:PH= PEEK (-16228): END

1500 TRY=0: IF CODE=ECOD THEN 2000

```

```
      :CODE=CODE+1: GOTO 1120
2000 PH= PEEK (-16228): END
2500 AD=768:DGT=0
2510 IF PEEK (AD+DGT)=255 THEN RETURN

2520 COUNT= PEEK (AD+DGT)
2525 PRINT COUNT;
2527 IF COUNT=0 THEN COUNT=10
2530 FOR PULSE=1 TO COUNT
2540 POKE -16228,0:DLY=ONTIME: GOSUB
      6000
2550 POKE -16227,0:DLY=OFFTIME: GOSUB
      6000
2560 NEXT PULSE
2570 DLY=INTERDIG: GOSUB 6000:DGT=
      DGT+1: GOTO 2510
2600 POKE 2,200: POKE 3,2: POKE
      0,47: POKE 1,73: CALL 2560
2610 POKE 2,100: POKE 3,2: RETURN

2700 POKE 0,65: POKE 1,73: CALL
      2560
2710 RETURN
2800 POKE 2,0: POKE 3,7
2810 POKE 0,0: POKE 1,112: CALL
      2560
2815 POKE 2,100: POKE 3,2
2820 RETURN
2999 CALL -936
3000 POKE -16167, PDL (0)/3:T=( PEEK
      (-16238)>127): VTAB 10: PRINT
      "TONE=";T: PRINT "VAL="; PDL
      (0)/3: GOTO 3000
4500 POKE -16231,9: GOSUB 7000: PRINT
      "T=";T: GOTO 4500
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
      AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
```

```
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127): VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (TBL+(2* PEEK (AD+
      DGT))):TONB= PEEK (TBL+(2* PEEK
      (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END
```

1 REM AUTO DIAL VER 1.0 10/77

```

10 DIM A$(10)
20 CON=8192:AQ=4:AW=8704
30 DIALPULSE=2500:ONTIME=4:OFFTIME=
  4
40 INTERDIG=75:BEEP=2800
50 TTN=3072
55 MF=3104
60 KP=2600:ST=2700
100 POKE 2,100: POKE 3,2
110 AD=768: FOR A=0 TO 255: POKE
  AD+A,255: NEXT A
120 CALL -936
200 VTAB 4: INPUT "AREA,PRE,NUMB"
  ,AC,PRE,NUMB
210 IF AC=0 THEN 310
300 NUM=AC:DIG=4: GOSUB 5000
310 NUM=PRE:DIG=3: GOSUB 5000
320 NUM=NUMB:DIG=4: GOSUB 5000
325 ADR=AD+1
400 INPUT "START,END CODE",CODE,
  ECOD
1000 PH= PEEK (-16228):DLY=350: GOSUB
  6000
1100 PH= PEEK (-16227)
1110 DL1=100: POKE -16231,4:T1=10
  : GOSUB 7000: IF T<10 THEN
  1000
1112 DLY=350: GOSUB 6000
1115 GOSUB DIALPULSE: PRINT
1118 INPUT "WANNA REDIAL?",A$: IF
  A$(1,1)="Y" THEN 1000
1120 GOSUB BEEP:TBL=MF:DLY=350: GOSUB
  6000: GOSUB KP:AD=769: GOSUB
  8210: GOSUB ST
1130 TBL=TTN
1140 DL1=800:T1=4: POKE -16231,7
  : GOSUB 7000
1145 IF T<T1 THEN 1120
1150 AD=ADR:NUM=CODE:DIG=4: GOSUB
  5000:AD=ADR: GOSUB 8210
1155 PRINT "CODE=";CODE
1160 POKE -16231,21:DL1=100:T1=10
  : GOSUB 7000: IF T=T1 THEN
  1500
1168 PRINT "";
1170 DL1=100:T1=20: POKE -16231,
  6: GOSUB 7000: IF T<T1 THEN
  1120:TRY=TRY+1: PRINT "TRYING "
  ;TRY: IF TRY<5 THEN 1120
1180 FOR A=1 TO 50: PRINT "":; NEXT
  A: PRINT "WHOOPIE!!!! CODE="
  ;CODE:PH= PEEK (-16228): END
1500 TRY=0: IF CODE=ECOD THEN 2000

```

```
:CODE=CODE+1: GOTO 1120
2000 PH= PEEK (-16228): END
2500 AD=768:DGT=0
2510 IF PEEK (AD+DGT)=255 THEN RETURN

2520 COUNT= PEEK (AD+DGT)
2525 PRINT COUNT;
2527 IF COUNT=0 THEN COUNT=10
2530 FOR PULSE=1 TO COUNT
2540 POKE -16228,0:DLY=ONTIME: GOSUB
      6000
2550 POKE -16227,0:DLY=OFFTIME: GOSUB
      6000
2560 NEXT PULSE
2570 DLY=INTERDIG: GOSUB 6000:DGT=
      DGT+1: GOTO 2510
2600 POKE 2,200: POKE 3,3: POKE
      0,47: POKE 1,73: CALL 2560
2610 POKE 2,100: POKE 3,2: RETURN

2700 POKE 0,65: POKE 1,73: CALL
      2560
2710 RETURN
2800 POKE 2,0: POKE 3,7
2810 POKE 0,0: POKE 1,112: CALL
      2560
2815 POKE 2,100: POKE 3,2
2820 RETURN
2999 CALL -936
3000 POKE -16167, PDL (0)/3:T=( PEEK
      (-16238)>127): VTAB 10: PRINT
      "TONE=";T: PRINT "VAL="; PDL
      (0)/3: GOTO 3000
4500 POKE -16231,9: GOSUB 7000: PRINT
      "T=";T: GOTO 4500
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
      AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
```

```
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127): VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (TBL+(2* PEEK (AD+
      DGT))):TONB= PEEK (TBL+(2* PEEK
      (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END
```

```

900L
0900-  A0 00      LDY   #$00
0902-  A9 05      LDA   #$05
0904-  85 0A      STA   $0A
0906-  A9 14      LDA   #$14
0908-  85 0B      STA   $0B
090A-  B1 0C      LDA   ($0C),Y
090C-  91 0A      STA   ($0A),Y
090E-  C8          INY
090F-  C9 1E      CMP   #$1E
0911-  D0 F7      BNE   $090A
0913-  60          RTS
0914-  A5 0C      LDA   $0C
0916-  85 0E      STA   $0E
0918-  A5 0D      LDA   $0D
091A-  85 0F      STA   $0F
091C-  A0 00      LDY   #$00
091E-  A9 1E      LDA   #$1E
0920-  91 0E      STA   ($0E),Y
0922-  E6 0E      INC   $0E
0924-  D0 F8      BNE   $091E
*L
0926-  E6 0F      INC   $0F
0928-  A5 0F      LDA   $0F
092A-  C9 60      CMP   #$60
092C-  D0 F0      BNE   $091E
092E-  A9 00      LDA   #$00
0930-  8D 00 40   STA   $4000
0933-  60          RTS
0934-  A5 0C      LDA   $0C
0936-  85 0E      STA   $0E
0938-  A5 0D      LDA   $0D
093A-  85 0F      STA   $0F
093C-  A9 8E      LDA   #$8E
093E-  85 0A      STA   $0A
0940-  A9 14      LDA   #$14
0942-  85 0B      STA   $0B
0944-  A9 00      LDA   #$00
0946-  85 11      STA   $11
0948-  85 12      STA   $12
094A-  A0 00      LDY   #$00
094C-  B1 0E      LDA   ($0E),Y
*L
094E-  C9 1E      CMP   #$1E
0950-  F0 0A      BEQ   $095C
0952-  C9 00      CMP   #$00
0954-  F0 25      BEQ   $097B
0956-  20 7C 03   JSR   $037C
0959-  4C 4C 03   JMP   $034C
095C-  20 7C 03   JSR   $037C
095F-  18          CLC
0960-  A5 0E      LDA   $0E
0962-  91 0A      STA   ($0A),Y
0964-  C8          INY
0965-  A5 0F      LDA   $0F
0967-  91 0A      STA   ($0A),Y

```


0969-	A5 0A	LDA	\$0A
096B-	69 02	ADC	#\$02
096D-	85 0A	STA	\$0A
096F-	90 02	BCC	\$0973
0971-	E6 0B	INC	\$0B
0973-	E6 11	INC	\$11
0975-	D0 D3	BNE	\$094A
*L			
0977-	E6 12	INC	\$12
0979-	D0 CF	BNE	\$094A
097B-	60	RTS	
097C-	E6 0E	INC	\$0E
097E-	D0 FB	BNE	\$097B
0980-	E6 0F	INC	\$0F
0982-	D0 F7	BNE	\$097B
0984-	60	RTS	
0985-	A9 16	LDA	#\$16
0987-	85 12	STA	\$12
0989-	A5 0C	LDA	\$0C
098B-	85 0E	STA	\$0E
098D-	A5 0F	LDA	\$0F
098F-	85 0B	STA	\$0B
0991-	A9 8A	LDA	#\$8A
0993-	85 0A	STA	\$0A
0995-	A9 14	LDA	#\$14
0997-	85 0B	STA	\$0B
0999-	18	CLC	
099A-	A5 10	LDA	\$10
*L			
099C-	2A	ROL	
099D-	65 0A	ADC	\$0A
099F-	85 0A	STA	\$0A
09A1-	90 02	BCC	\$09A5
09A3-	E6 0B	INC	\$0B
09A5-	A9 00	LDA	#\$00
09A7-	85 11	STA	\$11
09A9-	A0 00	LDY	#\$00
09AB-	E6 11	INC	\$11
09AD-	A5 11	LDA	\$11
09AF-	20 C1 FB	JSR	\$FBC1
09B2-	B1 0A	LDA	(\$0A),Y
09B4-	85 0E	STA	\$0E
09B6-	C8	INY	
09B7-	B1 0A	LDA	(\$0A),Y
09B9-	85 0F	STA	\$0F
09BB-	A0 00	LDY	#\$00
09BD-	B1 0E	LDA	(\$0E),Y
09BF-	C9 1E	CMP	#\$1E
09C1-	F0 08	BEQ	\$09CB
*L			
09C3-	C9 00	CMP	#\$00
09C5-	F0 03	BEQ	\$09CA
09C7-	91 28	STA	(\$28),Y
09C9-	C8	INY	
09CA-	D0 F1	BNE	\$09BD
09CC-	60	RTS	

09CD-	18	CLC	
09CE-	C6 12	DEC	\$12
09D0-	F0 FA	BEQ	\$09CC
09D2-	A5 0A	LDA	\$0A
09D4-	69 02	ADC	#\$02
09D6-	85 0A	STA	\$0A
09D8-	90 CF	BCC	\$09A9
09DA-	E6 0B	INC	\$0B
09DC-	B0 CB	BCS	\$09A9
09DE-	00	BRK	
09DF-	00	BRK	
09E0-	00	BRK	
09E1-	00	BRK	
09E2-	00	BRK	
*L			
09E3-	00	BRK	
09E4-	00	BRK	
09E5-	00	BRK	
09E6-	00	BRK	
09E7-	00	BRK	
09E8-	00	BRK	
09E9-	00	BRK	
09EA-	00	BRK	
09EB-	00	BRK	
09EC-	00	BRK	
09ED-	00	BRK	
09EE-	00	BRK	
09EF-	00	BRK	
09F0-	00	BRK	
09F1-	00	BRK	
09F2-	00	BRK	
09F3-	00	BRK	
09F4-	00	BRK	
09F5-	00	BRK	
09F6-	00	BRK	
*L			
09F7-	00	BRK	
09F8-	00	BRK	
09F9-	00	BRK	
09FA-	00	BRK	
09FB-	00	BRK	
09FC-	00	BRK	
09FD-	00	BRK	
09FE-	00	BRK	
09FF-	00	BRK	
0A00-	A9 0C	LDA	#\$0C
0A02-	85 08	STA	\$08
0A04-	A5 02	LDA	\$02
0A06-	85 06	STA	\$06
0A08-	A5 03	LDA	\$03
0A0A-	85 07	STA	\$07
0A0C-	A9 00	LDA	#\$00
0A0E-	85 04	STA	\$04
0A10-	85 05	STA	\$05
0A12-	18	CLC	
0A13-	A5 04	LDA	\$04

```

*L
0A15- 65 00      ADC   $00
0A17-  A8        TAY
0A18- 85 04      STA   $04
0A1A- A5 05      LDA   $05
0A1C- 65 01      ADC   $01
0A1E-  AA        TAX
0A1F- 85 05      STA   $05
0A21- 18         CLC
0A22- A9 00      LDA   # $00
0A24- 7D 00 0B   ADC   $0B00,X
0A27- 79 00 0B   ADC   $0B00,Y
0A2A- 6A         ROR
0A2B- 8D 9A C0   STA   $C09A
0A2E- 20 3A 0A   JSR   $0A3A
0A31- C6 06      DEC   $06
0A33- D0 DD      BNE   $0A12
0A35- C6 07      DEC   $07
0A37- D0 D9      BNE   $0A12
0A39- 60         RTS
0A3A- A5 08      LDA   $08
*L
0A3C- 85 09      STA   $09
0A3E- C6 09      DEC   $09
0A40- D0 FC      BNE   $0A3E
0A42- EA        NOP
0A43- EA        NOP
0A44- EA        NOP
0A45- EA        NOP
0A46- EA        NOP
0A47- 60         RTS
0A48- 00         BRK
0A49- 00         BRK
0A4A- 00         BRK
0A4B- 00         BRK
0A4C- 00         BRK
0A4D- 00         BRK
0A4E- 00         BRK
0A4F- 00         BRK
0A50- 00         BRK
0A51- 00         BRK
0A52- 00         BRK
*A52.AFF
0A52- 00 00 00 00 00 00 00
0A58- 00 00 00 00 00 00 00 00
0A60- 00 00 00 00 00 00 00 00
0A68- 00 00 00 00 00 00 00 00
0A70- 00 00 00 00 00 00 00 00
0A78- 00 00 00 00 00 00 00 00
0A80- 00 00 00 00 00 00 00 00
0A88- 00 00 00 00 00 00 00 00
0A90- 00 00 00 00 00 00 00 00
0A98- 00 00 00 00 00 00 00 00
0AA0- 00 00 00 00 00 00 00 00
0AA8- 00 00 00 00 00 00 00 00
0AB0- 00 00 00 00 00 00 00 00

```

0AB8- 00 00 00 00 00 00 00 00
 0AC0- 00 00 00 00 00 00 00 00
 0AC8- 00 00 00 00 00 00 00 00
 0AD0- 00 00 00 00 00 00 00 00
 0AD8- 00 00 00 00 00 00 00 00
 0AE0- 00 00 00 00 00 00 00 00
 0AE8- 00 00 00 00 00 00 00 00
 0AF0- 00 00 00 00 00 00 00 00
 0AF8- 00 00 00 00 00 00 00 00

*B00.BFF

0B00- 80 83 86 89 8C 8F 92 95
 0B08- 98 9C 9F A2 A5 A8 AB AE
 0B10- B0 B3 B6 B9 BC BF C1 C4
 0B18- C7 C9 CC CE D1 D3 D5 D8
 0B20- DA DC DE E0 E2 E4 E6 E8
 0B28- EA EC ED EF F0 F2 F3 F5
 0B30- F6 F7 F8 F9 FA FB FC FC
 0B38- FD FE FE FF FF FF FF FF
 0B40- FF FF FF FF FF FF FE FE
 0B48- FD FC FC FB FA F9 F8 F7
 0B50- F6 F5 F3 F2 F0 EF ED EC
 0B58- EA E8 E6 E4 E2 E0 DE DC
 0B60- DA D8 D5 D3 D1 CE CC C9
 0B68- C7 C4 C1 BF BC B9 B6 B3
 0B70- B0 AE AB A8 A5 A2 9F 9C
 0B78- 98 95 92 8F 8C 89 86 83
 0B80- 7F 7C 79 76 73 70 6D 6A
 0B88- 67 63 60 5D 5A 57 54 51
 0B90- 4F 4C 49 46 43 40 3E 3B
 0B98- 38 36 33 31 2E 2C 2A 27
 0BA0- 25 23 21 1F 1D 1B 19 17
 0BA8- 15 13 12 10 0F 0D 0C 0A
 0BB0- 09 08 07 06 05 04 03 03
 0BB8- 02 01 01 00 00 00 00 00
 0BC0- 00 00 00 00 00 00 01 01
 0BC8- 02 03 03 04 05 06 07 08
 0BD0- 09 0A 0C 0D 0F 10 12 13
 0BD8- 15 17 19 1B 1D 1F 21 23
 0BE0- 25 27 2A 2C 2E 31 33 36
 0BE8- 38 3B 3E 40 43 46 49 4C
 0BF0- 4F 51 54 57 5A 5D 60 63
 0BF8- 67 6A 6D 70 73 76 79 7C

*C00.CFF

0C00- 39 28 1E 34 1E 39 1E 40
 0C08- 21 34 21 39 21 40 25 34
 0C10- 25 39 25 40 27 40 1E 46
 0C18- 21 46 25 46 27 46 00 00
 0C20- 41 38 1E 27 1E 2F 2F 27
 0C28- 38 1E 38 27 38 2F 41 1E
 0C30- 41 27 41 2F 2F 49 41 49
 0C38- 38 49 1E 49 27 49 00 00
 0C40- 00 00 00 00 00 00 00 00
 0C48- 00 00 00 00 00 00 00 00
 0C50- 00 00 00 00 00 00 00 00
 0C58- 00 00 00 00 00 00 00 00
 0C60- 00 00 00 00 00 00 00 00

0C68- 00 00 00 00 00 00 00 00
0C70- 00 00 00 00 00 00 00 00
0C78- 00 00 00 00 00 00 00 00
0C80- 00 00 00 00 00 00 00 00
0C88- 00 00 00 00 00 00 00 00
0C90- 00 00 00 00 00 00 00 00
0C98- 00 00 00 00 00 00 00 00
0CA0- 00 00 00 00 00 00 00 00
0CA8- 00 00 00 00 00 00 00 00
0CB0- 00 00 00 00 00 00 00 00
0CB8- 00 00 00 00 00 00 00 00
0CC0- 00 00 00 00 00 00 00 00
0CC8- 00 00 00 00 00 00 00 00
0CD0- 00 00 00 00 00 00 00 00
0CD8- 00 00 00 00 00 00 00 00
0CE0- 00 00 00 00 00 00 00 00
0CE8- 00 00 00 00 00 00 00 00
0CF0- 00 00 00 00 00 00 00 00
0CF8- 00 00 00 00 00 00 00 00

*

```

1 REM      DIMENSION VER 1.4 8/12/7
7
20 CON=8192:AQ=4:AW=8704
100 POKE 2,100: POKE 3,2
110 AD=768: FOR A=0 TO 255: POKE
AD+A,255: NEXT A
120 CALL -936
200 VTAB 4: INPUT "AREA,PRE,NUMB"
,AC,PRE,NUMB
210 IF AC=0 THEN 310
300 NUM=AC:DIG=3: GOSUB 5000
310 NUM=PRE:DIG=3: GOSUB 5000
320 NUM=NUMB:DIG=4: GOSUB 5000
325 ADR=AD+1
400 INPUT "START,END CODE",CODE,
ECOD
1000 PH= PEEK (-16228):DLY=350: GOSUB
6000
1100 PH= PEEK (-16227)
1110 DL1=100: POKE -16231,5:T1=10
: GOSUB 7000: IF T1<10 THEN
1000
1115 AD=768: GOSUB 8210
1140 DL1=800:T1=20: POKE -16231,
5: GOSUB 7000
1145 IF T<T1 THEN 1000
1150 AD=ADR:NUM=CODE:DIG=4: GOSUB
5000:AD=ADR: GOSUB 8210
1155 PRINT "CODE=";CODE
1160 POKE -16231,19:DL1=100:T1=10
: GOSUB 7000: IF T=T1 THEN
1500
1168 PRINT "";
1170 DL1=100:T1=20: POKE -16231,
6: GOSUB 7000: IF T<T1 THEN
1000:TRY=TRY+1: PRINT "TRYING "
;TRY: IF TRY<5 THEN 1000
1180 FOR A=1 TO 50: PRINT "":; NEXT
A: PRINT "WHOOPIE!!!! CODE="
;CODE:PH= PEEK (-16228): END

