To the Graduate Council:

I am submitting herewith a thesis written by Kia Jom Kwang entitled "Microcomputer-Controlled Assembly Voting System." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Electrical Engineering.

Robert W. Rochelle

Robert W. Rochelle, Major Professor

We have read this thesis
and recommend its acceptance:

Accepted for the Council:

The Graduate School

MICROCOMPUTER-CONTROLLED ASSEMBLY VOTING SYSTEM

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Kia Jom Kwang

June 1984

ACKNOWLEDGEMENT

I would like to thank my parents, sisters and brother-in-law for giving me the opportunity to further my career. I would like to give special thanks to Dr. Robert W. Rochelle for his patience, guidance, and encouragement. His warm and friendly attitude will always be remembered. I would like to thank Dr. R. E. Bodenheimer and Dr. H. P. Neff for serving on the committees. I also would like to thank Mr. Russell Rochelle and Mr. Seiko Harimoto for their assistance.

Most of all, I owe a great deal to my wife, Lin, my daughters Jane and Angela. I owe them my complete appreciation and love for their sacrifice while I am working on the thesis.

ABSTRACT

The application of microcomputer in modern control system is a common tool in today's technology. But one using a microcomputer to control the orderly transaction of business in an assembly is not. A microcomputer-controlled assembly voting system is introduced for this purpose.

The main task of a voting system is to enhance legislative and policy making bodies in making decision on certain vital issues. The decision making process could be time consuming and complicated if there is a limitation on the capability of the type of system used.

The AIM-65 microcomputer is used in this particular application. It is specifically programmed to process the gathered information which shows the progress and the results of a certain discussed issue. The author describes the microcomputer-controlled system in detail and developes procedures to test and operate the system.

TABLE OF CONTENTS

CHAPTER		PAGE
I.	INTRODUCTION	1
II.	THE ORIGINAL INSTALLATION	3
III.	THE REVISED SYSTEM	7
	AIM-65 Overview System Configuration System Descriptions and Operations Modes Of System's Operations	
IV.	THE REVISED SYSTEM'S ELEMENTS	20
	Master Control Unit Voter Control Unit Display and Buzzer Unit Printer Unit Power Supply Unit	
V.	SOFTWARE AND HARDWARE CONSIDERATIONS	35
	Software Considerations Hardware Considerations Programming of The 2532 PROM Computer I/O Port Interface Computer sourcing capability	
VI.	SYSTEM POWER-UP INITIALIZATION AND OPERATING PROCEDURE	42
VII.	SUMMARY AND CONCLUSION	45
LIST OF REFERENCES		47
APPENDIXES		49
A.	Supplementary Information on MC14469	
B.	System Flow Charts	
C.	Tables Of Memory Contents In Computer	
D.	Program Listing Of System	
VITA		117

LIST OF TABLES

TABLE		PAGE
1.	Voters' Addressing Addresses (First Transmission, V or D Mode)	79
2.	Discussion Mode Voters' Addressing Addresses (Second Transmission)	80
3.	Voting Mode Voters' Addressing Addresses (Second Transmission)	81
4.	Display Panels Receivers' Addresses	82
5.	Display Panels Command Bit Assignment	83
6.	Transmit Registers Bit Assignment	84
7.	Receive Registers Bit Assignment	84
8.	Voting Results Storage Registers	85
9.	Memory Work Space (Voting Mode)	85
10.	Memory Work Space (Discussion Mode)	86

LIST of FIGURES

FIGURE		PAGE
2.1	Schematic Diagram of The Original Electronic Voting System	4
3.1	The AIM-65 Microcomputer Devices and Peripherals	8
3.2	The AIM-65 Microcomputer Block Diagram	9
3.3	The Revised Voting System Configuration	13
3.4	Circuit Diagram of Single Line Simplex Data Transmission	16
4.1	Schematic Diagram of The Voter Control Panel	22
4.2	Block Diagram of MC14469	24
4.3	Circuit Diagram of Voter Control Box	27
4.4	Schematic Diagram of The Display Panel	30
4.5	Block Diagram of The Display Panel	31
4.6	Circuit Diagram of The Display Panel	32
5.1	Schematic Diagram of The Computer I/O Port Interface	39
5.2	Test Circuit and The Observed Waveform	40
A1.	Block Diagram of MC14469 (AUART) and Pin-Out Assignment	55
A2.	Transmit and Receive Signal of The AUART and Computer	56
A3.	Schematic Diagram of a Typical Receive and Send Cycle	57
A4.	Flow Chart of MC14469 Operation	58

CHAPTER I

INTRODUCTION

Participation in group activities is often an inescapable part of modern life. Groups known as deliberative assemblies are the Government legislature, semipublic associations such as trade unions and corporate boards of directors, or voluntary groups such as parent-teacher-association or honor societies. They share the common practices of adopting parliamentary procedure to conduct meetings to accomplish certain objectives. Their objectives may be to adopt constitutions, secure charters, promulgate by-laws and rules, hold elections of officers, or introduce proposals to be studied and debated systematically.

The term "rules of order" refers to written rules of parliamentary procedure. Robert's Rules of Order, first published in 1876 by U.S. Army General Henry M. Robert, is generally used. Such rules relate to the orderly transaction of business and to the duties of officers in that connection. The objectives of rules of order are to facilitate the smooth functioning of the assembly and to provide a firm basis for resolving questions of procedure that may arise.

Motions are the basic units of parliamentary procedure. They are the means by which business is brought before the assembly. They are proposals on which the legislative or organization takes action or expresses its views on the subject. Debate can come only after a motion is made and the chair has pronounced the motion to be the pending question. When the debate appears to have closed, and if no one arises

to claim the floor, the chair then proceeds to put the question to a vote.

The regular methods of voting on motions are:

1. By voice (viva voce) - the normal method of voting on a motion;

2. By rising - used in verifying an inconclusive voice vote, and in voting on motions requiring a two-thirds vote for adoption; and

3. By show of hands - an alternate method that can be used in place of a rising vote in very small assemblies if no member objects. In some small groups, a vote by show of hands is also used in place of a voice vote as a normal method of voting.

Other methods of voting are voting by ballot and voting by electrical roll-call installation. In electrical voting, some legislative bodies are equipped with an installation providing for push-button voting from each member's seat, and electrical tabulation, which takes the place of a roll call.

In 1979, the Department of Traffic Engineering of the city of Knoxville developed an electronic voting system to record and display the results of a vote for the City-County Government. However, this system encountered several problems. Thus a revised system was developed to replace the old system.

CHAPTER II

THE ORIGINAL INSTALLATION

A schematic diagram of all the elements contained in the electronic voting system is shown in Figure 2.1. The elements are interconnected to each other by bundles of wires and the process of the meeting is controlled by the master control center. A brief description of the function of each of the elements is given below.

Master Control Center

This device is used by the chairperson to control the debates and to initiate the voting process. It also indicates when individual legislative members are ready to vote on any particular issue. The chairperson has the responsibity to call for a vote on the issue.

The master control center allows the chairperson to select either the automatic or manual mode of voting. The automatic mode of voting has a time limit on the voting period and the result of a vote is automatically displayed on the display panels and on the recorder's display panel. The manual mode of voting requires the chairperson to terminate the voting process and to display the result of a vote by activating a manual "display" switch.

The master control center also controls the power supply to the system. Power supply to the system elements is available only when a key is used to turn the supply switch to the "ON" position.

Recorder's Display Panel

The recorder's display aids the recorder in transferring the vote

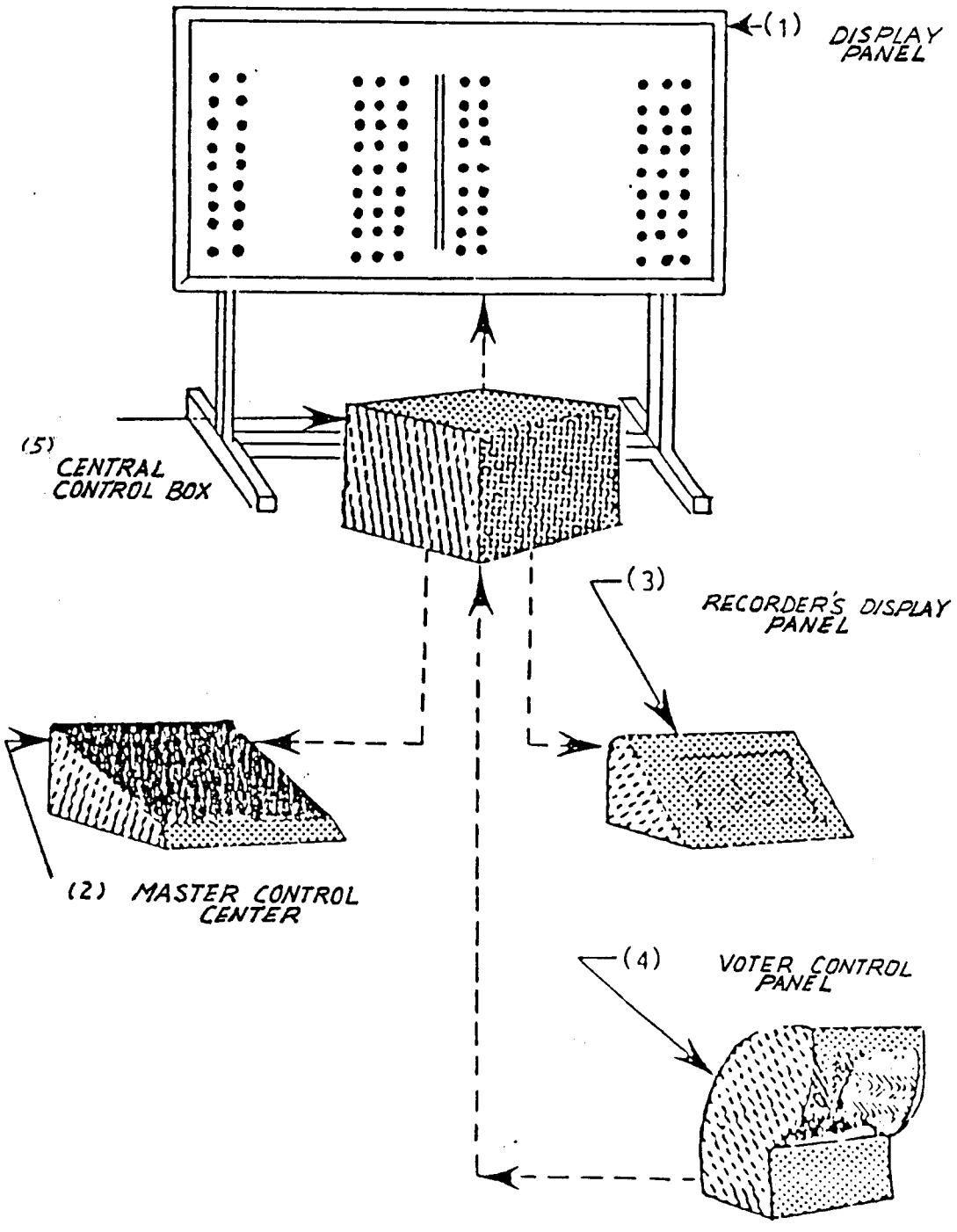


Figure 2.1 Schematic Diagram of The Original Electronic Voting System

to a permanent tally sheet. The lights on the recorder's panel may remain illuminated while those on the main display panels can be turned off so that the voting body may proceed to the next issue without having to wait for the recorder to complete the transfer process.

Voter Control Panel

This panel is the heart of the system. All voting is performed from these panels and all members of the voting body can vote simultaneously. The voting member can vote "yes", "no", or "abstain" by pushing a toggle switch in the appropriate direction until a light corresponding to the desired vote is illuminated. The voting member may also call to be recognized by the chairperson during a debate and ask to call for the question by pushing a similar toggle switch.

Main Display Panels

The main display panels use a series of lights to indicate to the audience and other members of the voting body how the individual members have voted on the issue. The voting member's names are on magnetized strips in-line with the voting lights. The display panels indicate when members have voted on an issue as well as when members of the voting body have called for the floor during discussion on an issue and also indicates which members are present at the meeting.

Central Control Box

This box provides the link between the voter control panel, master control center, recorder's display panel and main display panels. This box also serves as the power source for the system. Each voter control panel is turned on and off from this box.

This system was developed by the Department of Traffic Engineering of the City of Knoxville. It has been used since 1979 and it has functioned satisfactorily. However, several problems were encountered. The main problem faced was due to the large number of wires used. Multi-pin connectors were used to terminate these large number of wires to provide inter-connection between system elements. However, due to the frequent plugging and unplugging of these connectors to reconfigure the system elements, those pins were often damaged and they required regular services and replacement. As a result, the system's cost of maintenance was increased.

CHAPTER III

THE REVISED SYSTEM

The revised system describes the application of a Rockwell AIM-65 microcomputer to control the debate and voting process. The chairperson is the master controller of the microcomputer. Communication between the chairperson and the voters is accomplished via the addressable asynchronous transmitter-receiver (AUART).

The main advantage of using a microcomputer is that of its speed of operation and reliability. It is more efficient and it allows 4-wire serial connections between the voter control boxes and the display panels. It is this feature which gives the added reliability over the original system which has as many as 75-wires in a single connector.

AIM-65 Overview

The AIM-65 computer is a complete microcomputer system which consists of the master module and the keyboard module that are interconnected by a short plug-in ribbon cable. The master module holds a 20-column printer, a 20-character LED display and the micro-computer components. Figure 3.1 identifies the peripherals and microcomputer devices and shows where they are located. The on and off-board expansion capabilities enchance the usability of the AIM-65. The dedicated R6522 versatile interface adapter provides user application ease. The major components are shown in the AIM-65 block diagram in Figure 3.2.

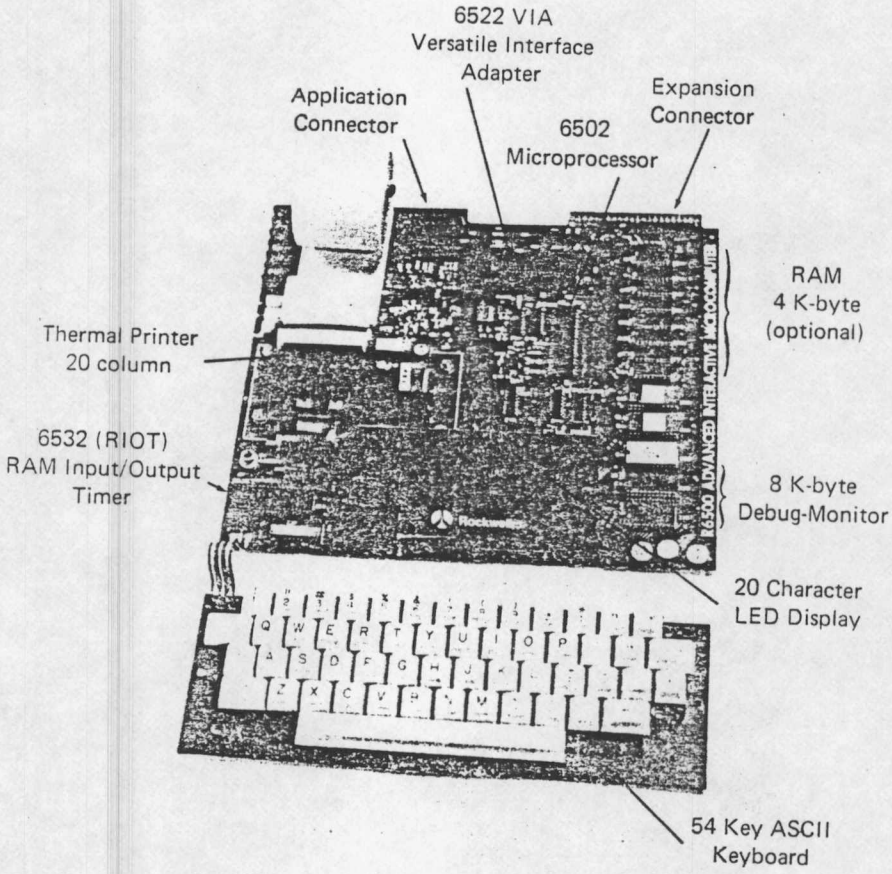


Figure 3.1 The AIM-65 Microcomputer Devices and Peripherals

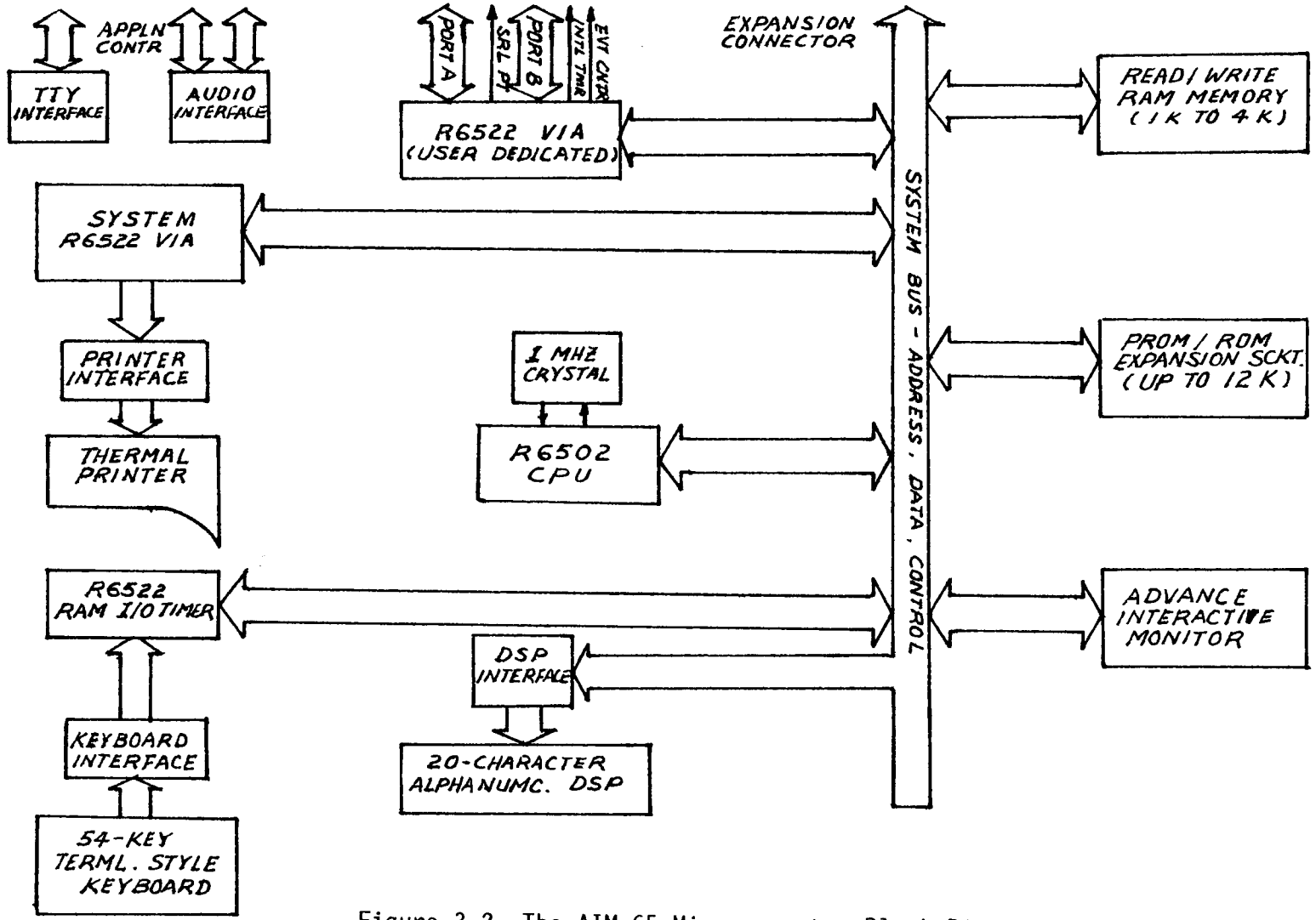


Figure 3.2 The AIM-65 Microcomputer Block Diagram

R6502 CPU. The R6502 central processing unit (CPU) is the heart of the Aim-65. The R6502 is a very widely used and powerful 8-bit microprocesser. It operates at 1 MHz in the Aim-65 to provide a minimum instruction execution time of two micro-seconds. The R6502 has 56 instructions and 13 addressing modes. The R6502 can address 4K bytes of RAM and 20K bytes of ROM on the master module plus an additional 40K bytes of user provided external RAM, ROM or I/O.

Other R6500 Devices. The other R6500 devices on the AIM-65 include the R6522 versatile interface adapter (VIA), the R6532 RAM Input/Output Timer (RIOT), the R6520 peripheral interface adapter (PIA), the R2332 read only memory (ROM), and the R2114 read/write random access memory (RAM).

Thermal Printer. The 20-column thermal printer provides a permanent record of user commands, data, and programs as well as AIM-65 status, prompts, and messages. Printing at 120 lines per minute, the 5 x 7 dot matrix printer provides rapid, quiet, and reliable operation. It can print the complete ASCII 64-character format.

Display. The display provides visual feedback during keyboard operations. The display length of 20 characters allows complete correspondence with the printer. The display uses a 16-segment font to provide a full 64-character ASCII set. The high contrast characters are distinct and easy to read.

Keyboard. The AIM-65 uses a 54-key full-size keyboard which

provides 70 functions (26 alphabetic, 10 numeric, 22 special, 9 control, and 3 user-defined).

AIM-65 Operation. The operation of the AIM-65 is controlled by a ROM-resident 8K monitor which provides a comprehensive set of easy-to-use single keystroke commands. The monitor commands enable the user to enter R6502 instructions in mnemonic form for direct translation to object code, disassemble R6502 instructions from object code to mnemonic form, execute user-written programs with debugging aids such as instruction trace, register trace, and breakpoints. The monitor commands also has the capability of display and alter memory and registers, and transfer object code to and from one or two audio cassette recorders or a teletype. It also allows user-defined functions to interface with user-provided peripherals, and it has an editor which allows easy entry, editing, and listing of R6502 source instructions, data, or general text.

The master module has three spare ROM sockets to allow the addition of 12K bytes of optional AIM-65 programs or user-developed programs.

An optional R6502 assembler, resident in a 4K R2332 ROM may be installed in one of the spare ROM sockets. By using the assembler and editor, the user can rapidly code, edit, assemble and debug programs. Optional AIM-65 8K basic interpreter may be installed in the remaining spare sockets.

More RAM, ROM, PROM, I/O, or other peripheral chips can be added by extending the AIM-65 address, data, and control bus lines to an expansion connector.

The user R6502 has 16 bi-directional input/output lines, four control lines, and two timers for directly interfacing to external equipment.

System Configuration

The voting system block diagram and its configuration are shown in Figure 3.3. The typical triangular arrangement of the master controller and voter stations with the two display panels and with the audience surrounding them offer good co-ordination and visibility.

The chairperson seated at the microcomputer is the master controller of the debate and voting process. A T-shape 4-wire coaxial cable provides the communication link between the chairperson and the voters. Because of the parallel connection of the voter boxes to the coaxial cable, each voter can log on or off the computer by activating the selection switch on the box. It also allows additional voter control stations to be added without much difficulty if the system is to be expanded at a later date.

The dotted block is the microcomputer unit containing the display and the printer. The two main display panels are connected to the computer by the 4-wire bus in the same way as the voter control boxes.

An external TV monitor is shown as an optional feature that can be implemented by interfacing to the computer.

System Descriptions And Operations

The essential task in the design is to program the computer to communicate with the addressable universal asynchronous transmitter and

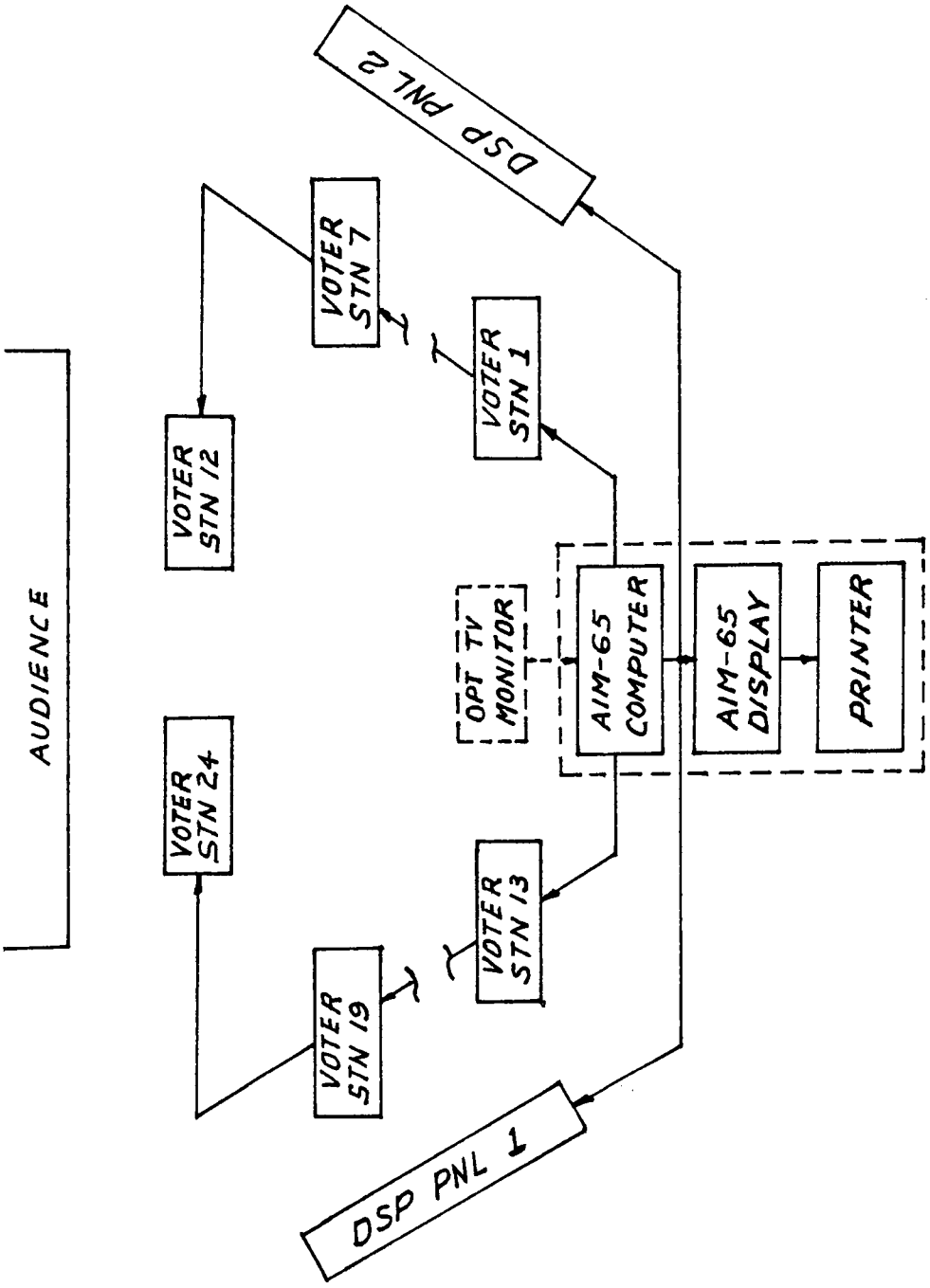


Figure 3.3 The Revised Voting System Configuration

receiver on the voter control boxes. Successful system operation is almost impossible if communication cannot be established.

By making use of the capability of the computer to shift data bit-streams in and out of the I/O port serially, the computer becomes the main transmitter and receiver of the system. The computer communicates by transmitting a 22-bit data stream of which the first 11 bits contains the address of the voter station and the remaining bits contain the information bits. The first 11 bits is made up of 7 address bits, an address identifier bit, a parity bit, and a stop bit. The second 11 bits are made up of 7 information bits, a command identifier bit, a parity bit and a stop bit. Transmission of the 22-bit data stream is done by the shift registers of the computer. The shift registers can only transmit 8 bits at one time. Two dummy bits are added to the 22-bit data to form three 8-bit bytes for serial transmission. Transmission is performed by a software routine which loads each 8-bit byte to the shift registers. The bit time is controlled by an internal timer and the shift register flag is used to signal for fetching the next byte of data to be transmitted when 8 bits have been shifted. Reception of data from the AUART is carried out in a similar way except that the shift registers are programmed to shift in the data serially. The first 8-bit of data is shifted-in the shift registers and stored in memory when a start bit (LOGIC "0") is detected. When the nineth bit of data is shifted-in, the previous 7-bit of data and the nineth bit is again stored in the next memory location. This procedure is repeated for the next word in the incoming data stream. Four bytes of memory are allocated for the received data.