1500 TRY=0: IF CODE=ECOD THEN 2000
:CODE=CODE+1: GOTO 1000
2000 PH= PEEK (-16228): END
2999 CALL -936
3000 POKE -16167, PDL (0)/3:T=( PEEK
(-16238)>127): VTAB 10: PRINT
"STONE=";T: PRINT "VAL="; PDL
(0)/3: GOTO 3000
4500 POKE -16231,9: GOSUB 7000: PRINT
"T=";T: GOTO 4500
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN

```

```
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127): VTAB
10: PRINT "VAL="; PDL (0)/4
+PH: PRINT "TONE=";Y: GOTO
8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (3072+(2* PEEK (
      AD+DGT))):TONB= PEEK (3072+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END
```

```
10 DIM A$(128),L(256),B$(40)
15 DIM C$(40)
20 DIALPULSE=1000:ONTIME=2:OFFTIME=
  2:INTERDIG=75:BEEP=2600
30 TTN=3072:MF=3104:KP=1100:ST=
  1500:POKE 2,100:POKE 3,2
40 AD=2048:L(1)=16384:REENTER=
  820:POKE 12,0:POKE 13,64:
  STUFF=768
50 CALL REENTER
60 PTR=PEEK(14)+256*PEEK(15)
  ):LINE=PEEK(17)+256*PEEK
  (18)+1
70 IF LINE<=0 THEN LINE=1
80 TONEDIAL=8210:PLL=7000
90 PW=0
100 REM -----ROUTINES
110 INIT=1110:ENTNUM=1205:LOOKUP=
  1310
130 NUMS=3328:GOSUB INIT
140 INPUT "PRINT NUMBERS?",B$:IF
  B$(1,1)="N" THEN 200:INPUT
  "CODE=",A$
150 IF A$="ASDFGHJKL" THEN PW=1

200 CALL -936:INPUT "NAME?",B$

210 IF B$="EN" THEN 1400:REM ENTER

215 IF B$#"69" THEN 220
218 PW=1:GOTO 200
220 IF B$="FIX" THEN 0:REM FIX
225 IF B$#"00" THEN 230
228 PW=0:GOTO 200
230 IF B$="DIS" THEN 1500:REM DISP
  LAY
240 IF B$="END" THEN END
250 IF B$="LOOK" THEN 2000
260 IF B$(1,1)#"- " THEN 300
270 FLAG=1:B$=B$(2)
300 FOR L=1 TO LINE:POKE 12,L(
  L) MOD 256:POKE 13,L(L)/256
  :CALL STUFF
310 IF B$=A$ THEN 500:NEXT L:PRINT
  "SORRY, NO SUCH ENTRY"
320 INPUT "WANNA SEE THE NAMES?"
  ,B$:IF B$(1,1)="Y" THEN 200

330 INPUT "ENTER NUMBER--ANY FORMAT:
  ",B$:GOTO 590
480 GOSUB INIT
500 PRINT A$;" ";:GOSUB LOOKUP
510 IF PW=1 THEN PRINT B$:PRINT
  :IF FLAG#1 THEN 520
515 DLY=900:GOSUB 6000:FLAG=0:
  GOTO 200
```



```

520 PT=NUMS+6*(L-1):PX= PEEK (PT+
    2)+256* PEEK (PT+3)
530 IF PX#0 THEN 590
540 VTAB 20: PRINT "NO PHONE NUMBER
    FOR ",A$:DLY=350: GOSUB 6000
    : GOTO 200
590 VTAB 20
600 INPUT "WANNA USE A WATS EXTENDER
    ?",A$: IF A$(1,1)="Y" THEN
    670: IF A$(1,1)="P" THEN 200

610 START=2048:A$=B$: GOSUB ENTNUM:
    AD=2048
615 PRINT "PLEASE PICK UP PHONE"

620 POKE -16228,0:DLY=350: GOSUB
    6000: POKE -16227,0
630 DL1=100: POKE -16231,4:T1=5
    : GOSUB PLL
640 IF T<T1 THEN 620: GOSUB DIALPULS
    E: PRINT
650 INPUT "WANNA TRY AGAIN?",A$
    :AD=2048: IF A$(1,1)="Y" THEN
    620
660 GOTO 200
670 CALL -936: PRINT "WHICH ONE?"

672 PRINT "1. NEW JERSEY #1"
673 PRINT "2. NEW JERSEY #2"
674 PRINT "3. ATLANTA"
675 PRINT "4. FLORIDA"
676 PRINT "5. MINNIAPOLIS"
680 INPUT "TYPE 1-5 :",WH: IF WH<
    1 OR WH>5 THEN 670
690 IF WH=1 THEN A$="18006310725*555
    5*85*"
700 IF WH=2 THEN A$="18006310744*134
    3*85*"
710 IF WH=3 THEN A$="18002413466*551
    1*85*"
720 IF WH=4 THEN A$="18003273282*370
    2*81*"
730 IF WH=5 THEN A$="18003288308*106
    0*85*"
740 A$(1+ LEN(A$))=B$
750 GOSUB INIT: PRINT "PLEASE PICK U
    P PHONE"
760 START=2048: GOSUB ENTNUM
770 AD=2048
780 POKE -16228,0:DLY=350: GOSUB
    6000: POKE -16227,0
790 DL1=100: POKE -16231,4:T1=5
    : GOSUB PLL
800 IF T<T1 THEN 780: GOSUB DIALPULS
    E:TBL=TTN
810 DL1=600:T1=10: POKE -16231,

```

```
6: GOSUB 7000
820 IF T=T1 THEN 850
830 CALL -936: INPUT "YOUR CALL FAIL
ED--WANNA TRY AGAIN",C$
840 IF C$(1,1)="Y" THEN 770: IF
C$(1,1)="T" THEN 670: GOTO
200
850 AD=AD+1: GOSUB TONEDIAL:DL1=
50: POKE -16231,20:T1=10: GOSUB
PLL
860 IF T<T1 THEN 880
870 PRINT "THE CODE WAS CHANGED-TRY
ANOTHER WATS EXTENDER":DLY=
800: GOSUB 6000: GOTO 670
880 DL1=100: POKE -16231,6:T1=5
: GOSUB PLL: IF T=T1 THEN 850
: PRINT
890 INPUT "IS YOUR CALL SUCESSFUL?"
,C$: IF C$(1,1)="N" THEN 770
: IF C$(1,1)="T" THEN 670
900 GOTO 200
1000 REM ---DIAL PULSE ROUTINE--
1010 IF PEEK (AD)=255 THEN RETURN

1020 COUNT= PEEK (AD)
1025 IF PW=1 THEN PRINT COUNT;
1030 IF COUNT=0 THEN COUNT=10
1040 FOR PULSE=1 TO COUNT
1050 POKE -16228,0:DLY=ONTIME: GOSUB
6000
1060 POKE -16227,0:DLY=OFFTIME: GOSUB
6000
1070 NEXT PULSE
1080 DLY=INTERDIG: GOSUB 6000:AD=
AD+1: GOTO 1010
1110 FOR A=2048 TO 2303: POKE A,
255: NEXT A: RETURN
1205 ADS=START
1210 FOR P=1 TO LEN (A$)
1220 VAL= ASC (A$(P))
1230 IF VAL=170 THEN 1260
1240 IF VAL<176 OR VAL>185 THEN
1270
1250 POKE ADS,VAL-176
1260 ADS=ADS+1
1270 NEXT P: RETURN
1310 POINT=NUMS+6*(L-1):AC= PEEK
(PPOINT)+256* PEEK (PPOINT+1)

1320 PRE= PEEK (PPOINT+2)+256* PEEK
(PPOINT+3)
1330 NUMB= PEEK (PPOINT+4)+256* PEEK
(PPOINT+5)
1340 AD=5777:DIG=4:NUM=AC: GOSUB
5000
1350 DIG=3:NUM=PRE: GOSUB 5000
```

```
1360 DIG=4:NUM=NUMB: GOSUB 5000:
      ADR=AD+1: RETURN
1400 CALL -936
1410 INPUT "NAME:",A$: IF A$="END"
      THEN 200
1420 FOR I=1 TO LEN(A$): POKE PTR,
      ASC(A$(I)):PTR=PTR+1: NEXT
      I
1430 LINE=LINE+1: POKE PTR,30:PTR=
      PTR+1:L(LINE)=PTR: POKE PTR,
      0
1440 INPUT "NUMBER--AC,PRE,NUMB:"
      ,AC,PRE,NUMB
1450 PT=NUMS+6*(LINE-2)
1460 POKE PT,AC MOD 256: POKE PT+
      1,AC/256
1470 POKE PT+2,PRE MOD 256: POKE
      PT+3,PRE/256
1480 POKE PT+4,NUMB MOD 256: POKE
      PT+5,NUMB/256
1490 GOTO 1410
1500 CALL -936
1505 VTAB 22: PRINT "TYPE 'E' TO EXIT
      ": INPUT "LINE=",L: GOTO 1590

1510 KEY= PEEK (-16384): POKE -16368
      ,0
1520 IF KEY=197 THEN 200
1530 IF KEY=160 THEN 1580
1540 IF KEY=204 THEN 1500
1550 IF KEY=208 THEN 1570
1560 GOTO 1510
1570 L=L+21
1580 L=L+1
1585 CALL -936
1590 IF L>LINE THEN 1505: IF L<=
      0 THEN L=1
1595 POKE 16,L: CALL 901
1598 GOTO 1510
2000 CALL -936: IF PW=0 THEN 200

2010 INPUT "LINE #",L: IF L>LINE OR
      L<0 THEN 2010
2020 KEY= PEEK (-16384): POKE -16368
      ,0
2030 IF KEY=197 THEN 200: IF KEY=
      193 THEN 2050
2035 IF KEY=195 THEN 2130
2040 IF KEY=204 THEN 2010
2045 GOTO 2020
2050 L=L+1
2055 CALL -936
2060 POKE 12,L(L) MOD 256: POKE
      13,L(L)/256: CALL STUFF
2070 VTAB 2: PRINT "
```

```

2080 POINT=NUMS+6*(L-1)
2090 AC= PEEK (POINT)+256*(( PEEK
    (POINT+1)) MOD 127)
2100 PRE= PEEK (POINT+2)+256*(( PEEK
    (POINT+3)) MOD 127)
2110 NUMB= PEEK (POINT+4)+256*((
    PEEK (POINT+5)) MOD 127)
2120 VTAB 2: PRINT L;" ";A$;" ";
    AC;"-";PRE;"-";NUMB: GOTO 2020

2130 VTAB 4: PRINT "OLD=";AC;"-"
    ;PRE;"-";NUMB
2140 VTAB 5: INPUT "NEW=",AC,PRE,
    NUMB
2145 POKE POINT,AC MOD 256: POKE
    POINT+1,AC/256
2150 POKE POINT+2,PRE MOD 256: POKE
    POINT+3,PRE/256
2160 POKE POINT+4,NUMB MOD 256: POKE
    POINT+5,NUMB/256: GOTO 2020

5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A+176
    :AD=AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW: POKE AD,30
5030 RETURN
6000 FOR D=1 TO DLY: NEXT D: RETURN

7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
8000 KEY= PEEK (-16384): POKE -16368
    ,0: IF KEY>127 THEN PRINT KEY:
    GOTO 8000
8210 REM TONEDIAL ROUTINE
8220 IF PEEK (AD)=255 THEN RETURN

8225 IF PW=1 THEN PRINT PEEK (AD)
    ;
8230 TONA= PEEK (TBL+(2* PEEK (AD)
    )):TONB= PEEK (TBL+(2* PEEK
    (AD) )+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
    2560
8260 AD=AD+1: GOTO 8220
30000 END

```

```

10 DIM A$(128),L(256),B$(40)
15 DIM C$(40)
20 DIALPULSE=1000:ONTIME=2:OFFTIME=
  2:INTERDIG=75:BEEP=2600
30 TTN=3072:MF=3104:KP=1100:ST=
  1500:POKE 2,100:POKE 3,2
40 AD=2048:L(1)=16384:REENTER=
  820:POKE 12,0:POKE 13,64:
  STUFF=768
50 CALL REENTER
60 PTR=PEEK(14)+256*PEEK(15)
  ):LINE=PEEK(17)+256*PEEK
  (18)+1
70 IF LINE<=0 THEN LINE=1
80 TONEDIAL=8210:PLL=7000
90 PW=0
100 REM -----ROUTINES
110 INIT=1110:ENTNUM=1205:LOOKUP=
  1310
130 NUMS=3328:GOSUB INIT
140 INPUT "PRINT NUMBERS?",B$:IF
  B$(1,1)="N" THEN 200:INPUT
  "CODE=",A$
150 IF A$="ASDFGHJKL" THEN PW=1

200 CALL -936:INPUT "NAME?",B$

210 IF B$="EN" THEN 1400:REM ENTER

215 IF B$#"69" THEN 220
218 PW=1:GOTO 200
220 IF B$="FIX" THEN 0:REM FIX
225 IF B$#"00" THEN 230
228 PW=0:GOTO 200
230 IF B$="DIS" THEN 1500:REM DISP
  LAY
240 IF B$="END" THEN END
250 IF B$="LOOK" THEN 2000
260 IF B$(1,1)#"-" THEN 300
270 FLAG=1:B$=B$(2)
300 FOR L=1 TO LINE:POKE 12,L(
  L) MOD 256:POKE 13,L(L)/256
  :CALL STUFF
310 IF B$=A$ THEN 500:NEXT L:PRINT
  "SORRY, NO SUCH ENTRY"
320 INPUT "WANNA SEE THE NAMES?"
  ,B$:IF B$(1,1)="Y" THEN 200

330 INPUT "ENTER NUMBER--ANY FORMAT:
  ",B$:GOTO 590
480 GOSUB INIT
500 PRINT A$;" ";:GOSUB LOOKUP
510 IF PW=1 THEN PRINT B$:PRINT
  :IF FLAG#1 THEN 520
515 DLY=900:GOSUB 6000:FLAG=0:
  GOTO 200

```

```
520 PT=NUMS+6*(L-1):PX= PEEK (PT+
    2)+256* PEEK (PT+3)
530 IF PX#0 THEN 590
540 VTAB 20: PRINT "NO PHONE NUMBER
    FOR ",A$:DLY=350: GOSUB 6000
    : GOTO 200
590 VTAB 20
600 INPUT "WANNA USE A WATS EXTENDER
    ?",A$: IF A$(1,1)="Y" THEN
    670: IF A$(1,1)="P" THEN 200

610 START=2048:A$=B$: GOSUB ENTNUM:
    AD=2048
615 PRINT "PLEASE PICK UP PHONE"

620 POKE -16228,0:DLY=350: GOSUB
    6000: POKE -16227,0
630 DL1=100: POKE -16231,4:T1=5
    : GOSUB PLL
640 IF T<T1 THEN 620: GOSUB DIALPULS
    E: PRINT
650 INPUT "WANNA TRY AGAIN?",A$
    :AD=2048: IF A$(1,1)="Y" THEN
    620
660 GOTO 200
670 CALL -936: PRINT "WHICH ONE?"

672 PRINT "1. NEW JERSEY #1"
673 PRINT "2. NEW JERSEY #2"
674 PRINT "3. ATLANTA"
675 PRINT "4. FLORIDA"
676 PRINT "5. MINNIAPOLIS"
680 INPUT "TYPE 1-5 : ",WH: IF WH<
    1 OR WH>5 THEN 670
690 IF WH=1 THEN A$="18006310725*555
    5*85*"
700 IF WH=2 THEN A$="18006310744*134
    3*85*"
710 IF WH=3 THEN A$="18002413466*551
    1*85*"
720 IF WH=4 THEN A$="18003273282*370
    2*81*"
730 IF WH=5 THEN A$="18003288308*106
    0*85*"
740 A$(1+ LEN(A$))=B$
750 GOSUB INIT: PRINT "PLEASE PICK U
    P PHONE"
760 START=2048: GOSUB ENTNUM
770 AD=2048
780 POKE -16228,0:DLY=350: GOSUB
    6000: POKE -16227,0
790 DL1=100: POKE -16231,4:T1=5
    : GOSUB PLL
800 IF T<T1 THEN 780: GOSUB DIALPULS
    E:TBL=TTN
810 DL1=600:T1=10: POKE -16231,
```

```

6: GOSUB 7000
820 IF T=T1 THEN 850
830 CALL -936: INPUT "YOUR CALL FAIL
ED--WANNA TRY AGAIN",C$
840 IF C$(1,1)="Y" THEN 770: IF
C$(1,1)="T" THEN 670: GOTO
200
850 AD=AD+1: GOSUB TONEDIAL:DL1=
50: POKE -16231,20:T1=10: GOSUB
PLL
860 IF T<T1 THEN 880
870 PRINT "THE CODE WAS CHANGED-TRY
ANOTHER WATS EXTENDER":DLY=
800: GOSUB 6000: GOTO 670
880 DL1=100: POKE -16231,6:T1=5
: GOSUB PLL: IF T=T1 THEN 850
: PRINT
890 INPUT "IS YOUR CALL SUCESSFUL?"
,C$: IF C$(1,1)="N" THEN 770
: IF C$(1,1)="T" THEN 670
900 GOTO 200
1000 REM ---DIAL PULSE ROUTINE--
1010 IF PEEK (AD)=255 THEN RETURN

1020 COUNT= PEEK (AD)
1025 IF PW=1 THEN PRINT COUNT;
1030 IF COUNT=0 THEN COUNT=10
1040 FOR PULSE=1 TO COUNT
1050 POKE -16228,0:DLY=ONTIME: GOSUB
6000
1060 POKE -16227,0:DLY=OFFTIME: GOSUB
6000
1070 NEXT PULSE
1080 DLY=INTERDIG: GOSUB 6000:AD=
AD+1: GOTO 1010
1110 FOR A=2048 TO 2303: POKE A,
255: NEXT A: RETURN
1205 ADS=START
1210 FOR P=1 TO LEN(A$)
1220 VAL= ASC(A$(P))
1230 IF VAL=170 THEN 1260
1240 IF VAL<176 OR VAL>185 THEN
1270
1250 POKE ADS,VAL-176
1260 ADS=ADS+1
1270 NEXT P: RETURN
1310 POINT=NUMS+6*(L-1):AC= PEEK
(PPOINT)+256* PEEK (POINT+1)

1320 PRE= PEEK (POINT+2)+256* PEEK
(PPOINT+3)
1330 NUMB= PEEK (POINT+4)+256* PEEK
(PPOINT+5)
1340 AD=5777: DIG=4: NUM=AC: GOSUB
5000
1350 DIG=3: NUM=PRE: GOSUB 5000

```

```

1360 DIG=4:NUM=NUMB: GOSUB 5000:
      ADR=AD+1: RETURN
1400 CALL -936
1410 INPUT "NAME:",A$: IF A$="END"
      THEN 200
1420 FOR I=1 TO LEN(A$): POKE PTR,
      ASC(A$(I)):PTR=PTR+1: NEXT
      I
1430 LINE=LINE+1: POKE PTR,30:PTR=
      PTR+1:L(LINE)=PTR: POKE PTR,
      0
1440 INPUT "NUMBER--AC,PRE,NUMB:"
      ,AC,PRE,NUMB
1450 PT=NUMS+6*(LINE-2)
1460 POKE PT,AC MOD 256: POKE PT+
      1,AC/256
1470 POKE PT+2,PRE MOD 256: POKE
      PT+3,PRE/256
1480 POKE PT+4,NUMB MOD 256: POKE
      PT+5,NUMB/256
1490 GOTO 1410
1500 CALL -936
1505 VTAB 22: PRINT "TYPE 'E' TO EXIT
      ": INPUT "LINE=",L: GOTO 1590

1510 KEY= PEEK (-16384): POKE -16368
      ,0
1520 IF KEY=197 THEN 200
1530 IF KEY=160 THEN 1580
1540 IF KEY=204 THEN 1500
1550 IF KEY=208 THEN 1570
1560 GOTO 1510
1570 L=L+21
1580 L=L+1
1585 CALL -936
1590 IF L>LINE THEN 1505: IF L<=
      0 THEN L=1
1595 POKE 16,L: CALL 901
1598 GOTO 1510
2000 CALL -936: IF PW=0 THEN 200

2010 INPUT "LINE #",L: IF L>LINE OR
      L<0 THEN 2010
2020 KEY= PEEK (-16384): POKE -16368
      ,0
2030 IF KEY=197 THEN 200: IF KEY=
      193 THEN 2050
2035 IF KEY=195 THEN 2130
2040 IF KEY=204 THEN 2010
2045 GOTO 2020
2050 L=L+1
2055 CALL -936
2060 POKE 12,L(L) MOD 256: POKE
      13,L(L)/256: CALL STUFF
2070 VTAB 2: PRINT "

```



```

2080 POINT=NUMS+6*(L-1)
2090 AC= PEEK (POINT)+256*(( PEEK
    (POINT+1)) MOD 127)
2100 PRE= PEEK (POINT+2)+256*(( PEEK
    (POINT+3)) MOD 127)
2110 NUMB= PEEK (POINT+4)+256*((
    PEEK (POINT+5)) MOD 127)
2120 VTAB 2: PRINT L;" ";A$;" ";
    AC;"-";PRE;"-";NUMB: GOTO 2020

2130 VTAB 4: PRINT "OLD=";AC;"-"
    ;PRE;"-";NUMB
2140 VTAB 5: INPUT "NEW=",AC,PRE,
    NUMB
2145 POKE POINT,AC MOD 256: POKE
    POINT+1,AC/256
2150 POKE POINT+2,PRE MOD 256: POKE
    POINT+3,PRE/256
2160 POKE POINT+4,NUMB MOD 256: POKE
    POINT+5,NUMB/256: GOTO 2020

5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A+176
    :AD=AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW: POKE AD,30
5030 RETURN
6000 FOR D=1 TO DLY: NEXT D: RETURN

7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
8000 KEY= PEEK (-16384): POKE -16368
    ,0: IF KEY>127 THEN PRINT KEY:
    GOTO 8000
8210 REM TONEDIAL ROUTINE
8220 IF PEEK (AD)=255 THEN RETURN

8225 IF PW=1 THEN PRINT PEEK (AD)
    ;
8230 TONA= PEEK (TBL+(2* PEEK (AD)
    )):TONB= PEEK (TBL+(2* PEEK
    (AD) )+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
8260 AD=AD+1: GOTO 8220
30000 END

```

```

10 DIM A$(128),L(256),B$(40)
20 DIALPULSE=1000:ONTIME=4:OFFTIME=
  4:INTERDIG=75:BEEP=2600
30 TTN=3072:MF=3104:KP=1100:ST=
  1500:POKE 2,100:POKE 3,2
40 AD=2048:L(1)=16384:REENTER=
  820:POKE 12,0:POKE 13,64:
  STUFF=768
50 CALL REENTER
60 PTR=PEEK(14)+256*PEEK(15)
  ):LINE=PEEK(17)+256*PEEK
  (18)+1
70 IF LINE<=0 THEN LINE=1
80 TONEDIAL=8210:PLL=7000
90 PW=0
100 REM -----ROUTINES
110 INIT=1110:ENTNUM=1205:LOOKUP=
  1310
130 NUMS=3328:GOSUB INIT
140 INPUT "PRINT NUMBERS?",B$:IF
  B$(1,1)="N" THEN 200:INPUT
  "CODE=",A$
150 IF A$="ASDFGHJKL" THEN PW=1

200 CALL -936:INPUT "NAME?",B$

210 IF B$="EN" THEN 1400:REM ENTER

220 IF B$="FIX" THEN 0:REM FIX
230 IF B$="DIS" THEN 1500:REM DISP
  LAY
240 IF B$="END" THEN END
250 IF B$="LOOK" THEN 2000
300 FOR L=1 TO LINE:POKE 12,L(
  L) MOD 256:POKE 13,L(L)/256
  :CALL STUFF
310 IF B$=A$ THEN 500:NEXT L:PRINT
  "SORRY, NO SUCH ENTRY"
320 INPUT "WANNA SEE THE NAMES?"
  ,B$:IF B$(1,1)="Y" THEN 200

330 INPUT "ENTER NUMBER--ANY FORMAT:
  ",B$:GOTO 510
480 GOSUB INIT
500 PRINT A$;" ";:GOSUB LOOKUP
510 IF PW=1 THEN PRINT B$
512 PRINT
515 VTAB 20
600 INPUT "WANNA USE A WATS EXTENDER
  ?",A$:IF A$(1,1)="Y" THEN
  670
610 START=2048:A$=B$:GOSUB ENTNUM:
  AD=2048
620 POKE -16228,0:DLY=350:GOSUB
  6000:POKE -16227,0
630 DL1=100:POKE -16231,4:T1=5

```

```

: GOSUB PLL
640 IF T<T1 THEN 620: GOSUB DIALPULS
E: PRINT
650 INPUT "WANNA TRY AGAIN?",A$
:AD=2048: IF A$(1,1)="Y" THEN
620
660 GOTO 200
670 CALL -936: PRINT "WHICH ONE?"

672 PRINT "1. NEW JERSEY #1"
673 PRINT "2. NEW JERSEY #2"
674 PRINT "3. ATLANTA"
675 PRINT "4. FLORIDA"
676 PRINT "5. MINNIAPOLIS"
680 INPUT "TYPE 1-5 :","WH: IF WH<
1 OR WH>5 THEN 670
690 IF WH=1 THEN A$="18006310725*555
5*85*"
700 IF WH=2 THEN A$="18006310744*134
3*85*"
710 IF WH=3 THEN A$="18002413466*551
1*85*"
720 IF WH=4 THEN A$="18003273282*370
2*81*"
730 IF WH=5 THEN A$="18003288308*106
0*85*"
740 A$(1+ LEN(A$))=B$
750 GOSUB INIT: PRINT "PLEASE PICK U
P PHONE"
760 START=2048: GOSUB ENTNUM
770 AD=2048
780 POKE -16228,0:DLY=350: GOSUB
6000: POKE -16227,0
790 DL1=100: POKE -16231,4:T1=5
: GOSUB PLL
800 IF T<T1 THEN 780: GOSUB DIALPULS
E:TBL=TTN
810 DL1=800:T1=10: POKE -16231,
6: GOSUB 7000
820 IF T=T1 THEN 850
830 CALL -936: INPUT "YOUR CALL FAIL
ED--WANNA TRY AGAIN",B$
840 IF B$(1,1)="Y" THEN 770: GOTO
200
850 AD=AD+1: GOSUB TONEDIAL:DL1=
100: POKE -16231,20:T1=5: GOSUB
PLL
860 IF T<T1 THEN 880
870 PRINT "THE CODE WAS CHANGED-TRY
ANOTHER WATS EXTENDER": GOTO
670
880 DL1=100: POKE -16231,6:T1=5
: GOSUB PLL: IF T=T1 THEN 850
: PRINT
890 INPUT "IS YOUR CALL SUCESSFUL?"
,B$: IF B$(1,1)="N" THEN 770

```

```
900 GOTO 200
1000 REM ---DIAL PULSE ROUTINE--
1010 IF PEEK (AD)=255 THEN RETURN

1020 COUNT= PEEK (AD)
1025 PRINT COUNT;
1030 IF COUNT=0 THEN COUNT=10
1040 FOR PULSE=1 TO COUNT
1050 POKE -16228,0:DLY=ONTIME: GOSUB
    6000
1060 POKE -16227,0:DLY=OFFTIME: GOSUB
    6000
1070 NEXT PULSE
1080 DLY=INTERDIG: GOSUB 6000:AD=
    AD+1: GOTO 1010
1110 FOR A=2048 TO 2303: POKE A,
    255: NEXT A: RETURN
1205 ADS=START
1210 FOR P=1 TO LEN (A$)
1220 VAL= ASC (A$(P))
1230 IF VAL=170 THEN 1260
1240 IF VAL<176 OR VAL>185 THEN
    1270
1250 POKE ADS,VAL-176
1260 ADS=ADS+1
1270 NEXT P: RETURN
1310 POINT=NUMS+6*(L-1):AC= PEEK
    (POINT)+256* PEEK (POINT+1)

1320 PRE= PEEK (POINT+2)+256* PEEK
    (POINT+3)
1330 NUMB= PEEK (POINT+4)+256* PEEK
    (POINT+5)
1340 AD=5777:DIG=4:NUM=AC: GOSUB
    5000
1350 DIG=3:NUM=PRE: GOSUB 5000
1360 DIG=4:NUM=NUMB: GOSUB 5000:
    ADR=AD+1: RETURN
1400 PT=NUMS
1410 FOR L=1 TO LINE: POKE 12,L(
    L) MOD 256: POKE 13,L(L)/256

1420 CALL STUFF
1430 PRINT L;" ";A$;"::: INPUT
    "AC,PRE,NUMB:",AC,PRE,NUMB
1440 IF AC=-1 THEN 1450: IF AC=-
    2 THEN 200
1445 POKE PT,AC MOD 256: POKE PT+
    1,AC/256: POKE PT+2,PRE MOD
    256
1447 POKE PT+3,PRE/256: POKE PT+
    4,NUMB MOD 256: POKE PT+5,NUMB/
    256
1450 PT=PT+6
1460 NEXT L: GOTO 200
```

```

1500 CALL -936
1505 VTAB 22: PRINT "TYPE 'E' TO EXIT
      "
1510 KEY= PEEK (-16384): POKE -16368
      ,0
1520 IF KEY=197 THEN 200
1530 P= PDL (0)/2: IF PDL (0)/2=
      P THEN 1510
1540 CALL -936:L= PDL (0)/2-6: IF
      L>LINE THEN 1510: IF L<=0 THEN
      L=1
1550 POKE 16,L: CALL 901
1560 GOTO 1510
2000 CALL -936: IF PW=0 THEN 200

2010 INPUT "LINE #",L
2020 KEY= PEEK (-16384): POKE -16368
      ,0
2030 IF KEY=197 THEN 200: IF KEY=
      193 THEN 2055
2040 IF KEY=204 THEN 2010
2045 GOTO 2020
2055 L=L+1: CALL -936
2060 POKE 12,L(L) MOD 256: POKE
      13,L(L)/256: CALL STUFF
2070 VTAB 2: PRINT "
      "

2080 POINT=NUMS+6*(L-1)
2090 AC= PEEK (POINT)+256*(( PEEK
      (POINT+1)) MOD 127)
2100 PRE= PEEK (POINT+2)+256*(( PEEK
      (POINT+3)) MOD 127)
2110 NUMB= PEEK (POINT+4)+256*((
      PEEK (POINT+5)) MOD 127)
2120 VTAB 2: PRINT L;" ";A$;" ";
      AC;"-";PRE;"-";NUMB: GOTO 2020

5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A+176
      :AD=AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW: POKE AD,30
5030 RETURN
6000 FOR D=0 TO DLY: NEXT D: RETURN

7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN I=T+1
7040 IF T=TN1 THEN RETURN
7060 NEXT DL
7070 RETURN
8210 REM TONEDIAL ROUTINE
8220 IF PEEK (AD)=255 THEN RETURN

8230 TONA= PEEK (TBL+(2* PEEK (AD)

```

```
    ):TONB= PEEK (TBL+(2* PEEK  
    (AD) )+1)  
8240 POKE 0,TONA: POKE 1,TONB: CALL  
    2560  
8260 AD=AD+1: GOTO 8220  
30000 END
```

```

10 DIM A$(128),L(256),B$(40)
15 DIM C$(40)
20 DIALPULSE=1000:ONTIME=2:OFFTIME=
  2:INTERDIG=75:BEEP=2600
30 TTN=3072:MF=3104:KP=1100:ST=
  1500:POKE 2,100:POKE 3,2
40 AD=2048:L(1)=16384:REENTER=
  820:POKE 12,0:POKE 13,64:
  STUFF=768
50 CALL REENTER
60 PTR=PEEK(14)+256*PEEK(15)
  ):LINE=PEEK(17)+256*PEEK
  (18)+1
70 IF LINE<=0 THEN LINE=1
80 TONEDIAL=8210:PLL=7000
90 PW=0
100 REM -----ROUTINES
110 INIT=1110:ENTNUM=1205:LOOKUP=
  1310
130 NUMS=3328:GOSUB INIT
140 INPUT "PRINT NUMBERS?",B$:IF
  B$(1,1)="N" THEN 200:INPUT
  "CODE=",A$
150 IF A$="ASDFGHJKL" THEN PW=1

200 CALL -936:INPUT "NAME?",B$

210 IF B$="EN" THEN 1400:REM ENTER

215 IF B$#"69" THEN 220
218 PW=1:GOTO 200
220 IF B$="FIX" THEN 0:REM FIX
225 IF B$#"00" THEN 230
228 PW=0:GOTO 200
230 IF B$="DIS" THEN 1500:REM DISP
  LAY
240 IF B$="END" THEN END
250 IF B$="LOOK" THEN 2000
260 IF B$(1,1)#"-" THEN 300
270 FLAG=1:B$=B$(2)
300 FOR L=1 TO LINE:POKE 12,L(
  L) MOD 256:POKE 13,L(L)/256
  :CALL STUFF
310 IF B$=A$ THEN 500:NEXT L:PRINT
  "SORRY, NO SUCH ENTRY"
320 INPUT "WANNA SEE THE NAMES?"
  ,B$:IF B$(1,1)="Y" THEN 200

330 INPUT "ENTER NUMBER--ANY FORMAT:
  ",B$:GOTO 590
480 GOSUB INIT
500 PRINT A$;" ";:GOSUB LOOKUP
510 IF PW=1 THEN PRINT B$:PRINT
  :IF FLAG#1 THEN 520
515 DLY=900:GOSUB 6000:FLAG=0:
  GOTO 200

```

```

520 PT=NUMS+6*(L-1):PX= PEEK (PT+
    2)+256* PEEK (PT+3)
530 IF PX#0 THEN 590
540 VTAB 20: PRINT "NO PHONE NUMBER
    FOR ",A$:DLY=350: GOSUB 6000
    : GOTO 200
590 VTAB 20
600 INPUT "WANNA USE A WATS EXTENDER
    ?",A$: IF A$(1,1)="Y" THEN
    670: IF A$(1,1)="P" THEN 200

610 START=2048:A$=B$: GOSUB ENTNUM:
    AD=2048
615 PRINT "PLEASE PICK UP PHONE"

620 POKE -16228,0:DLY=350: GOSUB
    6000: POKE -16227,0
630 DL1=100: POKE -16231,4:T1=5
    : GOSUB PLL
640 IF T<T1 THEN 620: GOSUB DIALPULS
    E: PRINT
650 INPUT "WANNA TRY AGAIN?",A$
    :AD=2048: IF A$(1,1)="Y" THEN
    620
660 GOTO 200
670 CALL -936: PRINT "WHICH ONE?"

672 PRINT "1. NEW JERSEY #1"
673 PRINT "2. NEW JERSEY #2"
674 PRINT "3. ATLANTA"
675 PRINT "4. FLORIDA"
676 PRINT "5. MINNIAPOLIS"
680 INPUT "TYPE 1-5 :",WH: IF WH<
    1 OR WH>5 THEN 670
690 IF WH=1 THEN A$="18006310725*555
    5*85*"
700 IF WH=2 THEN A$="18006310744*134
    3*85*"
710 IF WH=3 THEN A$="18002413466*551
    1*85*"
720 IF WH=4 THEN A$="18003273282*370
    2*81*"
730 IF WH=5 THEN A$="18003288308*106
    0*85*"
740 A$(1+ LEN(A$))=B$
750 GOSUB INIT: PRINT "PLEASE PICK U
    P PHONE"
760 START=2048: GOSUB ENTNUM
770 AD=2048
780 POKE -16228,0:DLY=350: GOSUB
    6000: POKE -16227,0
790 DL1=100: POKE -16231,4:T1=5
    : GOSUB PLL
800 IF T<T1 THEN 780: GOSUB DIALPULS
    E:TBL=TTN
810 DL1=600:T1=10: POKE -16231,

```



```
6: GOSUB 7000
820 IF T=T1 THEN 850
830 CALL -936: INPUT "YOUR CALL FAIL
ED--WANNA TRY AGAIN",C$
840 IF C$(1,1)="Y" THEN 770: IF
C$(1,1)="T" THEN 670: GOTO
200
850 AD=AD+1: GOSUB TONEDIAL:DL1=
50: POKE -16231,20:T1=10: GOSUB
PLL
860 IF T<T1 THEN 880
870 PRINT "THE CODE WAS CHANGED-TRY
ANOTHER WATS EXTENDER":DLY=
800: GOSUB 6000: GOTO 670
880 DL1=100: POKE -16231,6:T1=5
: GOSUB PLL: IF T=T1 THEN 850
: PRINT
890 INPUT "IS YOUR CALL SUCESSFUL?"
,C$: IF C$(1,1)="N" THEN 770
: IF C$(1,1)="T" THEN 670
900 GOTO 200
1000 REM ---DIAL PULSE ROUTINE--
1010 IF PEEK (AD)=255 THEN RETURN

1020 COUNT= PEEK (AD)
1025 IF PW=1 THEN PRINT COUNT;
1030 IF COUNT=0 THEN COUNT=10
1040 FOR PULSE=1 TO COUNT
1050 POKE -16228,0:DLY=ONTIME: GOSUB
6000
1060 POKE -16227,0:DLY=OFFTIME: GOSUB
6000
1070 NEXT PULSE
1080 DLY=INTERDIG: GOSUB 6000:AD=
AD+1: GOTO 1010
1110 FOR A=2048 TO 2303: POKE A,
255: NEXT A: RETURN
1205 ADS=START
1210 FOR P=1 TO LEN(A$)
1220 VAL= ASC(A$(P))
1230 IF VAL=170 THEN 1260
1240 IF VAL<176 OR VAL>185 THEN
1270
1250 POKE ADS,VAL-176
1260 ADS=ADS+1
1270 NEXT P: RETURN
1310 POINT=NUMS+6*(L-1):AC= PEEK
(POINT)+256* PEEK (POINT+1)

1320 PRE= PEEK (POINT+2)+256* PEEK
(POINT+3)
1330 NUMB= PEEK (POINT+4)+256* PEEK
(POINT+5)
1340 AD=5777:DIG=4:NUM=AC: GOSUB
5000
1350 DIG=3:NUM=PRE: GOSUB 5000
```

```

1360 DIG=4: NUM=NUMB: GOSUB 5000:
ADR=AD+1: RETURN
1400 CALL -936
1410 INPUT "NAME:", A$: IF A$="END"
THEN 200
1420 FOR I=1 TO LEN(A$): POKE PTR,
ASC(A$(I)): PTR=PTR+1: NEXT
I
1430 LINE=LINE+1: POKE PTR, 30: PTR=
PTR+1: L(LINE)=PTR: POKE PTR,
0
1440 INPUT "NUMBER--AC,PRE,NUMB:"
,AC,PRE,NUMB
1450 PT=NUMS+6*(LINE-2)
1460 POKE PT,AC MOD 256: POKE PT+
1,AC/256
1470 POKE PT+2,PRE MOD 256: POKE
PT+3,PRE/256
1480 POKE PT+4,NUMB MOD 256: POKE
PT+5,NUMB/256
1490 GOTO 1410
1500 CALL -936
1505 VTAB 22: PRINT "TYPE 'E' TO EXIT
": INPUT "LINE=", L: GOTO 1590

1510 KEY= PEEK (-16384): POKE -16368
,0
1520 IF KEY=197 THEN 200
1530 IF KEY=160 THEN 1580
1540 IF KEY=204 THEN 1500
1550 IF KEY=208 THEN 1570
1560 GOTO 1510
1570 L=L+21
1580 L=L+1
1585 CALL -936
1590 IF L>LINE THEN 1505: IF L<=
0 THEN L=1
1595 POKE 16,L: CALL 901
1598 GOTO 1510
2000 CALL -936: IF PW=0 THEN 200

2010 INPUT "LINE #",L: IF L>LINE OR
L<0 THEN 2010
2020 KEY= PEEK (-16384): POKE -16368
,0
2030 IF KEY=197 THEN 200: IF KEY=
193 THEN 2050
2035 IF KEY=195 THEN 2130
2040 IF KEY=204 THEN 2010
2045 GOTO 2020
2050 L=L+1
2055 CALL -936
2060 POKE 12,L(L) MOD 256: POKE
13,L(L)/256: CALL STUFF
2070 VTAB 2: PRINT "

```

```

2080 POINT=NUMS+6*(L-1)
2090 AC= PEEK (POINT)+256*(( PEEK
      (POINT+1)) MOD 127)
2100 PRE= PEEK (POINT+2)+256*(( PEEK
      (POINT+3)) MOD 127)
2110 NUMB= PEEK (POINT+4)+256*((
      PEEK (POINT+5)) MOD 127)
2120 VTAB 2: PRINT L;" ";A$;" ";
      AC;"-";PRE;"-";NUMB: GOTO 2020

2130 VTAB 4: PRINT "OLD=";AC;"-"
      ;PRE;"-";NUMB
2140 VTAB 5: INPUT "NEW=",AC,PRE,
      NUMB
2145 POKE POINT,AC MOD 256: POKE
      POINT+1,AC/256
2150 POKE POINT+2,PRE MOD 256: POKE
      POINT+3,PRE/256
2160 POKE POINT+4,NUMB MOD 256: POKE
      POINT+5,NUMB/256: GOTO 2020