The unique feature of the AUART is that on receiving the transmitted signal from the computer, it performs its own address match with the incoming address. The received information is then processed if valid address was received. Synchronization of operation between computer and AUART is accomplished by setting the bit rate of the transmitted signal from the computer equal to the operating frequency of the AUART. One bit time of the computer transmitted signal is 0.14 ms which is equivalent to the AUART receiver frequency of 7.143 KHz.

Simplex communication is used in this system. A circuit diagram is shown in Figure 3.4. Simplex communication permits communication in both directions, but only one direction at a time. An AUART will transmit its own status immediately upon receiving its address and a command word. Therefore, the interrogations of each voter's station is delayed by 5 milli-seconds to avoid communication collision. The computer transmission is also delayed by 500 milli-seconds if the printer is activated. The reason for the delay is that the printer takes 500 milli-seconds to print a line and this delay time is necessary to prevent the computer and the printer from operating at the same time which might cause corrupted transmission.

Modes Of System's Operations

Discussion Mode. This mode of operation is normally selected by the chairperson at the very beginning of the meeting and later at the choice of the chairperson. This mode is selected by depressing the 'D' key on the keyboard.

When this mode is activated, the voters are notified by the lighting of the discussion mode LED indicator lights on the voter

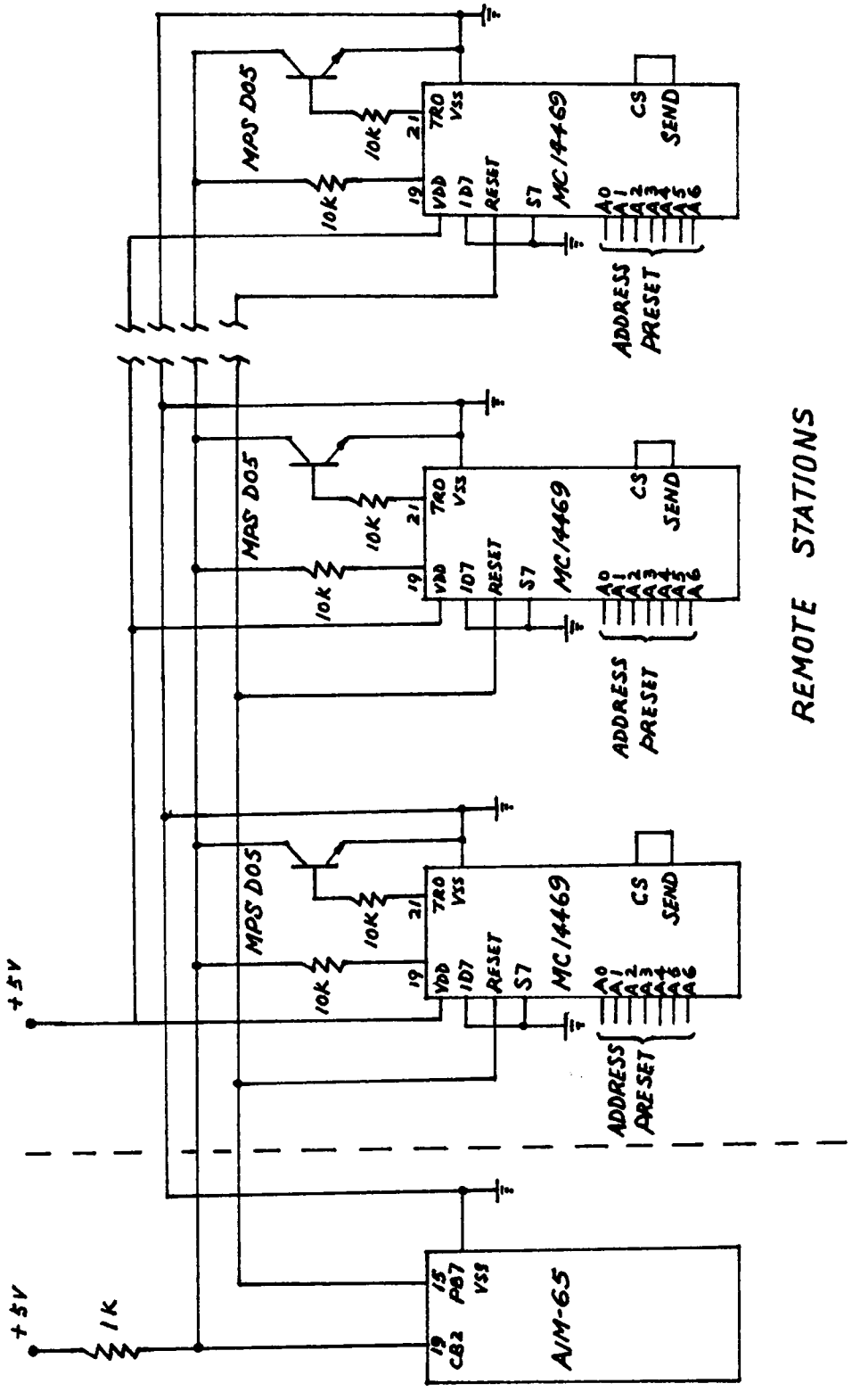


Figure 3.4 Circuit Diagram of Single Line Simplex Data Transmission

control box. They indicate that the issue of the meeting is open to questions and requests from the floor. The voter may signal to the chairperson by depressing either the question or the request button on the control box. When the question or request button is depressed, the incandescent light beside the voter's name on the main display will flash to serve as a visual signal to those present in the meeting. The printer will print the voter's initials when the request button is depressed. Cancellation of the question or request signal is done by depressing the "CANCEL" button. However, it is required that any of the three buttons selected be depressed and held until the corresponding LED light is turned on. The time required for the LED light to turn on is approximately 1 second. The "QUESTION" button is depressed if a voter would like to put a vote to the discussion issue and the "REQUEST" button is depressed to obtain permission to address the floor.

In this mode of operation, the computer will also execute the present-state subroutine to update the voters presence status. The required condition for updating is that the voter box has to be in the "ON" condition.

Voting Mode. In the event that the chairperson notices on the main display panel that six persons have signalled for a vote on a particular issue, he or she would then set the computer to the voting mode by depressing the 'V' key on the keyboard. He may want to get the members attention by talking into the microphone before he sets this particular mode of operation.

When this mode is selected, the voting mode LED indicator light on each voter control box will be illuminated as an indication to the

voter. The voter is allowed a voting time of 10 seconds to cast his vote. A vote cast after 10 seconds will not be registered by the computer. The voter can vote either 'YES', 'ABSTAIN', or 'NO'. The voter is allowed to vote as many times as he likes within the 10-second period but the computer will only register the last vote that is cast. If the voter is present but does not cast a vote, the computer will register that voter as absent during voting.

To serve as an audible and visual signal to all present in the meeting during the voting mode of operation, the buzzer will sound and an incandescent light on the main display panel will flash every 1 second for 10 seconds. The important feature of this mode is that the computer will execute the present-state subroutine only once at the beginning of the voting routine. This is to prevent voters from coming in while the voting process is in progress. The present state indicates those participating voters and any one coming in during the voting period to cast a vote will be displayed as absent even if the display shows his vote. This allows the chairperson to distinguish between a legal and an illegal vote.

Display Mode. The computer will immediately go into this mode of operation once the 10-second voting time is over. The results of the vote cast by all the voters will be displayed on the main display panels. Each and every voter's vote is displayed by an incandescent light so that all the audience can see how the votes were cast.

Print Mode. This mode of operation is executed when the display mode is completed. In this mode, the printer is activated and the

computer will print each voter's name and his vote on hard copy paper. The sum of all the votes will also be printed.

When the print mode is over, the display panel of the computer will again indicate to the chairperson that it is ready to be initiated to go into either the voting or discussion mode. The system circuitry will be re-initialized by the computer before going into the selected mode of operation.

CHAPTER IV

THE REVISED SYSTEM'S ELEMENTS

The basic units of the final system consist mainly of the master controller unit, voter control unit, display and buzzer unit, and the power supply unit.

Master Control Unit

The AIM-65 microcomputer has those features mentioned earlier to meet the desired task.

The terminal keyboard enables the chairperson or an assistant to enter the voter's name into the memory of the computer. It also allows the chairperson to put the computer into either the voting or discussion mode depending on the progress of the meeting by depressing the assigned keys on the keyboard.

The single-row 20-character alphanumeric display enables the chairperson to view the state of progress of the meeting which could either be the discussion or voting modes. The display also enables the assistant to make a correction if an error is detected during the entry of names.

The hard-copy printer prints the voter's name, the vote that has been cast, and the sum of all the three different types of votes on the issue immediately after the voting and display modes are over. During the discussion mode, the printer responds to the voter's request for the floor by printing the voter's name.

The computer reset button is for the chairperson to initialize the state of the computer as well as the state of the voter control box

and the main displays. When this button is activated, the list of voters' names has to be re-entered and the mode must be selected again.

Voter Control Unit

The main purpose of the control box is for the voter to participate in the meeting. Turning on this unit enables the computer to update the voter's present state as well as to communicate to the chairperson. The schematic diagram of the voter control panel is shown in Figure 4.1.

The voter control unit consists of one AUART IC chip, one on/off switch, six push-buttons and nine LED indicator lights. The functions of the components are described below.

On/off Switch. This switch allows the voter to log on or off the computer. The on and off state is indicated by an LED indicator light on the box.

Voting and Discussion Indicators. These indicator lights serve to inform the voter the current operating mode of the computer. Only one of the mode indicator lights is turned on at any one time and the "ON" state indicates
the operating mode.

Q, C, R and Y, A, N Indicators. Q, C, and R stand for question, cancel, and request respectively. Y, A, and N stand for yes, abstain, and no respectively. The Q, C, and R push-button circuitry is activated only during the discussion mode and deactivated during the voting mode. The Y, A, and N push-button circuitry operates vice versa. As a check

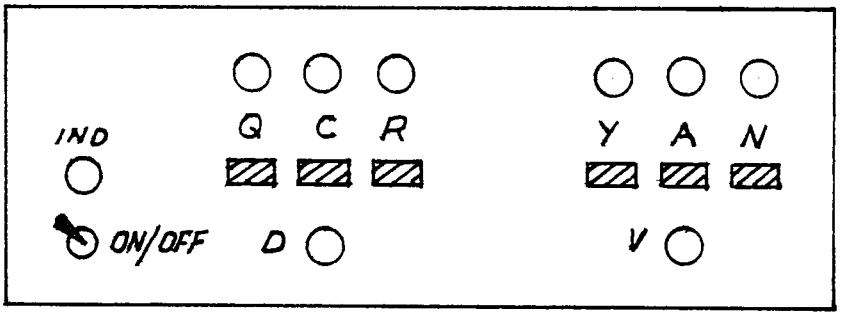


Figure 4.1 Schematic Diagram of The Voter Control Panel

for proper push-button operation, the corresponding indicator light will be turned on when the push-button is depressed and held for 1 second. If at any one time two or three push-buttons were held depressed togather, the indicator lights will not be turned on as an indication that an invalid selection has been made.

AUART Chip. This is the communication switching center for the voter. Its main function is to transmit the voter station address and the encoded messages to the computer for processing. Coding is done when any of the push-buttons are depressed.

The MC14469 is a "UART" specially designed for a remote station. It is a CMOS chip, and it uses very little current. It has a self-contained oscillator and an address comparator that permits the selection of a station when multiple stations are on the same link. A UART protocol is supported in which the frame has even parity and one stop bit. The baud rate is determined by the ceramic resonator connected between pins one and two. The resonator frequency is divided by sixty-four to set the baud rate. A diagram of the MC14469 is shown in Figure 4.2.

The receiver is a standard UART receiver with an address comparator. A seven-bit address is sent as the low order seven bits of an eight-bit word, the most significant bit being true. The station has a seven-bit address, which is selected by strapping pins ten to four low if a zero is needed, or leaving them open if a one bit is needed in the address (these pins have an internal pull-up resistor to make them high if they are not connected). If the incoming address is equal to the station address, the valid address pulse VAP (pin thirty-one) is made

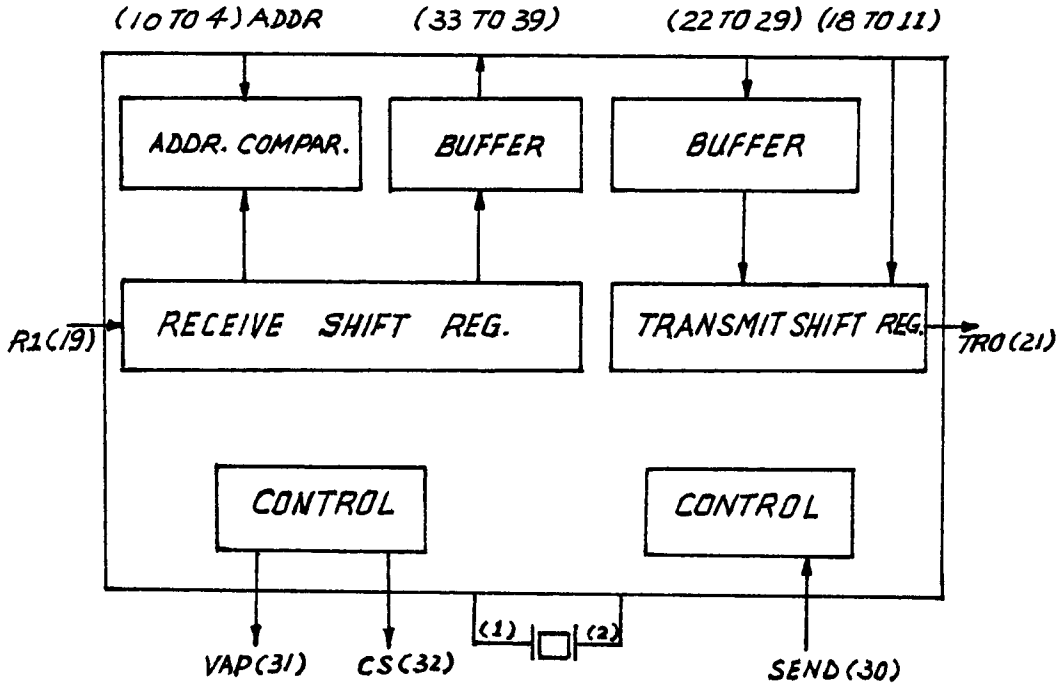


Figure 4.2 Block Diagram of MC14469

high momentarily and the station is said to be selected. A seven-bit data word is sent as the low order seven bits of a word, the most significant bit being false. A station that has been selected will put any data word into its receive buffer when the word is completely shifted in, and make a command strobe CS (pin thirty-two) high momentarily just after this happens. Error status is not available on a pin, but if a parity or framing error is detected, an address will not select a station, data will not be transfered to the receive buffer, and VAP or CS will not be pulsed.

A typical message will consist of a frame with an adrress (most significant bit true) followed by zero or more frames with data (most significant bit false). A single address frame or an address frame followed by a number of data frames can be used to trigger a remote station to do something. The VAP and CS command pulses may be used to command something to be done with the data.

The transmitter is a conventional UART transmitter that is modified to send out sixteen bits of data in two consecutive frames if SEND is made high when VAL or CS is asserted (or within eight data-bit time units after that) and if it is not currently transmitting a pair of frames. Sixteen bits are sent from the signals on pins eleven to eighteen and twenty-nine to twenty-two by transferring the data on pins eleven to eighteen directly into the transmitter shift register, and simultaneously transferring the data on pins twenty- nine to twenty-two into the transmitter buffer. The data in the shift register is sent out (pin eleven data first) in the first UART frame, and the data in the buffer (pin twenty-nine data first) is sent out immediately after that

in the next frame. The data appears on the transmitter output TRO (pin twenty-one) in negative logic. This output is in negative logic so that it can drive a transistor, which inverts the signal, to power the link out of the voter station. A detail description of the AUART is given in appendix A.

Voter Control Unit Hardware Circuit Design. The circuit design of the voter control unit is shown in Figure 4.3.

The AUART chip is controlled by a 455-KHz ceramic oscillator connected between pins 1 and 2. Pin 3 is connected to pin 15 of the computer I/O port as a reset. Pin 4 to pin 10 are the address preset for the voter station. Pin 11 to pin 17 are strapped high and pin 18 low to avoid erroneous communication since it is intended for each voter to communicate to the chairperson only.

Pin 19 is the receiving pin for the incoming signal from the computer. Though it is connected to the AUART chip transmitting pin 21, the receiver does not respond to it's own transmission because no address word is transmitted (pin 18 low). A two-position switch is connected to the receiving pin as a voter-control-unit enable switch. The ground position of the switch disables transmission and the other position enables transmission as well as turning on the voter-control-unit indicator light. Pin 19 (CB2) of the computer I/O port is connected to these two pins to provide a communication link. Noise on the communication bus is filtered by the 0.01 micro-farad capacitor.

The AUART chip ground is pin 20 which is also connected to system ground. Pin 22 to 29 are the command pins which are used as functions

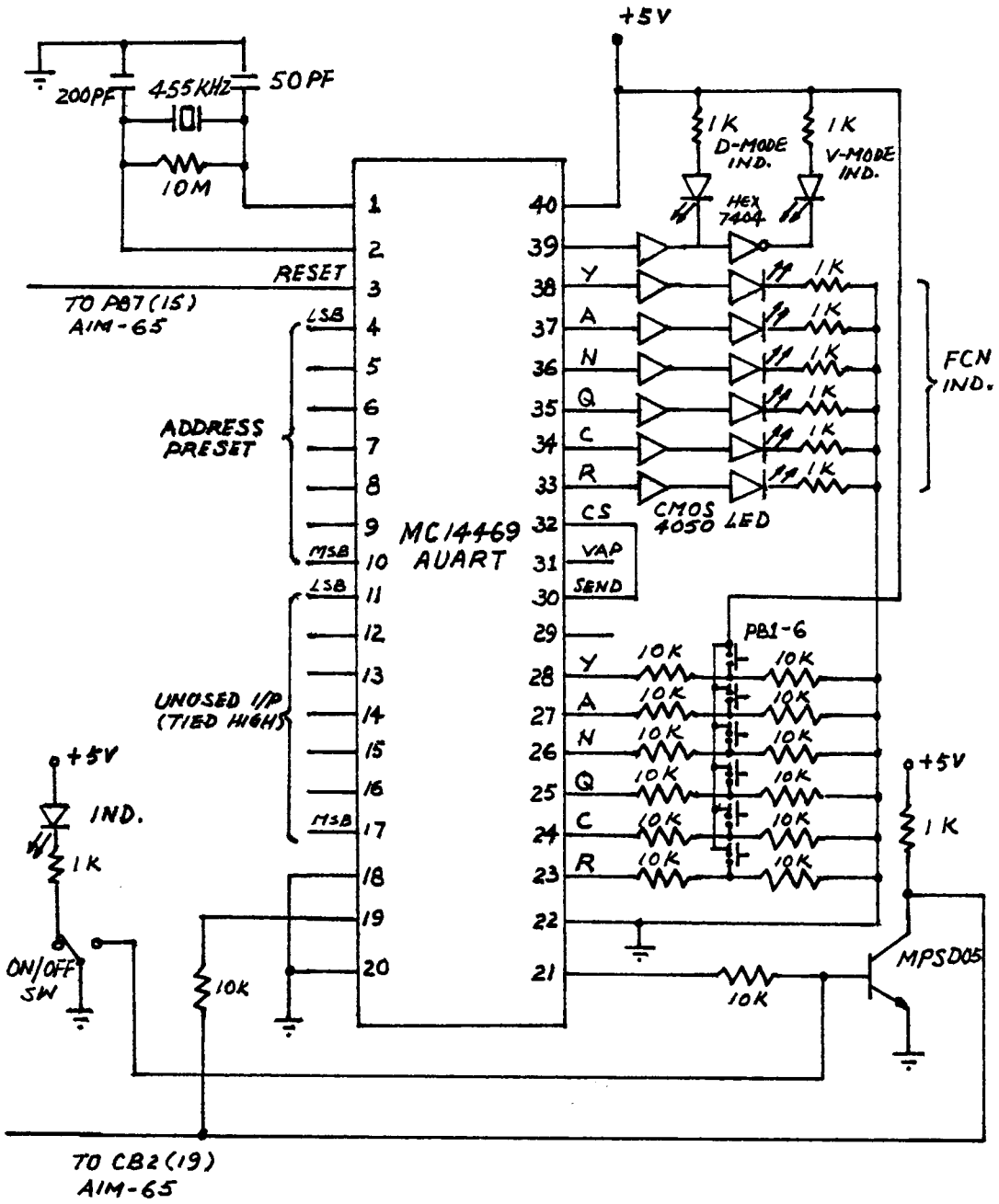


Figure 4.3 Circuit Diagram of Voter Control Box

selector by the voter. Pin 18 and pin 22 are tied low and high respectively in this particular system operation. Pin 31 is connected to pin 30 to initiate transmitter operation. Pin 33 to pin 38 are buffered to drive the voter-selected function indicators. Light emitting diodes (LED) are used. A 7404 HEX inverter is used as the buffer device. Pin 40 is the UART chip supply input. Positive 5 volts is used from the power supply. One of the important features in the design is that the push-buttons are not debounced. This is accomplished because of the use of the polling technique. The requirement that the depressed button be held for 1 second makes the debouncing circuit unnecessary.

Display And Buzzer Unit

The purpose of the display panels is to display the presence state of the voters, the vote cast by the voters, and the question or request signal. The flashing of the warning lights during the voting period is also displayed.

Communication between the computer and the display panels is via the AUART chips too; however, they are being used as a receiver only. Two of these chips are used and they are called the data and command chips respectively. The data chip is responsible for receiving data to be shifted into the shift registers for display. The command chip is to receive the clock signal which moves the display data from one shift register to the other. The clock disable signal and the relay activation signal are also being sent to the command chip for proper display circuit operation.

A special feature of the display is that the data will not be displayed until all the 12 shift registers are filled. When the 24

voters' processed data is sent, a relay activation signal is sent by the computer to turn the display on. Incandescent lamps are used to enable visibility from a distance of 50 feet.

The configuration of the incandescent lamps on the display panel is shown in Figure 4.4.

The buzzer unit and the warning lights are also located on the display panels.

Display Panel Hardware Circuit Design. The block diagram of the display panel is shown in Figure 4.5. The design circuit diagram is shown in Figure 4.6. The circuit diagram consists of three parts. The first part of the circuit is to display the voting result and the question or request signal. The circuit components include data chip 1, command chip 1, 12 8-bit parallel shift registers, a relay circuit and the incandescent displays. The shift registers' output are buffered to ensure sufficient driving and sinking capability when the switching transistors are turned on and off to light the lamps. For simplicity, two voter's circuitry are shown. The remaining voters' circuitry are similar and they are implemented by 11 additional shift registers cascaded in series. Pin 33 of the command chip is buffered by a parallel buffer circuit to prevent overloading since it is used to provide the clock pulses for the 12 shift registers.

Data received from the computer will be latched on pins 33 to 38 of the data chip. Clock pulses are received by the command chip and are latched onto pin 33 to shift the received data. The computer sends the data for display in 12 consecutive rounds. For every round of data sent, a clock pulse is generated to shift the data from one shift