5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A+176
      :AD=AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW: POKE AD,30
5030 RETURN
6000 FOR D=1 TO DLY: NEXT D: RETURN

7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
8000 KEY= PEEK (-16384): POKE -16368
      ,0: IF KEY>127 THEN PRINT KEY:
      GOTO 8000
8210 REM TONEDIAL ROUTINE
8220 IF PEEK (AD)=255 THEN RETURN

8225 IF PW=1 THEN PRINT PEEK (AD)
      ;
8230 TONA= PEEK (TBL+(2* PEEK (AD)
      )):TONB= PEEK (TBL+(2* PEEK
      (AD))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 AD=AD+1: GOTO 8220
30000 END

```

```

0900- 18 CLC
0901- A2 16 LDX # $16
0903- A5 00 LDA $00
0905- 85 04 STA $04
0907- A5 02 LDA $02
0909- 85 06 STA $06
090B- A5 01 LDA $01
090D- 85 05 STA $05
090F- A5 03 LDA $03
0911- 85 07 STA $07
0913- A0 00 LDY # $00
0915- B1 04 LDA ($04), Y
0917- C9 1E CMP # $1E
0919- F0 08 BEQ $0923
091B- 91 06 STA ($06), Y
091D- C8 INY
091E- D0 F5 BNE $0915
0920- 84 08 STY $08
0922- 60 RTS
0923- 18 CLC
0900- 18 A2 16 A5 00 85 04 A5
*L
0924- C8 INY
0925- 98 TYA
0926- 65 04 ADC $04
0928- 85 04 STA $04
092A- 90 03 BCC $092F
092C- 18 CLC
092D- E6 05 INC $05
092F- A9 80 LDA # $80
0931- 65 06 ADC $06
0933- 85 06 STA $06
0935- 90 03 BCC $093A
0937- E6 07 INC $07
0939- 18 CLC
093A- CA DEX
093B- D0 D6 BNE $0913
093D- 60 RTS
093E- A5 00 LDA $00
0940- 85 02 STA $02
0942- A5 01 LDA $01
0944- 85 03 STA $03
0946- A0 00 LDY # $00
0948- A9 1E LDA # $1E
094A- 91 02 STA ($02), Y
094C- E6 02 INC $02
094E- D0 F8 BNE $0948
0950- E6 03 INC $03
0952- A5 03 LDA $03
0954- C9 60 CMP # $60
0956- D0 F0 BNE $0948
0958- 60 RTS
0959- 00 BRK
095A- 00 BRK
095B- 00 BRK
095C- 00 BRK

```

113

095D-	00	BRK
095E-	00	BRK
095F-	00	BRK
0960-	00	BRK
0961-	00	BRK
0962-	00	BRK
0963-	00	BRK
0964-	00	BRK
0965-	00	BRK
0966-	00	BRK
0967-	00	BRK
0968-	00	BRK
0969-	00	BRK
096A-	00	BRK
096B-	00	BRK
096C-	00	BRK
096D-	00	BRK
096E-	00	BRK
096F-	00	BRK
0970-	00	BRK
0971-	00	BRK
0972-	00	BRK
0973-	00	BRK
0974-	00	BRK
0975-	00	BRK
0976-	00	BRK

```
100 REM -----INIT-----  
200 PUTDIS=768:DISP=1024:BUFF=16384  
   :PAGE=880:CLRA=807  
300 POKE 0,BUFF MOD 256: POKE 1  
   ,BUFF/256: POKE 4,0: POKE 5  
   ,8192/256  
500 CALL CLRA: END
```

```

1 REM          SCANNER VER 2.5  8/4/77

100 POKE 2,100: POKE 3,2: DIM A$(
    (3)
110 DT=5:BY=18:RG=16:MEM=4096
120 AD=768: FOR A=0 TO 255: POKE
    AD+A,255: NEXT A
130 CALL -936: INPUT "CLEAR AREA"
    ,A$: IF A$(1,1)="Y" THEN GOSUB
    4000
140 INPUT "AC,PRE",AC,PRE
150 INPUT "STARTING BANK:",BANK
160 CALL -936
170 VTAB 3: PRINT "STATUS:"
180 VTAB 4: PRINT "NUMB=";AC;"-"
    ;PRE;"-";BANK*10
190 NUMB=BANK*10: VTAB 5: PRINT
    "          "
1000 PH= PEEK (-16228):DLY=350: GOSUB
    6000
1010 PH= PEEK (-16227)
1020 REM LOOK FOR DIAL TONE
1030 POKE -16231,DT:T1=5:DL1=100
    : GOSUB 7000
1040 IF T1<5 THEN 1000
1100 REM DIAL NUMBER
1110 AD=768: IF AC=0 THEN 1130
1120 NUM=AC:DIG=3: GOSUB 5000
1130 NUM=PRE:DIG=3: GOSUB 5000
1140 ADR=AD
1150 NUM=NUMB:DIG=4: GOSUB 5000
1160 AD=768: GOSUB 8210: REM DIAL NU
    MBER
2000 KEY= PEEK (-16384): POKE -16368
    ,0
2010 IF KEY=197 THEN 3000: REM ENTER
    NUMBER
2020 IF KEY=194 THEN 3500: REM BACK
    UP
2025 IF KEY=196 THEN 1000: REM DIAL
    NUM
2030 IF KEY=201 THEN 3600: REM INCR
    & DIAL
2040 IF KEY=204 THEN 4100: REM LOOK
    AT BANKS
2045 IF KEY=211 THEN 4500: REM STORE
    NEW BANKS
2050 IF KEY=193 THEN 4200: REM ADVAN
    CE
2055 IF KEY=218 THEN 4210: REM (Z) B
    ACK-UP
2100 GOTO 2000
3000 REM ENTER BANK IN MEM
3005 PRINT " ";
3010 POKE MEM,BANK MOD 256: POKE
    MEM+1,BANK/256:MEM=MEM+2: VTAB

```

```

5: GOTO 4100
3500 REM ----BACK UP
3510 BANK=BANK-1: GOTO 170
3600 REM INCREMENT ----
3610 BANK=BANK+1: GOTO 170
4000 REM CLEAR AREA
4010 FOR M=4096 TO 5120: POKE M,
255: NEXT M: RETURN
4100 CALL -936:ME=4096: VTAB 6
4110 IF ME=MEM THEN 4130
4120 NU= PEEK (ME)+ PEEK (ME+1)*
256: PRINT NU;" "":ME=ME+2
: GOTO 4110
4130 PRINT : GOTO 2000
4200 BANK=BANK+1: VTAB 4: TAB 14
: PRINT BANK*10: GOTO 2000
4210 BANK=BANK-1: VTAB 4: TAB 14
: PRINT BANK*10: GOTO 2000
4500 REM ----STORE NEW BANKS----
4510 VTAB 8: PRINT "INPUT BANK,0 IF
DONE"
4520 INPUT BA: IF BA=0 THEN 2000
: POKE MEM,BA MOD 256: POKE
MEM+1,BA/256:MEM=MEM+2: GOTO
4520
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
(-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7005 REM SUBR THAT RETURNS LENGTH O
F TONE
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
( PEEK (-16238)>127): VTAB
10: PRINT "VAL="; PDL (0)/4
+PH: PRINT "TONE=";Y: GOTO
8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

```



```
8230 TONA= PEEK (3072+(2* PEEK (
      AD+DGT))):TONB= PEEK (3072+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END
```

1 REM SCANNER VER 2.6 8/4/77

```

100 POKE 2,100: POKE 3,2: DIM A$
    (3)
110 DT=5:BY=18:RG=16:MEM=4096
120 AD=768: FOR A=0 TO 255: POKE
    AD+A,255: NEXT A
130 CALL -936: INPUT "CLEAR AREA"
    ,A$: IF A$(1,1)="Y" THEN GOSUB
    4000
140 INPUT "AC,PRE",AC,PRE
150 INPUT "STARTING BANK:",BANK
160 CALL -936
170 VTAB 3: PRINT "STATUS:"
180 VTAB 4: PRINT "NUMB=";AC;"-"
    ;PRE;"-";BANK*10
190 NUMB=BANK*10: VTAB 5: PRINT
    "
    "
500 PH= PEEK (-16227)
1000 AD=768: IF AC=0 THEN 1020
1010 NUM=AC:DIG=3: GOSUB 5000
1020 NUM=PRE:DIG=3: GOSUB 5000
1030 NUM=NUMB:DIG=4: GOSUB 5000
1040 AD=768: GOSUB 8210
2000 KEY= PEEK (-16384): POKE -16368
    ,0
2010 IF KEY=197 THEN 3000: REM ENTER
    NUMBER
2020 IF KEY=194 THEN 3500: REM BACK
    UP
2025 IF KEY=196 THEN 1000: REM DIAL
    NUM
2030 IF KEY=201 THEN 3600: REM INCR
    & DIAL
2040 IF KEY=204 THEN 4100: REM LOOK
    AT BANKS
2045 IF KEY=211 THEN 4500: REM STORE
    NEW BANKS
2050 IF KEY=193 THEN 4200: REM ADVAN
    CE
2055 IF KEY=218 THEN 4210: REM (Z) B
    ACK-UP
2100 GOTO 2000
3000 REM ENTER BANK IN MEM
3005 PRINT "";
3010 POKE MEM,(BANK-1) MOD 256: POKE
    MEM+1,(BANK-1)/256:MEM=MEM+
    2: VTAB 5: GOTO 4100
3500 REM ----BACK UP
3510 BANK=BANK-1: GOTO 170
3600 REM INCREMENT ----
3610 BANK=BANK+1: GOTO 170
4000 REM CLEAR AREA
4010 FOR M=4096 TO 5120: POKE M,
    255: NEXT M: RETURN
4100 CALL -936:ME=4096: VTAB 6

```

```

4110 IF ME=MEM THEN 4130
4120 NU= PEEK (ME) + PEEK (ME+1)*
      256: PRINT NU;" ";:ME=ME+2
      : GOTO 4110
4130 PRINT : GOTO 2000
4200 BANK=BANK+1: VTAB 4: TAB 14
      : PRINT BANK*10: GOTO 2000
4210 BANK=BANK-1: VTAB 4: TAB 14
      : PRINT BANK*10: GOTO 2000
4500 REM ----STORE NEW BANKS----
4510 VTAB 8: PRINT "INPUT BANK,0 IF
      DONE"
4520 INPUT BA: IF BA=0 THEN 2000
      : POKE MEM,BA MOD 256: POKE
      MEM+1,BA/256:MEM=MEM+2: GOTO
      4520
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
      AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7005 REM SUBR THAT RETURNS LENGTH O
      F TONE
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127): VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (3072+(2* PEEK (
      AD+DGT))):TONB= PEEK (3072+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END

```

```

1 REM          SCANNER VER 2.7 8/12/7
7
100 POKE 2,100: POKE 3,2: DIM A$
(3)
110 DT=5:BY=18:RG=16:MEM=4096
120 AD=768: FOR A=0 TO 255: POKE
AD+A,255: NEXT A
130 CALL -936: INPUT "CLEAR AREA"
,A$: IF A$(1,1)="Y" THEN GOSUB
4000
140 INPUT "AC,PRE",AC,PRE
150 INPUT "STARTING BANK:",BANK
160 CALL -936
170 VTAB 3: PRINT "STATUS:"
180 VTAB 4: PRINT "NUMB=";AC;"-"
;PRE;"-";BANK*10
190 NUMB=BANK*10: VTAB 5: PRINT
"
"
500 PH= PEEK (-16227)
1000 GOSUB 5500
1040 AD=768: GOSUB 8210
2000 KEY= PEEK (-16384): POKE -16368
,0
2010 IF KEY=197 THEN 3000: REM ENTER
NUMBER
2020 IF KEY=194 THEN 3500: REM BACK
UP
2025 IF KEY=196 THEN 1000: REM DIAL
NUM
2030 IF KEY=201 THEN 3600: REM INCR
& DIAL
2040 IF KEY=204 THEN 4100: REM LOOK
AT BANKS
2045 IF KEY=211 THEN 4500: REM STORE
NEW BANKS
2050 IF KEY=193 THEN 4200: REM ADVAN
CE
2055 IF KEY=218 THEN 4210: REM (Z) B
ACK-UP
2060 IF KEY=198 THEN 3100: REM FIND
#'S
2100 GOTO 2000
3000 REM ENTER BANK IN MEM
3005 PRINT "";
3010 POKE MEM,(BANK-1) MOD 256: POKE
MEM+1,(BANK-1)/256:MEM=MEM+
2: VTAB 5: GOTO 4100
3100 CALL -936: VTAB 3: PRINT "DIALIN
G NUMBERS":MEM=4096
3103 REM FIND #'S IN BANKS
3105 KEY= PEEK (-16384): POKE -16368
,0
3110 IF KEY=196 THEN 3230: REM DIA
L
#
3115 IF KEY=201 THEN 3300: REM INC A
ND DIAL

```

```

3120 IF KEY=195 THEN 3310: REM (C) D
ATA
3125 IF KEY=212 THEN 3320: REM (T) T
ONE
3130 IF KEY=210 THEN 3330: REM (R) R
E-ENTER NUMB
3135 IF KEY=213 THEN 3305: REM (U) DE
C AND DIAL
3199 GOTO 3105
3200 REM GET BANK
3205 INC=0
3210 BANK= PEEK (MEM)+ PEEK (MEM+
1)*256:MEM=MEM+2
3220 NUMB=BANK*10+INC
3230 GOSUB 5500
3235 VTAB 2: PRINT "NUM=";AC;"-"
;PRE;"-";NUMB
3240 AD=768: GOSUB 8210: GOTO 3105

3300 INC=INC+1: IF INC=10 THEN 3200
: GOTO 3220
3305 INC=INC-1: IF INC<0 THEN 3330
: GOTO 3220
3310 VTAB 4: PRINT AC;"-";PRE;"-"
;NUMB-1;" DATA": GOTO 3105

3320 VTAB 5: PRINT AC;"-";PRE;"-"
;NUMB-1;" TONE": GOTO 3105

3330 VTAB 6: INPUT "NUMB=",NUMB:
BANK=NUMB/10:INC=NUMB-BANK*
10: GOTO 3105

3500 REM ----BACK UP
3510 BANK=BANK-1: GOTO 170
3600 REM INCREMENT ----
3610 BANK=BANK+1: GOTO 170
4000 REM CLEAR AREA
4010 FOR M=4096 TO 5120: POKE M,
255: NEXT M: RETURN
4100 CALL -936:ME=4096: VTAB 6
4110 IF ME=MEM THEN 4130
4120 NU= PEEK (ME)+ PEEK (ME+1)*
256: PRINT NU;" "":ME=ME+2
: GOTO 4110
4130 PRINT : GOTO 2000
4200 BANK=BANK+1: VTAB 4: TAB 14
: PRINT BANK*10: GOTO 2000
4210 BANK=BANK-1: VTAB 4: TAB 14
: PRINT BANK*10: GOTO 2000
4500 REM ----STORE NEW BANKS----
4510 VTAB 8: PRINT "INPUT BANK,0 IF
DONE"
4520 INPUT BA: IF BA=0 THEN 2000
: POKE MEM,BA MOD 256: POKE
MEM+1,BA/256:MEM=MEM+2: GOTO
4520

```

```

5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
      AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
5500 AD=768: IF AC=0 THEN 5520
5510 NUM=AC:DIG=3: GOSUB 5000
5520 NUM=PRE:DIG=3: GOSUB 5000
5530 NUM=NUMB:DIG=4: GOSUB 5000
5540 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7005 REM SUBR THAT RETURNS LENGTH O
      F TONE
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127): VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN
8230 TONA= PEEK (3072+(2* PEEK (
      AD+DGT))):TONB= PEEK (3072+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9000 KQ= PEEK (-16384): IF KQ>127
      THEN 9010: POKE -16368,0: GOTO
      9000
9010 PRINT KQ: GOTO 9000
9999 END

```

```

1 REM      DIMENSION VER 1.4 8/12/7
7
20 CON=8192:AQ=4:AW=8704
100 POKE 2,100: POKE 3,2
110 AD=768: FOR A=0 TO 255: POKE
AD+A,255: NEXT A
120 CALL -936
200 VTAB 4: INPUT "AREA,PRE,NUMB"
,AC,PRE,NUMB
210 IF AC=0 THEN 310
300 NUM=AC:DIG=3: GOSUB 5000
310 NUM=PRE:DIG=3: GOSUB 5000
320 NUM=NUMB:DIG=4: GOSUB 5000
325 ADR=AD+1
400 INPUT "START,END CODE",CODE,
ECOD
1000 PH= PEEK (-16228):DLY=350: GOSUB
6000
1100 PH= PEEK (-16227)
1110 DL1=100: POKE -16231,5:T1=10
: GOSUB 7000: IF T1<10 THEN
1000
1115 AD=768: GOSUB 8210
1140 DL1=800:T1=20: POKE -16231,
5: GOSUB 7000
1145 IF T<T1 THEN 1000
1150 AD=ADR:NUM=CODE:DIG=4: GOSUB
5000:AD=ADR: GOSUB 8210
1155 PRINT "CODE=";CODE
1160 POKE -16231,19:DL1=100:T1=10
: GOSUB 7000: IF T=T1 THEN
1500
1168 PRINT "";
1170 DL1=100:T1=20: POKE -16231,
6: GOSUB 7000: IF T<T1 THEN
1000:TRY=TRY+1: PRINT "TRYING "
;TRY: IF TRY<5 THEN 1000
1180 FOR A=1 TO 50: PRINT "":NEXT
A: PRINT "WHOOPIE!!!! CODE="
;CODE:PH= PEEK (-16228): END

1500 TRY=0: IF CODE=ECOD THEN 2000
:CODE=CODE+1: GOTO 1000
2000 PH= PEEK (-16228): END
2999 CALL -936
3000 POKE -16167, PDL (0)/3:T=( PEEK
(-16238)>127): VTAB 10: PRINT
"STONE=";T: PRINT "VAL="; PDL
(0)/3: GOTO 3000
4500 POKE -16231,9: GOSUB 7000: PRINT
"T=";T: GOTO 4500
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN

```

```
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7005 REM SUBR THAT RETURNS LENGTH O
      F TONE
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127):VTAB
10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (2304+(2* PEEK (
      AD+DGT))):TONB= PEEK (2304+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END
```



```

1 REM      DIMENSION VER 1.4 8/12/7
7
20 CON=8192:AQ=4:AW=8704
100 POKE 2,100: POKE 3,2
110 AD=768: FOR A=0 TO 255: POKE
    AD+A,255: NEXT A
120 CALL -936
200 VTAB 4: INPUT "AREA,PRE,NUMB"
    ,AC,PRE,NUMB
210 IF AC=0 THEN 310
300 NUM=AC:DIG=3: GOSUB 5000
310 NUM=PRE:DIG=3: GOSUB 5000
320 NUM=NUMB:DIG=4: GOSUB 5000
325 ADR=AD+1
400 INPUT "START,END CODE",CODE,
    ECOD
1000 PH= PEEK (-16228):DLY=350: GOSUB
    6000
1100 PH= PEEK (-16227)
1110 DL1=100: POKE -16231,5:T1=10
    : GOSUB 7000: IF T1<10 THEN
    1000
1115 AD=768: GOSUB 8210
1140 DL1=800:T1=20: POKE -16231,
    5: GOSUB 7000
1145 IF T<T1 THEN 1000
1150 AD=ADR:NUM=CODE:DIG=4: GOSUB
    5000:AD=ADR: GOSUB 8210
1155 PRINT "CODE=";CODE
1160 POKE -16231,19:DL1=100:T1=10
    : GOSUB 7000: IF T=T1 THEN
    1500
1168 PRINT "";
1170 DL1=100:T1=20: POKE -16231,
    6: GOSUB 7000: IF T<T1 THEN
    1000:TRY=TRY+1: PRINT "TRYING "
    ;TRY: IF TRY<5 THEN 1000
1180 FOR A=1 TO 50: PRINT " ";: NEXT
    A: PRINT "WHOOPIE!!!! CODE="
    ;CODE:PH= PEEK (-16228): END

1500 TRY=0: IF CODE=ECOD THEN 2000
    :CODE=CODE+1: GOTO 1000
2000 PH= PEEK (-16228): END
2999 CALL -936
3000 POKE -16167, PDL (0)/3:T=( PEEK
    (-16238)>127): VTAB 10: PRINT
    "TONE=";T: PRINT "VAL="; PDL
    (0)/3: GOTO 3000
4500 POKE -16231,9: GOSUB 7000: PRINT
    "T=";T: GOTO 4500
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
    AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN

```

```
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127): VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (2304+(2* PEEK (
      AD+DGT))):TONB= PEEK (2304+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END
```

```

1 REM      DIMENSION VER 1.4 8/12/7
  7
20 CON=8192:AQ=4:AW=8704
100 POKE 2,100: POKE 3,2
110 AD=768: FOR A=0 TO 255: POKE
    AD+A,255: NEXT A
120 CALL -936
200 VTAB 4: INPUT "AREA,PRE,NUMB"
    ,AC,PRE,NUMB
210 IF AC=0 THEN 310
300 NUM=AC:DIG=3: GOSUB 5000
310 NUM=PRE:DIG=3: GOSUB 5000
320 NUM=NUMB:DIG=4: GOSUB 5000
325 ADR=AD+1
400 INPUT "START,END CODE",CODE,
    ECOD
1000 PH= PEEK (-16228):DLY=350: GOSUB
    6000
1100 PH= PEEK (-16227)
1110 DL1=100: POKE -16231,5:T1=10
    : GOSUB 7000: IF T1<10 THEN
    1000
1115 AD=768: GOSUB 8210
1140 DL1=800:T1=20: POKE -16231,
    5: GOSUB 7000
1145 IF T<T1 THEN 1000
1150 AD=ADR:NUM=CODE:DIG=4: GOSUB
    5000:AD=ADR: GOSUB 8210
1155 PRINT "CODE=";CODE
1160 POKE -16231,19:DL1=100:T1=10
    : GOSUB 7000: IF T=T1 THEN
    1500
1168 PRINT "";
1170 DL1=100:T1=20: POKE -16231,
    6: GOSUB 7000: IF T<T1 THEN
    1000:TRY=TRY+1: PRINT "TRYING "
    ;TRY: IF TRY<5 THEN 1000
1180 FOR A=1 TO 50: PRINT "":NEXT
    A: PRINT "WHOOPIE!!!! CODE="
    ;CODE:PH= PEEK (-16228): END

1500 TRY=0: IF CODE=ECOD THEN 2000
    :CODE=CODE+1: GOTO 1000
2000 PH= PEEK (-16228): END
2999 CALL -936
3000 POKE -16167, PDL (0)/3:T=( PEEK
    (-16238)>127): VTAB 10: PRINT
    "TONE=";T: PRINT "VAL="; PDL
    (0)/3: GOTO 3000
4500 POKE -16231,9: GOSUB 7000: PRINT
    "T=";T: GOTO 4500
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
    AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN

```

```
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127):VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (3072+(2* PEEK (
      AD+DGT))):TONB= PEEK (3072+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END
```

```

1 REM          SCANNER VER 2.6 8/4/77

100 POKE 2,100: POKE 3,2: DIM A$
    (3)
110 DT=5:BY=18:RG=16:MEM=4096
120 AD=768: FOR A=0 TO 255: POKE
    AD+A,255: NEXT A
130 CALL -936: INPUT "CLEAR AREA"
    ,A$: IF A$(1,1)="Y" THEN GOSUB
    4000
140 INPUT "AC,PRE",AC,PRE
150 INPUT "STARTING BANK:",BANK
160 CALL -936
170 VTAB 3: PRINT "STATUS:"
180 VTAB 4: PRINT "NUMB=";AC;"-"
    ;PRE;"-";BANK*10
190 NUMB=BANK*10: VTAB 5: PRINT
    "          "

500 PH= PEEK (-16227)
1000 AD=768: IF AC=0 THEN 1020
1010 NUM=AC: DIG=3: GOSUB 5000
1020 NUM=PRE: DIG=3: GOSUB 5000
1030 NUM=NUMB: DIG=4: GOSUB 5000
1040 AD=768: GOSUB 8210
2000 KEY= PEEK (-16384): POKE -16368
    ,0
2010 IF KEY=197 THEN 3000: REM  ENTER
    NUMBER
2020 IF KEY=194 THEN 3500: REM  BACK
    UP
2025 IF KEY=196 THEN 1000: REM  DIAL
    NUM
2030 IF KEY=201 THEN 3600: REM  INCR
    & DIAL
2040 IF KEY=204 THEN 4100: REM  LOOK
    AT BANKS
2045 IF KEY=211 THEN 4500: REM  STORE
    NEW BANKS
2050 IF KEY=193 THEN 4200: REM  ADVAN
    CE
2055 IF KEY=218 THEN 4210: REM  (Z) B
    ACK-UP
2100 GOTO 2000
3000 REM  ENTER BANK IN MEM
3005 PRINT " ";
3010 POKE MEM,(BANK-1) MOD 256: POKE
    MEM+1,(BANK-1)/256: MEM=MEM+
    2: VTAB 5: GOTO 4100
3500 REM  ----BACK UP
3510 BANK=BANK-1: GOTO 170
3600 REM  INCREMENT ----
3610 BANK=BANK+1: GOTO 170
4000 REM  CLEAR AREA
4010 FOR M=4096 TO 5120: POKE M,
    255: NEXT M: RETURN
4100 CALL -936: ME=4096: VTAB 6

```

```

4110 IF ME=MEM THEN 4130
4120 NU= PEEK (ME) + PEEK (ME+1)*
      256: PRINT NU;" ";:ME=ME+2
      : GOTO 4110
4130 PRINT : GOTO 2000
4200 BANK=BANK+1: VTAB 4: TAB 14
      : PRINT BANK*10: GOTO 2000
4210 BANK=BANK-1: VTAB 4: TAB 14
      : PRINT BANK*10: GOTO 2000
4500 REM ----STORE NEW BANKS----
4510 VTAB 8: PRINT "INPUT BANK,0 IF
      DONE"
4520 INPUT BA: IF BA=0 THEN 2000
      : POKE MEM,BA MOD 256: POKE
      MEM+1,BA/256:MEM=MEM+2: GOTO
      4520
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
      AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7005 REM SUBR THAT RETURNS LENGTH O
      F TONE
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127): VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN
8230 TONA= PEEK (3072+(2* PEEK (
      AD+DGT))):TONB= PEEK (3072+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END

```

```

1 REM      DIMENSION VER 1.3 8/4/77
20 CON=8192:AQ=4:AW=8704
100 POKE 2,100: POKE 3,2
110 AD=768: FOR A=0 TO 255: POKE
    AD+A,255: NEXT A
120 CALL -936
200 VTAB 4: INPUT "AREA,PRE,NUMB"
    ,AC,PRE,NUMB
210 IF AC=0 THEN 310
300 NUM=AC:DIG=3: GOSUB 5000
310 NUM=PRE:DIG=3: GOSUB 5000
320 NUM=NUMB:DIG=4: GOSUB 5000
325 ADR=AD+1
400 INPUT "START,END CODE",CODE,
    ECOD
1000 PH= PEEK (-16228):DLY=350: GOSUB
    6000
1100 PH= PEEK (-16227)
1110 DL1=100: POKE -16231,5:T1=10
    : GOSUB 7000: IF T1<10 THEN
    1000
1115 AD=768: GOSUB 8210
1140 DL1=800:T1=20: POKE -16231,
    6: GOSUB 7000
1145 IF T<T1 THEN 1000
1150 AD=ADR:NUM=CODE:DIG=4: GOSUB
    5000:AD=ADR: GOSUB 8210
1155 PRINT "CODE=";CODE
1160 POKE -16231,19:DL1=100: GOSUB
    6500: IF NOT TN1 THEN 1500
1168 PRINT "";
1170 DL1=100:T1=20: POKE -16231,
    6: GOSUB 7000: IF T<T1 THEN
    1000:TRY=TRY+1: PRINT "TRYING "
    ;TRY: IF TRY<5 THEN 1000
1180 FOR A=1 TO 50: PRINT "":NEXT
    A: PRINT "WHOOPIE!!!! CODE="
    ;CODE:PH= PEEK (-16228): END

1500 TRY=0: IF CODE=ECOD THEN 2000
    :CODE=CODE+1: GOTO 1000
2000 PH= PEEK (-16228): END
2999 CALL -936
3000 POKE -16167, PDL (0)/3:T=( PEEK
    (-16238)>127): VTAB 10: PRINT
    "TONE=";T: PRINT "VAL="; PDL
    (0)/3: GOTO 3000
4500 POKE -16231,9: GOSUB 7000: PRINT
    "T=";T: GOTO 4500
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
    AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ

```

```
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7005 REM SUBR THAT RETURNS LENGTH O
      F TONE
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231,-PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127):VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (2304+(2* PEEK (
      AD+DGT))):TONB= PEEK (2304+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9999 END
```



```

1 REM          SCANNER VER 2.7 8/12/7
7
100 POKE 2,100: POKE 3,2: DIM A$
(3)
110 DT=5:BY=18:RG=16:MEM=4096
120 AD=768: FOR A=0 TO 255: POKE
AD+A,255: NEXT A
130 CALL -936: INPUT "CLEAR AREA"
,A$: IF A$(1,1)="Y" THEN GOSUB
4000
140 INPUT "AC,PRE",AC,PRE
150 INPUT "STARTING BANK:",BANK
160 CALL -936
170 VTAB 3: PRINT "STATUS:"
180 VTAB 4: PRINT "NUMB=";AC;"-"
;PRE;"-";BANK*10
190 NUMB=BANK*10: VTAB 5: PRINT
"
"
300 PH= PEEK (-16228):DLY=350: GOSUB
6000
500 PH= PEEK (-16227)
550 T1=20:DL1=150: GOSUB 7000: IF
T1<20 THEN 300
1000 GOSUB 5500
1040 AD=768: GOSUB 8210
2000 KEY= PEEK (-16384): POKE -16368
,0
2010 IF KEY=197 THEN 3000: REM ENTER
NUMBER
2020 IF KEY=194 THEN 3500: REM BACK
UP
2025 IF KEY=196 THEN 300: REM DIAL
NUM
2030 IF KEY=201 THEN 3600: REM INCR
& DIAL
2040 IF KEY=204 THEN 4100: REM LOOK
AT BANKS
2045 IF KEY=211 THEN 4500: REM STORE
NEW BANKS
2050 IF KEY=193 THEN 4200: REM ADVAN
CE
2055 IF KEY=218 THEN 4210: REM (Z) B
ACK-UP
2060 IF KEY=198 THEN 3100: REM FIND
#'S
2100 GOTO 2000
3000 REM ENTER BANK IN MEM
3005 PRINT " ";
3010 POKE MEM,(BANK) MOD 256: POKE
MEM+1,(BANK)/256:MEM=MEM+2:
VTAB 5: GOTO 4100
3100 CALL -936: VTAB 3: PRINT "DIALIN
G NUMBERS":MEM=4096
3103 REM FIND #'S IN BANKS
3105 KEY= PEEK (-16384): POKE -16368
,0

```

```

3110 IF KEY=196 THEN 3230: REM DIA
      L
      #
3115 IF KEY=201 THEN 3300: REM INC A
      ND DIAL
3120 IF KEY=195 THEN 3310: REM (C) D
      ATA
3125 IF KEY=212 THEN 3320: REM (T) T
      ONE
3130 IF KEY=210 THEN 3330: REM (R) R
      E-ENTER NUMB
3135 IF KEY=213 THEN 3305: REM (U) DE
      C AND DIAL
3199 GOTO 3105
3200 REM GET BANK
3205 INC=0
3210 BANK= PEEK (MEM)+ PEEK (MEM+
      1) *256: MEM=MEM+2
3220 NUMB=BANK*10+INC
3230 GOSUB 5500
3235 VTAB 2: PRINT "NUM=";AC;"-"
      ;PRE;"-";NUMB
3240 AD=768: GOSUB 8210: GOTO 3105

3300 INC=INC+1: IF INC=10 THEN 3200
      : GOTO 3220
3305 INC=INC-1: IF INC<0 THEN 3330
      : GOTO 3220
3310 VTAB 4: PRINT AC;"-";PRE;"-"
      ;NUMB-1;" DATA": GOTO 3105

3320 VTAB 5: PRINT AC;"-";PRE;"-"
      ;NUMB-1;" TONE": GOTO 3105

3330 VTAB 6: INPUT "NUMB=",NUMB:
      BANK=NUMB/10:INC=NUMB-BANK*
      10: GOTO 3105
3500 REM ----BACK UP
3510 BANK=BANK-1: GOTO 170
3600 REM INCREMENT ----
3610 BANK=BANK+1: GOTO 170
4000 REM CLEAR AREA
4010 FOR M=4096 TO 5120: POKE M,
      255: NEXT M: RETURN
4100 CALL -936:ME=4096: VTAB 6
4110 IF ME=MEM THEN 4130
4120 NU= PEEK (ME)+ PEEK (ME+1) *
      256: PRINT NU;" "":ME=ME+2
      : GOTO 4110
4130 PRINT : GOTO 2000
4200 BANK=BANK+1: VTAB 4: TAB 14
      : PRINT BANK*10: GOTO 2000
4210 BANK=BANK-1: VTAB 4: TAB 14
      : PRINT BANK*10: GOTO 2000
4500 REM ----STORE NEW BANKS----
4510 VTAB 8: PRINT "INPUT BANK,0 IF
      DONE"

```

```
4520 INPUT BA: IF BA=0 THEN 2000
      : POKE MEM,BA MOD 256: POKE
      MEM+1,BA/256:MEM=MEM+2: GOTO
      4520
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
      AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
5500 AD=768: IF AC=0 THEN 5520
5510 NUM=AC:DIG=4: GOSUB 5000
5520 NUM=PRE:DIG=3: GOSUB 5000
5530 NUM=NUMB:DIG=4: GOSUB 5000
5540 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7005 REM SUBR THAT RETURNS LENGTH O
      F TONE
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
      ( PEEK (-16238)>127): VTAB
      10: PRINT "VAL="; PDL (0)/4
      +PH: PRINT "TONE=";Y: GOTO
      8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN
8230 TONA= PEEK (3072+(2* PEEK (
      AD+DGT))):TONB= PEEK (3072+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9000 KQ= PEEK (-16384): IF KQ>127
      THEN 9010: POKE -16368,0: GOTO
      9000
9010 PRINT KQ: GOTO 9000
9999 END
```

```

1 REM          LOOKER VER 5.6  8/3/77

20 CON=8192:AQ=4:AW=8704
100 POKE 2,100: POKE 3,2
110 AD=768: FOR A=0 TO 255: POKE
    AD+A,255: NEXT A
120 CALL -936
130 FOR A=8192 TO 8704: POKE A,
    0: NEXT A
140 FOR A=8704 TO 9216: POKE A,
    0: NEXT A
200 VTAB 4: INPUT "AREA,PRE,NUMB,ENU
    M",AC,PRE,NUMB,ENUM
210 IF AC=0 THEN 310
300 NUM=AC:DIG=3: GOSUB 5000
10 NUM=PRE:DIG=3: GOSUB 5000
315 ADR=AD
320 NUM=NUMB:DIG=4: GOSUB 5000
330 CALL -936
1000 PH= PEEK (-16228):DLY=350: GOSUB
    6000
1100 PH= PEEK (-16227):DL1=1000:
    POKE -16231,6:T1=10: GOSUB
    7000: IF T<9 THEN 1000
1110 AD=768: GOSUB 8210
1115 VTAB 2: PRINT "NUMBER DIALED="
    ;PRE;"-";NUMB;" ";
1130 DL1=300: POKE -16231,20:T1=
    20: GOSUB 7000: IF T>18 THEN
    1800
1140 DL1=300: POKE -16231,5:T1=50
    : GOSUB 7000
1141 PRINT T;"          ";
1145 PRINT
1150 IF T<49 THEN 1500
1160 REM -----FOUND-----
1170 POKE CON,NUMB MOD 256: POKE
    CON+1,NUMB/256:CON=CON+2: IF
    CON>8703 THEN END
1180 VTAB AQ: PRINT "NUMB=";PRE;
    "-";NUMB:AQ=AQ+1
1500 REM -----NOT FOUND-----
1510 IF NUMB=ENUM THEN 2000:NUMB=
    NUMB+1
1520 AD=ADR:NUM=NUMB:DIG=4: GOSUB
    5000
1530 GOTO 1000
1800 REM -----BUSY-----
1810 POKE AW,NUMB MOD 256: POKE
    AW+1,NUMB/256: IF AW>9216 THEN
    END :AW=AW+2: VTAB 2: TAB 24
    : PRINT "BUSY": GOTO 1500
2000 PH= PEEK (-16228): END
2999 CALL -936
3000 POKE -16167, PDL (0)/3:T=( PEEK
    (-16238)>127): VTAB 10: PRINT

```

```
"TONE=";T: PRINT "VAL="; PDL
(0)/3: GOTO 3000
4500 POKE -16231,9: GOSUB 7000: PRINT
"T=";T: GOTO 4500
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
(-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 T=0
7005 REM SUBR THAT RETURNS LENGTH O
F TONE
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
7999 INPUT PH: CALL -936
8000 POKE -16231, PDL (0)/4+PH:Y=
( PEEK (-16238)>127): VTAB
10: PRINT "VAL="; PDL (0)/4
+PH: PRINT "TONE=";Y: GOTO
8000
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN
8230 TONA= PEEK (2304+(2* PEEK (
AD+DGT))):TONB= PEEK (2304+
(2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
2560
8260 DGT=DGT+1: GOTO 8220
9999 END
```

```

1 REM    DIALER VER 1.6 7/31/77
100 POKE 2,100: POKE 3,2
110 AD=768: FOR A=0 TO 255: POKE
    AD+A,255: NEXT A
200 CALL -936
210 PRINT "TYPE FOLLOWING KEYS FOR F
    UNCTIONS:"
220 PRINT "E:  EXIT": PRINT "I:  INP
    UT NUMBER"
224 PRINT "D:  DIAL NUMBER"
226 PRINT "H:  HANG UP PHONE"
230 PRINT "P:  PICK UP PHONE": PRINT
    "K:  KILL AUTO FEATURE": PRINT
    "A:  ENABLE AUTO FEATURE"
235 PRINT "S:  PRINT AUTO FEATURE ST
    ATUS"
240 PRINT "R:  PROGRAM NON-STANDARD
    TONES"
300 KEY= PEEK (-16384): POKE -16368
    ,0
310 IF KEY=197 THEN END
315 IF KEY=203 THEN 700
320 IF KEY=201 THEN 1000
325 IF KEY=193 THEN 710
330 IF KEY=196 THEN 1500
335 IF KEY=211 THEN 720
340 IF KEY=208 THEN 2000
345 IF KEY=212 THEN 3000
350 IF KEY=200 THEN 2500
355 IF KEY=210 THEN 4000
360 IF KEY=206 THEN 1800
365 IF KEY=178 THEN 2600
370 IF KEY=177 THEN 2200
380 IF KEY=176 THEN 2400
400 GOTO 300
700 VTAB 16:K=1: PRINT "AUTO FEATURE
    KILLED":DLY=1500: GOSUB
    6000: GOTO 200
710 VTAB 16:K=0: PRINT "AUTO FEATURE
    ENABLED":DLY=1500:
    GOSUB 6000: GOTO 200
720 IF K=0 THEN 710
730 IF K=1 THEN 700
740 GOTO 200
1000 CALL -936: TAB 2: PRINT "ENTER A
    C,PRE,NUMB": INPUT "(0 IF NO ARE
    A CODE):",AC,PRE,NUMB: IF AC=
    0 THEN 1110
1010 IF AC>99 AND AC<1000 THEN 1100
    : PRINT "INVALID AREA CODE-PLEAS
    E RE-ENTER": GOTO 1000
1100 NUM=AC:DIG=3: GOSUB 5000
1110 NUM=PRE:DIG=3: GOSUB 5000
1120 NUM=NUMB:DIG=4: GOSUB 5000
1130 GOTO 200
1500 PH= PEEK (-16228):DLY=350: GOSUB

```