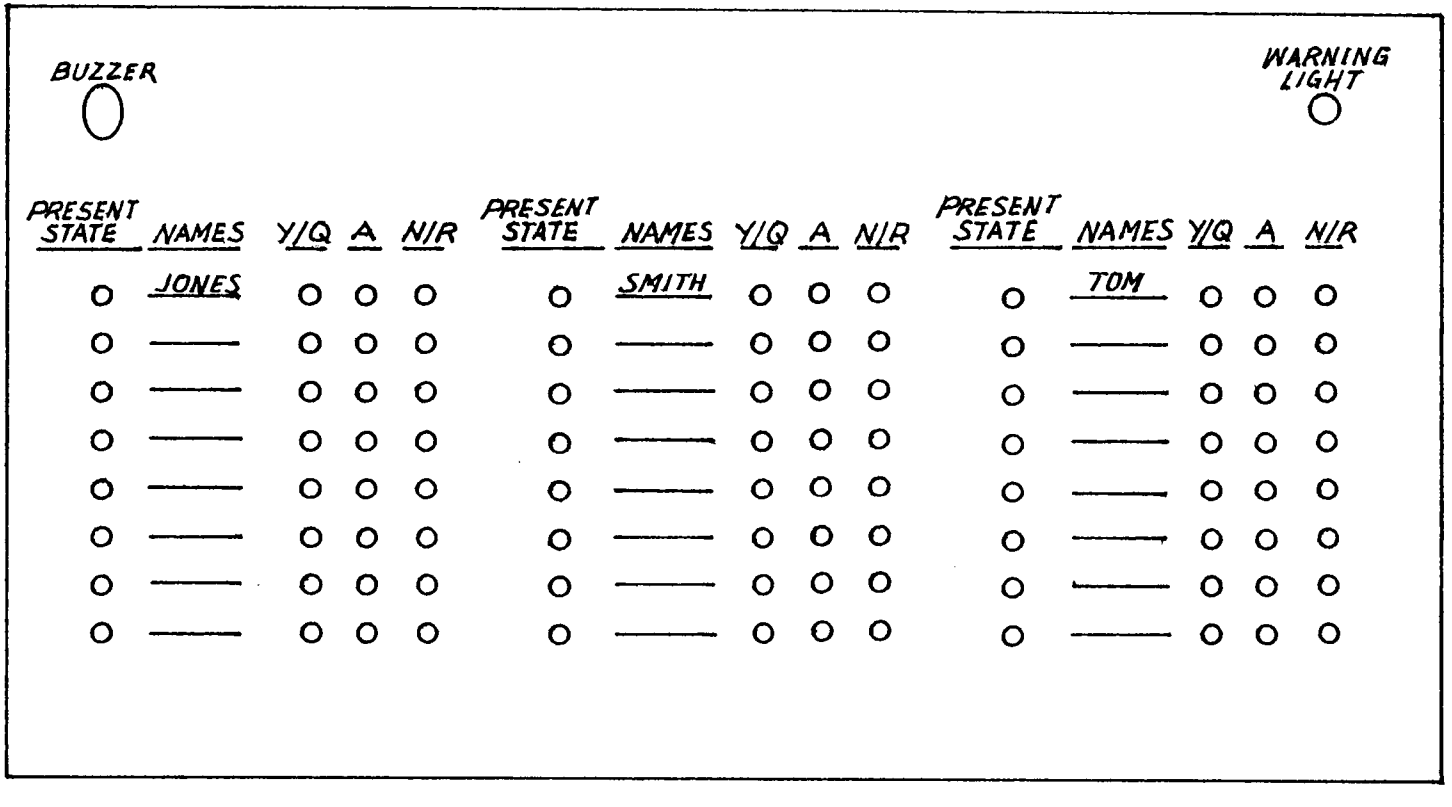


Figure 4.4 Schematic Diagram of The Display Panel

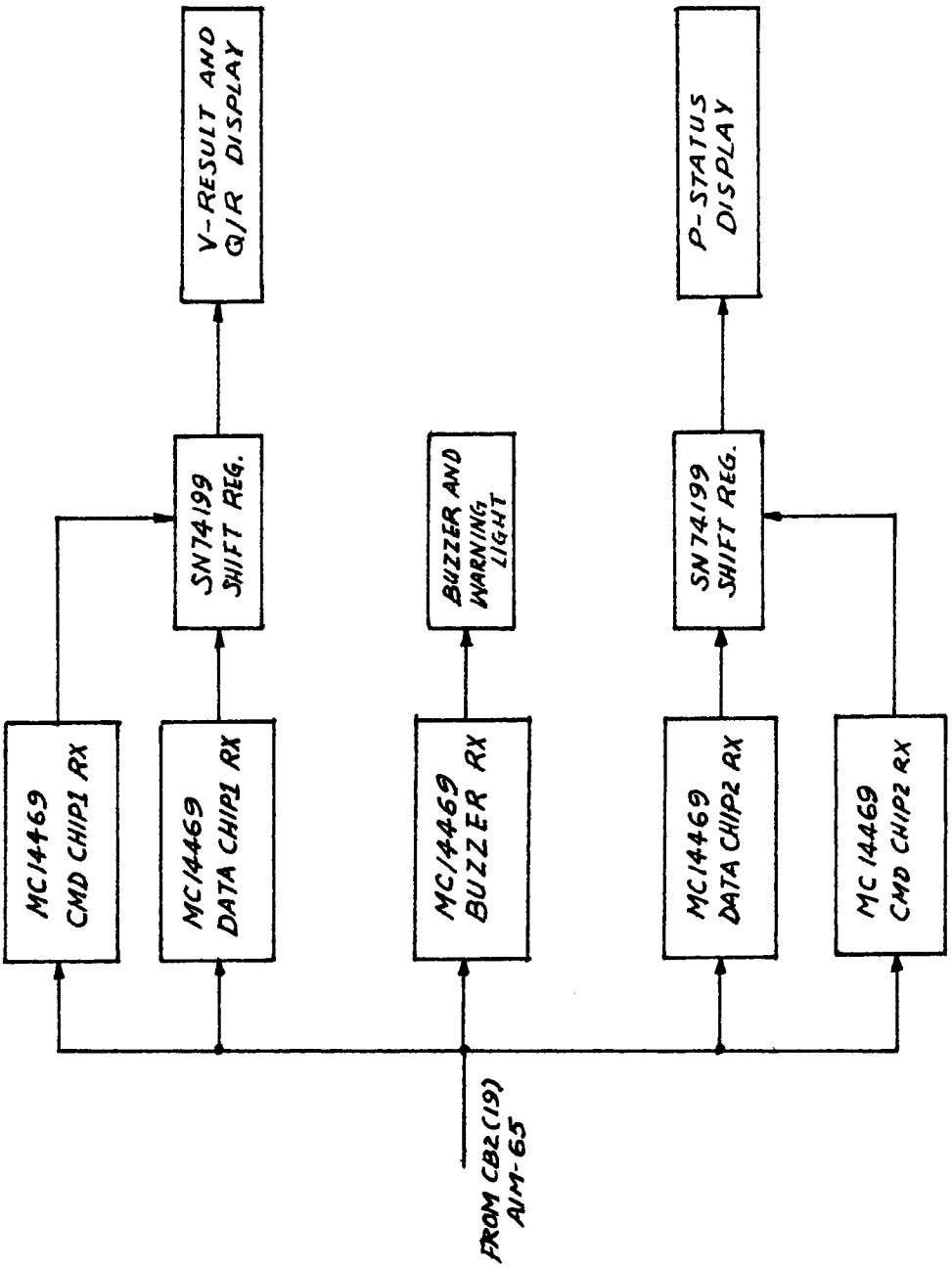


Figure 4.5 Block Diagram of The Display Panel

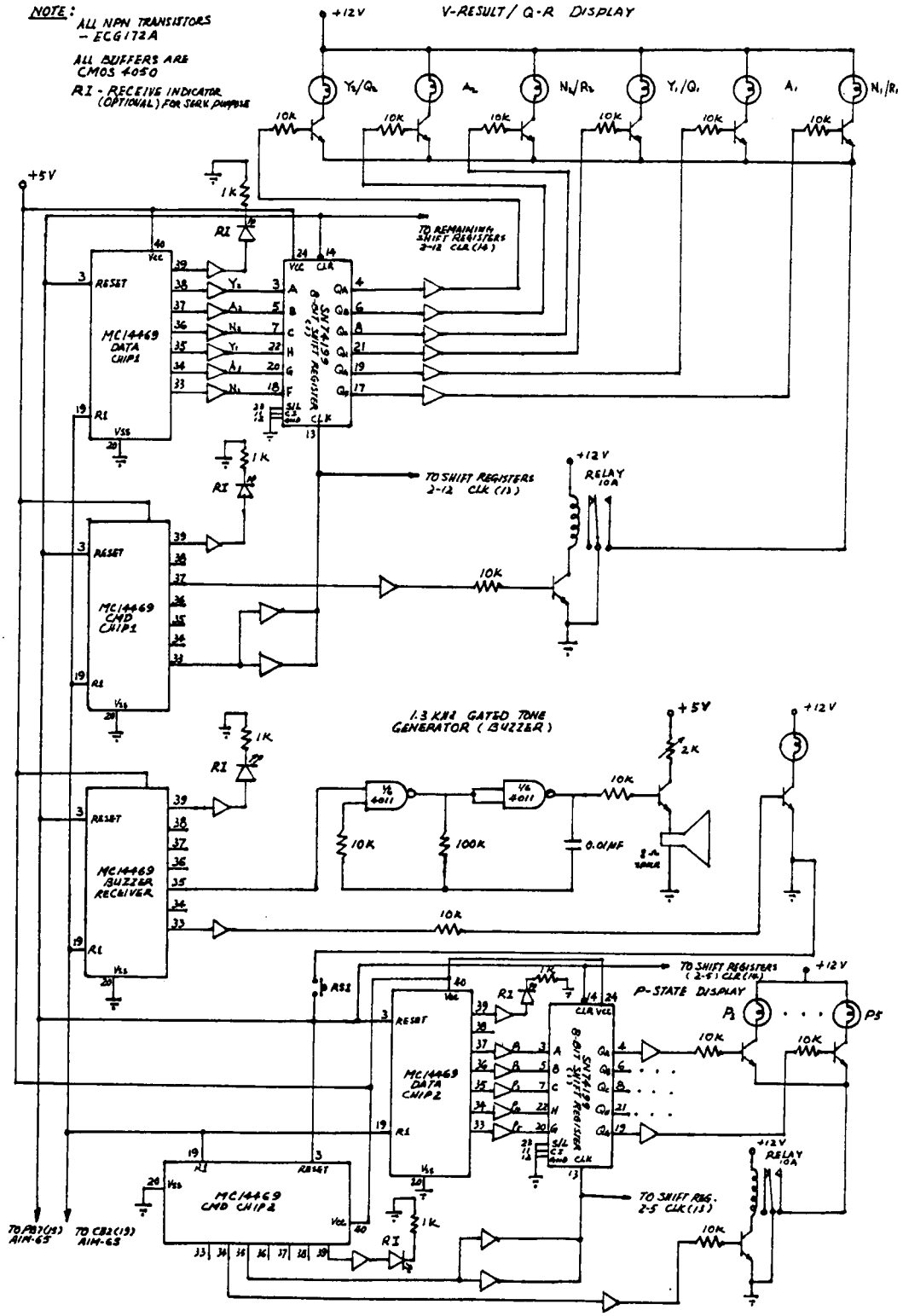


Figure 4.6 Circuit Diagram of The Display Panel

register to the next. The incandescent lamps are not turned on until all the shift registers are filled. As soon as all the data has been received, a relay activation pulse (logic '1') is generated on pin 37 of the command chip to turn on the relay which completes the emitters' path of the transistors to ground. Grounding the emitters' path causes only the display lights to turn on since the base voltages are controlled by the shift registers contents. Switching transistors with a gain of 7000 are used to provide the switching speed for the circuit.

The second part of the display circuitry consists of the buzzer (a tone generator) and warning lights. Pin 35 of the receiver chip is used to drive a gated tone generator which produces a 1.3 KHz tone on a 8 ohms speaker. The same type of switching transistor is used. The 1.3 KHz tone and the warning lights are activated when pin 33 and 35 of the receiver chip is high (logic '1') and de-activated when it is low (logic '0'). The duration of the "ON" period of the incandescent lamp is the same as the tone generator. The 1-second tone and flashing light is simulated by the computer software routine which drives pin 33 and 35 high and low sequentially for the desired interval.

The third part of the display circuitry is similar to the first since it is also used for display purposes but for a different function. It is used in the presence-state display. To display a total of 24 states, five 8-bit parallel shift registers are used. The circuit description and operation are similar to that described in the first part.

The display circuitry can be reset during initial system power-up by depressing the reset switch (RS1). Testing of all the incandescent

lamps is done by computer software routine. The power supply voltages used are 5 and 12 volts.

Printer Unit

The printer is a built-in unit in the microcomputer. It is turned on by depressing the "CTRL" and the "PRINT" key on the keyboard at the same time. The task of the printer is to print the voters' votes and prints a list of those voters who request for the floor.

Power Supply Unit

A power supply capable of supplying 3 different level of voltages is necessary to operate the system. The supply voltage and current for the AIM-65 microcomputer is regulated 5 volts and 2.0 amperes respectively. The supply voltage and current for the printer is unregulated 24 volts and 2.5 amperes peak respectively. The supply voltage and current for the display and buzzer is unregulated 12 volts and 4.5 amperes respectively. The supply voltage for the CMOS AUART chips and LED indicators is also 5 volts. However, the current consumption of these two components is in the order of 10 micro-amperes and 3 milli-amperes which increases the overall 5 volts supply current to 2.5 amperes. The above estimation is the minimum required supply voltage and current for satisfactory system operation.

CHAPTER V

SOFTWARE AND HARDWARE COSIDERATIONS

The development of system operation begins with the forming of the software routines for communication between the computer and the AUARTS. The computer is the main transmitter and receiver and the AUARTS are the auxillary transmitters and receivers. When communication link is established, subsequent routines are developed to perform data-bit tests and other functional tasks. The computer is programmed to address a AUART once and then followed by another address and data transmission to ensure reliable communication.

Software Considerations

The sequence of system operation is determined by the software routines configuration. The beginning of the system software routine is to allow the maintenance personnel to perform a functional test on all the incandescent lamps on the display panels. When the test is completed, the next routine allows voters' names to be entered into the computer memory. The computer will go into the "SEARCH" routine looking for either voting or discussion commands when the termination command (/) is received at the end of the entering process. If the discussion mode is selected, the computer will execute the "STATE-check" routine to check the presence of each voter. The computer will then address the voters and inform the voters its current mode of operation. On the second round of transmission, the computer will immediately go into reception and is ready to process the incoming data. The received data

is processed for validity. If the data is valid, further processing is carried out to update or retrieve data from the computer memory. When data processing is completed, the computer acknowledges the voter by another transmission. The computer steps through the sequence of polling from the first voter station to the twenty-forth voter station. Once the polling of the last voter station is completed, the computer will check for a voting command from the chairperson. If a voting command is given, the computer will proceed to the voting mode; otherwise, it will continue the poll sequence all over again.

The tasks of the voting-mode routine is to process the voting results, display the results on the display panels and print the results on the printer. The voting results stored in the computer memory are updated once in every second until the allocated 10 seconds of voting time expires. The computer will sound the buzzer and flash the warning lights during each round of the poll sequence. The "DISPLAY" routine is executed to show the result of each voter's vote on the main display panels. When the display is completed, the computer proceeds to execute the "PRINT" routine to print the voters' vote on hard-copy paper. When execution is completed, the computer will again execute the "SEARCH" routine waiting for the command to go into the voting or discussion mode.

Hardware Considerations

The design motive is to use as little hardware components as possible to reduce cost and wiring problems. The reduction in the hardware components will reduce the voltage and current supply of the

system. The low-power components used are CMOS AUART chips, light-emitting diodes (LED), 7404 HEX buffers and shift registers (SN74199). The mechanical devices used are pushbuttons and relays.

Incandescent lamps are used on the display panels to provide the necessary brightness. CMOS AUART is used as the communication device because of its low power consumption and its addressable feature.

Programming Of The 2532 PROM

The 2532 PROM is used as the firm-ware storage device for the final version of the software routine. The starting address of the PROM is B000 (HEX) and it is plugged into the spare socket (Z-26) of the AIM-65 microcomputer. Execution of the system program has to start from this address.

The burning of the system software program into the PROM is done by the Department of Electrical Engineering's TEXAS INSTRUMENT (TI) computer. The UT Computing Center DEC-10 computer is used to edit and cross-assembled the system software program. To burn the program into the PROM, the assembled program in the DEC computer has to be transferred into the TEXAS INSTRUMENT computer. The transferring process is accomplished through the use of another TEXAS INSTRUMENT microcomputer (TM-990). The task of the TI microcomputer is to receive the transferred data from a modem and store it in an allocated area of the main TI computer. The received data which is in ASCII format is converted to binary format by the microcomputer before storing in the main TI computer. Detection of errors is performed by the main TI computer.

When the transfer between computers is completed, the operator may double check for error by matching the old and transferred file. The "Error-Free" file is then loaded into a specific memory location of the computer to be down-loaded into the PROM. The programming of the PROM takes about 3 minutes to complete.

COMPUTER I/O Port Interface

The interface between the computer, the voter stations, and display panels is shown in Figure 5.1. Interface of the microcomputer to the AUART is simplified as a result of using only four wires. The R6522 versatile interface adapter (VIA) is programmed to be the input/output port to provide system communication.

Pin 15 of the J1 connector is programmed to reset the system (ACTIVE LOW). Pin 19 is programmed to be bidirectional for transmission and reception. Pin 1 is the system ground.

COMPUTER Sourcing Capability

The circuit developed for testing the driving capabilty of the microcomputer is shown in Figure 5.2(a). It is basically a resistive and capacitive network. They are actually the load and the input capaci- tance of the AUART chips. Each AUART chip has an input capacitance of 5 pico-farads.

The test is carried out by programming the microcomputer to shift a continuous series of rectangular pulses on pin 19 of the output port to the test circuit. The output is then observed on the oscilloscope for distortion. The over-loading condition of the microcomputer is

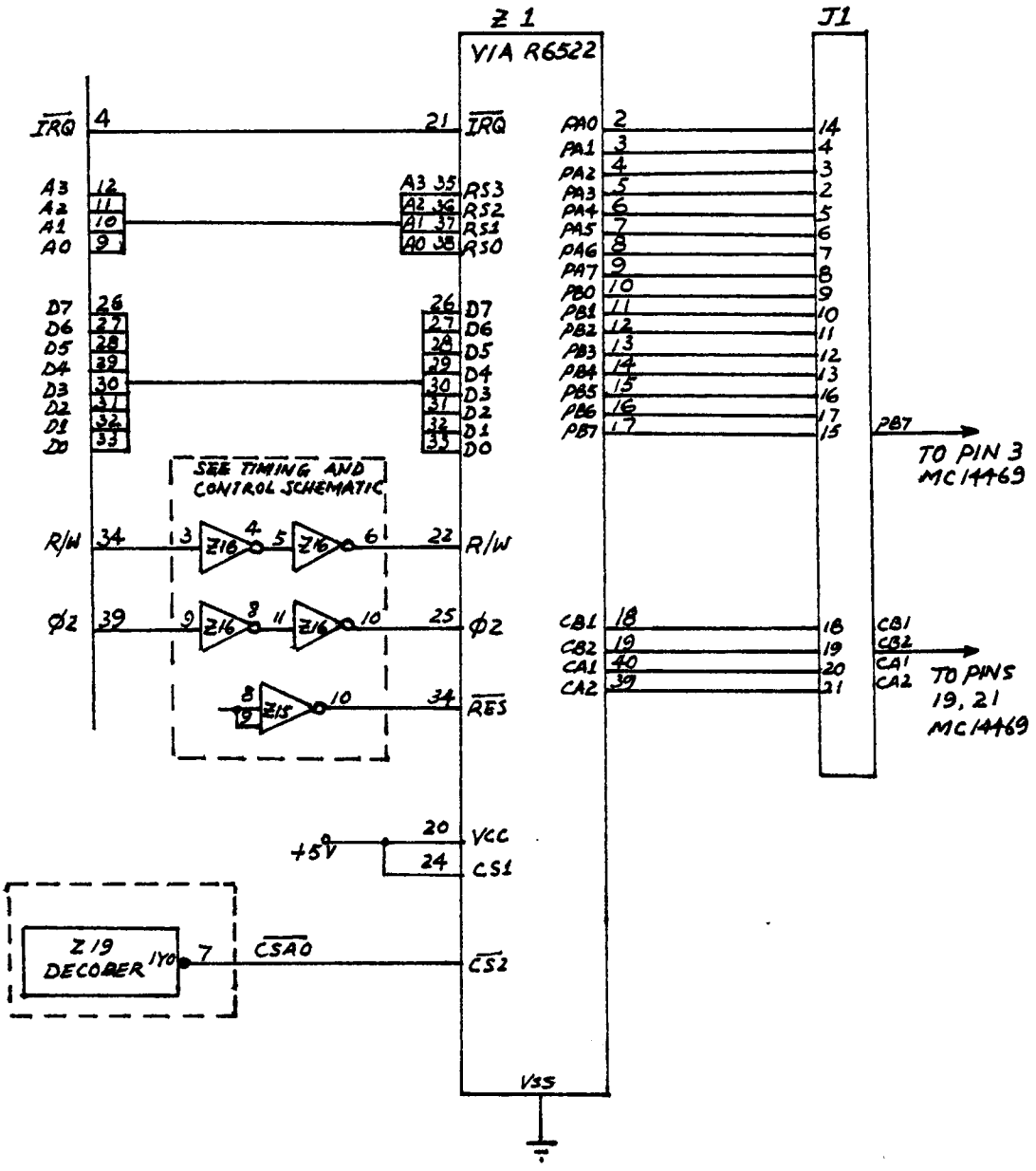


Figure 5.1 Schematic Diagram of The Computer I/O Port Interface

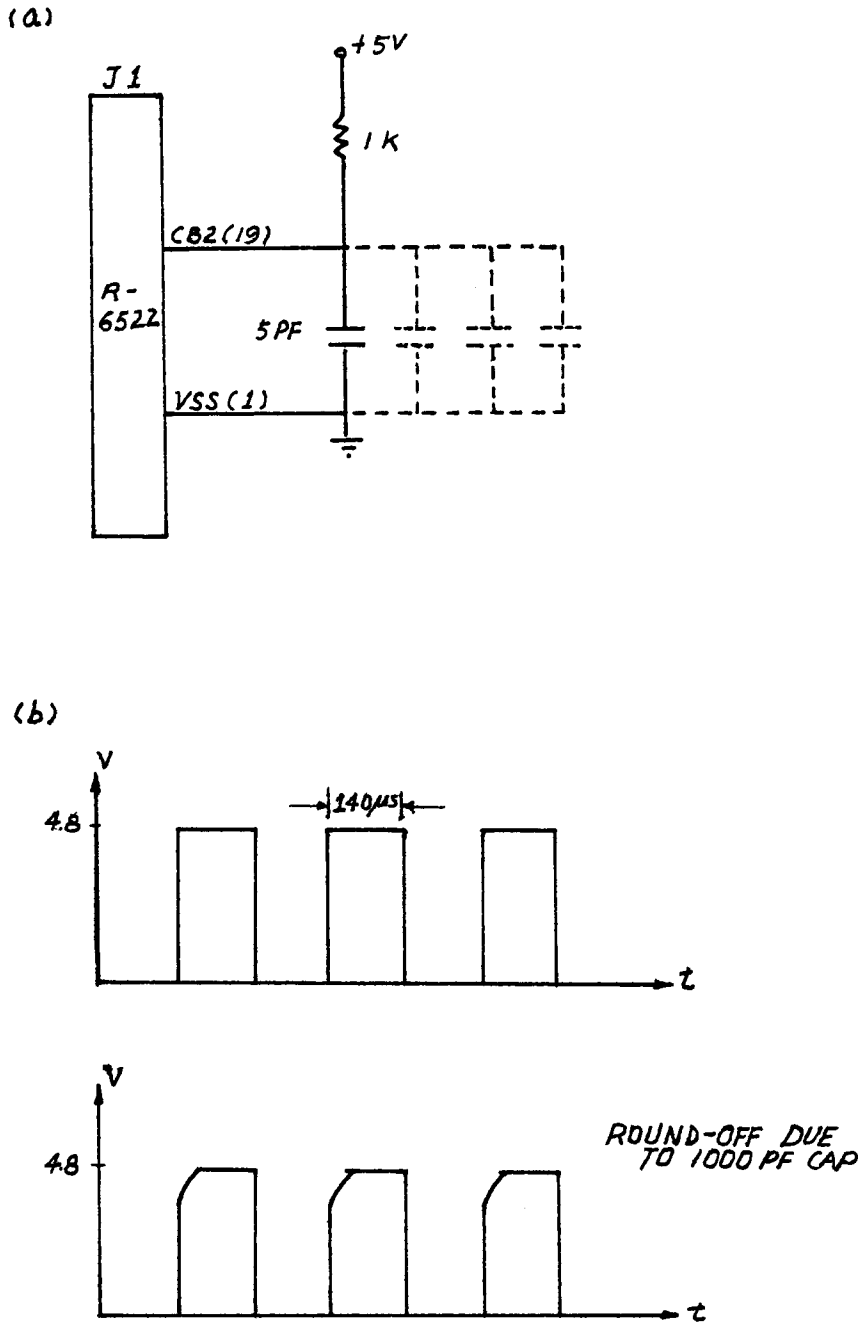


Figure 5.2 Test Circuit and The Observed Waveform

simulated by subsequently increasing the capacitance value and observing the distortion in the output waveform. The sum of all the voter stations input capacitances is about 140 pico-farads. Capacitance value ranging from 5 pico-farads to 1000 pico- farads is applied during the test. Observations show that the output waveforms do not change until a capacitance value of 1000 pico-farad is reached. The effect of this over-loading condition, however, causes only a minor round-off at the rising edge top-corner of the output waveform. This effect does not affect the system operation. The waveform is shown in Figure 5.2(b).

The conclusion of the test performed show that the computer is capable of driving additional voter stations without being overloaded. Therefore, the system can be expanded by adding more voter stations to the system bus.

CHAPTER VI

SYSTEM POWER-UP INITIALIZATION AND OPERATING PROCEDURE

Power is applied to the voting system when the power supply on/off switch is switched to the "ON" position. The computer will initialize itself to the monitor mode by displaying " ∠ " when both the 5 volts and 24 volts are available. The computer will display "PRINTER DOWN" if only the 5 volts is available.

During cold start (i.e., initial power-up), the states of the display and voter stations circuitry are not defined. They are either in the "ON" (logic '1') or the "OFF" (logic '0') state. The "ON" state may cause the display panels incandescent lamps to be turned on if the relay contact is closed. The undefined states can be initialized to "0" state by depressing the manual system reset pushbutton on the computer. To reset the computer, a manual computer reset switch is used. It is necessary to perform these two tasks after the power to the system has been turned on. Illumination test is performed to check all the incandescent lamps by microcomputer software routine. The maintenance personnel are responsible for carrying out the task mentioned above. The system functional test is then carried out at the computer and the voter stations. The steps to be followed are:

Enter the command to execute the system program by typing "* B000" and depress the "RETURN" key. The computer responds by displaying "*=B000".

Type "G" and depress the "RETURN" key. The computer responds by

displaying "ITST OR NAME". In this mode the computer is waiting for the operator to select either illumination test or enter voters' names mode.

Type "T" to perform illumination test on all the incandescent lamps on the display panels. Six lamps are tested for illumination every second. They are turned on for 1 second and then turned off before the next six lamps are illuminated. The process is continued until all the lamps are tested. When the test is completed, the computer will display "ITST or NAME" to allow the operator to select again.

Type "N" to select names-entering mode. The computer responds by displaying "ENTER NAME". Type "E" to enter the voter's names into the computer memory. The computer will echo the names being entered on the display. Any error made before the third initial is entered can be corrected by using the "DEL" key. If error is encountered after the third initial is typed, the whole list of names has to be re-entered again. In this situation, depress the "ESC" key to enter the monitor mode again.

On completion of the list of names, type "/" to terminate the names-entering mode. The computer responds and displays "V OR D Mode" for mode selection. At this point, the printer is turned on by depressing the "CTRL" and "PRINT" keys at the same time.

The discussion mode is tested by depressing the "D" key. When this key is depressed, the "PRESENT" incandescent lamps will be turned on. It is essential to turn on all the voter control boxes for this test. A check should then be carried out to see that all voter-stations' discussion indicator lights are turned on. The three

push-buttons assigned to this mode should be checked for malfunctioning. On the computer display, "D-MODE¦V?" is displayed to indicate its current mode of operation and prompting for selection of voting mode. The computer can be initiated to go into the voting mode while it is in the discussion mode by depressing the "Y" key. When this key is depressed, the computer displays "V-MODE" to indicate that it is in the voting mode. It should be noted that the discussion mode LED indicators are turned on during initial power-up.

The voting mode can also be tested by depressing the "V" key when it is prompting the operator to select either "V OR D Mode". When the "V" key is depressed, check that all voter stations voting indicator lights are lighted. Check the sounding of the buzzer and the flashing of warning lights. When the buzzer stops, the display circuitry will be activated followed by the operation of the printer at the computer. When the printer stops printing, The computer will dipsplay "V or D MODE " again. At this stage, the system functional test is complete. The computer is in the ready state for the chairperson to start the meeting.

At the end of the meeting, depress the manual computer reset button and the manual system reset button to clear the state of the system before turning the power switch off. In the case of system breakdown due to power failure, the computer display will black-out. If system failure is due to communication breakdown, the computer will display "BIT ERROR" to indicate that validity test performed by the computer has failed and the system is "DOWN." Under this situation, the meeting will have to be continued by the old method of hand raising and using the microphone allocated to each of the voters at their station.

CHAPTER VII

SUMMARY AND CONCLUSION

The development of the revised system is based upon the background information of the original electronic voting system. Even though the procedure of operations are different, they serve the same purpose to the users.

The major problem encountered in this project is the synchronization of the AUART and the computer during reception by the computer. The continous shifting-in of three 8-bit bytes of data stream from the AUART causes the computer to run out-of synchronization. This problem is solved when synchronization is established on the detection of each start-bit in the 24-bit data stream. Four bytes of memory are allocated to store the received data.

A unique feature found on the AUART is that it will respond and operate only when it is being addressed. Operation is thus configured to address the AUART by sending only its address bits and then followed by sending its address and information bits. There are two precautions taken in the disign. The first being that a 5 milli-seconds interval is imposed between each transmission. This interval of time is necessary to achieved simplex communication. The second being that a 500 milli-seconds delay imposed on the computer's transmission. This interval of time is necessary to prevent concurrent operations of the printer and the computer which might cause transmission error.

The revised system is functionally tested for its reliability and durability by long hours of operation. Infinite loop test is carried

out on the voting and discussion mode respectively for malfunction. All the voters' stations are tested for correct operations and all push-buttons are tested for malfunction. The overall system operation is tested by inserting the PROM in socket Z26 and it performs satisfactorily.

Even though the author ended his design effort at this point, there are still some improvements which could be made. One is the addition of a protection circuit to save the memory content of the microcomputer during voting mode when the power supply to the system is interrupted. The other is to interface an external TV monitor to the computer to increase computer display capability.

LIST of REFERENCES

LIST OF REFERENCES

Sarah Corbin Robert, "Robert's Rules Of Order Newly Revised," Scott, Foresman and Company, 1970.

Collier's Encyclopedia, Macmillan Educational Company, New York, P. F. Collier, INC., London and New York, Vol. 18, pp. 454-457, 1982.

Encyclopedia Britannica, Encyclopedia Britannica INC., William Benton Publisher, Vol. 19, p. 385, 1768.

AIM-65 Microcomputer User's Guide, Rockwell International Corporation, 1979.

AIM-65 Microcomputer Programming Manual, Rockwell International Corporation, 1979.

AIM-65 Microcomputer Monitor Program Listing Manual, Rockwell International Corporation, 1979.

AIM-65 Microcomputer Hardware Manual, Rockwell International Corporation, 1978.

Lance A Leventhal, "6502 Assembly Language Programming," Aosborne/McGraw Hill, Berkeley California, 1979.

Motorola CMOS Data Manual, Motorola INC., 1978.

The TTL Data Book For Design Engineer, Second Edition, Texas Instrument INC., 1976.

Hert Taub, Donald Schilling, "Digital Integrated Electronics," McGraw Hill, INC., pp. 262-277, 1977.

R. C. Camp, T. A. Smay, C. J. Triska, "Microcomputer System Engineering," Matrix Publishers INC., Portland Oregon, 1979.

G. Jack Lipovski, "Microcomputer Interfacing Principles and Practices," DC Heath And Company, Lexington, Massachusetts, Toronto, pp. 384-387, 1980.

APPENDIXES

APPENDIX A

SUPPLEMENTARY INFORMATION ON MC14469

ADDRESSABLE UNIVERSAL ASYNCHRONOUS TRANSMITTER RECEIVER

The MC14469 addressable asynchronous receiver transmitter is constructed with MOS P-channel and N-channel enhancement devices in a single monolithic structure (CMOS). the MC14469 receives one or two eleven-bit words in a serial data stream. The first incoming word contains the address and when the address matches, the MC14469 is enable to transmit two data words. Each of the transmitted word containes eight data bits, even parity bit, start and stop bit, in UART compatible format.

The address word contains seven address bits and the address of the MC14469 is set on seven pins. Thus 128 units can be interconnected in simplex or or full duplex data transmission. In addition to the address received, seven command bits may optionally be received for the data and control used.

The special features of the MC14469 is that it has a wide supply voltage range from 4.5 volts dc to 18 volts dc and at 5 volts dc, it has a low quiescent current of 75 micro-amps dc maximum. The data rates of the MC14469 can be as high as 4800 baud. Data is received by the MC14469 receiver serially and is converted to parallel form, and data is converted from parallel to serial form for transmission. The operation of the MC14469 can either be crystal or resonator controlled.

In this particular voting system application, the MC14469 is being configured to operate in simplex transmission and is ceramic resonator controlled.

Device Operation

OSCILLATOR (Osc1, Osc2; pin 1, 2) -- These pins are the oscillator input and output.