```

6000:PH= PEEK (-16227)
1510 POKE -16231,6:DL1=1000: GOSUB
6500: IF TN1 THEN 1500
1520 AD=768: GOSUB 8210
1525 IF K THEN 300
1530 POKE -16231,18:DL1=400: GOSUB
6500: IF NOT TN1 THEN 1600
1540 POKE -16231,13:DL1=1000: GOSUB
6500: IF NOT TN1 THEN 1700
1560 GOTO 200
1600 CALL -936: PRINT "NUMBER "
;AC;"-";PRE;"-";NUMB: PRINT
"IS BUSY!! WILL KEEP": PRINT
"TRYING": GOTO 1500
1700 CALL -936: FOR A=0 TO 30: PRINT
"": NEXT A: PRINT "RINGING"
:DLY=1500: GOSUB 6000: GOTO
200
1800 VTAB 16: PRINT "NUMBER=";AC;
"-";PRE;"-";NUMB:DLY=1500: GOSUB
6000: GOTO 200
2000 PH= PEEK (-16227): GOTO 200

2200 NUMB=NUMB+1: GOSUB 7000: PRINT
"NUMBER=";AC;"-";PRE;"-";NUMB:
GOTO 1500
2400 NUMB=NUMB+10: GOSUB 7000: PRINT
"NUMBER=";AC;"-";PRE;"-";NUMB:
GOTO 1500
2500 PH= PEEK (-16228): GOTO 200

2600 NUMB=NUMB-10: GOSUB 7000: PRINT
"NUMBER=";AC;"-";PRE;"-";NUMB:
GOTO 1500
3000 CALL -936
3005 PRINT "I=INCR": PRINT "D=DECR"
: PRINT "E=EXIT"
3007 PRINT "N=INPUT NU,DL1": PRINT
"R=REPEAT TEST"
3010 KEY= PEEK (-16384): POKE -16368
,0
3011 IF KEY=201 THEN 3200
3020 IF KEY=196 THEN 3210
3030 IF KEY=197 THEN 200
3035 IF KEY=210 THEN 3110
3040 IF KEY=206 THEN 3100
3050 GOTO 3010
3100 CALL -936: INPUT "NU,DL1",NU,
DL1
3110 CALL -936: POKE -16231,NU: GOSUB
6500: VTAB 10: PRINT "NU=";
NU;" "; "TONE=";TN1: GOTO 3010

3200 NU=NU+1: GOTO 3110
3210 NU=NU-1: GOTO 3110
4000 CALL -936

```

```
4010 POKE -16231, PDL (0)/3: VTAB
      10: PRINT "VAL="; PDL (0)/3
      : PRINT "TONE="; ( PEEK (-16238
      )>127): GOTO 4010
5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A:AD=
      AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW
5030 RETURN
6000 FOR DQ=0 TO DLY
6010 NEXT DQ
6020 RETURN
6500 FOR DL=0 TO DL1:TN1=( PEEK
      (-16238)>127)
6510 IF TN1=0 THEN RETURN
6520 NEXT DL
6530 RETURN
7000 AD=768:NUM=AC:DIG=3: GOSUB
      5000
7010 NUM=PRE:DIG=3: GOSUB 5000
7020 NUM=NUMB:DIG=4: GOSUB 5000
7030 RETURN
8210 DGT=0
8220 IF PEEK (AD+DGT)=255 THEN RETURN

8230 TONA= PEEK (2304+(2* PEEK (
      AD+DGT))):TONB= PEEK (2304+
      (2* PEEK (AD+DGT))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 DGT=DGT+1: GOTO 8220
9500 POKE 0, PDL (0): POKE 1, PDL
      (1)
9510 CALL 2560
9520 GOTO 9500
9999 END
10000 KE= PEEK (-16384): POKE -16368
      ,0: IF KE<128 THEN 10000: PRINT
      KE: GOTO 10000
32000 END
```



```
10 DIM A$(128),L(256),B$(40)
15 DIM C$(40)
20 DIALPULSE=1000:ONTIME=2:OFFTIME=
  2:INTERDIG=75:BEEP=2600
30 TTN=3072:MF=3104:KP=1100:ST=
  1500:POKE 2,100:POKE 3,2
40 AD=2048:L(1)=16384:REENTER=
  820:POKE 12,0:POKE 13,64:
  STUFF=768
50 CALL REENTER
60 PTR=PEEK(14)+256*PEEK(15)
  ):LINE=PEEK(17)+256*PEEK
  (18)+1
70 IF LINE<=0 THEN LINE=1
80 TONEDIAL=8210:PLL=7000
90 PW=0
100 REM -----ROUTINES
110 INIT=1110:ENTNUM=1205:LOOKUP=
  1310
130 NUMS=3328:GOSUB INIT
140 INPUT "PRINT NUMBERS?",B$:IF
  B$(1,1)="N" THEN 200:INPUT
  "CODE=",A$
150 IF A$="ASDFGHJKL" THEN PW=1

200 CALL -936:INPUT "NAME?",B$

210 IF B$="EN" THEN 1400:REM ENTER

215 IF B$#"69" THEN 220
218 PW=1:GOTO 200
220 IF B$="FIX" THEN 0:REM FIX
225 IF B$#"00" THEN 230
228 PW=0:GOTO 200
230 IF B$="DIS" THEN 1500:REM DISP
  LAY
240 IF B$="END" THEN END
250 IF B$="LOOK" THEN 2000
260 IF B$(1,1)#"-" THEN 300
270 FLAG=1:B$=B$(2)
300 FOR L=1 TO LINE:POKE 12,L(
  L)MOD 256:POKE 13,L(L)/256
  :CALL STUFF
310 IF B$=A$ THEN 500:NEXT L:PRINT
  "SORRY, NO SUCH ENTRY"
320 INPUT "WANNA SEE THE NAMES?"
  ,B$:IF B$(1,1)="Y" THEN 200

330 INPUT "ENTER NUMBER--ANY FORMAT:
  ",B$:GOTO 590
480 GOSUB INIT
500 PRINT A$;" ";:GOSUB LOOKUP
510 IF PW=1 THEN PRINT B$:PRINT
  :IF FLAG#1 THEN 520
515 DLY=900:GOSUB 6000:FLAG=0:
  GOTO 200
```

```

520 PT=NUMS+6*(L-1):PX= PEEK (PT+
2)+256* PEEK (PT+3)
530 IF PX#0 THEN 590
540 VTAB 20: PRINT "NO PHONE NUMBER
FOR ",A$:DLY=350: GOSUB 6000
: GOTO 200
590 VTAB 20
600 INPUT "WANNA USE A WATS EXTENDER
?",A$: IF A$(1,1)="Y" THEN
670: IF A$(1,1)="P" THEN 200

610 START=2048:A$=B$: GOSUB ENTNUM:
AD=2048
615 PRINT "PLEASE PICK UP PHONE"

620 POKE -16228,0:DLY=350: GOSUB
6000: POKE -16227,0
630 DL1=100: POKE -16231,4:T1=5
: GOSUB PLL
640 IF T<T1 THEN 620: GOSUB DIALPULS
E: PRINT
650 INPUT "WANNA TRY AGAIN?",A$
:AD=2048: IF A$(1,1)="Y" THEN
620
660 GOTO 200
670 CALL -936: PRINT "WHICH ONE?"

672 PRINT "1. NEW JERSEY #1"
673 PRINT "2. NEW JERSEY #2"
674 PRINT "3. ATLANTA"
675 PRINT "4. FLORIDA"
676 PRINT "5. MINNIAPOLIS"
680 INPUT "TYPE 1-5 :",WH: IF WH<
1 OR WH>5 THEN 670
690 IF WH=1 THEN A$="18006310725*555
5*85*"
700 IF WH=2 THEN A$="18006310744*134
3*85*"
710 IF WH=3 THEN A$="18002413466*551
1*85*"
720 IF WH=4 THEN A$="18003273282*370
2*81*"
730 IF WH=5 THEN A$="18003288308*106
0*85*"
740 A$(1+ LEN(A$))=B$
750 GOSUB INIT: PRINT "PLEASE PICK U
P PHONE"
760 START=2048: GOSUB ENTNUM
770 AD=2048
780 POKE -16228,0:DLY=350: GOSUB
6000: POKE -16227,0
790 DL1=100: POKE -16231,4:T1=5
: GOSUB PLL
800 IF T<T1 THEN 780: GOSUB DIALPULS
E:TBL=TTN
810 DL1=600:T1=10: POKE -16231,

```

```
6: GOSUB 7000
820 IF T=T1 THEN 850
830 CALL -936: INPUT "YOUR CALL FAIL
ED--WANNA TRY AGAIN",C$
840 IF C$(1,1)="Y" THEN 770: IF
C$(1,1)="T" THEN 670: GOTO
200
850 AD=AD+1: GOSUB TONEDIAL:DL1=
50: POKE -16231,20:T1=10: GOSUB
PLL
860 IF T<T1 THEN 880
870 PRINT "THE CODE WAS CHANGED-TRY
ANOTHER WATS EXTENDER":DLY=
800: GOSUB 6000: GOTO 670
880 DL1=100: POKE -16231,6:T1=5
: GOSUB PLL: IF T=T1 THEN 850
: PRINT
890 INPUT "IS YOUR CALL SUCESSFUL?"
,C$: IF C$(1,1)="N" THEN 770
: IF C$(1,1)="T" THEN 670
900 GOTO 200
1000 REM ---DIAL PULSE ROUTINE--
1010 IF PEEK (AD)=255 THEN RETURN

1020 COUNT= PEEK (AD)
1025 IF PW=1 THEN PRINT COUNT;
1030 IF COUNT=0 THEN COUNT=10
1040 FOR PULSE=1 TO COUNT
1050 POKE -16228,0:DLY=ONTIME: GOSUB
6000
1060 POKE -16227,0:DLY=OFFTIME: GOSUB
6000
1070 NEXT PULSE
1080 DLY=INTERDIG: GOSUB 6000:AD=
AD+1: GOTO 1010
1110 FOR A=2048 TO 2303: POKE A,
255: NEXT A: RETURN
1205 ADS=START
1210 FOR P=1 TO LEN(A$)
1220 VAL= ASC(A$(P))
1230 IF VAL=170 THEN 1260
1240 IF VAL<176 OR VAL>185 THEN
1270
1250 POKE ADS,VAL-176
1260 ADS=ADS+1
1270 NEXT P: RETURN
1310 POINT=NUMS+6*(L-1):AC= PEEK
(POINT)+256* PEEK (POINT+1)

1320 PRE= PEEK (POINT+2)+256* PEEK
(POINT+3)
1330 NUMB= PEEK (POINT+4)+256* PEEK
(POINT+5)
1340 AD=5777:DIG=4:NUM=AC: GOSUB
5000
1350 DIG=3:NUM=PRE: GOSUB 5000
```

```

1360 DIG=4:NUM=NUMB: GOSUB 5000:
      ADR=AD+1: RETURN
1400 CALL -936
1410 INPUT "NAME:",A$: IF A$="END"
      THEN 200
1420 FOR I=1 TO LEN(A$): POKE PTR,
      ASC(A$(I)):PTR=PTR+1: NEXT
      I
1430 LINE=LINE+1: POKE PTR,30:PTR=
      PTR+1:L(LINE)=PTR: POKE PTR,
      0
1440 INPUT "NUMBER--AC,PRE,NUMB:"
      ,AC,PRE,NUMB
1450 PT=NUMS+6*(LINE-2)
1460 POKE PT,AC MOD 256: POKE PT+
      1,AC/256
1470 POKE PT+2,PRE MOD 256: POKE
      PT+3,PRE/256
1480 POKE PT+4,NUMB MOD 256: POKE
      PT+5,NUMB/256
1490 GOTO 1410
1500 CALL -936
1505 VTAB 22: PRINT "TYPE 'E' TO EXIT
      ": INPUT "LINE=",L: GOTO 1590

1510 KEY= PEEK (-16384): POKE -16368
      ,0
1520 IF KEY=197 THEN 200
1530 IF KEY=160 THEN 1580
1540 IF KEY=204 THEN 1500
1550 IF KEY=208 THEN 1570
1560 GOTO 1510
1570 L=L+21
1580 L=L+1
1585 CALL -936
1590 IF L>LINE THEN 1505: IF L<=
      0 THEN L=1
1595 POKE 16,L: CALL 901
1598 GOTO 1510
2000 CALL -936: IF PW=0 THEN 200

2010 INPUT "LINE #",L: IF L>LINE OR
      L<0 THEN 2010
2020 KEY= PEEK (-16384): POKE -16368
      ,0
2030 IF KEY=197 THEN 200: IF KEY=
      193 THEN 2050
2035 IF KEY=195 THEN 2130
2040 IF KEY=204 THEN 2010
2045 GOTO 2020
2050 L=L+1
2055 CALL -936
2060 POKE 12,L(L) MOD 256: POKE
      13,L(L)/256: CALL STUFF
2070 VTAB 2: PRINT "

```

```

2080 POINT=NUMS+6*(L-1)
2090 AC= PEEK (POINT)+256*(( PEEK
      (POINT+1)) MOD 127)
2100 PRE= PEEK (POINT+2)+256*(( PEEK
      (POINT+3)) MOD 127)
2110 NUMB= PEEK (POINT+4)+256*((
      PEEK (POINT+5)) MOD 127)
2120 VTAB 2: PRINT L;" ";A$;" ";
      AC;"-";PRE;"-";NUMB: GOTO 2020

2130 VTAB 4: PRINT "OLD=";AC;"-";
      ;PRE;"-";NUMB
2140 VTAB 5: INPUT "NEW=",AC,PRE,
      NUMB
2145 POKE POINT,AC MOD 256: POKE
      POINT+1,AC/256
2150 POKE POINT+2,PRE MOD 256: POKE
      POINT+3,PRE/256
2160 POKE POINT+4,NUMB MOD 256: POKE
      POINT+5,NUMB/256: GOTO 2020

5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A+176
      :AD=AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW: POKE AD,30
5030 RETURN
6000 FOR D=1 TO DLY: NEXT D: RETURN

7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
8000 KEY= PEEK (-16384): POKE -16368
      ,0: IF KEY>127 THEN PRINT KEY:
      GOTO 8000
8210 REM TONEDIAL ROUTINE
8220 IF PEEK (AD)=255 THEN RETURN

8225 IF PW=1 THEN PRINT PEEK (AD)
      ;
8230 TONA= PEEK (TBL+(2* PEEK (AD)
      )):TONB= PEEK (TBL+(2* PEEK
      (AD))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
      2560
8260 AD=AD+1: GOTO 8220
30000 END

```

```

10 DIM A$(128),L(256),B$(40)
15 DIM C$(40)
20 DIALPULSE=1000:ONTIME=4:OFFTIME=
  4:INTERDIG=75:BEEP=2600
30 TTN=3072:MF=3104:KP=1100:ST=
  1500:POKE 2,100:POKE 3,2
40 AD=2048:L(1)=16384:REENTER=
  820:POKE 12,0:POKE 13,64:
  STUFF=768
50 CALL REENTER
60 PTR=PEEK(14)+256*PEEK(15)
  ):LINE=PEEK(17)+256*PEEK
  (18)+1
70 IF LINE<=0 THEN LINE=1
80 TONEDIAL=8210:PLI=7000
90 PW=0
100 REM -----ROUTINES
110 INIT=1110:ENTNUM=1205:LOOKUP=
  1310
130 NUMS=3328:GOSUB INIT
140 INPUT "PRINT NUMBERS?",B$:IF
  B$(1,1)="N" THEN 200:INPUT
  "CODE=",A$
150 IF A$="ASDFGHJKL" THEN PW=1

200 CALL -936:INPUT "NAME?",B$

210 IF B$="EN" THEN 1400:REM ENTER

215 IF B$#"69" THEN 220
218 PW=1:GOTO 200
220 IF B$="FIX" THEN 0:REM FIX
225 IF B$#"00" THEN 230
228 PW=0:GOTO 200
230 IF B$="DIS" THEN 1500:REM DISP
  LAY
240 IF B$="END" THEN END
250 IF B$="LOOK" THEN 2000
260 IF B$(1,1)#"- " THEN 300
270 FLAG=1:B$=B$(2)
300 FOR L=1 TO LINE:POKE 12,L(
  L) MOD 256:POKE 13,L(L)/256
  :CALL STUFF
310 IF B$=A$ THEN 500:NEXT L:PRINT
  "SORRY, NO SUCH ENTRY"
320 INPUT "WANNA SEE THE NAMES?"
  ,B$:IF B$(1,1)="Y" THEN 200

330 INPUT "ENTER NUMBER--ANY FORMAT:
  ",B$:GOTO 590
480 GOSUB INIT
500 PRINT A$;" ";:GOSUB LOOKUP
510 IF PW=1 THEN PRINT B$:PRINT
  :IF FLAG#1 THEN 520
515 DLY=900:GOSUB 6000:FLAG=0:
  GOTO 200

```

```

520 PT=NUMS+6*(L-1):PX= PEEK (PT+
    2)+256* PEEK (PT+3)
530 IF PX#0 THEN 590
540 VTAB 20: PRINT "NO PHONE NUMBER
    FOR ",A$:DLY=350: GOSUB 6000
    : GOTO 200
590 VTAB 20
600 INPUT "WANNA USE A WATS EXTENDER
    ?",A$: IF A$(1,1)="Y" THEN
    670: IF A$(1,1)="P" THEN 200

610 START=2048:A$=B$: GOSUB ENTNUM:
    AD=2048
615 PRINT "PLEASE PICK UP PHONE"

620 POKE -16228,0:DLY=350: GOSUB
    6000: POKE -16227,0
630 DL1=100: POKE -16231,4:T1=5
    : GOSUB PLL
640 IF T<T1 THEN 620: GOSUB DIALPULS
    E: PRINT
650 INPUT "WANNA TRY AGAIN?",A$
    :AD=2048: IF A$(1,1)="Y" THEN
    620
660 GOTO 200
670 CALL -936: PRINT "WHICH ONE?"