RESET (Reset, pin 3) -- When this pin is pulled low, the circuit is reset and ready for operation.

ADDRESS (A0-A6; pin 4, 5, 6, 7, 8, 9, 10) -- These are the address setting pins which contain the address for the received signal.

INPUT DATA (Id0-Id7; pins 11, 12, 13, 14, 15, 16, 17, 18) -- These pins contain the input data for the first eight bits of data to be transmitted.

RECEIVE INPUT (R1; pin 19) -- This is the receive input pin.

NEGATIVE POWER SUPPLY (Vss; pin 20) -- This pin is the negative power supply connection. Normally this pin is system ground.

TRANSMIT REGISTER OUTPUT SIGNAL (TRO inverted; pin 21) -- This pin transmits the outgoing signal. Note that it is inverted from the incoming signal. It must go through one stage of inversion if it is to drive another MC14469.

SECOND or STATUS INPUT DATA (S0-S7; pins 22, 23, 24, 25, 25, 26, 27, 28, 29) -- These pins contained the input data for the second eight bits of data to be transmitted.

SEND (Send; pin 30) -- This pin accepts the send command after receipt of an address.

VALID ADDRESS PULSE (VAP; pin 31) -- This is the output for the valid address pulse upon receipt of a matched incoming address.

COMMAND STROBE (CS; pin 32) -- This is the output for the command strobe signifying a valid set of command data on pins 33-39.

COMMAND WORD (C0-C6; pins 33, 34, 35, 36, 37, 38, 39) -- These pins are the read out of the command word which is the second word of the receive signal.

POSITIVE POWER SUPPLY (VDD; pin 40) -- this pin is the package positive power-supply pin.

Operating Characteristic

The receipt of a start bit on the receive input (R1) line causes the receive clock to start at a frequency equal to that of the oscillator devided by 64 (in the voting system application, the receiver frequency is 7.11 kHz). All received data is strobed in at the center of a receive clock period. The start bit is followed by eight data bits. Seven of the bits are compared against states of the address of the particular circuit (A0-A6), while the eighth bits signifies an address word '1', or a command word '0'. Next a parity bit is received and checked by the internal logic for even parity. Finally a stop bit is received. At the completion of the cycle if the address compared, a valid address pulse (VAP) occurs. Immediately following the address word, a command word is received. It also contains a start bit, eight data bits, even parity bit, and a stop bit. The eight data bits are composed of a seven-bit command, and a '0' which indicates a command word. At the end of the command word a command strobe pulse (CS) occurs.

A negative transition on the send input initiates the transmit sequence. Again the transmitted data is made up of two eleven-bit words. The data portion of the first word is made up from input data inputs (ID0-ID7), and the data for the second word from second input

data (S0-S7) inputs. This data is latched before the start of transmit of the first of the two words. The transmitted signal is the inversion of the standard UART TRO, which allow the use of an inverting amplifier to drive the lines.

The oscillator can be crystal controlled or ceramic resonator controlled for required accuracy. Pin 1 may be driven from an external clock source. (Ceramic resonator of 455 kHz is used in the voting system application.)

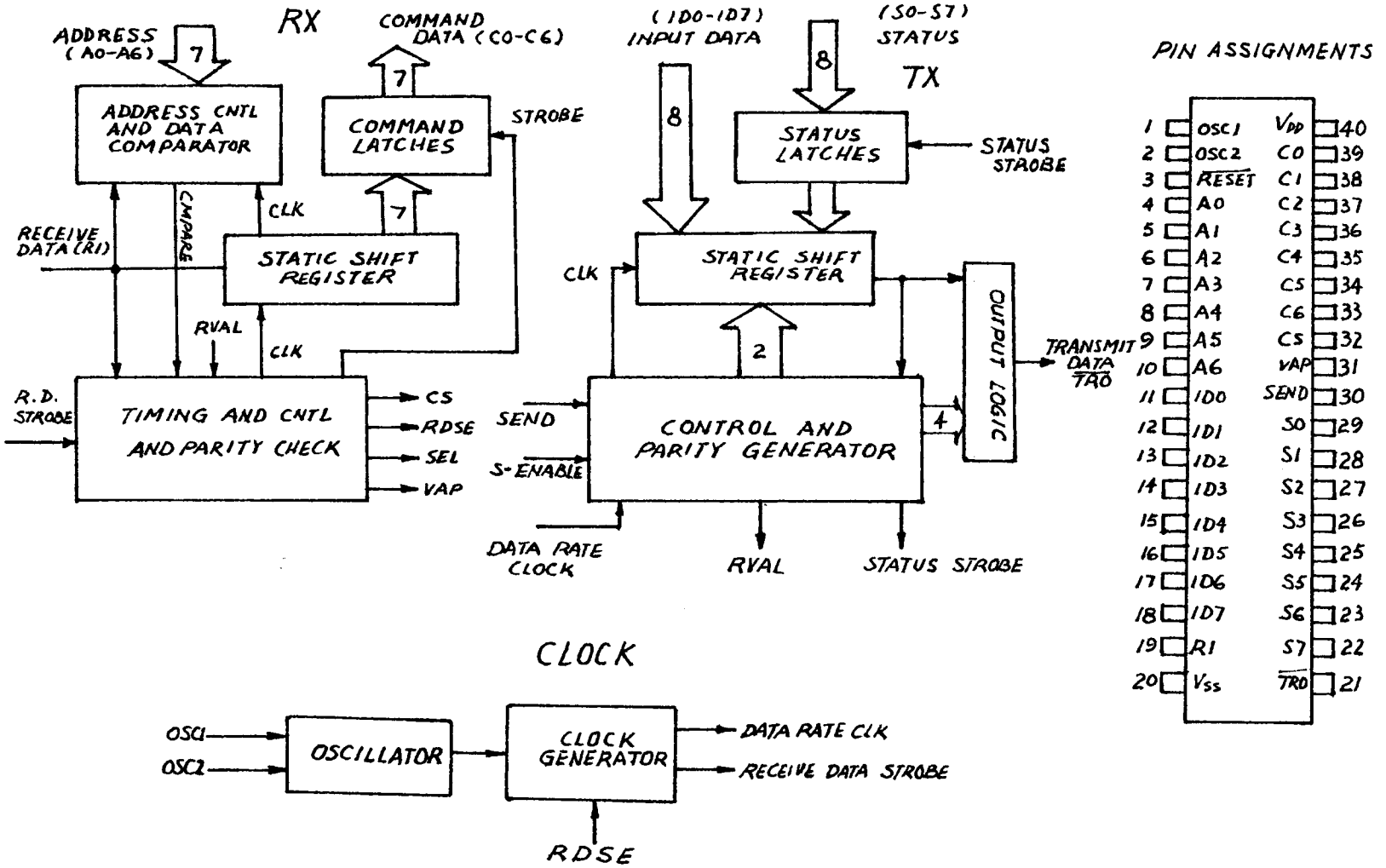
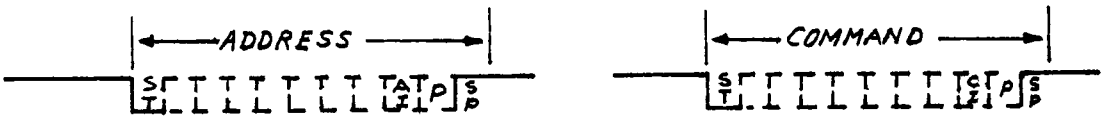


Figure A1. Block Diagram of MC14469 (AUART) and Pin-Out Assignment

CORRESPONDING DATA POSITION AND PINS FOR MC14469

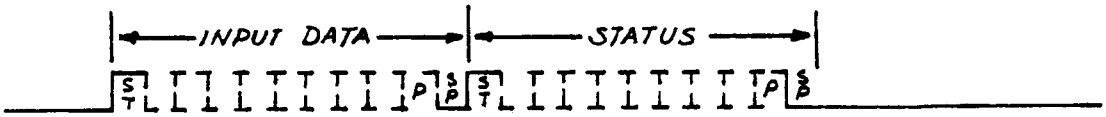
RECEIVE DATA R1(19)



MC14469 PIN NO. 4 5 6 7 8 9 10 — 39 38 37 36 35 34 33

PIN DESTINATION A_0 A_1 A_2 A_3 A_4 A_5 A_6 — C_0 C_1 C_2 C_3 C_4 C_5 C_6

TRANSMIT DATA $\overline{TRO}$ (21)



MC 14469 PIN NO. 11 12 13 14 15 16 17 18 — 29 28 27 26 25 24 23 22

PIN DESTINATION ID_0 ID_1 ID_2 ID_3 ID_4 ID_5 ID_6 ID_7 — S_0 S_1 S_2 S_3 S_4 S_5 S_6 S_7

ST = START BIT
P = PARITY BIT
SP = STOP BIT

A_0–A_6 = ADDRESS BITS
C_0–C_6 = COMMAND BITS
AI = ADDRESS IDENTIFIER

ID_0–ID_7 = IDENTIFICATION CODE
S_0–S_7 = STATUS CODE
CI = COMMAND IDENTIFIER

AIM-65 TRANSMIT DATA CB2 (P19)

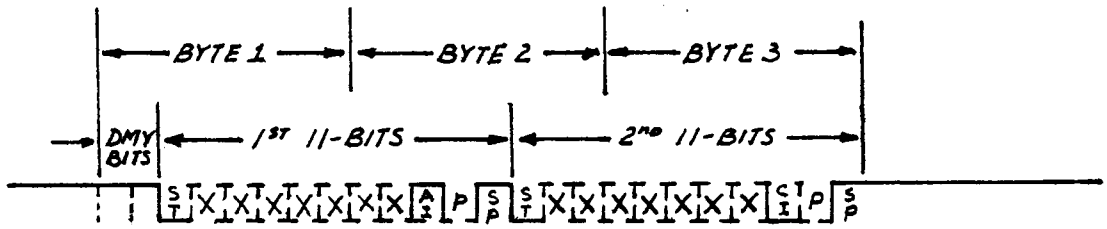


Figure A2. Transmit and Receive Signal of The AUART and Computer

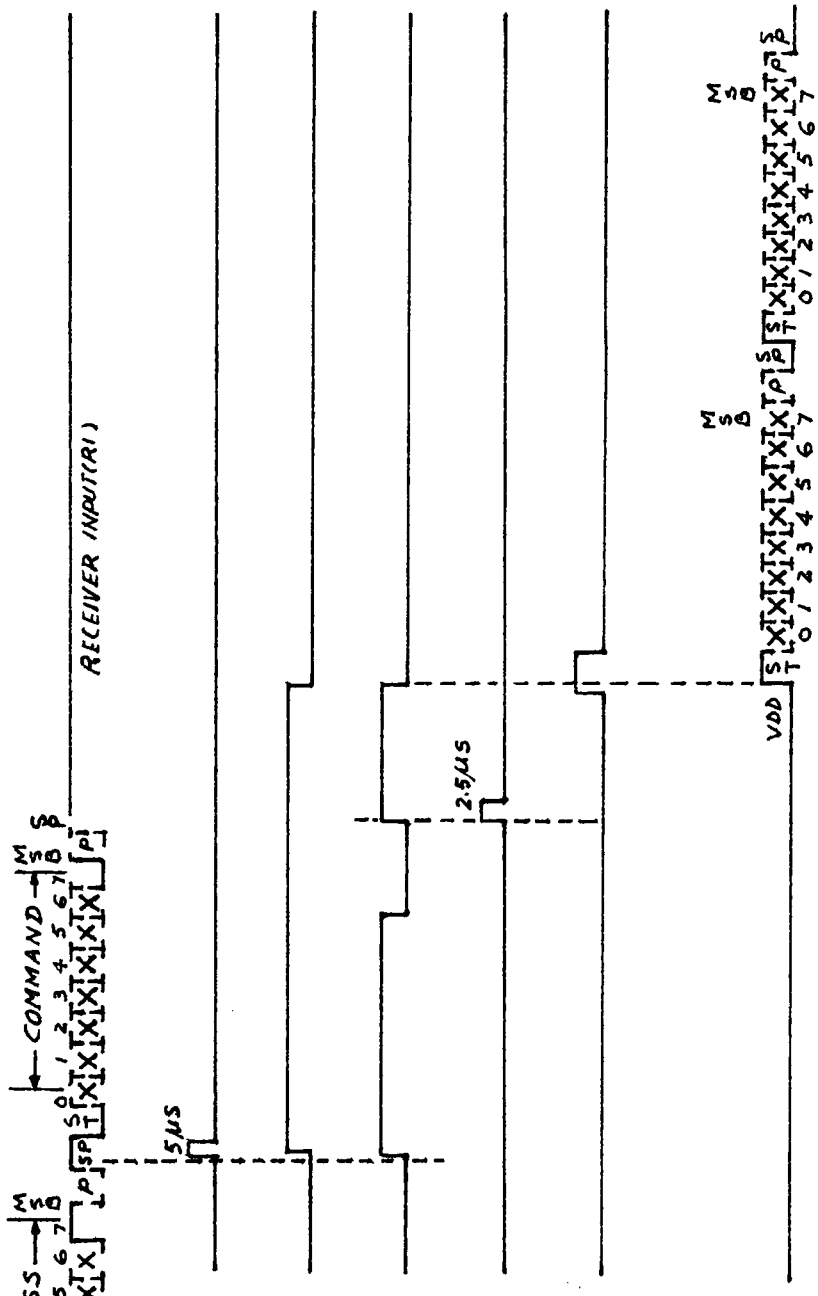


Figure A3. Schematic Diagram of a Typical Receive and Send Cycle

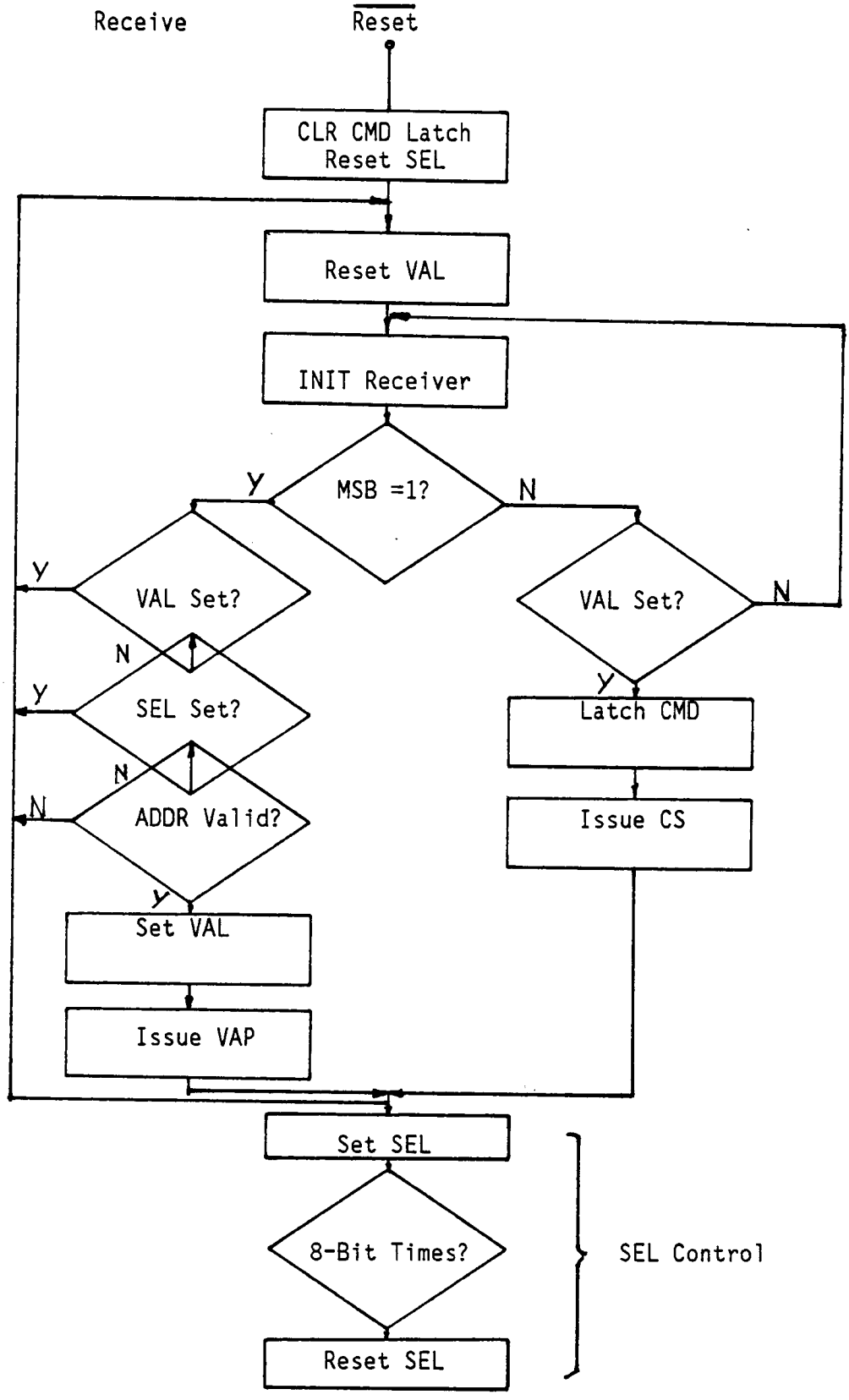
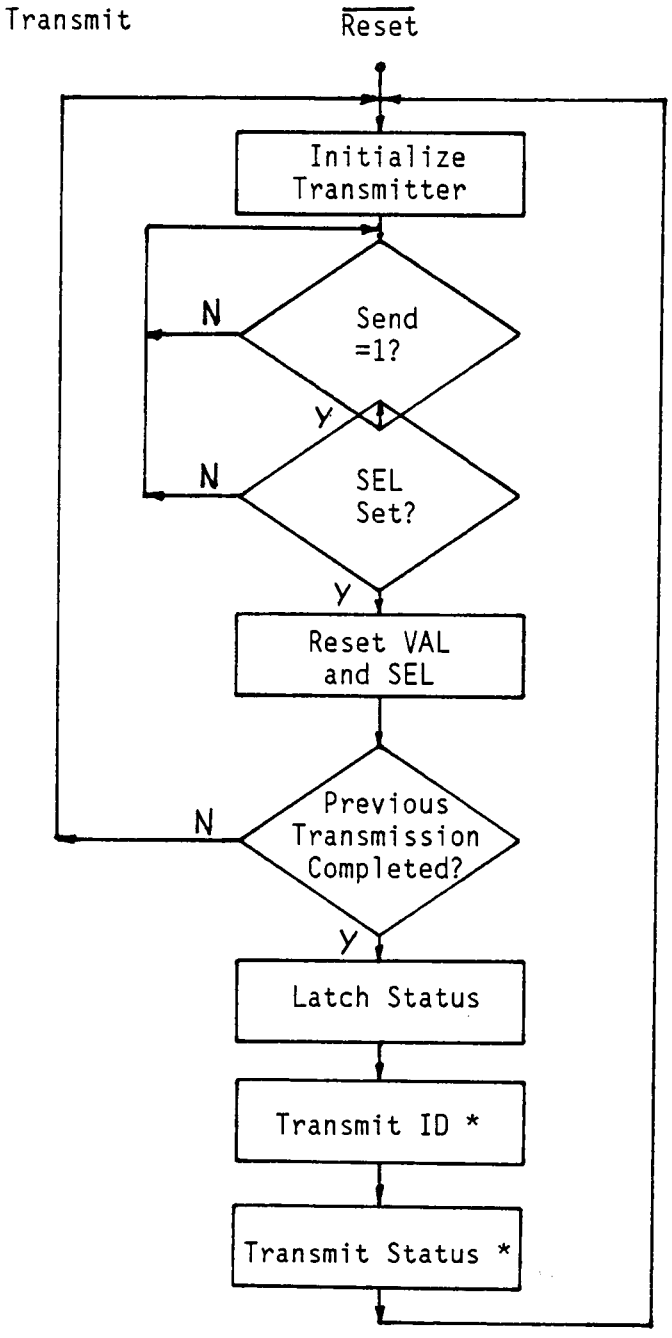


Figure A4. Flow Chart of MC14469 Operation



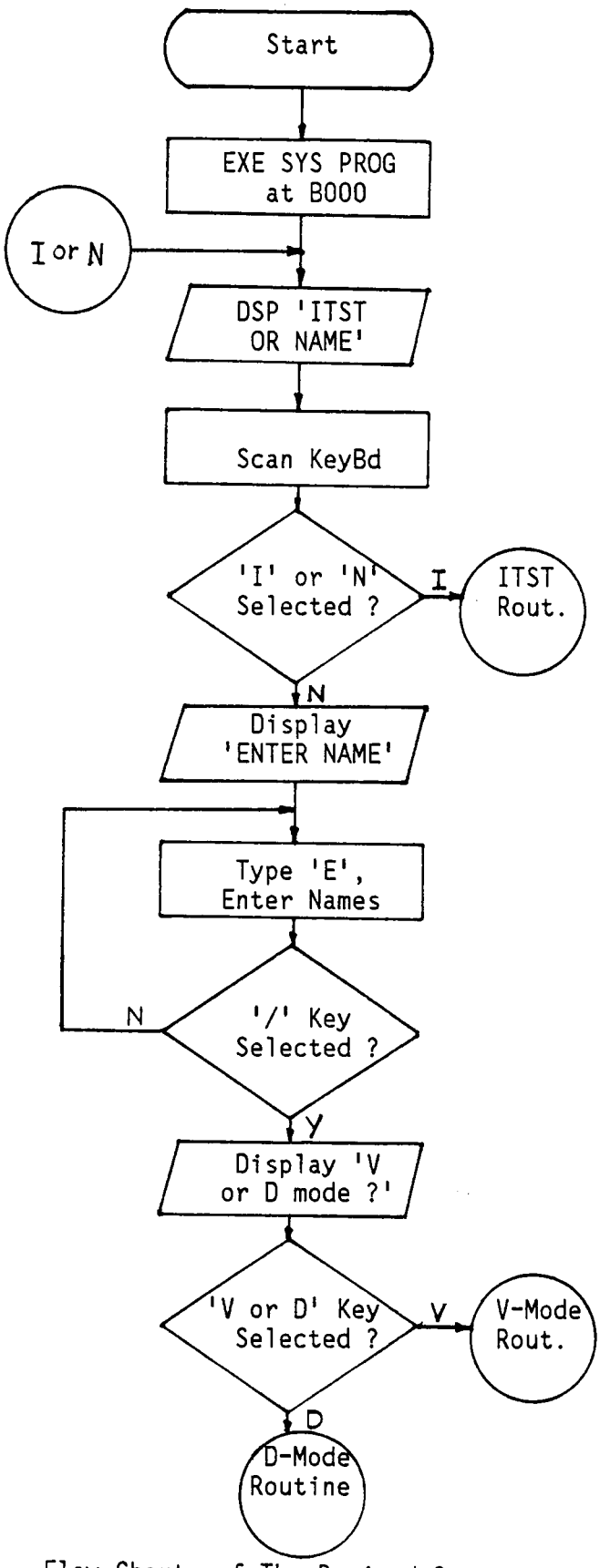
* Data format for both transmit and receive consists of :

1 Start Bit
8 Data Bits
1 Even Parity Bit
1 Stop Bit

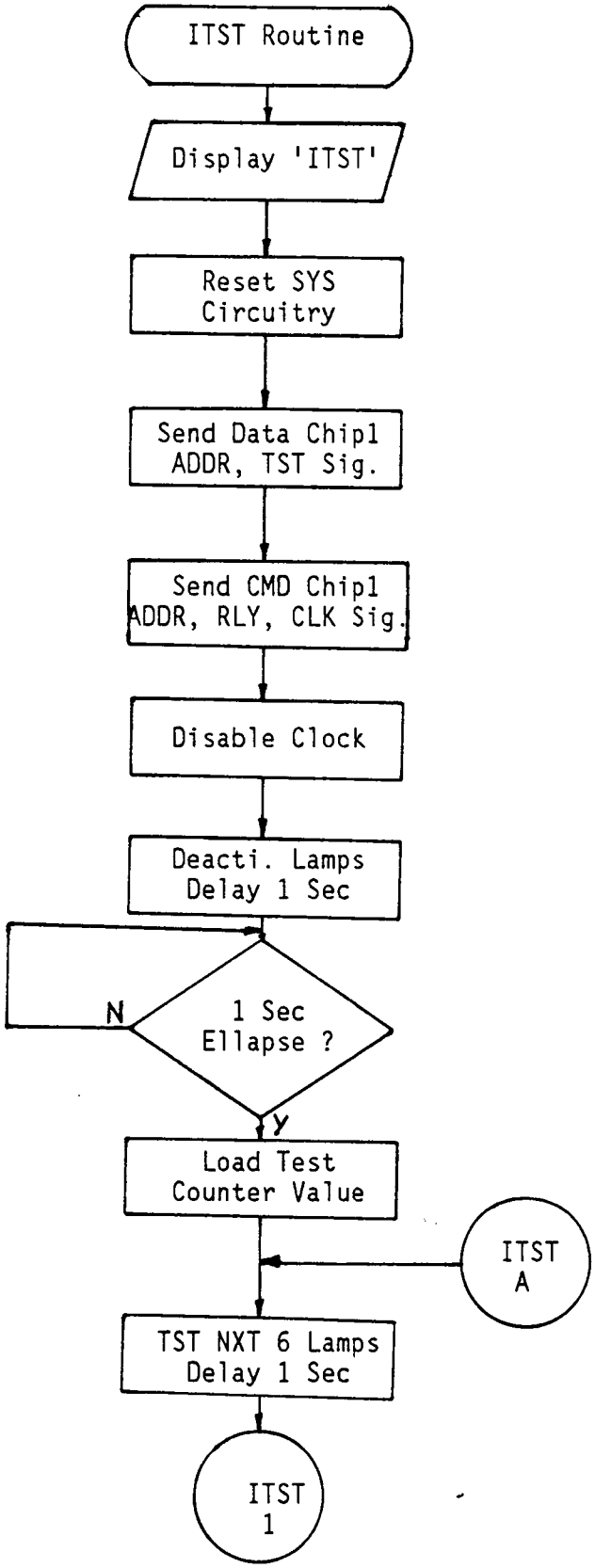
Figure A4. (Continued)

APPENDIX B

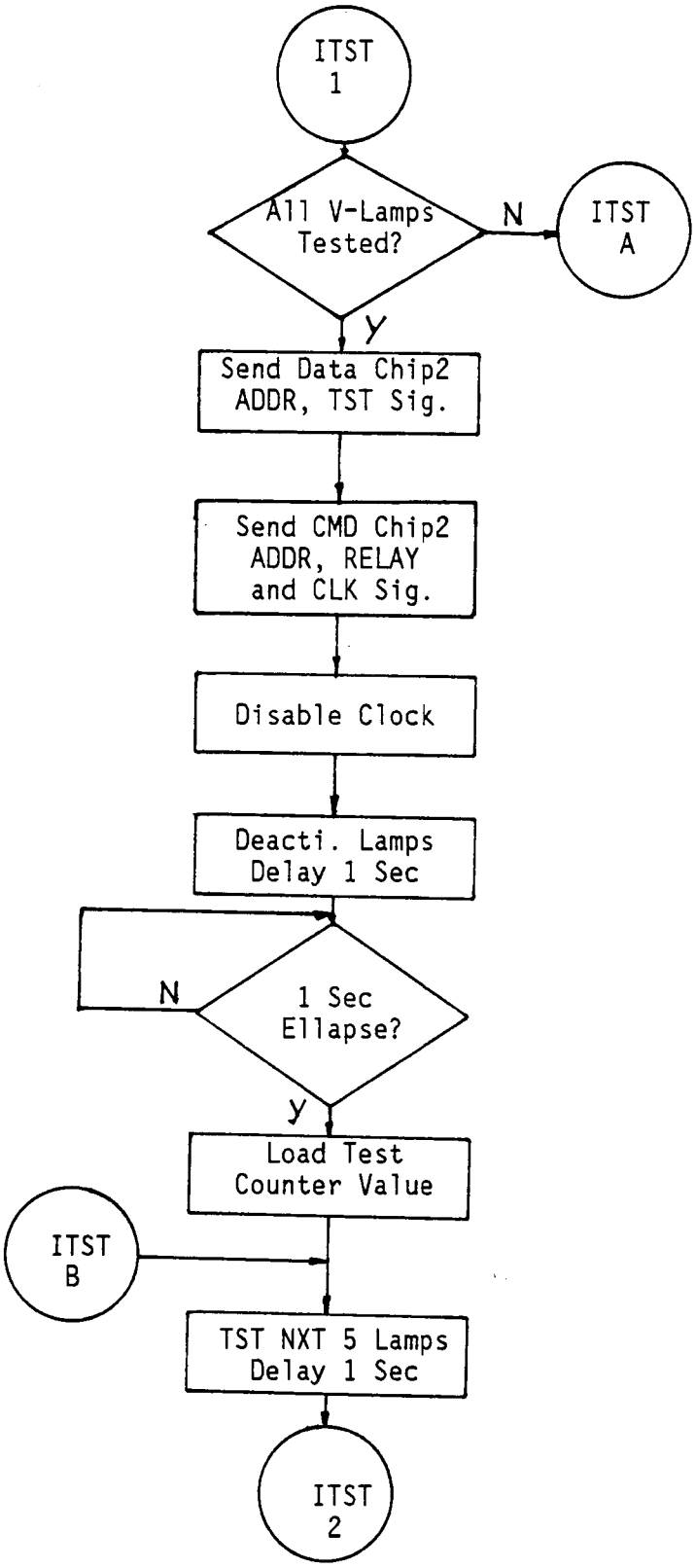
SYSTEM FLOW CHARTS



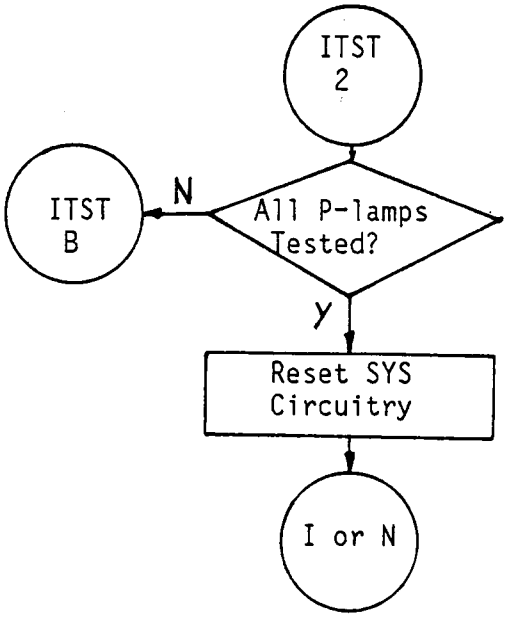
Flow Charts of The Revised System



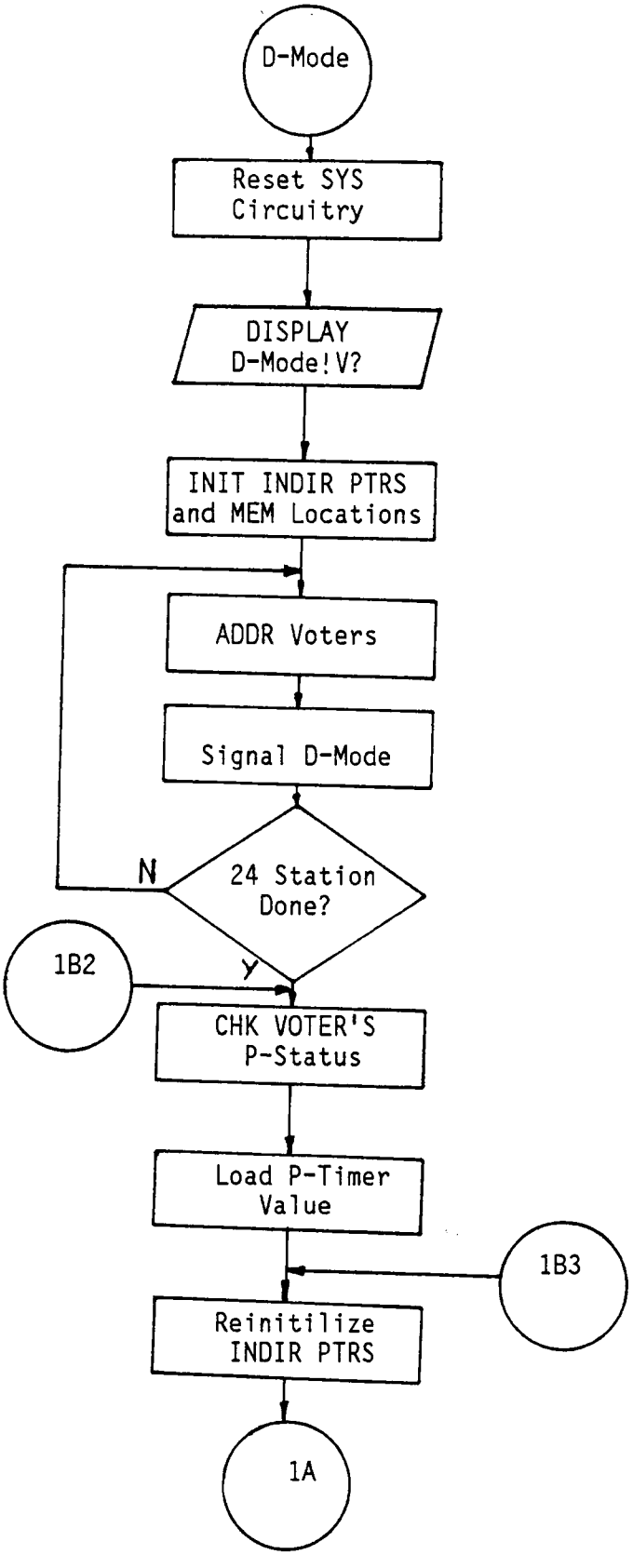
Lamps Illumination Test Routine



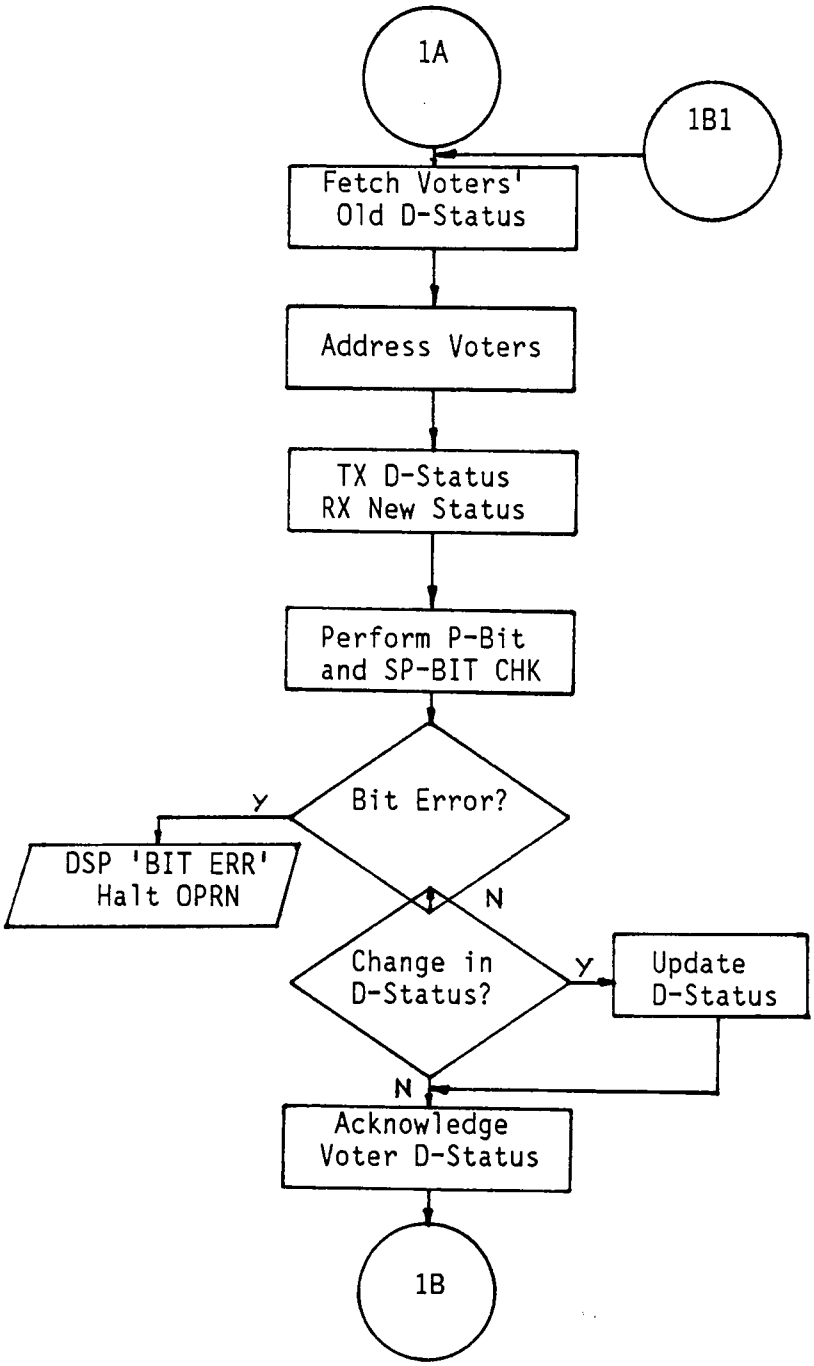
Lamps Illumination Test Routine (Continued 1)



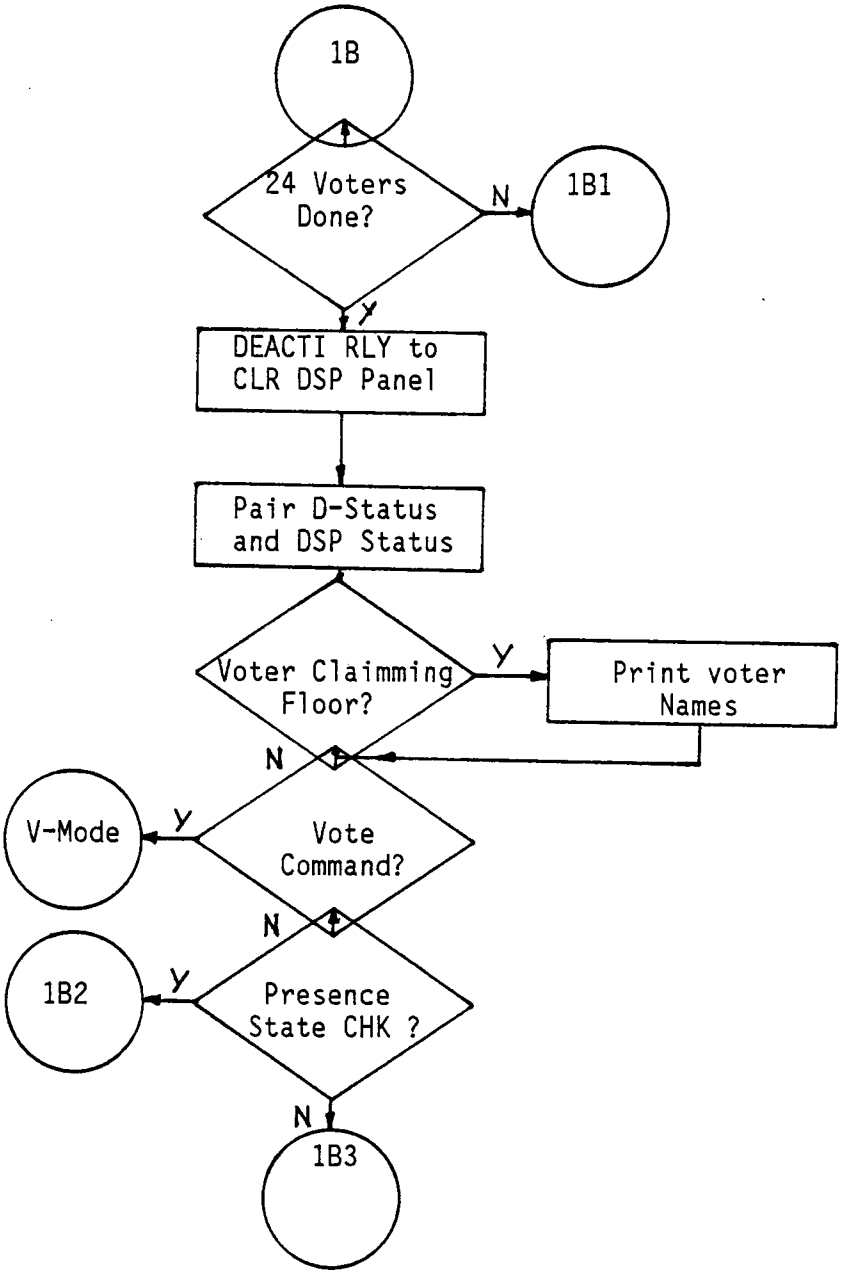
Lamps Illumination Test Routine (Continued 2)



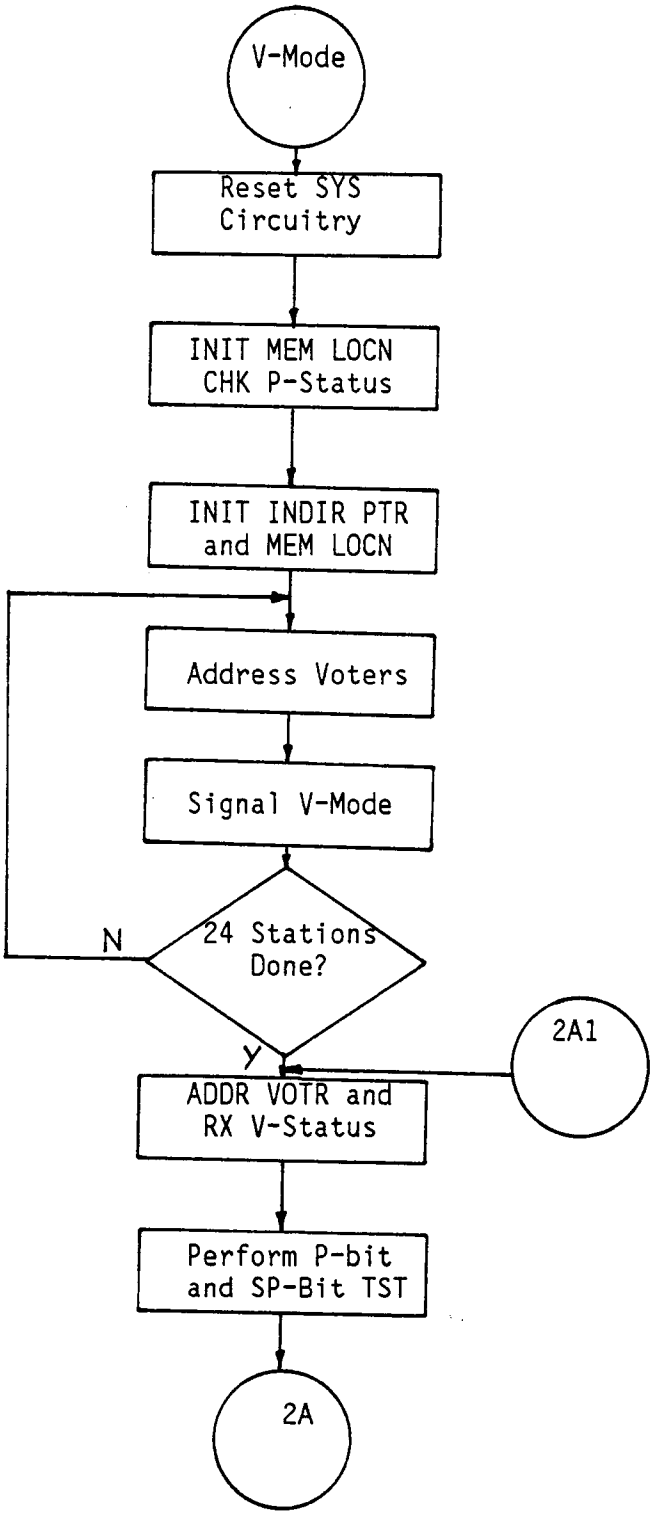
Discussion Mode



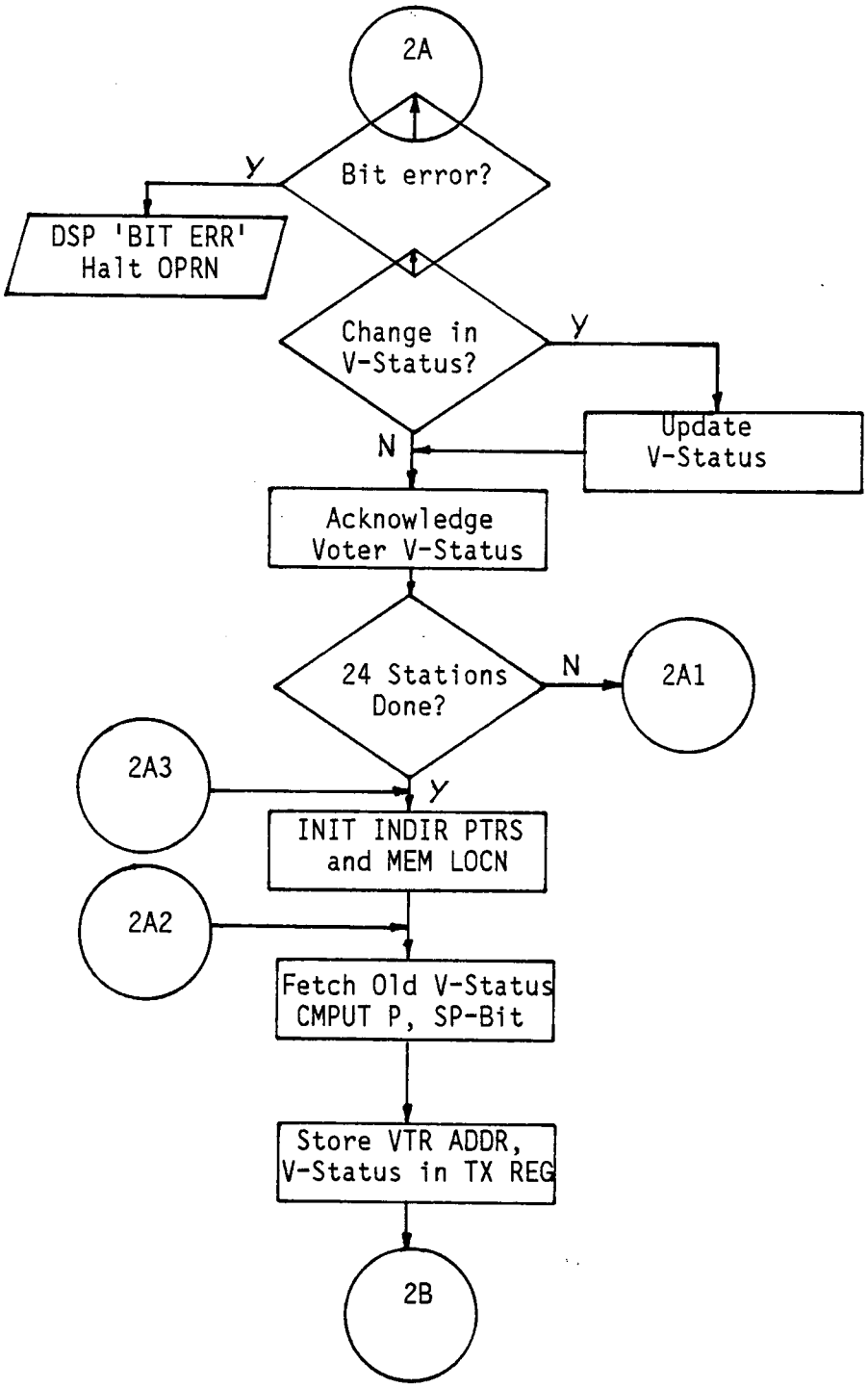
Discussion Mode (Continued 1)



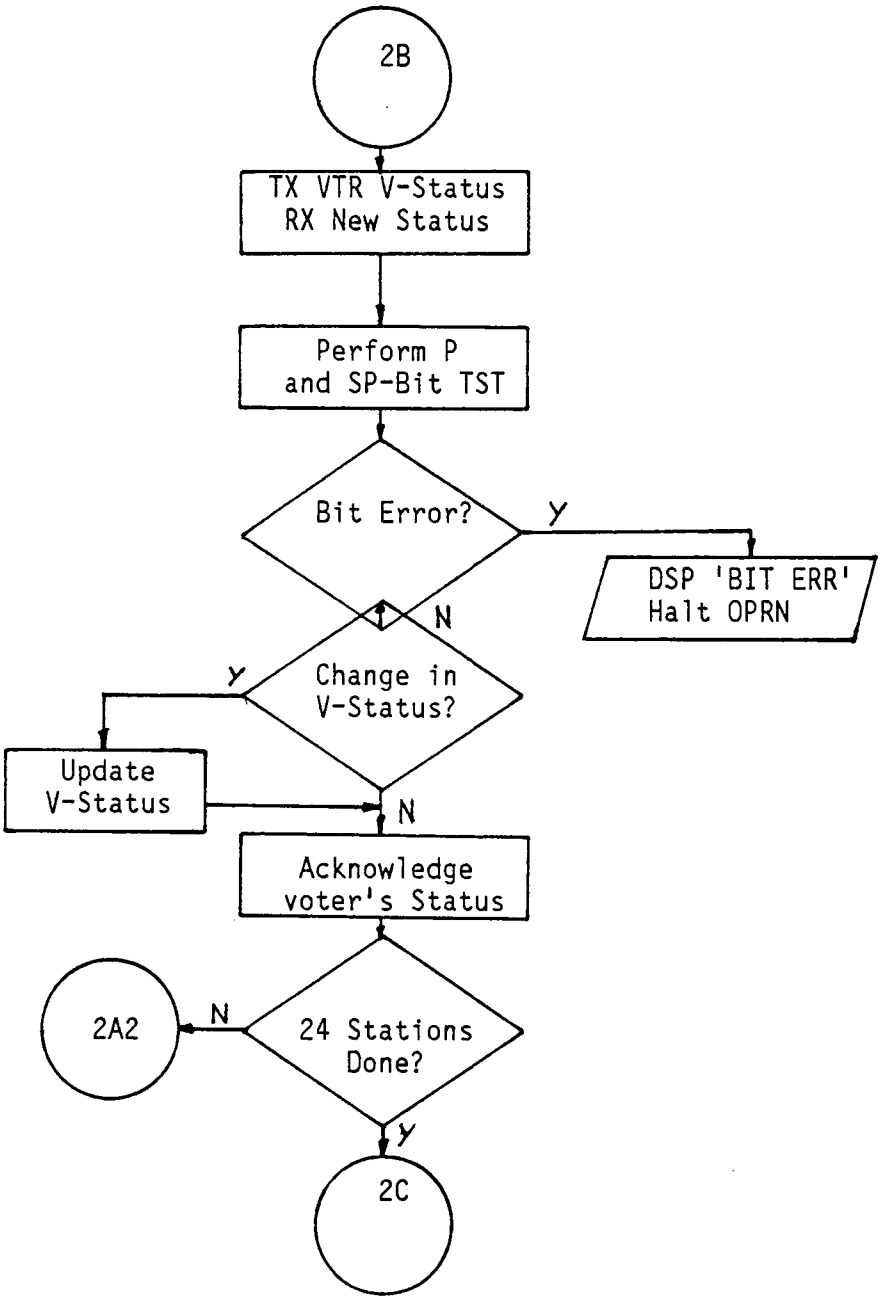
Discussion Mode (Continued 2)


V-Mode
Reset SYS Circuitry
INIT MEM LOCN CHK P-Status
INIT INDIR PTR and MEM LOCN
Address Voters
Signal V-Mode
24 Stations Done?
N
Y
2A1
ADDR VOTR and RX V-Status
Perform P-bit and SP-Bit TST
2A

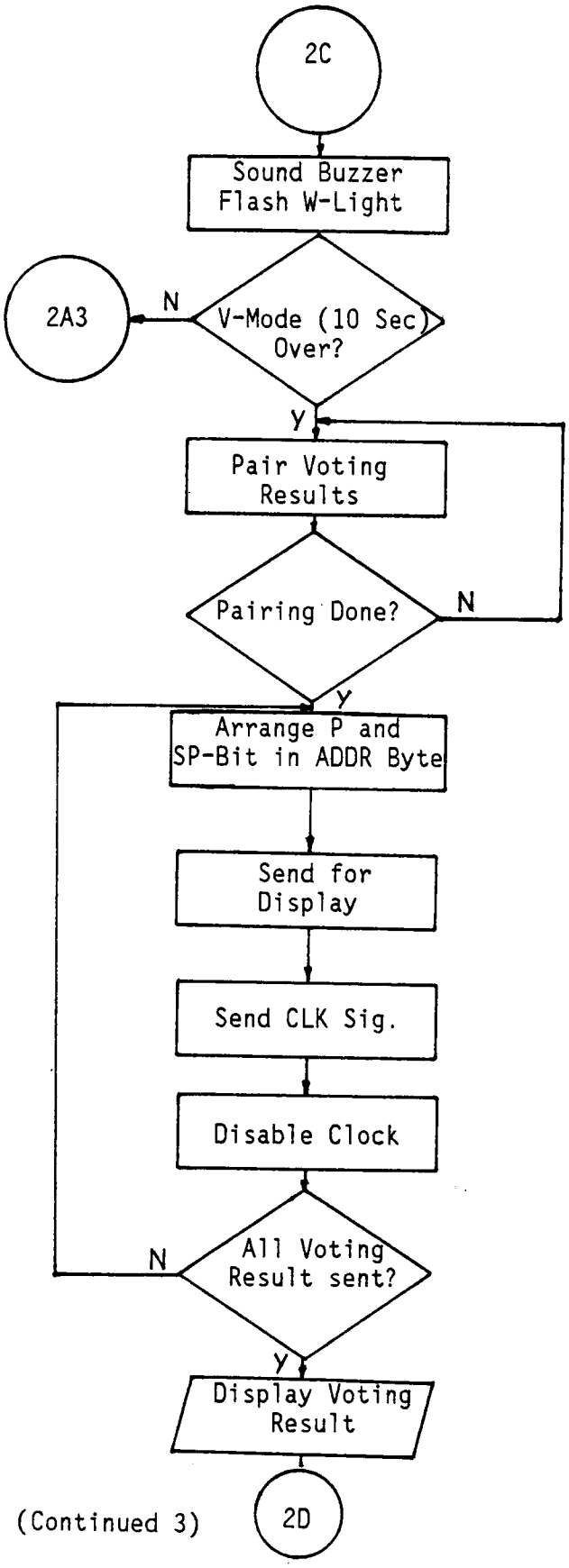
Voting Mode



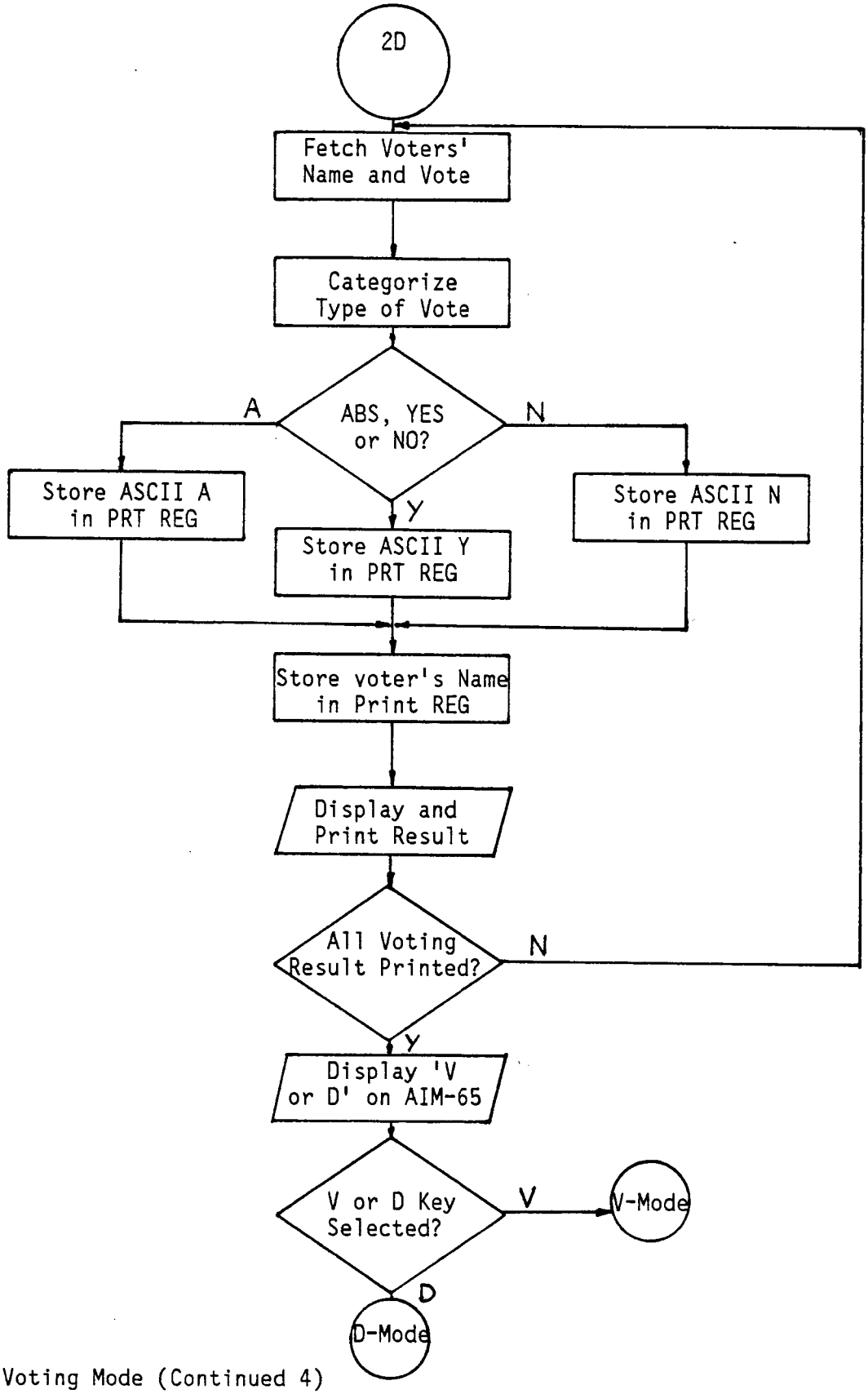
Voting Mode (Continued 1)