672 PRINT "1. NEW JERSEY #1"
673 PRINT "2. NEW JERSEY #2"
674 PRINT "3. ATLANTA"
675 PRINT "4. FLORIDA"
676 PRINT "5. MINNIAPOLIS"
680 INPUT "TYPE 1-5 :",WH: IF WH<
    1 OR WH>5 THEN 670
690 IF WH=1 THEN A$="18006310725*555
    5*85*"
700 IF WH=2 THEN A$="18006310744*134
    3*85*"
710 IF WH=3 THEN A$="18002413466*551
    1*85*"
720 IF WH=4 THEN A$="18003273282*370
    2*81*"
730 IF WH=5 THEN A$="18003288308*106
    0*85*"
740 A$(1+ LEN(A$))=B$
750 GOSUB INIT: PRINT "PLEASE PICK U
    P PHONE"
760 START=2048: GOSUB ENTNUM
770 AD=2048
780 POKE -16228,0:DLY=350: GOSUB
    6000: POKE -16227,0
790 DL1=100: POKE -16231,4:T1=5
    : GOSUB PLL
800 IF T<T1 THEN 780: GOSUB DIALPULS
    E:TBL=TTN
810 DL1=600:T1=10: POKE -16231,

```

```

6: GOSUB 7000
820 IF T=T1 THEN 850
830 CALL -936: INPUT "YOUR CALL FAIL
ED--WANNA TRY AGAIN",C$
840 IF C$(1,1)="Y" THEN 770: IF
C$(1,1)="T" THEN 670: GOTO
200
850 AD=AD+1: GOSUB TONEDIAL:DL1=
100: POKE -16231,20:T1=5: GOSUB
PLL
860 IF T<T1 THEN 880
870 PRINT "THE CODE WAS CHANGED-TRY
ANOTHER WATS EXTENDER":DLY=
800: GOSUB 6000: GOTO 670
880 DL1=100: POKE -16231,6:T1=5
: GOSUB PLL: IF T=T1 THEN 850
: PRINT
890 INPUT "IS YOUR CALL SUCESSFUL?"
,C$: IF C$(1,1)="N" THEN 770
: IF C$(1,1)="T" THEN 670
900 GOTO 200
1000 REM ---DIAL PULSE ROUTINE--
1010 IF PEEK (AD)=255 THEN RETURN

1020 COUNT= PEEK (AD)
1025 IF PW=1 THEN PRINT COUNT;
1030 IF COUNT=0 THEN COUNT=10
1040 FOR PULSE=1 TO COUNT
1050 POKE -16228,0:DLY=ONTIME: GOSUB
6000
1060 POKE -16227,0:DLY=OFFTIME: GOSUB
6000
1070 NEXT PULSE
1080 DLY=INTERDIG: GOSUB 6000:AD=
AD+1: GOTO 1010
1110 FOR A=2048 TO 2303: POKE A,
255: NEXT A: RETURN
1205 ADS=START
1210 FOR P=1 TO LEN(A$)
1220 VAL= ASC(A$(P))
1230 IF VAL=170 THEN 1260
1240 IF VAL<176 OR VAL>185 THEN
1270
1250 POKE ADS,VAL-176
1260 ADS=ADS+1
1270 NEXT P: RETURN
1310 POINT=NUMS+6*(L-1):AC= PEEK
(PPOINT)+256* PEEK (PPOINT+1)

1320 PRE= PEEK (PPOINT+2)+256* PEEK
(PPOINT+3)
1330 NUMB= PEEK (PPOINT+4)+256* PEEK
(PPOINT+5)
1340 AD=5777: DIG=4: NUM=AC: GOSUB
5000
1350 DIG=3: NUM=PRE: GOSUB 5000

```



```

1360 DIG=4:NUM=NUMB: GOSUB 5000:
      ADR=AD+1: RETURN
1400 CALL -936
1410 INPUT "NAME:",A$: IF A$="END"
      THEN 200
1420 FOR I=1 TO LEN(A$): POKE PTR,
      ASC(A$(I)):PTR=PTR+1: NEXT
      I
1430 LINE=LINE+1: POKE PTR,30:PTR=
      PTR+1:L(LINE)=PTR: POKE PTR,
      0
1440 INPUT "NUMBER--AC,PRE,NUMB:"
      ,AC,PRE,NUMB
1450 PT=NUMS+6*(LINE-2)
1460 POKE PT,AC MOD 256: POKE PT+
      1,AC/256
1470 POKE PT+2,PRE MOD 256: POKE
      PT+3,PRE/256
1480 POKE PT+4,NUMB MOD 256: POKE
      PT+5,NUMB/256
1490 GOTO 1410
1500 CALL -936
1505 VTAB 22: PRINT "TYPE 'E' TO EXIT
      ": INPUT "LINE=",L: GOTO 1590

1510 KEY= PEEK (-16384): POKE -16368
      ,0
1520 IF KEY=197 THEN 200
1530 IF KEY=160 THEN 1580
1540 IF KEY=204 THEN 1500
1550 IF KEY=208 THEN 1570
1560 GOTO 1510
1570 L=L+21
1580 L=L+1
1585 CALL -936
1590 IF L>LINE THEN 1505: IF L<=
      0 THEN L=1
1595 POKE 16,L: CALL 901
1598 GOTO 1510
2000 CALL -936: IF PW=0 THEN 200

2010 INPUT "LINE #",L: IF L>LINE OR
      L<0 THEN 2010
2020 KEY= PEEK (-16384): POKE -16368
      ,0
2030 IF KEY=197 THEN 200: IF KEY=
      193 THEN 2050
2035 IF KEY=195 THEN 2130
2040 IF KEY=204 THEN 2010
2045 GOTO 2020
2050 L=L+1
2055 CALL -936
2060 POKE 12,L(L) MOD 256: POKE
      13,L(L)/256: CALL STUFF
2070 VTAB 2: PRINT "

```

```

2080 POINT=NUMS+6*(L-1)
2090 AC= PEEK (POINT)+256*(( PEEK
    (POINT+1)) MOD 127)
2100 PRE= PEEK (POINT+2)+256*(( PEEK
    (POINT+3)) MOD 127)
2110 NUMB= PEEK (POINT+4)+256*((
    PEEK (POINT+5)) MOD 127)
2120 VTAB 2: PRINT L;" ";A$;" ";
    AC;"-";PRE;"-";NUMB: GOTO 2020

2130 VTAB 4: PRINT "OLD=";AC;"-";
    ;PRE;"-";NUMB
2140 VTAB 5: INPUT "NEW=",AC,PRE,
    NUMB
2145 POKE POINT,AC MOD 256: POKE
    POINT+1,AC/256
2150 POKE POINT+2,PRE MOD 256: POKE
    POINT+3,PRE/256
2160 POKE POINT+4,NUMB MOD 256: POKE
    POINT+5,NUMB/256: GOTO 2020

5000 FOR POW=DIG-1 TO 0 STEP -1
5010 A=NUM/10 ^ POW: POKE AD,A+176
    :AD=AD+1:NUM=NUM-A*10 ^ POW
5020 NEXT POW: POKE AD,30
5030 RETURN
6000 FOR D=1 TO DLY: NEXT D: RETURN

7000 T=0
7010 FOR DL=0 TO DL1
7020 TN1=( PEEK (-16238)>127)
7030 IF NOT TN1 THEN T=T+1
7040 IF T=T1 THEN RETURN
7060 NEXT DL
7070 RETURN
8000 KEY= PEEK (-16384): POKE -16368
    ,0: IF KEY>127 THEN PRINT KEY:
    GOTO 8000
8210 REM TONEDIAL ROUTINE
8220 IF PEEK (AD)=255 THEN RETURN

8225 IF PW=1 THEN PRINT PEEK (AD)
    ;
8230 TONA= PEEK (TBL+(2* PEEK (AD)
    )):TONB= PEEK (TBL+(2* PEEK
    (AD))+1)
8240 POKE 0,TONA: POKE 1,TONB: CALL
    2560
8260 AD=AD+1: GOTO 8220
30000 END

```

```

1 REM EDITOR VER 1
10 DIM A$(128),L(256),B$(40)
20 BUFF=16384:LINE=1:STUFF=768
   :CLR=788:L(1)=16384:TAB=1
30 DELAY=5000:PTR=BUFF:REENTER=
   820:POKE 0,0:POKE 1,64
32 CALL -936
35 CALL REENTER
38 PTR=PEEK(2)+256*PEEK(3)
   :LINE=PEEK(6)+256*PEEK(
   7)+1
40 IF LINE<=0 THEN LINE=1
42 PRINT "TYPE ? FOR LIST OF COMM
   ANDS":GOSUB DELAY
45 CALL -936
50 VTAB 22:INPUT "=>",B$
110 IF B$(1,2)="EX" THEN END
120 IF B$(1,2)="CL" THEN 2000
130 IF B$(1,2)="LI" THEN 1500
140 IF B$(1,2)="TE" THEN 2500
150 IF B$(1,2)="EN" THEN 1000
990 GOTO 45
1000 CALL -936
1010 VTAB TAB:PRINT LINE;:INPUT
   " ",A$:IF A$="END" THEN 32

1020 FOR I=1 TO LEN(A$):POKE PTR,
   ASC(A$(I)):PTR=PTR+1:NEXT
   I
1025 TAB=TAB+1:IF TAB>21 THEN 1040

1030 LINE=LINE+1:POKE PTR,30:PTR=
   PTR+1:L(LINE)=PTR:POKE PTR,
   0:GOTO 1010
1040 TAB=1:CALL -936:GOTO 1030

1500 CALL -936
1505 VTAB 22:PRINT "TYPE 'E' TO EXIT
   "
1508 INPUT "PAGE #",P:PAGE=BUFF+
   P*64:POKE 0,PAGE MOD 256:POKE
   1,PAGE/256
1510 KEY=PEEK(-16384):POKE -16368
   ,0
1520 IF KEY=197 THEN 45
1530 P=PDL(0):IF PDL(0)=P THEN
   1510
1540 CALL -936:L=PDL(0)-5:IF
   L>LINE THEN 1510:IF L<1 THEN
   L=1
1545 PRINT L
1550 POKE 8,L:CALL 901
1560 GOTO 1510
2000 POKE 0,0:POKE 1,64:CALL 788

2010 LINE=1:GOTO 45

```

```
2500 CALL -936: VTAB 22: INPUT "LIN="
      ,L: POKE 8,L: CALL 901
2510 END
5000 FOR DL=1 TO 1000: NEXT DL: RETURN
30000 END
```

```

1 REM EDITOR VER 1
10 DIM A$(128),L(256),B$(40)
20 BUFF=16384:LINE=1:STUFF=768
   :CLR=788:L(1)=16384:TAB=1
30 DELAY=5000:PTR=BUFF:REENTER=
   820:POKE 0,0:POKE 1,64
32 CALL -936
35 CALL REENTER
38 PTR=PEEK(2)+256*PEEK(3)
   :LINE=PEEK(6)+256*PEEK(
   7)+1
40 IF LINE<=0 THEN LINE=1
42 PRINT "TYPE ? FOR LIST OF COMM
   ANDS":GOSUB DELAY
45 CALL -936
50 VTAB 22:INPUT "=>",B$
110 IF B$(1,2)="EX" THEN END
120 IF B$(1,2)="CL" THEN 2000
130 IF B$(1,2)="LI" THEN 1500
140 IF B$(1,2)="TE" THEN 2500
150 IF B$(1,2)="EN" THEN 1000
990 GOTO 45
1000 CALL -936
1010 VTAB TAB:PRINT LINE;:INPUT
   " ",A$:IF A$="END" THEN 32

1020 FOR I=1 TO LEN(A$):POKE PTR,
   ASC(A$(I)):PTR=PTR+1:NEXT
   I
1025 TAB=TAB+1:IF TAB>21 THEN 1040

1030 LINE=LINE+1:POKE PTR,30:PTR=
   PTR+1:L(LINE)=PTR:POKE PTR,
   0:GOTO 1010
1040 TAB=1:CALL -936:GOTO 1030

1500 CALL -936
1505 VTAB 22:PRINT "TYPE 'E' TO EXIT
   "
1508 INPUT "PAGE #",P:PAGE=BUFF+
   P*64:POKE 0,PAGE MOD 256:POKE
   1,PAGE/256
1510 KEY=PEEK(-16384):POKE -16368
   ,0
1520 IF KEY=197 THEN 45
1530 P=PDL(0):IF PDL(0)=P THEN
   1510
1540 CALL -936:L=PDL(0)-5:IF
   L>LINE THEN 1510:IF L<1 THEN
   L=1
1545 PRINT L
1550 POKE 8,L:CALL 901
1560 GOTO 1510
2000 POKE 0,0:POKE 1,64:CALL 788

2010 LINE=1:GOTO 45

```

```
2500 CALL -936: VTAB 22: INPUT "LIN="
      ,L: POKE 8,L: CALL 901
2510 END
5000 FOR DL=1 TO 1000: NEXT DL: RETURN
30000 END
```

```

1 REM EDITOR VER 1
10 DIM A$(128),L(256),B$(40)
20 BUFF=16384:LINE=1:STUFF=768
   :CLR=788:L(1)=16384:TAB=1
30 DELAY=5000:PTR=BUFF:REENTER=
   820:POKE 0,0:POKE 1,64
32 CALL -936
35 CALL REENTER
38 PTR=PEEK(2)+256*PEEK(3)
   :LINE=PEEK(6)+256*PEEK(
   7)+1
40 IF LINE<=0 THEN LINE=1
42 PRINT "TYPE ? FOR LIST OF COMM
   ANDS":GOSUB DELAY
45 CALL -936
50 VTAB 22:INPUT "=>",B$
110 IF B$(1,2)="EX" THEN END
120 IF B$(1,2)="CL" THEN 2000
130 IF B$(1,2)="LI" THEN 1500
140 IF B$(1,2)="TE" THEN 2500
150 IF B$(1,2)="EN" THEN 1000
990 GOTO 45
1000 CALL -936
1010 VTAB TAB:PRINT LINE;:INPUT
   " ",A$:IF A$="END" THEN 32

1020 FOR I=1 TO LEN(A$):POKE PTR,
   ASC(A$(I)):PTR=PTR+1:NEXT
   I
1025 TAB=TAB+1:IF TAB>21 THEN 1040

1030 LINE=LINE+1:POKE PTR,30:PTR=
   PTR+1:L(LINE)=PTR:POKE PTR,
   0:GOTO 1010
1040 TAB=1:CALL -936:GOTO 1030

1500 CALL -936
1505 VTAB 22:PRINT "TYPE 'E' TO EXIT
   "
1508 INPUT "PAGE #",P:PAGE=BUFF+
   P*64:POKE 0,PAGE MOD 256:POKE
   1,PAGE/256
1510 KEY=PEEK(-16384):POKE -16368
   ,0
1520 IF KEY=197 THEN 45
1530 P=PDL(0):IF PDL(0)=P THEN
   1510
1540 CALL -936:L=PDL(0)-5:IF
   L>LINE THEN 1510:IF L<1 THEN
   L=1
1545 PRINT L
1550 POKE 8,L:CALL 901
1560 GOTO 1510
2000 POKE 0,0:POKE 1,64:CALL 788