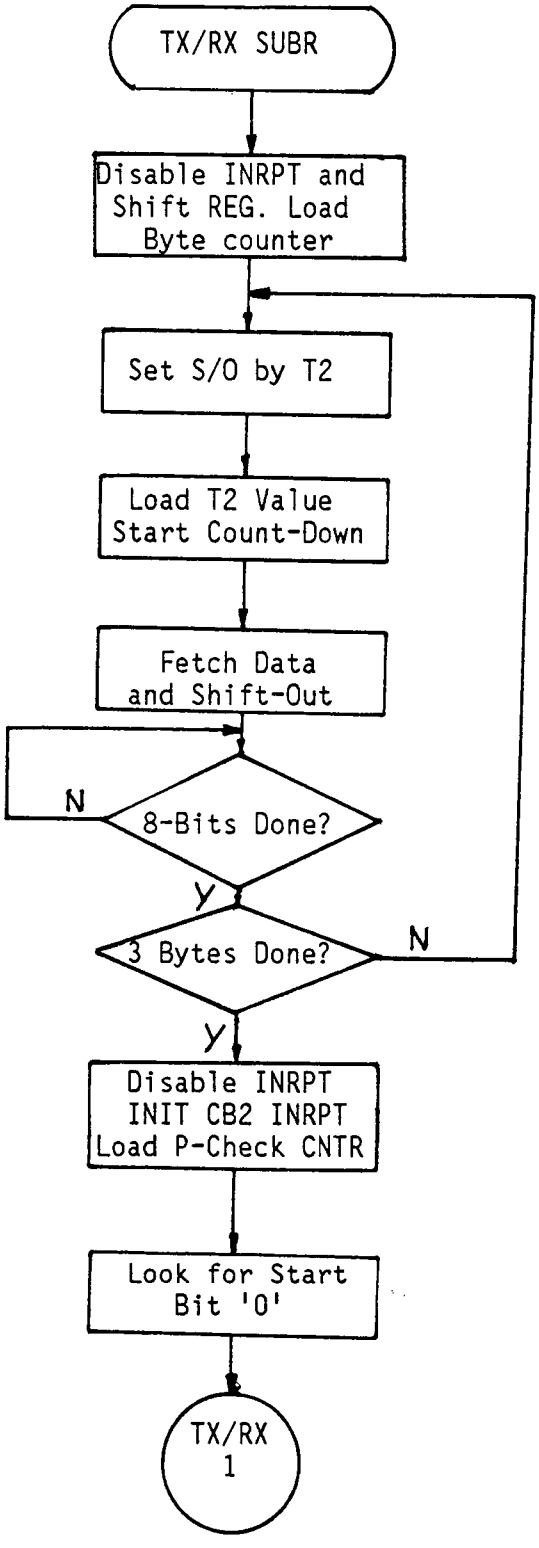
Voting Mode (Continued 2)



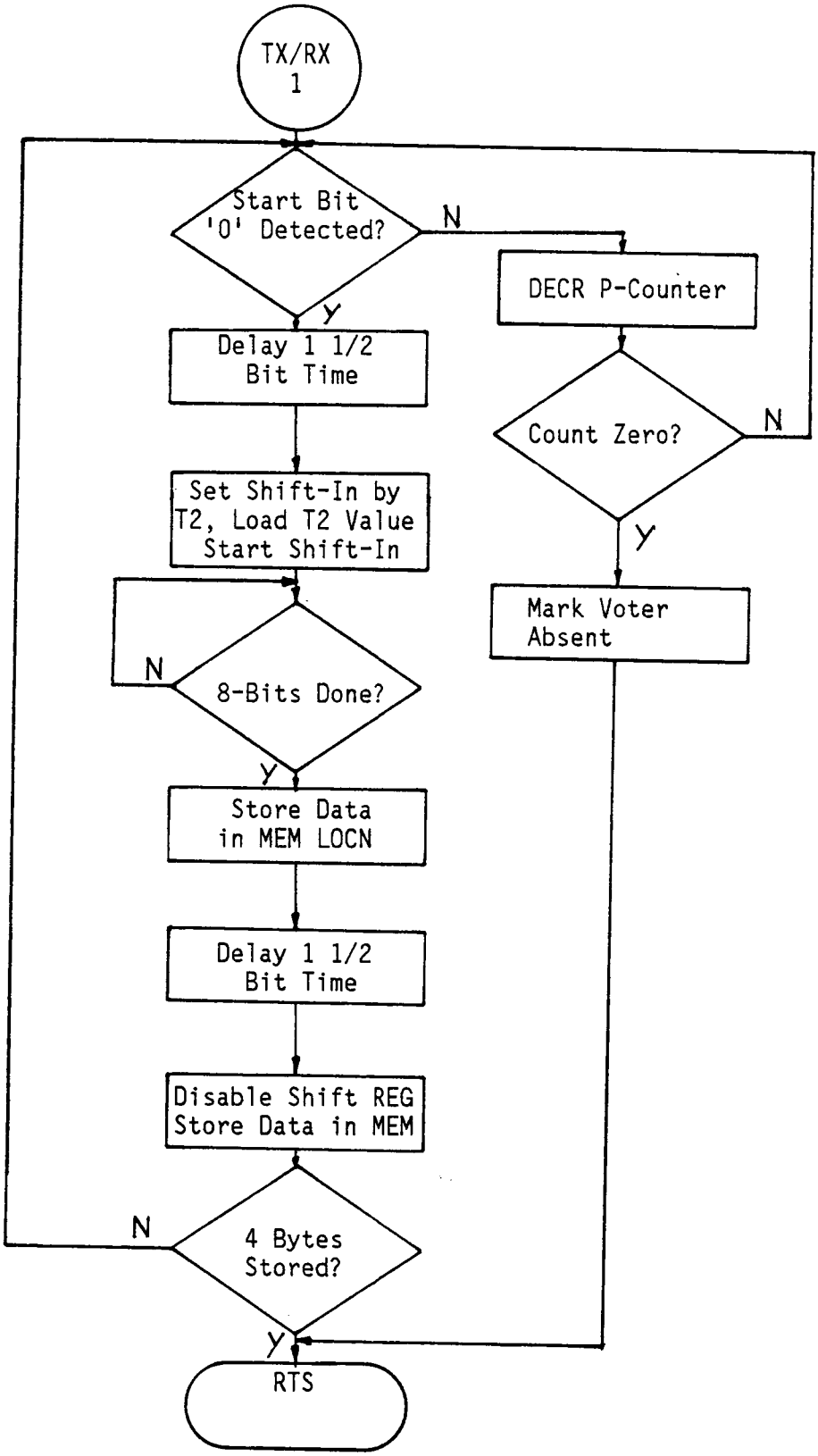
Voting Mode (Continued 3)



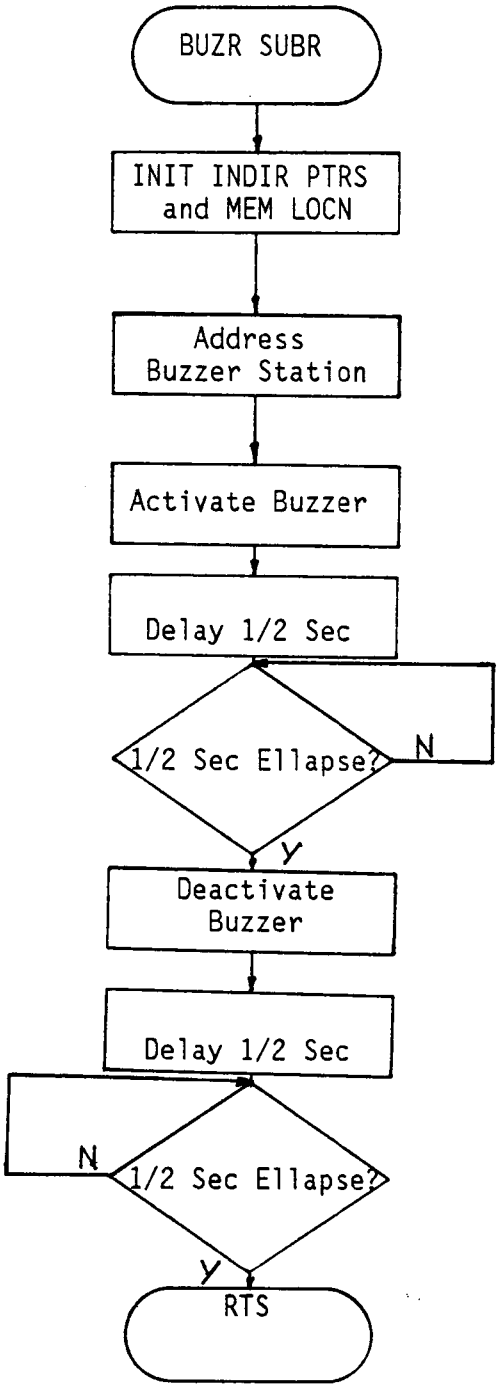
Voting Mode (Continued 4)



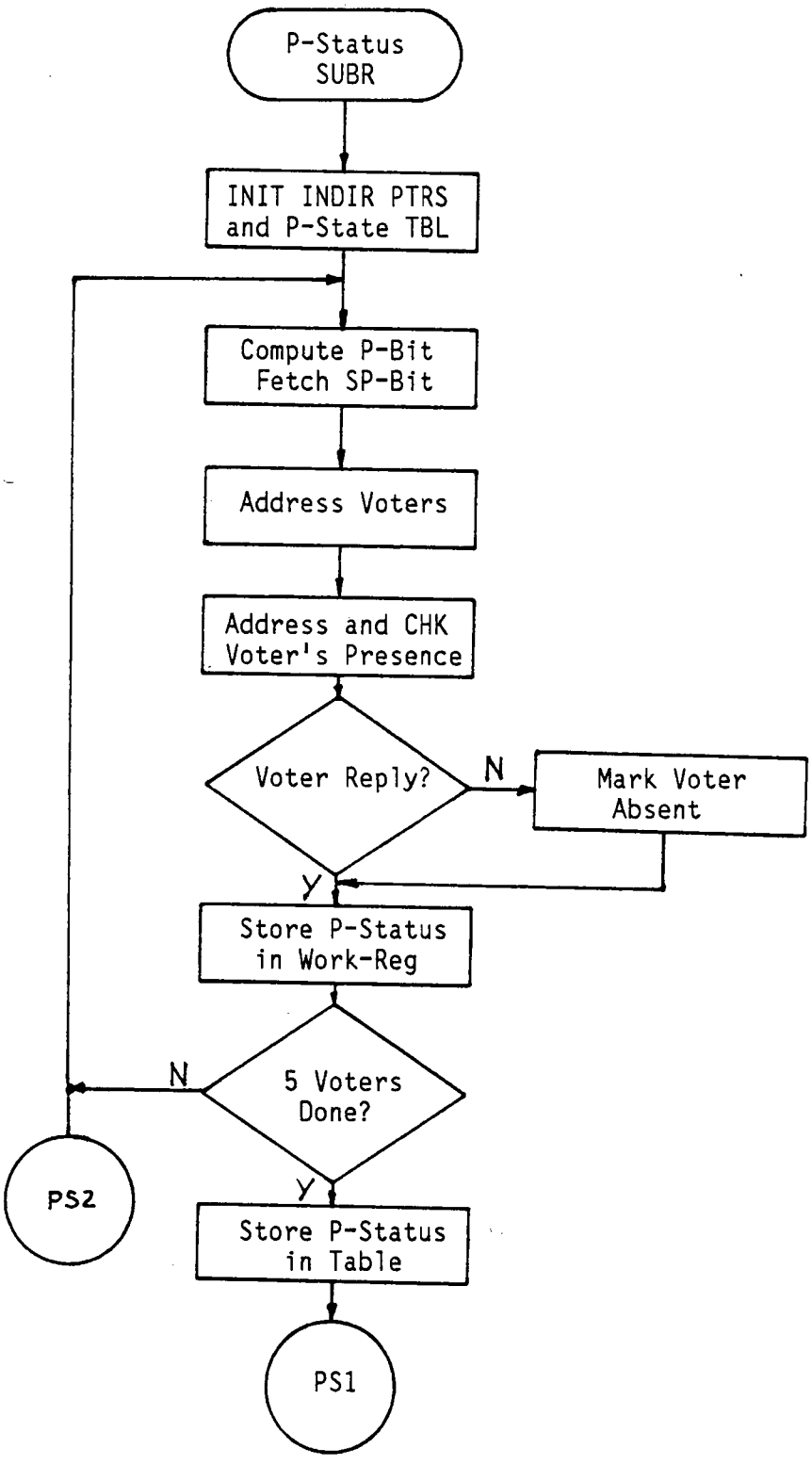
Transmit/Receive Routine



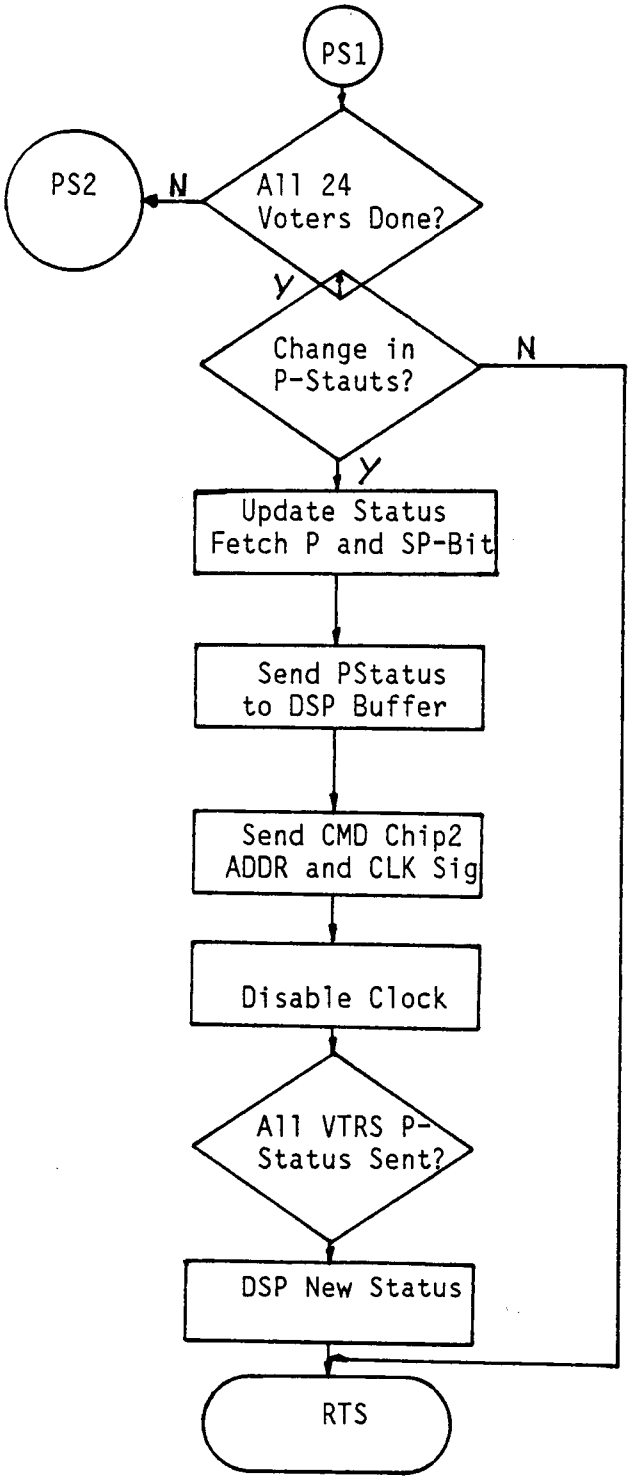
Transmit/Receive Routine (Continued)



Buzzer Routine



Present Status-Check Routine



Present Status-Check Routine (Continued)

APPENDIX C

TABLES OF MEMORY CONTENTS IN COMPUTER

Table 1. Voters' Addressing Addresses
(First Transmission, V or D Mode)

Station	Hex Code	* Bit Pattern	Station	Hex Code	* Bit Pattern
1	D0	11010000	2	C8	11001000
	2B	00101011		2B	00101011
3	D8	11011000	4	C4	11000100
	3B	00111011		2B	00101011
5	D4	11010100	6	CC	11001100
	3B	00111011		3B	00111011
7	DC	11011100	8	C2	11000010
	2B	00101011		2B	00101011
9	D2	11010010	10	CA	11001010
	3B	00111011		3B	00111011
11	DA	11011010	12	C6	11000110
	2B	00101011		3B	00111011
13	D6	11010110	14	CE	11001110
	2B	00101011		2B	00101011
15	DE	11011110	16	C1	11000001
	3B	00111011		2B	00101011
17	D1	11010001	18	C9	11001001
	3B	00111011		3B	00111011
19	D9	11011001	20	C5	11000101
	2B	00101011		3B	00111011
21	D5	11010101	22	CD	11001101
	2B	00101011		2B	00101011
23	DD	11011101	24	C3	11000011
	3B	00111011		3B	00111011

* Bit Pattern Assignment (Applicable to Table 2, 3, and 4)

1 1 ST A0 A1 A2 A3 A4
A5 A6 AI P SP ST M 1

ST=Start Bit
AI=Addr. ID
SP=Stop Bit

A=Addr. Bit
P=Parity Bit
M=Mode ID
1=V-Mode
0=D-Mode

Table 2. Discussion Mode Voters' Addressing Addresses (Second Transmission)

Station	Hex Code	* Bit Pattern	Station	Hex Code	* bit Pattern
1	DO	11010000	2	C8	11001000
	28	00101000		28	00101000
3	D8	11011000	4	C4	11000100
	38	00111000		28	00101000
5	D4	11010100	6	CC	11001100
	38	00111000		38	00111000
7	DC	11011100	8	C2	11000010
	28	00101000		28	00101000
9	D2	11001010	10	CA	11001010
	38	00111000		38	00111000
11	DA	11011010	12	C6	11000110
	28	00101000		38	00111000
13	D6	11010110	14	CE	11001110
	28	00101000		28	00101000
15	DE	11011110	16	C1	11000001
	38	00111000		28	00101000
17	D1	11010001	18	C9	11001001
	38	00111000		38	00111000
19	D9	11011001	20	C5	11000101
	28	00101000		38	00111000
21	D5	11010101	22	CD	11001101
	28	00101000		28	00101000
23	DD	11011101	24	C3	11000011
	38	00111000		38	00111000

Table 3. Voting Mode Voters' Addressing Addresses (Second Transmission)

Station	Hex Code	* Bit Pattern	Station	Hex Code	* Bit Pattern
1	D0	11010000	2	C8	11001000
	2A	00101010		2A	00101010
3	D8	11011000	4	C4	11000100
	3A	00111010		2A	00101010
5	D4	11010100	6	CC	11001100
	3A	00111010		3A	00111010
7	DC	11011100	8	C2	11000010
	2A	00101010		2A	00101010
9	D2	11010010	10	CA	11001010
	3A	00111010		3A	00111010
11	DA	11011010	12	C6	11000110
	2A	00101010		3A	00111010
13	D6	11010110	14	CE	11001110
	2A	00101010		2A	00101010
15	DE	11011110	16	C1	11000001
	3A	00111010		2A	00101010
17	D1	11010001	18	C9	11001001
	3A	00111010		3A	00111010
19	D9	11011001	20	C5	11000101
	2A	00101010		3A	00111010
21	D5	11010101	22	CD	11001101
	2A	00101010		2A	00101010
23	DD	11011101	24	C3	11000011
	3A	00111010		3A	00111010

Table 4. Display Panels Receivers' Addresses

Receiver	Station	Hex Code	* Bit Pattern
Command Chip 1	25	D3	11010011
		2B	00101011
Data Chip 1	26	CB	11001011
		28	00101000
Data Chip 2	27	DB	11011011
		38	00111000
Command Chip 2	28	C7	11000111
		2B	00101011
Buzzer	00	C0	11000000
		3B	00111011

Table 5. Display Panels Command Bit Assignment

Function (Activation or De-activation)	Bit Assignment S2	S3	S4	S5	S6	CI	P	SP
Buzzer	0	0	0	0	B	0	P	1
Present-Lamps	0	0	0	R1	0	0	P	1
Clock-Pulse	0	0	C1	0	0	0	P	1
Voting-Lamps	R2	0	0	0	0	0	P	1
Clock-Pulse	0	0	0	0	C2	0	P	1

B=Buzzer Bit
S*=Status Bits
CI=Command Identifier
P=Parity Bit
P=1 If Status Bits are Odd
P=0 If Status Bits are Even
R1 and R2 =Relay Bits
C1 and C2 =Clock Bits
B, R1, R2, C1, C2=1 (Activation)
B, R1, R2, C1, C2=0 (Deactivation)

Table 6. Transmit Registers Bit Assignment

Location	Bit Pattern
41	1 1 ST A0 A1 A2 A3 A4
42	A5 A6 AI P SP ST S0 S1
43	S2 S3 S4 S5 S6 CI P SP

Table 7. Receive Registers Bit Assignment

Location	Bit Pattern
50	A0 A1 A2 A3 A4 A5 A6 AI
51	A1 A2 A3 A4 A5 A6 AI P
52	S0 S1 S2 S3 S4 S5 S6 CI
53	S1 S2 S3 S4 S5 S6 CI P

A*=Address Bits
S*=Status Bits
P=Parity Bit
AI=Address Identifier
CI=Command Identifier
SP=Stop Bit

Table 8. Voting Results Storage Registers

Location	Function
80	Storage for Total "YES"
81	Storage for Total "ABSTAIN"
82	Storage for Total "NO"
83	Storage for Total "ABSENT"

Table 9. Memory Work Space (Voting Mode)

Location	Function
0C01-30	Voters' Addressing Addr1.
0C31-32	Command Chip 1 Addressing Addr1.
0C40-60	Voting Result Table
0C60-90	Voters' Addressing Addr2.
0C91-92	Data Chip 1 Address
0C93-94	Command Chip1 Addressing Addr2.
0CA1-AA	Voting Result Pairs
0D40-B8	Voting Result Printout
0E10-56	Voters' Names Table
0E91-95	Voters' Presence State Table Work-Registers (A1-A5)

Table 10. Memory Work Space (Discussion Mode)

Location	Function
0C01-30	Voters' Addressing Addr1.
0C31-32	Command Chip 1 Addressing Addr1.
0C60-90	Voters' Addressing Addr2.
0C91-92	Data Chip 1 Address
0C93-94	Command Chip 1 Addressing Addr2.
0CA1-AA	Discussion Status Pairs
0DD0-E8	Question or Request Status Table
0E10-56	Voters' Names Table
0E70-88	Voters' "REQUEST" Printout Table
0E91-95	Voters' Presence State Table Work-Registers (A1-A5)