2010 LINE=1:GOTO 45

```

```
2500 CALL -936: VTAB 22: INPUT "LIN="
      ,L: POKE 8,L: CALL 901
2510 END
5000 FOR DL=1 TO 1000: NEXT DL: RETURN
30000 END
```


2 TODD (MV) 1415-968-4275
3 DANA (PITTS) 1617-623-3217
4 BILL (LI) 1516-694-5859
5 PHIL (BOS) 1617-648-2900
6 LARRY (NY) 1212-934-8825
7 ELI (NY) 1212-339-8610
8 BILL (SJ) 1408-225-7843
9 CHUCK (SD) 1415-845-1415
10 BOB D. (LA) 1213-661-6982
11 JOEL (LI) 1516-928-390
12 WAYNE (NY) 1914-897-5513
13 WAYNE (NJ) 1609-881-8658
14 DAVE (NY) 1914-471-4958
15 SCOTT O. DONNELL 1415-367-1539
16 DAVE WIGAND 1408-733-4332
17 PETE PONCINI 1415-969-5636
18 JOE TOM 0-0-0
19 DAVE BANGLE 1415-591-920
20 DAN BROOKMAN 1213-277-4670
21 MARVE (LA) 1213-376-7467
22 MARK (CHI) 1312-432-6574
23 LEON (CHI) 1312-234-2397
24 ROB (CHI) 1312-234-2584
25 WDME 1207-564-2642
26 ROGER SANDS 1617-275-2516
27 HOWARD MELWORM 1408-295-7751
28 THOR 1213-462-1353
29 RORY 1415-843-1332
30 ISAAC 1415-658-7017
31 DAVE LUTZ 1408-244-1097
32 NANCY 1415-968-8004
33 GRAHAM BEATON 1212-986-8793
34 DANA (CUP) 0-0-0
35 CHARLEY (ORE) 1503-689-4630
36 GEORGE BATES 1415-321-8952
37 DON ERICKSON 1714-687-5910
38 JOHN WALKER 1609-881-3066
39 STEVE CALLAHAN 1213-274-756
40 MMIKE SPENCER 1213-374-2617
41 CHUCK SWEENEY 1415-969-8695
42 SCOTT LANGBRIDGE 1415-687-266
43 MIKE WILLIS 1415-676-4746
44 MIKE DOWLER 1415-585-3408
45 KIM HARRIS 1415-324-1069
46 TIM O HARE 1415-523-7396
47 MMIKE (CHI) 1312-945-7187
48 JOE E. 1303-832-3456
49 JEFF SIELAFF 1408-275-6672
50 RANDY WIGGENTON 1408-247-9747
51 JIM GARD 1415-968-2731
52 CRIS ESPINOSA 1408-257-1044
53 GEORGE (CHI) 1312-287-4254
54 ANDY (BERK) 1415-548-325
55 DAVE KARP 1415-841-4035
56 JOHN GRAHAME 1415-673-6549
57 ED BERGMAN 1415-728-7264

58 ERIC BOSS 1805-985-1028
59 JEFF SHIRMAN 1415-965-3235
60 SCOTT ELLIS 1206-363-5724
61 GRANT 1206-523-9839
62 BOB GEDGEL 1415-365-1412
63 JIM ANDERSON 1212-453-4668
64 WALTER WARREN 1404-458-2651
65 JIM KORN 1415-282-6869
66 ED MATSON 1415-969-7531
67 DON MEDLEY 1408-739-2593
68 RICHARD ROTH 1516-549-8218
69 DAVE DMENT 1415-232-8553
70 NIEL REVELLO 1415-756-9445
71 STEVE SAWYER 0-0-0
72 GARY TRICE 1601-762-8878
73 DONN PARKER 1415-948-9039
74 STEVE WOZ 1408-248-9994
75 AARON BRAM 1215-696-5826
76 JOHN SAWYER 1415-931-7387
77 JJ (DEN) 1303-499-5918
78 JJ (NJ) 0-0-0
79 BOB GILLESPI 1415-494-2746
80 FRED (SEA) 1206-523-3464
81 DAVE ASHTON 1213-472-7177
82 ADAM BAUMAN 1213-276-5262
83 JOHN ARTHUR 1800-538-1767
84 DIANE 1301-655-1285
85 DAVE COLE 1503-345-3178
86 DAVE PACTER 1408-732-8894
87 CURT 1503-484-1055
88 BURRELL SMITH 1415-969-2468
89 PETE (NJ) 1201-372-2425
90 CRIS (OHIO) 1513-767-1062
91 Y.E.S 1415-965-1215
92 Y.E.S #1 1415-965-1555
93 JEFF (UTAH) 1801-224-5965
94 BEN FRANKLIN 1415-526-7738
95 JOHN JAMES 1415-527-6679
96 DAVE & RUSS 1415-326-1347
97 DAVE & RUSS (HOME) 1415-329-1586
98 MARK (MV) 1415-961-657
99 GREG (PA) 1415-327-7890
100 BOB ROACH 1415-549-2450
101 MAX (BERK) 1415-548-4841
102 MIKE MCCAWE 1415-664-9123
103 JIM HOLLIDAY 1415-964-9625
104 ROBERT (PA) 1415-321-6259
105 BOB (BERK) 1415-642-6277
106 RICH (BOS) 1617-767-3983
107 APPLE 1408-996-1020
108 C-1 1800-323-7140
109 C-2 1800-323-7268
110 N-1 1800-221-3321
111 B-1 1800-638-4782
112 WAYNE (NJ-WK) 1609-424-4788
113 WAYNE TOGETHER 1609-881-4040

159

Nov 23 16:45 1977 repertory Page 3

114 STEVE LONG 1212-749-6132
115 JOEL (WRK) 1516-345-4800
116 0-12720-0



Bell Laboratories

subject: Evidence Examination
File 39855-4
Pennsylvania State Police
Incident N6-39474

date: December 14, 1977

from: K. D. Hopper
W. W. Heinze

PART 3

Examination Report, Pennsylvania State Police Property Record, parts of Items 18 and 20.

In Item 18, a cardboard carton marked, "Miscellaneous Papers Seized from Draper's Bedroom" there is an original pencil drawing of a logical circuit for a red box coin telephone fraud device. The switches are labeled "25", "5", and "10" corresponding to 25-cent, 5-cent, and 10-cent coin deposit signals. The output lead is labeled "2200", corresponding to the 2200 Hz tone signal used to indicate coin deposit from single-slot coin telephones. The logical circuit diagram is included as Figure 1.

Referring to Figure 1, shown in the upper left corner of the drawing are two NOR gates connected as a multivibrator. This circuit produces a square wave output at the lead designated "2200".

The rectangular box in the top-center of the diagram labeled "4024" is an integrated circuit (full designation CD4024AE) containing a 7-stage binary counter and is used as a frequency divider. The output identified as $f/64$ provides a square wave of $1/64$ of the input frequency (2200 Hz), approximately 34 Hz, having a period of 29 milliseconds. This represents a continuous train of pulses 15 milliseconds long, separated by 14-millisecond spaces. The output identified as $f/128$ similarly provides a square wave of approximately 17 Hz, having a period of 58 milliseconds, hence a train of 29 millisecond pulses separated by 29-millisecond spaces.

The rectangular box labeled "4017" is an integrated circuit (full designation CD4017AE) which functions as a decade counter/divider having ten outputs. Following reset, output 2 appears after two input cycles of the 34 Hz square wave have been completed, approximately 60 milliseconds. Output 4 appears after four input cycles,

approximately 120 milliseconds. Output 5 appears after five input cycles, approximately 150 milliseconds. The integrated circuits shown next to the push-button switches on the left are flip-flop circuits to be set by the push-buttons and cleared by the decade counter. The remaining integrated circuits shown on the drawing are used to process input and control signals as well as to combine the 5-cent, 10-cent, and 25-cent signals, delivering them to the output terminal.

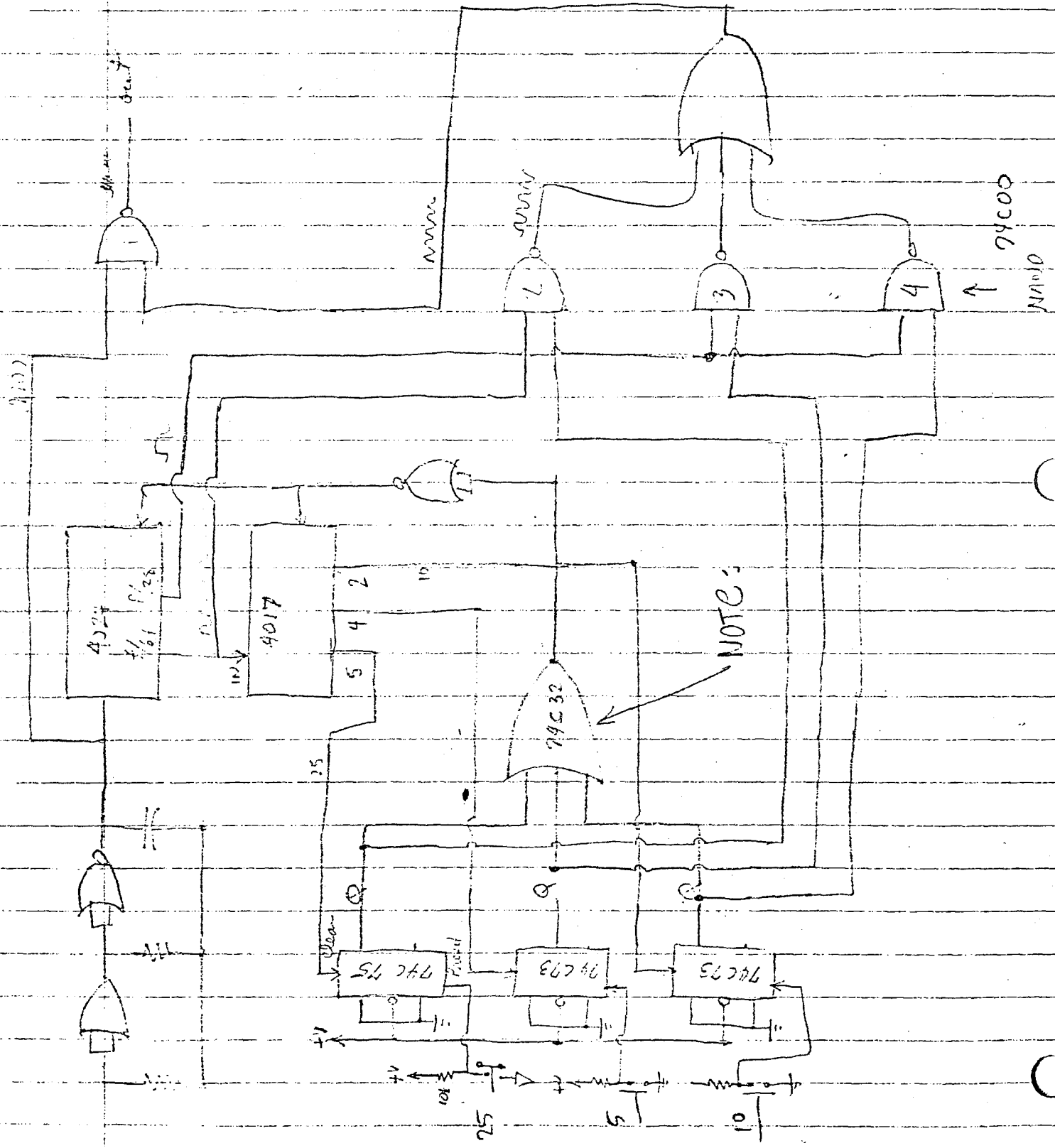
The intended function of the circuit appears to be as follows: When the push-button labeled "25" is closed, counter CD4017 starts and allows an output to occur for 150 milliseconds. Gate 2 opens and passes a train of five pulses derived from the 34 Hz square wave. This signal in turn controls Gate 1, which passes five 15-millisecond bursts of 2200 Hz. This represents the signal for a 25-cent deposit. Closing the other contacts ("5" and "10") similarly allow outputs of 120 and 60 milliseconds. Gates 3 and 4 pass pulses derived from the 17 Hz square wave resulting in one or two 29-millisecond bursts of 2200 Hz. These are the signals for 5-cent and 10-cent deposits, respectively.

The logical circuit design is similar to other red box coin telephone fraud circuits which we have examined over the last five years. On the drawing, the component parts appear to have been priced out at \$13.70 per device.

In Item 20, a small cardboard box contained in a paper bag, are seven plastic boxes, each measuring 3-1/4 inches x 2-1/8 inches x 1-1/8 inches, each mounting three push-button switches, a slide switch and a loudspeaker. These appear to be unfinished red boxes being built in accordance with the previously described logical circuit. Also in the cardboard box are other components which appear on the circuit diagram, e.g., 74C00 integrated circuits, 74C02 integrated circuits, 74C73 integrated circuits, transistor batteries, and additional speakers.

A device was constructed following the logical circuit by Mr. Alfred C. Bandini, a Senior Technical Associate assigned to our organization. He used integrated circuits identical or similar to those shown to perform the required logic functions. After routine deugging, the completed circuit provided one beep when the first button was depressed, two beeps when the second button was depressed, and five fast beeps when the third button was depressed. All beeps are 2200 Hz tone. This simulates the output signal of all Bell System single-slot coin telephones representing coin deposit.

163



74C00	1
74C93	3
74C93	1
74C93	1
4017	1
74C32	1
100pF	2
100pF	2
10k	7
5k	1
10k	1
Board	1
	13

IC's
Res
Cap
Cable
Spkr
100pF
5k
10k

PART 3

FIGURE 1-(Part of Item 18)



Bell Laboratories

subject: Evidence Examination
File 39865-4
Pennsylvania State Police
Incident N6-39474

date: December 13, 1977

from: K. D. Hopper

PART 4

Examination Report, Pennsylvania State Police Property
Record, Items 11-13, 15-18

Documents which appear to relate to methods of telephone toll fraud, privacy intrusion, telephone network subversion, toll evasion, and similar procedures are as follows:

Item 11 - A small memo notebook containing numbers and data.

Entry: under "G"
"Gary Trice 601-762-8878"
a known convicted toll defrauder at Pascagoula, Mississippi.

There are many entries which coincide with listings in repertory of "Apple Computer, Items 20-31 inclusive.

Item 12 - Clipboard containing miscellaneous papers with numbers and diagrams:

Handwritten invitation notice mast headed:
"CAPTN CRUNCH"

Mentions "Charley, the first home computer that has been taught to find demonstrators and WATS Extenders automatically with no intervention." Attached to this is a "Party Invitation List" having 29 persons listed including "Gary Trice" (see Item 11).

Listing of "IN-WATS Prefixes for PA." Thirty-seven entries appear.

Item 13 - Contains nothing pertinent.

7
9
1

PART 4 - 2

Item 15 - Blue Top Flight Notebook

Entry showing remote access usage:
"455-7701/3900/75/1-901-853-9187"

Item 16 - Brown Notebook with assorted papers.

Entry showing remote access usage:
"800-631-0725, 5555, 85, 10 secs.
800-328-8308, 1060, 85, 10 s.
800-327-3282 Code = 3702 DT 81+1+#
241-3466 5511 DT 85+1+#"
(Rubber banded open to this page)

Item 17 - AMSCO Blue Three Ring Binder containing assorted papers.

The information contained in this binder comprises a basic handbook on telephone toll fraud and privacy intrusion. Among the contents are the following:

Pages copied from Engineering handbooks listing multifrequency telephone signaling codes and tolerances.

Pages copied from the November 1960 issue of the Bell System Technical Journal, giving details of most telephone signaling systems of the world.

Pages copied from various newspapers, magazines, and trade journals, giving details of toll fraud cases, detection methods, and convictions.

Pages copied from "High Times" magazine entitled, "Captain Crunch, Super Phone Phreak".

Pages copied from the YIPster TIMES, "TelCo Battles Blue Boxes." Methods of blue box detection are underscored in red.

Handwritten notes of convicted telephone defrauders along with sentences they received.

Pages from 73 Magazine, giving construction information and circuits for toll fraud devices, e.g., red box coin telephone fraud, and blue box toll fraud.

Handwritten notes making references to:

- "Auto-verify circuit"
- "Bell Memo on Tandem Stacking"
- "KP2 for overseas calls"
- "WATTS EXTENDERS (100 banks in an 800 prefix)"
- Japanese service codes
- U.S. routing codes
- TWX area codes
- Plant test codes

Item 18 - Items in cardboard carton marked, "Miscellaneous Papers Seized from Draper's Bedroom."

Significant documents discovered in this container appear to be the following:

Letter dated September 27 addressed "Dear John" and signed "John". This letter makes reference to the UNIX Time Sharing System, invented by Bell Laboratories. There is discussion of encrypted passwords and accounting procedures.

Complete Xerox'd set of Volumes I and II of "Study of Vulnerability of Electronic Communication Systems to Electronic Interception" by the MITRE Corporation, MITRE Technical Report MTR-7439, produced under contract with the Office of Telecommunications Policy, contract TP6AC039, project number 2430, dated January, 1977. The facing page of Volume II is marked, "This document was prepared for authorized distribution. It has not been approved for public release." This report is undoubtedly the most complete discussion of interception techniques we have ever examined. Every transmission mode employed by the Bell System is covered, from loop plant to satellites and underseas cables. Surreptitious entry procedures are covered in detail.

One printed and one Xerox copy of Bell System Practice 817-102-150 marked "Advance Printing" and covering "Computerized Automatic Message Accounting CAMA-C". This equipment has toll fraud detecting capability and is one of the most recently introduced systems. Those parts of the practice relating to toll fraud detection are marked in pencil. The only other item in the same folder is a news clipping captioned, "Thieves have easy access to U.S. computer secrets."

Copy of Bell Telephone Laboratories drawing SD-96608-01-J15A, entitled "MF Oscillator (Main Frame), ATIS, 54 Type, Manually Controlled Interrogation System Schematic." This drawing details a Multi Frequency (MF)

PART 4 - 4

generator using a single stable frequency source and divider circuits.

Pencil schematic drawing of a red-box coin telephone fraud device. On the drawing, switches are labeled "25", "5", and "10", and the output frequency is shown at "2200". These values are completely consistent with all of the red box fraud devices we have examined to date. Component costs are also listed on the drawing. This circuit appears to relate to the seven partially completed red boxes and component parts included in inventory Item 20 (see also Part 3 of this report).

Lists of 800 numbers and accompanying notes. Indicate a systematic attempt to probe various IN-WATS lines. Some MF codes are also listed, e.g.: "KP 808-123 ST".

A description of the Bell System Digital Data System (DDS). This information is copied from pages 130 and 131 of an unknown document.

An original printed Pacific Telephone & Telegraph Co. route table showing operator dialing codes for all of California. The reverse side contains a penciled notation "2600" and an upper limit of "3400" and a lower limit of "500". Such a diagram could be used to explain guard-band blue box operation.

A bundle of loose note pages, mostly appearing to be from a pocket-size notebook. Entries include:

"Call up #NCIC
212-248-6818"

"ESS Mon #'s
329-1067
329-1381
493, 4, 355 ESS"

"1 2 3 4 5 6 7 8 9 0) (apparent credit
6th") card check letter
G U A Q R X Z L N E) format)

"0 1 2 3 4 5 6 7 8 9) (apparent credit
④ ") card check letter
R E M U J Q A W Z H) format)

A pencil schematic drawing entitled, "Phone Interface."
See Part 2 of examination report.

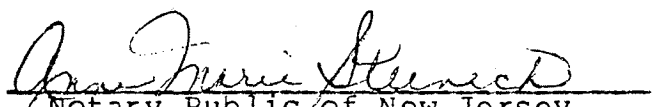
State of New Jersey)
) SS
County of Monmouth)

Kenneth D. Hopper, being duly sworn according to law upon his oath, deposes and says:

1. I am a Member of the Technical Staff of Bell Telephone Laboratories, Incorporated, a Corporation of the State of New York.
2. I have examined the items of evidence referenced herein and my examination report is true in every detail according to my best knowledge, information, and belief.


Kenneth D. Hopper

Sworn and subscribed
before me this 13th
day of December, 1977


Notary Public of New Jersey

My commission expires May 9, 1982



Bell Laboratories

subject: Evidence Examination
File 39365-4
Pennsylvania State Police
Incident N6-39474

date: December 14, 1977
from: W. W. Heinze
K. D. Hopper
P. Rabinowitz

PART 5

Test Procedure, Cresco Central Office, Cresco, Pennsylvania, December 1, 1977

The Apple Computer System and telephone line interface, Items 14, 20-31, inclusive, were set up at the Cresco central office. A telephone line, 595-7181, was provided for test purposes. This line was connected over cable facilities to a terminal approximately 3 miles from the central office and, at that point, was cross-connected to another cable pair going back to the central office to a standard telephone set. This arrangement gave us a test line in the same central office and having similar cable characteristics to that which the equipment was reported to have been connected at the time it was seized. A Hekimian Laboratories Model 51B Dialed Number Recorder was connected at the Main Distributing Frame (MDF) appearance of line 595-7181. This recorder is capable of producing a printed record of all line activity.

The "AUTO-DIAL" program was loaded into the Apple Computer System in the manner described in Part 2 of this report. The "RUN" command was entered by way of the keyboard and the program responded with a request on the television display for "AREA CODE, PREFIX, NUMBER". From the keyboard 1 800 241 3466 was entered. The program then requested a "START" and "END CODE" which was entered via the keyboard as "3248,8251". The telephone handset was then lifted, dial tone was heard. [For the purpose of this description, this dial tone should be considered the first dial tone.] The computer system then automatically dialed the 800 241 3466 IN-WATS telephone number. An operator came on the line and asked for the billing number. This was given and the call was advanced. While this was happening, the computer system asked "WANNA REDIAL". The answer "NO" was entered at the

keyboard, whereupon the computer system proceeded to output a burst of 2600 Hz SF tone followed by a redialing of the number in multi-frequency (MF) key-pulsing tone format. The call completed, was automatically answered by the customer's Remote Access Unit and a second dial tone was returned. The computer recognized the dial tone and output "3243" in TOUCH-TONE® format. This was recognized as the proper authorization code (password) at the Remote Access Unit and a third dial tone was returned. At this point, the computer recycled to the stage of outputting the 2600 Hz SF tone and MF key pulsing again, resulting in answer, second dial tone, authorization code, and third dial tone again. The procedure was done for a total of five successful third dial tones. [This took seven attempts as two attempts failed.] The computer then arrived at the decision that the proper authorization code had been found. The message "WHOOPIE!!!! CODE = 8248" was displayed, a series of audible beeps was heard from the computer loudspeaker, the telephone line was returned to an on-hook state, and the computer program ended. The Hekimian Dialed Number Recorder printout of the "AUTO-DIAL" procedure is included as Figure 1.

At this point, the "DIAL-DEMO" program was loaded. The authorization code on the cassette tape was "5511". As the customer's current authorization code was 8248, it was necessary to rewrite this authorization code from the keyboard. (It should be mentioned that the authorization code, 8248, was provided for the night of December 1, 1977, especially for this test by the IN-WATS customer, the KEM Company. The tests were made with their approval.)

The command "RUN" was entered from the keyboard, and three names and numbers were entered into the program's repertory memory. They were:

"TEST 1 717 595 7101"
(This is the number of another line in the Cresco central office)

"TROOPER 1 717 646 7886"
(This is the residence telephone of Trooper James R. Harris of the Pennsylvania State Police, the investigating officer)

"CHURA 1 717 424 1974"
(This is the residential telephone of Mr. E. B. Chura, Bell Telephone Business Office Manager at Stroudsburg)

PART 5 - 3

The instruction "TEST" was entered from the keyboard, followed by a selection of which IN-WATS line was to be used and the computer proceeded to place a call via KEM Company telephone facilities in Georgia back to the test line at the Cresco central office. This occurred at 7:10 p.m. Messrs. P. Rabinowitz and R. Previte conversed briefly over the connection.

The instruction "TROOPER" was entered from the keyboard, followed by the selection of IN-WATS line and a call was placed through the KEM Company telephone facilities in Georgia to the residence telephone of Trooper Harris at Pocono Summit. The call was answered at 7:15 p.m. and a short conversation was held between Mrs. Harris and Mrs. G. A. Orner.

The instruction "CHURA" was entered from the keyboard, followed by the selection of IN-WATS line and a call was placed through the KEM Company telephone facilities in Georgia to the residence of Mr. E. B. Chura at Stroudsburg. The call was answered at 7:22 p.m. and a short conversation was held between Mr. Chura and Mr. K. D. Hopper.

Throughout the three test calls just described, the television display showed the progress of dialing the IN-WATS number 800 241 3466 in rotary dial pulse format, outputting the authorization code 8248 in TOUCH-TONE format, selection of the out-trunk by digits 85 in TOUCH-TONE, and finally, the selected 717 area telephone number in TOUCH-TONE.

As the three test calls entered the network as 800 IN-WATS calls, there is no billing to the originating line. The KEM Company facilities were unavailable to other users while the test calls were in progress.



Bell Laboratories

PART 5
FIGURE 1

subject: Tests at Cresco, Pennsylvania C.O.
Hekimian Model 51B Printout

date: December 1, 1977

from: W. W. Heinze
K. D. Hopper
P. Rabinowitz

	Hour	Min.	Secs.	
Off Hook	17	47	48	
Rotary Dialing	18002413466			
2600 Hz SF tone	17	48	14	
MF Keypulsing	18002413466-8248			1 Touch Tone Password
2600 Hz SF tone	17	48	34	
MF Keypulsing	18002413466-8248			2 Touch Tone Password
2600 Hz SF tone	17	48	56	
MF Keypulsing	18002413466-8248			3 Touch Tone Password
2600 Hz SF tone	17	49	17	
MF Keypulsing	18002413466-8248			4 Touch Tone Password
2600 Hz SF tone	17	49	36	FAIL
MF Keypulsing	18002413466-			
2600 Hz SF tone	17	50	02	FAIL
MF Keypulsing	18002413466-			
2600 Hz SF tone	17	50	29	5
MF Keypulsing	18002413466-8248			"WHOOPIE" Touch Tone Password "CODE 8248"
On Hook	12	01	17	50 49

Month
Day

Initials of witnesses to test

KAH
PR
WWK
ghs

Dec. 1, 1977
at 1752 HRS
CRESCO, PA.

On line (717) 595-7181

Explanatory comments added by K. D. Hopper
December 8, 1977

KAH

"M" indicates manual call for last line

173

COMMONWEALTH OF PENNSYLVANIA

VS.

JOHN T. DRAPER

: IN THE COURT OF COMMON PLEAS OF
THE FORTY THIRD JUDICIAL DISTRICT

: MONROE COUNTY BRANCH CRIMINAL

: NO. 68 - 1978

GUILTY PLEA COLLOQUY MEMORANDUM

John Thomas Draper did possess an Apple Computer, ~~and~~ related software programs for that computer, *and a telephone line interface*, which were designed, adapted and used for the commission of theft of telecommunications services by means of the application multi-frequency tones and with the capability of:

1.
 - (A) Probing the telephone network in a manner to search out customer lines capable of being subverted.
 - (B) Having found the line(s) described in (1A), the line may then be further searched for its access code (password).
 - (C) Having found the access code (password) (1B, above), calls may then be placed over customers facilities, thereby perpetrating theft of telecommunications services.
2. Placing telephone calls in a manner permitting redirection of the calls to new destinations, and by-passing billing procedures. This method is known as "blue box" calling.
3. Automatic call placing to over 100 pre programmed numbers, or to any number entered manually, using a combination of 1 and 2 above.

Respectfully submitted,

Ralph A. Matergia, Esq.
Assistant District Attorney

COMMONWEALTH OF PENNSYLVANIA

VS.

JOHN T. DRAPER

: IN THE COURT OF COMMON PLEAS OF
THE FORTY THIRD JUDICIAL DISTRICT

: MONROE COUNTY BRANCH CRIMINAL

: NO. 68 - 1978

GUILTY PLEA COLLOUUY MEMORANDUM

John Thomas Draper did possess an Apple Computer, ~~and~~ related software programs for that computer, ^{and a telephone line interface} which were designed, adapted and used for the commission of theft of telecommunications services by means of the application multi-frequency tones and with the capability of:

1.
 - (A) Probing the telephone network in a manner to search out customer lines capable of being subverted.
 - (B) Having found the line(s) described in (1A), the line may then be further searched for its access code (password).
 - (C) Having found the access code (password) (1B, above), calls may then be placed over customers facilities, thereby perpetrating theft of telecommunications services.
2. Placing telephone calls in a manner permitting redirection of the calls to new destinations, and by-passing billing procedures. This method is known as "blue box" calling.
3. Automatic call placing to over 100 pre programmed numbers, or to any number entered manually, using a combination of 1 and 2 above.

Respectfully submitted,

Ralph A. Matergia, Esq.
Assistant District Attorney