APPENDIX D

PROGRAM LISTING OF SYSTEM

```
.MAIN.  CROSS G(33)  7-APR-84  09:53
C.KJK

                                ; ******************************************
                                ; *  MICROCOMPUTER-CONTROLLED ASSEMBLY     *
                                ; *  VOTING SYSTEM                         *
                                ; ******************************************
                                ;
    1       B000                .=$B000             ;INIT. SYS. STARTING ADDR.
                                ;
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ;
                                ; SUBR. FOR ILLUMINATION  TEST (LAMPS)
                                ;
    2  B000                     TON2:
    3  B000  20  13  EA                 JSR $EA13     ; CLR DISPLAY
    4  B003  A9  49                     LDA #$49
    5  B005  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'I'
    6  B008  A9  54                     LDA #$54
    7  B00A  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'T'
    8  B00D  A9  53                     LDA #$53
    9  B00F  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'S'
   10  B012  A9  54                     LDA #$54
   11  B014  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'T'
   12
   13  B017  20  3E  E8                 JSR $E83E     ; SKIP A SPACE
   14
   15  B01A  A9  4F                     LDA #$4F
   16  B01C  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'O'
   17  B01F  A9  52                     LDA #$52
   18  B021  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'R'
   19
   20  B024  20  3E  E8                 JSR $E83E     ; SKIP A SPACE
   21
   22  B027  A9  4E                     LDA #$4E
   23  B029  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'N'
   24  B02C  A9  41                     LDA #$41
   25  B02E  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'A'
   26  B031  A9  4D                     LDA #$4D
   27  B033  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'M'
   28  B036  A9  45                     LDA #$45
   29  B038  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'E'
   30  B03B  20  3E  E8                 JSR $E83E     ; SKIP A SPACE
   31  B03E  A9  3F                     LDA #$3F
   32  B040  20  05  EF                 JSR $EF05     ; DISPLAY ASCII '?'
   33
   34  B043  20  F8  E8                 JSR $E8F8     ; SELECT KEYBOARD
   35  B046  20  07  E9      TON1:      JSR $E907     ; SCAN KEYBOARD
   36  B049  C9  54                     CMP #$54      ; 'T' KEY I/P ?
   37  B04B  F0  0A                     BEQ ITST      ; YES, TEST LAMPS
   38  B04D  C9  4E                     CMP #$4E      ; 'N' KEY I/P ?
   39  B04F  F0  03                     BEQ NAME      ; YES, ENTER NAMES
   40  B051  4C  46  B0                 JMP TON1      ; NO, SCAN AGAIN
   41
   42  B054                      NAME:
   43  B054  4C  7D  B1                 JMP ENAM      ; GO TO NAME ROUTI.
   44
   45  B057                      ITST:               ; LAMPS TEST SUBR.
   46  B057  20  13  EA                 JSR $EA13     ; CLR DISPLAY
   47  B05A  A9  49                     LDA #$49
   48  B05C  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'I'
   49  B05F  A9  54                     LDA #$54
   50  B061  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'T'
   51  B064  A9  53                     LDA #$53
   52  B066  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'S'
   53  B069  A9  54                     LDA #$54
   54  B06B  20  05  EF                 JSR $EF05     ; DISPLAY ASCII 'T'
   55
   56  B06E  20  6A  B1                 JSR RESET     ; RESET SYS. CIRCUITRY
   57
   58  B071  A9  03                     LDA #$03
   59  B073  85  60                     STA $60       ; LOAD BYTES COUNTER
   60  B075  A9  CB                     LDA #$CB      ; FETCH DATA CHIP 1 ADDR.
   61  B077  85  41                     STA $41       ; STORE IN TX REGISTER
   62  B079  A9  2B                     LDA #$2B      ; FETCH THE NEXT BYTE
   63  B07B  85  42                     STA $42       ; STORE IN TX REGISTER
   64  B07D  A9  FB                     LDA #$FB      ; FETCH TEST BITS
   65  B07F  85  43                     STA $43       ; STORE IN TX REGISTER
   66  B081  20  78  BD                 JSR NOTIFY    ; SEND TO DISPLAY BUFFER
   67  B084  20  2C  ED                 JSR $ED2C     ; DELAY 5 MS
   68  B087  20  78  BD                 JSR NOTIFY    ; SEND AGAIN
```

```
 69
 70  B08A   20   0B   B1              JSR CLKS1      ; FETCH CLK AND RELAY SIG.
 71
 72  B08D   A9   03                   LDA #03
 73  B08F   85   60                   STA $60        ; LOAD BYTES COUNTER
 74  B091   A9   CB                   LDA #$CB       ; FETCH DATA CHIP 1 ADDR.
 75  B093   85   41                   STA $41        ; STORE IN TX REGISTER
 76  B095   A9   2A                   LDA #$2A       ; FETCH NEXT BYTE
 77  B097   85   42                   STA $42        ; STORE IN TX REGISTER
 78  B099   A9   03                   LDA #$03       ; FETCH LAMP DEACT. SIG.
 79  B09B   85   43                   STA $43        ; STORE IN TX REGISTER
 80  B09D   20   78   BD              JSR NOTIFY     ; SEND TO DISPLAY BUFFER
 81  B0A0   20   2C   ED              JSR $ED2C      ; DELAY 5 MS
 82  B0A3   20   78   BD              JSR NOTIFY     ; SEND AGAIN
 83
 84  B0A6   20   61   B1              JSR DLAY       ; DELAY 1 SECOND
 85
 86  B0A9   A9   0D                   LDA #$0D       ; LOAD TEST COUNTER
 87  B0AB   85   64                   STA $64
 88  B0AD   20   0B   B1    SFT1:     JSR CLKS1      ; PERFORM LAMPS TEST
 89  B0B0   20   61   B1              JSR DLAY       ; DELAY 1 SECOND
 90  B0B3   C6   64                   DEC $64        ; NEXT 6 LAMPS
 91  B0B5   A5   64                   LDA $64
 92  B0B7   C9   00                   CMP #$00       ; ALL LAMPS TESTED ?
 93  B0B9   D0   F2                   BNE SFT1       ; NO, TEST CONTINUE
 94
 95  B0BB                   PTST:                    ; PRESENCE LAMP TEST SUBR.
 96  B0BB   A9   DB                   LDA #$DB       ; FETCH DATA CHIP 2 ADDR.
 97  B0BD   85   41                   STA $41        ; STORE IN TX REGISTER
 98  B0BF   A9   3B                   LDA #$3B       ; FETCH NEXT BYTE
 99  B0C1   85   42                   STA $42        ; STORE IN TX REGISTER
100  B0C3   A9   FB                   LDA #$FB       ; FETCH TEST BITS
101  B0C5   85   43                   STA $43        ; STORE IN TX REGISTER
102  B0C7   A9   03                   LDA #$03       ; LOAD BYTES COUNTER
103  B0C9   85   60                   STA $60
104  B0CB   20   78   BD              JSR NOTIFY     ; SEND TO DISPLAY BUFFER
105  B0CE   20   2C   ED              JSR $ED2C      ; DELAY 5 MS
106  B0D1   20   78   BD              JSR NOTIFY     ; SEND AGAIN
107
108  B0D4   20   35   B1              JSR CLKS2      ; FETCH CLK AND RELAY SIG.
109
110  B0D7   A9   DB                   LDA #$DB       ; FETCH DATA CHIP 2 ADDR.
111  B0D9   85   41                   STA $41        ; STORE IN TX REGISTER
112  B0DB   A9   3A                   LDA #$3A       ; FETCH NEXT BYTE
113  B0DD   85   42                   STA $42        ; STORE IN TX REGISTER
114  B0DF   A9   03                   LDA #$03       ; FETCH LAMPS DEACT. SIG.
115  B0E1   85   43                   STA $43        ; STORE IN TX REGISTER
116  B0E3   A9   03                   LDA #$03       ; LOAD BYTES COUNTER
117  B0E5   85   60                   STA $60
118  B0E7   20   78   BD              JSR NOTIFY     ; SEND TO DISPLAY BUFFER
119  B0EA   20   2C   ED              JSR $ED2C      ; DELAY 5 MS
120  B0ED   20   78   BD              JSR NOTIFY     ; SEND AGAIN
121
122  B0F0   20   61   B1              JSR DLAY       ; DELAY 1 SECOND
123
124  B0F3   A9   06                   LDA #$06       ; LOAD TEST COUNTER
125  B0F5   85   64                   STA $64
126  B0F7   20   35   B1    SFT2:     JSR CLKS2      ; PERFORM LAMPS TEST
127  B0FA   20   61   B1              JSR DLAY       ; DELAY 1 SECOND
128  B0FD   C6   64                   DEC $64        ; NEXT 5 LAMPS
129  B0FF   A5   64                   LDA $64
130  B101   C9   00                   CMP #$00       ; ALL LAMPS TESTED ?
131  B103   D0   F2                   BNE SFT2       ; NO, TEST CONTINUE
132
133  B105   20   6A   B1              JSR RESET      ; YES, RESET SYS.
134
135  B108   4C   00   B0              JMP TON2       ; GO TO SELECT MODE
136
137
138  B10B                   CLKS1:                   ; VOTING LAMPS TEST ROU.
139  B10B   A9   D3                   LDA #$D3       ; FETCH CMD CHIP1 ADDR.
140  B10D   85   41                   STA $41        ; STORE IN TX REGISTER
141  B10F   A9   2B                   LDA #$2B       ; FETCH NEXT BYTE
142  B111   85   42                   STA $42        ; STORE IN TX REGISTER
143  B113   C6   60                   DEC $60        ; LOAD 2 BYTES TO BE SENT
144  B115   20   78   BD              JSR NOTIFY     ; ADDRESS CMD CHIP1
145  B118   20   2C   ED              JSR $ED2C      ; DELAY 5 MS
146  B11B   A9   89                   LDA #$89       ; FETCH CLK AND RELAY SIG.
147  B11D   85   43                   STA $43        ; STORE IN TX REGISTER
148  B11F   E6   60                   INC $60        ; LOAD 3 BYTES TO BE SENT
```

```
149  B121  20  78  BD          JSR NOTIFY     ; TEST LAMPS AND ROTATE
150  B124  20  2C  ED          JSR $ED2C      ; DELAY 5 MS
151  B127  A9  83               LDA #$83       ; FETCH CLK DISABLE SIG.
152  B129  85  43               STA $43        ; STORE IN TX REGISTER
153  B12B  20  78  BD          JSR NOTIFY     ; DISABLE CLOCK SIG.
154  B12E  20  2C  ED          JSR $ED2C      ; DELAY 5 MS
155  B131  20  78  BD          JSR NOTIFY     ; DISABLE AGAIN
156  B134  60                   RTS            ; RETURN TO MAIN ROUTINE
157
158  B135                  CLKS2:              ; PRESENCE LAMPS TEST ROU.
159  B135  A9  C7               LDA #$C7       ; FETCH CMD CHIP 2 ADDR.
160  B137  85  41               STA $41        ; STORE IN TX REGISTER
161  B139  A9  2B               LDA #$2B       ; FETCH NEXT BYTE
162  B13B  85  42               STA $42        ; STORE IN TX REGISTER
163  B13D  A9  02               LDA #$02       ; LOAD 2 BYTES TO BE SENT
164  B13F  85  60               STA $60
165  B141  20  78  BD          JSR NOTIFY     ; ADDRESS CMD CHIP 2
166  B144  20  2C  ED          JSR $ED2C      ; DELAY 5 MS
167  B147  A9  31               LDA #$31       ; FETCH CLK AND RLY SIG.
168  B149  85  43               STA $43        ; STORE IN TX REGISTER
169
170  B14B  E6  60               INC $60        ; LOAD 3 BYTES TO BE SENT
171  B14D  20  78  BD          JSR NOTIFY     ; TEST LAMPS AND ROTATE
172  B150  20  2C  ED          JSR $ED2C      ; DELAY 5 MS
173  B153  A9  13               LDA #$13       ; FETCH CLOCK DISABLE SIG.
174  B155  85  43               STA $43        ; STORE IN TX REGISTER
175  B157  20  78  BD          JSR NOTIFY     ; DISABLE CLOCK
176  B15A  20  2C  ED          JSR $ED2C      ; DELAY 5 MS
177  B15D  20  78  BD          JSR NOTIFY     ; DISABLE AGAIN
178  B160  60                   RTS            ; RETURN TO MAIN ROUTINE
179
180  B161                  DLAY:               ; 1 SECOND DELAY ROUTINE
181  B161  A0  FF               LDY #$FF       ; LOAD DELAY COUNTER
182  B163  20  2C  ED      DY:  JSR $ED2C      ; DELAY 5 MS
183  B166  88                   DEY            ; DECREMENT COUNTER
184  B167  D0  FA               BNE DY         ; COUNT DOWN ?
185  B169  60                   RTS            ; YES, RETURN TO MAIN ROU.
186
187  B16A                  RESET:              ; SYSTEM RESET SUBROUTINE
188  B16A  A9  FF               LDA #$FF       ; MAKE PORT B O/P PORT
189  B16C  8D  02  A0           STA $A002
190  B16F  8D  00  A0           STA $A000
191  B172  A9  00               LDA #$00       ; RESET SYSTEM'S CIRCUITRY
192  B174  8D  00  A0           STA $A000
193  B177  A9  FF               LDA #$FF       ; DISABLE RESET
194  B179  8D  00  A0           STA $A000
195  B17C  60                   RTS            ; RETURN TO MAIN ROUTINE
196
197
                               ;;;
```

```
225  B1AE  A9  45        LDA #$45
226  B1B0  20  05  EF     JSR $EF05     ; DISPALY ASCII 'E'
227  B1B3  A9  52        LDA #$52
228  B1B5  20  05  EF     JSR $EF05     ; DISPLAY ASCII 'R'
229  B1B8  20  3E  E8     JSR $E83E     ; SKIP A SPACE
230
231  B1BB  A9  4E        LDA #$4E
232  B1BD  20  05  EF     JSR $EF05     ; DISPLAY ASCII 'N'
233  B1C0  A9  41        LDA #$41
234  B1C2  20  05  EF     JSR $EF05     ; DISPLAY ASCII 'A'
235  B1C5  A9  4D        LDA #$4D
236  B1C7  20  05  EF     JSR $EF05     ; DISPLAY ASCII 'M'
237  B1CA  A9  45        LDA #$45
238  B1CC  20  05  EF     JSR $EF05     ; DISPLAY ASCII 'E'
239
240  B1CF  20  F8  E8     JSR $E8F8     ; SELECT KEYBOARD I/P
241  B1D2  20  07  E9  ETR2: JSR $E907  ; SCAN KEYBOARD
242  B1D5  20  83  FE     JSR $FE83     ; LOAD 1 CHAR. INTO ACCU.
243  B1D8  20  05  EF     JSR $EF05     ; DISPLAY ON COMPUTER PANL
244  B1DB  C9  45        CMP #$45      ; ENTER COMMAND (E) ?
245  B1DD  D0  F3        BNE ETR2      ; NO, SEARCH
246
247  B1DF  20  13  EA     JSR $EA13     ; YES, CLR DISPLAY
248  B1E2  A9  00     ETR3: LDA #$00   ; INIT. CHAR. COUNTER
249  B1E4  85  9A        STA $9A
250
251  B1E6  20  07  E9  ETR4: JSR $E907  ; SCAN KEYBOARD
252  B1E9  20  83  FE     JSR $FE83     ; READ 1 CHAR. INTO ACCU.
253  B1EC  20  05  EF     JSR $EF05     ; DISPLAY CHAR.
254  B1EF  C9  7F        CMP #$7F      ; 'DEL' KEY DEPRESSED ?
255  B1F1  F0  27        BEQ DEL       ; YES, DELETE CHARACTER
256  B1F3  C9  2F        CMP #$2F      ; TERMINATION COMMAND (/)?
257  B1F5  F0  34        BEQ MSELT     ; YES, GO TO MODE SEL ROU.
258  B1F7  A4  9A        LDY $9A
259  B1F9  91  98        STA ($98),Y   ; NO, STORE CHAR. TMPORARY
260  B1FB  E6  9A        INC $9A
261  B1FD  A4  9A        LDY $9A
262  B1FF  C0  03        CPY #$03      ; COMPLETE 3 CHARACTERS ?
263  B201  D0  E3        BNE ETR4      ; NO, NEXT CHARACTER
264
265  B203  A0  00        LDY #$00
266  B205  B1  98     ETR5: LDA ($98),Y ; STORE 3 CHAR. IN TABLE
267  B207  91  76        STA ($76),Y
268  B209  C8            INY
269  B20A  C0  03        CPY #$03      ; 3 CHAR. STORED ?
270  B20C  D0  F7        BNE ETR5      ; NO, NEXT
271
272  B20E  E6  76        INC $76       ; INCR. INDIRECT POINTER
273  B210  E6  76        INC $76
274  B212  E6  76        INC $76
275  B214  20  13  EA     JSR $EA13     ; CLR DISPLAY
276  B217  4C  E2  B1     JMP ETR3      ; FETCH NEXT NAME
277
278
279  B21A             DEL:             ; DELETE CHARACTER ROUTINE
280
281  B21A  A0  00        LDY #$00
282  B21C  A9  60        LDA #$60      ; LOAD ASCII ' '
283  B21E  91  98     DL1: STA ($98),Y ; CLR CHARACTER
284  B220  C8            INY
285  B221  C0  04        CPY #$04      ; ALL CHARS. CLEARED ?
286  B223  D0  F9        BNE DL1       ; NO, CONTINUE
287  B225  20  13  EA     JSR $EA13     ; CLR DISPLAY
288  B228  4C  E2  B1     JMP ETR3      ; FETCH NEXT NAME
289
290
291  B22B             MSELT:           ; MODE SELECTION ROUTINE
292
293  B22B  20  13  EA     JSR $EA13     ; CLR DISPLAY
294  B22E  20  44  EB     JSR $EB44     ; CLR DSP AND PRTR POINTER
295  B231  A9  56        LDA #$56
296  B233  20  05  EF     JSR $EF05     ; DISPLAY ASCII 'V'
297  B236  20  3E  E8     JSR $E83E     ; SKIP A SPACE
298
299  B239  A9  4F        LDA #$4F
300  B23B  20  05  EF     JSR $EF05     ; DISPLAY ASCII 'O'
301  B23E  A9  52        LDA #$52
302  B240  20  05  EF     JSR $EF05     ; DISPLAY ASCII 'R'
303  B243  20  3E  E8     JSR $E83E     ; SKIP A SPACE
304  B246  A9  44        LDA #$44
```

```
305  B248  20  05  EF          JSR $EF05      ; DISPLAY ASCII 'D'
306  B24B  A9  3F               LDA #$3F
307  B24D  20  05  EF          JSR $EF05      ; DISPLAY ASCII '?'
308
309  B250  20  F8  E8          JSR $E8F8      ; SELECT KEYBOARD INPUT
310  B253  20  07  E9  SLECT:  JSR $E907      ; SCAN KEYBOARD
311  B256  20  83  FE          JSR $FE83      ; LOAD CHAR. INTO ACCU.
312  B259  20  05  EF          JSR $EF05      ; DISPLAY ON PANEL
313  B25C  C9  56               CMP #$56       ; 'V' KEY DEPRESSED ?
314  B25E  F0  07               BEQ VGM1       ; YES, GO TO VOTING MODE
315  B260  C9  44               CMP #$44       ; 'D' KEY DEPRESSED ?
316  B262  F0  06               BEQ DISS       ; YES, GO TO DISCUSSN. MOD
317  B264  4C  53  B2           JMP SLECT      ; NO, SEARCH FOR SEL. CMD
318
319  B267               VGM1:
320
321  B267  4C  6D  B2           JMP VMODE      ; PROCEED TO VOTING SUBR.
322
323  B26A               DISS:
324
325  B26A  4C  65  B7           JMP DCUSN      ; PROCEED TO DCUSSN. SUBR.
326
327
                        ;;;
```

```
383  B2D4  20  A6  B3    NVTR1: JSR INTGN1    ; ADDR. VOTERS
384  B2D7  20  78  BD           JSR NOTIFY    ; INFORM VOTING MODE
385  B2DA  C6  60              DEC $60       ; GO TO NEXT VOTER
386  B2DC  E6  4E              INC $4E
387  B2DE  E6  4E              INC $4E
388  B2E0  E6  4C              INC $4C
389  B2E2  E6  4C              INC $4C
390  B2E4  C6  61              DEC $61
391  B2E6  A5  61              LDA $61
392  B2E8  C9  00              CMP #$00      ; ALL VOTERS DONE ?
393  B2EA  D0  E8              BNE NVTR1     ; NO, CONTINUE
394
395  B2EC  A2  30              LDX #$30
396  B2EE  C6  4E       VM2:   DEC $4E       ; REINIT. INDIRECT POINTER
397  B2F0  C6  4C              DEC $4C
398  B2F2  CA                  DEX
399  B2F3  D0  F9              BNE VM2       ; DONE ? IF NO, CONTINUE
400
401  B2F5  A9  18              LDA #$18      ; LOAD VOTERS COUNTER
402  B2F7  85  61              STA $61
403  B2F9  A9  02              LDA #$02      ; LOAD 2 BYTES TO BE SENT
404  B2FB  85  60              STA $60
405
406  B2FD  20  A6  B3    INRX1: JSR INTGN1    ; ADDR. VOTERS (2ND TX)
407  B300  20  A7  BD           JSR TXRX      ; TX AND RX VOTER'S STATUS
408  B303  20  92  B3           JSR RCHK1     ; CHECK FOR ACTIVE STATION
409  B306  20  CF  B3           JSR VERFY     ; PERFORM VALIDITY TEST
410  B309  20  6C  B4           JSR UDATV     ; PROCESS VOTING STATUS
411  B30C  20  78  BD           JSR NOTIFY    ; ACKNOWLEDGE VOTING STAT.
412  B30F  20  2C  ED           JSR $ED2C     ; DELAY 5 MS
413  B312  20  78  BD           JSR NOTIFY    ; ACKNOWLEDGE AGAIN
414
415  B315  C6  60       PRCD1: DEC $60       ; READY FOR NEXT VOTER
416  B317  E6  4E              INC $4E
417  B319  E6  4E              INC $4E
418  B31B  E6  4C              INC $4C
419  B31D  E6  4C              INC $4C
420  B31F  E6  4A              INC $4A
421
422  B321  C6  61              DEC $61
423  B323  A5  61              LDA $61
424  B325  C9  00              CMP #$00      ; ALL VOTERS DONE ?
425  B327  D0  D4              BNE INRX1     ; NO, CONTINUE POLLING
426
427  B329  A9  0A              LDA #$0A      ; LOAD BUZZER TIMING
428  B32B  85  63              STA $63
429
430  B32D  A2  FF       INRX3: LDX #$FF      ; INIT. STCK PTR END PAGE
431  B32F  9A                  TXS
432  B330  20  75  B3           JSR INIT1     ; INIT. INDIRECT POINTERS
433  B333  20  6C  B4    INRX2: JSR UDATV     ; FETCH OLD VOTING STATUS
434
435  B336  A9  03              LDA #$03      ; LOAD 3 BYTES TO BE SENT
436  B338  85  60              STA $60
437
438  B33A  20  78  BD           JSR NOTIFY    ; ADDR. VOTERS
439  B33D  20  2C  ED           JSR $ED2C     ; DELAY 5 MS
440  B340  20  A7  BD           JSR TXRX      ; TX AND RX VOTING STATUS
441  B343  20  9C  B3           JSR RCHK2     ; CHECK FOR ACTIVE STATUS
442  B346  20  CF  B3           JSR VERFY     ; PERFORM VALIDITY TEST
443  B349  20  6C  B4           JSR UDATV     ; PROCESS VOTING STATUS
444  B34C  20  78  BD           JSR NOTIFY    ; ACKNOWLEDGE VOTER STATUS
445  B34F  20  2C  ED           JSR $ED2C     ; DELAY 5 MS
446  B352  20  78  BD           JSR NOTIFY    ; ACKNOWLEDGE AGAIN
447
448  B355  E6  4C       PRCD2: INC $4C       ; READY TO POLL NEXT VOTER
449  B357  E6  4C              INC $4C
450  B359  E6  4E              INC $4E
451  B35B  E6  4E              INC $4E
452  B35D  E6  4A              INC $4A
453
454  B35F  C6  61              DEC $61
455  B361  A5  61              LDA $61
456  B363  C9  00              CMP #$00      ; ALL VOTERS DONE ?
457  B365  D0  CC              BNE INRX2     ; NO, CONTINUE POLLING
458  B367  20  FC  B6           JSR BUZZ      ; SOUND THE BUZZER
459  B36A  C6  63              DEC $63
460  B36C  A5  63              LDA $63
461  B36E  C9  00              CMP #$00      ; BUZZER TIMER UP ?
462  B370  D0  BB              BNE INRX3     ; NO, POLL FROM 1ST VOTER
```

```
463
464  B372  4C  8B  B4          JMP DSPVT     ; YES, GO TO DISPLAY ROUT.
465
466  B375                INIT1:              ; INIT. INDIRECT PTR ROU.
467  B375  A9  0C              LDA #$0C
468  B377  85  4B              STA $4B
469  B379  85  4D              STA $4D       ; ADDRESSING ADDR. PTR.
470  B37B  85  4F              STA $4F
471
472  B37D  A9  40              LDA #$40
473  B37F  85  4A              STA $4A       ; INDIR. PTR FOR RESLT TBL
474  B381  A9  60              LDA #$60
475  B383  85  4C              STA $4C
476  B385  A9  00              LDA #$00
477  B387  85  4E              STA $4E
478
479  B389  A9  FF              LDA #$FF
480  B38B  85  62              STA $62
481
482  B38D  A9  18              LDA #$18
483  B38F  85  61              STA $61       ; LOAD VOTERS COUNTER
484  B391  60                  RTS
485
486  B392                RCHK1:              ; ACTIVE STATION CHECK ROU.
487  B392  A5  51              LDA $51
488  B394  C9  00              CMP #$00      ; STATION ACTIVE ?
489  B396  F0  01              BEQ VRAB1     ; NO, GO TO NEXT STATION
490  B398  60                  RTS
491
492  B399                VRAB1:
493  B399  4C  15  B3          JMP PRCD1     ; PROCEED TO NEXT STATION
494
495  B39C                RCHK2:              ; ACTIVE STATION CHECK ROU
496  B39C  A5  51              LDA $51
497  B39E  C9  00              CMP #$00      ; STATION ACTIVE ?
498  B3A0  F0  01              BEQ VRAB2     ; NO, GO TO NEXT STATION
499  B3A2  60                  RTS
500  B3A3                VRAB2:
501  B3A3  4C  55  B3          JMP PRCD2     ; PROCEED TO NEXT STATION
502
503  B3A6                INTGN1:             ; ADDRESSING VOTERS ROU.
504  B3A6  20  B9  B3          JSR GAD1      ; FETCH TX1 ADDRESS
505  B3A9  20  78  BD          JSR NOTIFY    ; ADDRESS VOTER
506  B3AC  20  2C  ED          JSR $ED2C     ; DELAY 5 MS
507  B3AF  20  C4  B3          JSR GAD2      ; FETCH TX2 ADDRESS
508  B3B2  A9  03              LDA #$03      ; FETCH COMMAND BYTE
509  B3B4  85  43              STA $43       ; STORE IN TX REGISTER
510  B3B6  E6  60              INC $60       ; LOAD 3 BYTES TO BE SENT
511  B3B8  60                  RTS
512
513  B3B9                GAD1:               ; FETCHING ROUTINE FOR TX1
514  B3B9  A0  02              LDY #$02      ; LOAD INDIR. PTR
515  B3BB  B1  4E        GD1:  LDA ($4E),Y   ; LOAD TX1 ADDRESS
516  B3BD  99  40  00          STA $0040,Y   ; STORE IN TX REGISTER
517  B3C0  88                  DEY
518  B3C1  D0  F8              BNE GD1       ; NEXT BYTE IF NOT DONE
519  B3C3  60                  RTS
520
521  B3C4                GAD2:               ; FETCHING TX2 ADDRES ROU.
522  B3C4  A0  02              LDY #$02      ; LOAD INDIR. PTR.
523  B3C6  B1  4C        GD2:  LDA ($4C),Y   ; LOAD TX2 ADDRESS
524  B3C8  99  40  00          STA $0040,Y   ; STORE IN TX REGISTER
525  B3CB  88                  DEY
526  B3CC  D0  F8              BNE GD2       ; NEXT BYTE IF NOT DONE
527  B3CE  60                  RTS
528
529  B3CF                VERFY:              ; VALIDITY TEST ROUTINE
530
531  B3CF  A5  50              LDA $50       ; LOAD RECEIVED ADDR. BITS
532  B3D1  20  D4  BB          JSR CMPBT     ; COMPUTE PARITY BIT
533  B3D4  98                  TYA           ; TRANSFER P-BIT TO ACCU.
534  B3D5  18                  CLC           ; CLR CARRY BIT
535  B3D6  6A                  ROR A         ; STORE P-BIT IN CARRY
536  B3D7  A9  00              LDA #$00      ; CLR ACCU.
537  B3D9  2A                  ROL A         ; RETREIVE P-BIT
538  B3DA  85  5A              STA $5A       ; STORE IN TEMP. REGISTER
539
540  B3DC  A5  51              LDA $51       ; FETCH RX P-BIT
541  B3DE  18                  CLC           ; CLR CARRY BIT
542  B3DF  4A                  LSR A         ; STORE P-BIT IN CARRY
```

```
543  B3E0  A9  00          LDA #$00      ; CLR ACCU.
544  B3E2  2A              ROL A         ; RETREIVE P-BIT
545  B3E3  C5  5A          CMP $5A       ; VALID P-BIT ?
546  B3E5  D0  34          BNE BERR1     ; NO, DISPLAY AND HALT OPR
547
548  B3E7  A5  52          LDA $52       ; FETCH RECEIVED STAT. BYT
549  B3E9  20  D4  BB      JSR CMPBT     ; COMPUTE P-BIT
550  B3EC  98              TYA           ; TRANSFER P-BIT TO ACCU.
551  B3ED  18              CLC           ; CLR CARRY BIT
552  B3EE  6A              ROR A         ; STORE P-BIT IN CARRY
553  B3EF  A9  00          LDA #$00      ; CLR ACCU.
554  B3F1  2A              ROL A         ; RETREIVE P-BIT
555  B3F2  85  5B          STA $5B       ; STORE P-BIT IN TEMP REG.
556
557  B3F4  A5  53          LDA $53       ; FETCH RECEIVED P-BIT
558  B3F6  18              CLC
559  B3F7  4A              LSR A         ; STORE P-BIT IN CARRY
560  B3F8  A9  00          LDA #$00
561  B3FA  2A              ROL A         ; RETREIVE P-BIT
562  B3FB  C5  5B          CMP $5B       ; VALID P-BIT ?
563  B3FD  D0  36          BNE BERR2     ; NO, DISPLAY AND HALT OPR
564
565  B3FF  A5  52          LDA $52       ; FETCH VOTING STATUS
566  B401  85  49          STA $49       ; STORE IN WORK REGISTER
567  B403  06  49          ASL $49       ; DISCARD BIT S1
568  B405  18              CLC           ; CLR CARRY BIT
569  B406  A9  00          LDA #$00      ; CLR ACCU.
570  B408  A2  03          LDX #$03      ; LOAD POSITION COUNTER
571  B40A  26  49  VFY1:   ROL $49       ; REARRANGE BIT
572  B40C  2A              ROL A
573  B40D  CA              DEX
574  B40E  D0  FA          BNE VFY1      ; DONE ?
575  B410  85  49          STA $49       ; YES, STORE IN WORK REG.
576
577  B412  A0  01          LDY #$01
578  B414  11  4A          ORA ($4A),Y   ; PERFORM "OR" OPERATION
579  B416  D1  4A          CMP ($4A),Y   ; CHANGE IN STATUS?
580  B418  D0  35          BNE BTST      ; YES, TEST AND UPDATE STA
581  B41A  60    VRTN1:    RTS           ; NO, RETURN TO MAIN ROUT.
582
583  B41B        BERR1:                  ; ERROR MESSAGE (TST FAIL)
584
585  B41B  20  13  EA      JSR $EA13     ; CLR DISPLAY
586  B41E  A9  45          LDA #$45
587  B420  20  05  EF      JSR $EF05     ; DISPLAY ASCII 'E'
588  B423  A9  52          LDA #$52
589  B425  20  05  EF      JSR $EF05     ; DISPLAY ASCII "R'
590  B428  A9  52          LDA #$52
591  B42A  20  05  EF      JSR $EF05     ; DISPLAY ASCII 'R'
592  B42D  A9  31          LDA #$31
593  B42F  20  05  EF      JSR $EF05     ; DISPLAY ASCII '1'
594  B432  4C  32  B4  ERR1:  JMP ERR1   ; HALT SYSTEM OPERATION
595
596
597  B435        BERR2:                  ; ERROR MESSAGE (TST FAIL)
598
599  B435  20  13  EA      JSR $EA13     ; CLR DISPLAY
600  B438  A9  45          LDA #$45
601  B43A  20  05  EF      JSR $EF05     ; DISPLAY ASCII 'E'
602  B43D  A9  52          LDA #$52
603  B43F  20  05  EF      JSR $EF05     ; DISPLAY ASCII 'R'
604  B442  A9  52          LDA #$52
605  B444  20  05  EF      JSR $EF05     ; DISPLAY ASCII 'R'
606  B447  A9  32          LDA #$32
607  B449  20  05  EF      JSR $EF05     ; DISPLAY ASCII '2'
608  B44C  4C  4C  B4  ERR2:  JMP ERR2   ; HALT SYSTEM OPERATION
609
610
611  B44F        BTST:                   ; ROUTINE TST FOR P-BUTTN.
612
613  B44F  A5  49          LDA $49       ; FETCH VOTING STATUS
614  B451  20  D4  BB      JSR CMPBT     ; COMPUTE NO. PB. PUSHED
615  B454  C0  02          CPY #$02      ; TWO ?
616  B456  F0  0B          BEQ NVBTN     ; YES, GO TO INVALID ROUT.
617  B458  C0  03          CPY #$03      ; THREE ?
618  B45A  F0  07          BEQ NVBTN     ; YES, GO TO INVALID ROUT.
619  B45C  A0  01          LDY #$01      ; IT'S VALID PUSHBUTTON
620  B45E  91  4A          STA ($4A),Y   ; UPDATE VOTING STATUS
621  B460  4C  1A  B4      JMP VRTN1     ; RETURN TO MAIN ROUTINE
622
```

```
623  B463                          NVBTN:              ; INVALID PUSH-BUTTON ROU.
624
625  B463  A0  01                        LDY #$01      ; LOAD INDIR. PTR.
626  B465  A9  00                        LDA #$00
627  B467  91  4A                        STA ($4A),Y   ; CLR VOTING STATUS
628  B469  4C  1A  B4                    JMP VRTN1     ; RETURN TO MAIN ROUTINE
629
630  B46C                          UDATV:              ; ACKNOWLEDGE STATUS ROUT.
631
632  B46C  A0  01                        LDY #$01      ; LOAD INDIR. PTR.
633  B46E  B1  4C                        LDA ($4C),Y   ; FETCH VOTER ADDR. BYTE
634  B470  85  41                        STA $41       ; STORE IN TX REGISTER
635  B472  B1  4A                        LDA ($4A),Y   ; FETCH VOTING STATUS
636  B474  85  5C                        STA $5C       ; STORE IN WORK REGISTER
637  B476  C8                            INY
638  B477  B1  4C                        LDA ($4C),Y   ; NEXT ADDR. BYTE
639  B479  85  42                        STA $42
640
641  B47B  A5  5C                        LDA $5C       ; FETCH VOTING STATUS
642  B47D  20  D4  BB                    JSR CMPBT     ; COMPUTE P-BIT
643  B480  98                            TYA           ; TRANSFER P-BIT TO ACCU.
644  B481  6A                            ROR A         ; TEST P-BIT STATUS
645  B482  90  1A                        BCC PBONE     ; IF EVEN BIT, P-BIT'S '1'
646
647  B484  18                            CLC           ; IT'S ODD, P-BIT '0'
648  B485  66  5C                        ROR $5C       ; CONFIGURE P-BIT IN A BYT
649  B487  66  5C                        ROR $5C
650  B489  66  5C                        ROR $5C
651  B48B  66  5C                        ROR $5C
652  B48D  66  42                        ROR $42       ; PAD BIT S2 TO 2ND BYTE
653  B48F  18                            CLC           ; FETCH P-BIT
654  B490  26  5C                        ROL $5C
655  B492  26  42                        ROL $42
656  B494  46  5C                        LSR $5C
657  B496  38                            SEC           ; FETCH STOP BIT
658  B497  26  5C                        ROL $5C
659  B499  A5  5C                  UD1:  LDA $5C       ; LOAD VOTING STATUS
660  B49B  85  43                        STA $43       ; STORE IN TX REGISTER
661  B49D  60                            RTS
662
663  B49E                          PBONE:              ; P-BIT '1' ROUTINE
664
665  B49E  18                            CLC
666  B49F  66  5C                        ROR $5C       ; ARRANGE VOTING STAT. MSB.
667  B4A1  66  5C                        ROR $5C
668  B4A3  66  5C                        ROR $5C
669  B4A5  66  5C                        ROR $5C
670  B4A7  46  42                        LSR $42
671  B4A9  38                            SEC           ; FETCH VOTING-BIT '1'
672  B4AA  26  5C                        ROL $5C
673  B4AC  26  42                        ROL $42       ; PAD BIT S2 TO 2ND BYTE
674  B4AE  46  5C                        LSR $5C
675  B4B0  46  5C                        LSR $5C
676  B4B2  38                            SEC           ; FETCH P-BIT '1'
677  B4B3  26  5C                        ROL $5C
678  B4B5  38                            SEC           ; FETCH STOP BIT '1'
679  B4B6  26  5C                        ROL $5C
680  B4B8  4C  99  B4                    JMP UD1       ; RETURN TO MAIN ROUTINE
681
                                   ;;;
```

```
701  B4DB  9D  A0  0C          STA $0CA0,X   ; CLR WORK SPACE
702  B4DE  CA                  DEX
703  B4DF  D0  F8              BNE DP1       ; CONTINUE IF NOT DONE
704
705  B4E1  A9  0C              LDA #$0C      ; INIT. INDIR. POINTERS
706  B4E3  85  71              STA $71
707  B4E5  85  4B              STA $4B
708  B4E7  A9  A0              LDA #$A0
709  B4E9  85  70              STA $70
710  B4EB  A9  40              LDA #$40
711  B4ED  85  4A              STA $4A
712
713  B4EF  A2  FF              LDX #$FF      ; INIT. STCK PTR AT END PG
714  B4F1  9A                  TXS
715
716  B4F2  A9  0C              LDA #$0C      ; LOAD RESULT PAIR COUNTER
717  B4F4  85  64              STA $64
718
719  B4F6  A0  01        DP2:  LDY #$01
720  B4F8  B1  4A              LDA ($4A),Y   ; FETCH ODD VOTER'S VOTE
721  B4FA  85  65              STA $65       ; STORE IN WORK-REGISTER
722  B4FC  C8                  INY
723  B4FD  B1  4A              LDA ($4A),Y   ; FETCH EVEN VOTER'S VOTE
724  B4FF  18                  CLC
725  B500  A2  03              LDX #$03
726  B502  46  65        DP3:  LSR $65       ; ARRANGE INTO A SNGL. BYT
727  B504  2A                  ROL A
728  B505  CA                  DEX
729  B506  D0  FA              BNE DP3       ; DONE ? CONTINUE IF NOT
730
731  B508  88                  DEY
732  B509  91  70              STA ($70),Y   ; STORE RESULT PAIR NEWLOC
733  B508  E6  4A              INC $4A       ; NEXT PAIR
734  B50D  E6  4A              INC $4A
735  B50F  E6  70              INC $70
736  B511  C6  64              DEC $64
737  B513  A5  64              LDA $64
738  B515  C9  00              CMP #$00      ; ALL RESULTS DONE ?
739  B517  D0  DD              BNE DP2       ; CONTINUE IF NOT
740
741  B519  A2  0C              LDX #$0C
742  B51B  C6  70        DP4:  DEC $70       ; REINIT. INDIR. PTR.
743  B51D  CA                  DEX
744  B51E  D0  FB              BNE DP4
745
746  B520  A9  0C              LDA #$0C
747  B522  85  64              STA $64       ; LOAD VOTERS COUNTER
748
749  B524  A0  01        NXDSP: LDY #$01
750  B526  B1  70              LDA ($70),Y   ; FETCH RESULT PAIR
751  B528  20  D4  BB          JSR CMPBT     ; COMPUTE P-BIT
752  B52B  85  66              STA $66       ; SAVE RESULT PAIR IN REG.
753  B52D  98                  TYA           ; TRANSFER P-BIT TO ACCU.
754  B52E  18                  CLC
755  B52F  26  66              ROL $66       ;LOAD CMD ID BIT
756  B531  4A                  LSR A
757  B532  26  66              ROL $66       ; LOAD P-BIT
758
759  B534  A0  02              LDY #$02
760  B536  B9  90  0C          LDA $0C90,Y   ; FETCH 2ND ADDR. BYT DCHP
761  B539  4A                  LSR A         ; DISCARD LSB BIT
762  B53A  38                  SEC           ; FETCH STOP BIT '1'
763  B53B  26  66              ROL $66
764  B53D  2A                  ROL A
765  B53E  85  42              STA $42       ; STORE IN TX REGISTER
766
767  B540  A9  03              LDA #$03      ; LOAD 3 BYTES TO BE SENT
768  B542  85  60              STA $60
769
770  B544  A0  01              LDY #$01
771  B546  B9  90  0C          LDA $0C90,Y   ; FETCH DATA CHIP 1 ADDR.
772  B549  85  41              STA $41       ; STORE IN TX REGISTER
773  B548  A5  66              LDA $66       ; FETCH VOTING RESULT
774  B54D  85  43              STA $43       ; STORE IN TX REGISTER
775
776  B54F  20  78  BD          JSR NOTIFY    ; SENT TO DISPLAY BUFFER
777  B552  20  2C  ED          JSR $ED2C     ; DELAY 5 MS
778  B555  20  78  BD          JSR NOTIFY    ; SEND AGAIN
779
780  B558  A0  01              LDY #$01
```

```
781  855A  B9  30  0C          LDA $0C30,Y   ; FETCH TX1 CMD CHIP 1 ADDR
782  855D  85  41              STA $41       ; STORE IN TX REGISTER
783  855F  C8                  INY
784  8560  B9  30  0C          LDA $0C30,Y   ; FETCH NEXT BYTE
785  8563  85  42              STA $42
786
787  8565  C6  60              DEC $60       ; LOAD 2 BYTES TO BE SENT
788  8567  20  78  BD          JSR NOTIFY    ; ADDRESS COMMAND CHIP 1
789  856A  20  2C  ED          JSR $ED2C     ; DELAY 5 MS
790
791  856D  A0  03              LDY #$03
792  856F  B9  90  0C          LDA $0C90,Y   ; FETCH TX2 CMD CHIP 1 ADD
793  8572  85  41              STA $41       ; STORE IN TX REGISTER
794  8574  C8                  INY
795  8575  B9  90  0C          LDA $0C90,Y   ; FETCH NEXT BYTE
796  8578  85  42              STA $42
797  857A  A9  0B              LDA #$0B      ; FETCH CLOCK SIGNAL
798  857C  85  43              STA $43       ; STORE IN TX REGISTER
799  857E  E6  60              INC $60
800  8580  20  78  BD          JSR NOTIFY    ; SHIFT RSLT NEXT SFT-REG.
801
802  8583  A9  01              LDA #$01      ; FETCH CLOCK DISABLE SIG.
803  8585  85  43              STA $43       ; STORE IN TX REGISTER
804  8587  20  78  BD          JSR NOTIFY    ; DISABLE SHIFTING
805  858A  20  2C  ED          JSR $ED2C     ; DELAY 5 MS
806  858D  20  78  BD          JSR NOTIFY    ; DISABLE AGIAN
807
808  8590  E6  70              INC $70       ; POINT TO NEXT RESLT PAIR
809  8592  C6  64              DEC $64
810  8594  A5  64              LDA $64
811  8596  C9  00              CMP #$00      ; COMPLETE SENDING RESULT?
812  8598  D0  8A              BNE NXDSP     ; NO, FETCH NEXT PAIR
813
814  859A  A9  83              LDA #$83      ; FETCH RELAY ACTI. SIGNAL
815  859C  85  43              STA $43       ; STORE IN TX REGISTER
816  859E  20  78  BD          JSR NOTIFY    ; DISPALY VOTING RESULT
817  85A1  20  2C  ED          JSR $ED2C     ; DELAY 5 MS
818  85A4  20  78  BD          JSR NOTIFY    ; MAKE SURE RELAY IS ON
819  85A7  4C  AA  B5          JMP PRTER     ; PROCEED TO PRINTING MODE
820

                        ;;;
```

```
859  B5E4  A9  3F          LDA #$3F
860  B5E6  85  78          STA $78
861
862  B5E8  A9  18          LDA #$18      ; LOAD VOTERS COUNTER
863  B5EA  85  64          STA $64
864
865  B5EC  20  01  E9      JSR $E901     ; ACTIVATE PRINTER UNIT
866  B5EF  A0  01    PR2:  LDY #$01
867  B5F1  B1  78    PR3:  LDA ($78),Y   ; FETCH VTRS' NAME AND VOTE
868  B5F3  20  8C  E9      JSR $E98C     ; SEND TO PRINTER BUFFER
869  B5F6  C8              INY
870  B5F7  C0  06          CPY #$06      ; DONE ?
871  B5F9  D0  F6          BNE PR3       ; NO, CONTINUE
872
873  B5FB  20  24  EA      JSR $EA24     ; PRINT VOTING RESULT
874  B5FE  20  44  EB      JSR $EB44     ; CLR PRINTER POINTER
875  B601  A2  05          LDX #$05
876  B603  E6  78    PR4:  INC $78       ; READY FOR NEXT RESULT
877  B605  CA              DEX
878  B606  D0  F8          BNE PR4
879
880  B608  C6  64          DEC $64
881  B60A  A5  64          LDA $64
882  B60C  C9  00          CMP #$00      ; ALL VOTING RESULT DONE?
883  B60E  D0  DF          BNE PR2       ; NO, NEXT RESULT
884
885  B610  20  13  EA      JSR $EA13     ; CLR DISPLAY
886  B613  20  3B  E8      JSR $E83B     ; SKIP 3 SPACES
887  B616  20  3B  E8      JSR $E83B
888  B619  20  3B  E8      JSR $E83B
889
890  B61C  A9  59          LDA #$59
891  B61E  20  FC  EE      JSR $EEFC     ; DISPLAY ASCII 'Y'
892  B621  20  3B  E8      JSR $E83B     ; SKIP 2 SPACES
893
894  B624  A9  41          LDA #$41
895  B626  20  FC  EE      JSR $EEFC     ; DISPLAY ASCII 'A'
896  B629  20  3B  E8      JSR $E83B     ; SKIP 2 SPACES
897
898  B62C  A9  4E          LDA #$4E
899  B62E  20  FC  EE      JSR $EEFC     ; DISPLAY ASCII 'N'
900  B631  20  3B  E8      JSR $E83B     ; SKIP 2 SPACES
901
902  B634  A9  40          LDA #$40
903  B636  20  FC  EE      JSR $EEFC     ; DISPLAY ASCII '@' (ABST)
904
905  B639  20  24  EA      JSR $EA24     ; PRINT RESULT SUM
906  B63C  20  44  EB      JSR $EB44     ; CLR PRINTER POINTERS
907
908  B63F  A9  54          LDA #$54
909  B641  20  FC  EE      JSR $EEFC     ; DISPLAY ASCII 'T'
910  B644  A9  4F          LDA #$4F
911  B646  20  FC  EE      JSR $EEFC     ; DISPLAY ASCII 'O'
912  B649  A9  4C          LDA #$4C
913  B64B  20  FC  EE      JSR $EEFC     ; DISPLAY ASCII 'L'
914  B64E  20  3E  E8      JSR $E83E     ; SKIP 3 SPACES
915  B651  20  3B  E8      JSR $E83B
916
917  B654  A5  80          LDA $80       ; FETCH TOL 'YES' VOTE
918  B656  20  46  EA      JSR $EA46     ; CONVERT TO DECIMAL VALUE
919  B659  20  3E  E8      JSR $E83E     ; SKIP A SPACE
920
921  B65C  A5  81          LDA $81       ; LOAD TOL 'ABSTAIN' VOTE
922  B65E  20  46  EA      JSR $EA46     ; CONVERT TO DECIMAL VALUE
923  B661  20  3E  E8      JSR $E83E     ; SKIP A SPACE
924
925  B664  A5  82          LDA $82       ; LOAD TOL 'NO' VOTE
926  B666  20  46  EA      JSR $EA46     ; CONVERT TO DECIMAL VALUE
927  B669  20  3E  E8      JSR $E83E     ; SKIP A SPACE
928
929  B66C  A5  83          LDA $83       ; LOAD TOL NOT VOTING '@'
930  B66E  20  46  EA      JSR $EA46     ; CONVERT TO DECIMAL VALUE
931
932  B671  20  24  EA      JSR $EA24     ; PRINT TOL OF VOTING RESL
933  B674  20  44  EB      JSR $EB44     ; CLR PRINTER POINTER
934
935  B677  4C  2B  B2      JMP MSELT     ; PROCEED TO SELECT MODE
936
937
938  B67A          CMPVR:              ; ROUTINE ARR RESLT FORMAT
```

```
939
940  B67A  A2  03        CR1:  LDX #$03
941  B67C  A0  01              LDY #$01
942  B67E  B1  76        CR2:  LDA ($76),Y   ; FETCH VOTERS NAMES
943  B680  91  78              STA ($78),Y   ; STORE IN PRINT TABLE
944  B682  C8                 INY
945  B683  CA                 DEX
946  B684  D0  F8              BNE CR2       ; CONTINUE IF NOT DONE
947
948  B686  A9  60              LDA #$60      ; LEAVE A BLANK
949  B688  91  78              STA ($78),Y
950  B68A  A0  01              LDY #$01
951  B68C  B1  4A              LDA ($4A),Y   ; FETCH VOTE CAST
952  B68E  C9  00              CMP #$00      ; DID VOTER VOTE ?
953  B690  F0  22              BEQ BLANK     ; NO, MARK ABSENT
954
955  B692  18                  CLC
956  B693  6A                  ROR A         ; VOTE 'YES' ?
957  B694  B0  30              BCS NO        ; YES, UPDATE TOL 'YES'
958
959  B696  6A                  ROR A         ; VOTE 'ABSTAIN' ?
960  B697  B0  3F              BCS ABSN      ; YES, UPDATE TOL 'ABSTN'
961
962  B699  6A                  ROR A         ; VOTE 'YES' ?
963  B69A  B0  4E              BCS YES       ; YES, UPDATE TOL 'YES'
964
965  B69C  A2  03        CR6:  LDX #$03
966  B69E  E6  76        CR7:  INC $76       ; READY FOR NEXT VOTER
967  B6A0  E6  78              INC $78
968  B6A2  CA                  DEX
969  B6A3  D0  F9              BNE CR7
970
971  B6A5  E6  78              INC $78
972  B6A7  E6  78              INC $78
973  B6A9  E6  4A              INC $4A
974  B6AB  C6  64              DEC $64
975  B6AD  A5  64              LDA $64
976  B6AF  C9  00              CMP #$00      ; ALL VOTING RESULT CHECK?
977  B6B1  D0  C7              BNE CR1       ; NO NEXT RESULT
978  B6B3  60                  RTS
979
980
981  B6B4              BLANK:              ; ROUT. TO COMPU. NO PARTI
982
983  B6B4  A0  05              LDY #$05
984  B6B6  A9  2D              LDA #$2D      ; FETCH ASCII '-'
985  B6B8  91  78              STA ($78),Y   ; STORE IN PRINT TABLE
986  B6BA  18                  CLC
987  B6BB  F8                  SED           ; COMPUTE IN DECIMAL MODE
988  B6BC  A9  01              LDA #$01
989  B6BE  65  83              ADC $83       ; ADD 1 TO RESULT REGISTER
990  B6C0  85  83              STA $83       ; UPDATE TOTAL RESULT
991  B6C2  D8                  CLD           ; CLR DECIMAL MODE
992  B6C3  4C  9C  B6          JMP CR6       ; RETURN TO MAIN ROUTINE
993
994  B6C6              NO:                 ; ROUT. TO COMPU. TOL. 'NO'
995
996  B6C6  A0  05              LDY #$05
997  B6C8  A9  4E              LDA #$4E      ; FETCH ASCII 'N'
998  B6CA  91  78              STA ($78),Y   ; STORE IN PRINT TABLE
999  B6CC  18                  CLC
1000 B6CD  F8                  SED           ; COMPUTE IN DECIMAL MODE
1001 B6CE  A9  01              LDA #$01
1003 B6D0  65  82              ADC $82       ; ADD 1 TO OLD RESULT
1004 B6D2  85  82              STA $82       ; UPDATE RESULT REGISTER
1005 B6D4  D8                  CLD           ; CLR DECIMAL MODE
1006 B6D5  4C  9C  B6          JMP CR6       ; RETURN TO MAIN ROUTINE
1008
1009 B6D8              ABSN:               ; COMPU. ROUT. (ABSTAIN VO)
1010
1011 B6D8  A0  05              LDY #$05
1012 B6DA  A9  41              LDA #$41      ; FETCH ASCII 'A'
1013 B6DC  91  78              STA ($78),Y   ; STORE IN PRINT TABLE
1014 B6DE  18                  CLC
1015 B6DF  F8                  SED           ; COMPUTE IN DECIMAL MODE
1016 B6E0  A9  01              LDA #$01
1017 B6E2  65  81              ADC $81       ; ADD 1 TO RESULT REGISTER
1018 B6E4  85  81              STA $81       ; UPDATE NEW RESULT
1019 B6E6  D8                  CLD           ; CLR DECIMAL MODE
1020 B6E7  4C  9C  B6          JMP CR6       ; RETURN TO MAIN ROUTINE
```

```
1021
1022  B6EA                        YES:                ; COMPU. ROUT. (TOL 'YES')
1023
1024  B6EA  A0  05                      LDY #$05
1025  B6EC  A9  59                      LDA #$59        ; FETCH ASCII 'Y'
1026  B6EE  91  78                      STA ($78),Y     ; STORE IN PRINT TABLE
1027  B6F0  18                          CLC
1028  B6F1  F8                          SED             ; COMPUTE IN DECIMAL MODE
1029  B6F2  A9  01                      LDA #$01
1030  B6F4  65  80                      ADC $80         ; ADD 1 TO RESULT REGISTER
1031  B6F6  85  80                      STA $80         ; UPDATE NEW RESULT
1032  B6F8  D8                          CLD             ; CLR DECIMAL MODE
1033  B6F9  4C  9C  B6                  JMP CR6         ; RETURN TO MAIN ROUTINE
1034
1035                              ;;;
```

```
1099                                    ;;;
```

```
1177  8800  A5  61          LDA $61
1178  8802  C9  00          CMP #$00      ; ALL VOTERS DONE ?
1179  8804  D0  EA          BNE NXVTR     ; NO, NEXT VOTER
1180
1181  8806  A2  30          LDX #$30
1182  8808  C6  4E    LP3:  DEC $4E       ; POINT TO FIRST VOTER
1183  880A  C6  7A          DEC $7A
1184  880C  CA              DEX
1185  880D  D0  F9          BNE LP3
1186
1187  880F  A9  18          LDA #$18      ; LOAD VOTERS COUNTER
1188  8811  85  61          STA $61
1189  8813  20  74  BA HERE1: JSR INGN1   ; ADDR. VTRS, INFORM D-MODE
1190  8816  20  A7  BD      JSR TXRX      ; TX2 AND RX DISCN STATUS
1191  8819  20  A1  BA      JSR PCHK1     ; PERFORM VALIDITY TEST
1192  881C  20  B5  BA      JSR PCHK3
1193  881F  20  E1  BB      JSR GDAT      ; CHCK AND UPDATE D-STATUS
1194  8822  20  78  BD      JSR NOTIFY    ; ACKNOWLEDGE VOTER STATUS
1195  8825  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1196  8828  20  78  BD      JSR NOTIFY    ; ACKNOWLEDGE AGAIN
1197
1198  882B  A2  02    NXT1: LDX #$02
1199  882D  E6  4E    LP4:  INC $4E       ; POINT TO NEXT VOTER
1200  882F  E6  7A          INC $7A
1201  8831  E6  76          INC $76
1202  8833  CA              DEX
1203  8834  D0  F7          BNE LP4
1204
1205  8836  E6  4A          INC $4A
1206  8838  E6  76          INC $76
1207  883A  E6  4C          INC $4C
1208  883C  E6  78          INC $78
1209  883E  C6  61          DEC $61
1210  8840  A5  61          LDA $61
1211  8842  C9  00          CMP #$00      ; ALL VOTERS DONE ?
1212  8844  D0  CD          BNE HERE1     ; IF NOT, CHECK NEXT VOTER
1213
1214  8846  A2  FF  CSTAT:  LDX #$FF      ; INIT. STCK PTR AT END PG.
1215  8848  9A              TXS
1216
1217  8849  20  FC  B8      JSR STCHK     ; PERFORM PRESENCE CHECK
1218  884C  A9  0A          LDA #$0A      ; LOAD PRESENCE CHECK VALUE
1219  884E  85  80          STA $80       ; STORE IN TIMER
1220
1221  8850  20  CB  B8 REPT:  JSR IDPTR   ; INIT. INDIR. PTRS. VALUE
1222
1223  8853  A9  03          LDA #$03      ; LOAD 3 BYTES TO BE SENT
1224  8855  85  60          STA $60
1225  8857  A9  18          LDA #$18      ; LOAD VOTERS COUNTER
1226  8859  85  61          STA $61
1227
1228  885B  20  E1  BB HERE3: JSR GDAT    ; FETCH VOTER'S OLD D-STAT
1229  885E  20  78  BD      JSR NOTIFY    ; ACKNOWLEDGE VOTER
1230  8861  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1231  8864  20  A7  BD      JSR TXRX      ; CHECK VOTER'S NEW D-STAT
1232  8867  20  34  B8      JSR PCHK2     ; PERFORM VALIDITY TEST
1233  886A  20  B5  BA      JSR PCHK3
1234  886D  20  E1  BB      JSR GDAT      ; UPDATE NEW STATUS
1235  8870  20  78  BD      JSR NOTIFY    ; ACKNOWLEDGE VOTER'S STAT
1236  8873  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1237  8876  20  78  BD      JSR NOTIFY    ; ACKNOWLEDGE AGAIN
1238
1239  8879  A2  02    NXT2: LDX #$02
1240  887B  E6  4E    LP5:  INC $4E       ; POINT AT NEXT VOTER
1241  887D  E6  7A          INC $7A
1242  887F  E6  76          INC $76
1243  8881  CA              DEX
1244  8882  D0  F7          BNE LP5
1245
1246  8884  E6  4A          INC $4A
1247  8886  E6  4C          INC $4C
1248  8888  E6  76          INC $76
1249  888A  E6  78          INC $78
1250  888C  C6  61          DEC $61
1251  888E  A5  61          LDA $61
1252  8890  C9  00          CMP #$00      ; ALL VOTERS DONE ?
1253  8892  D0  C7          BNE HERE3     ; IF NOT, CHECK NEXT VOTER
1254
1255  8894  20  B7  B8 HERE2: JSR CLRLY   ; CLEAR DISPLAY
1256  8897  20  AD  BC      JSR BCHK1     ; DISPLAY VOTER'S Q/R STAT
```

```
1257  889A  20  E5  BC          JSR BCHK2
1258  889D  20  12  BC          JSR PRINT      ; PRINT VOTER'S REG. STAT.
1259
1260  88A0  A9  0A               LDA #$0A       ; LOAD KEYBOARD SRCH VALUE
1261  88A2  85  81               STA $81
1262  88A4  20  F8  E8          JSR $E8F8      ; SELECT KEYBOARD
1263  88A7  20  07  E9    KYB:  JSR $E907      ; SCAN KEYBOARD
1264  88AA  C9  59               CMP #$59       ; VOTING CMD GIVEN 'Y'?
1265  88AC  F0  19               BEQ VSEL       ; YES, PROCEED TO V-MODE
1266  88AE  C6  81               DEC $81
1267  88B0  A5  81               LDA $81
1268  88B2  C9  00               CMP #$00       ; SEARCH OVER ?
1269  88B4  D0  F1               BNE KYB        ; IF NOT, SEARCH AGAIN
1270
1271  88B6  C6  80               DEC $80
1272  88B8  A5  80               LDA $80
1273  88BA  C9  00               CMP #$00       ; TIME UPDATE PRESENCE ?
1274  88BC  F0  06               BEQ PDAT       ; IF YES, CHCK PRESENCE
1275
1276  88BE  A2  FF               LDX #$FF       ; INIT. STCK PTR. END PG
1277  88C0  9A                   TXS
1278  88C1  4C  50  88          JMP REPT       ; CONTINUE POLLING
1279  88C4  4C  46  88    PDAT: JMP CSTAT      ; PROCEED TO PRESENCE CK
1280
1281  88C7  4C  67  B2    VSEL: JMP VGM1       ; PROCEED TO VOTING MODE
1282  88CA  00                   BRK            ; TRAP POINT
1283
1284
1285  88CB                IDPTR:               ; INDIR. PTRS. LOAD ROUT
1286  88CB  A9  70               LDA #$70       ; LOAD INDIR. PTRS. VALU
1287  88CD  85  78               STA $78        ; STORE IN WORK SPACE
1288  88CF  A9  0D               LDA #$0D       ; NEXT VALUE
1289  88D1  85  4D               STA $4D
1290  88D3  85  7B               STA $7B
1291  88D5  A9  D0               LDA #$D0
1292  88D7  85  4C               STA $4C
1293  88D9  A9  00               LDA #$00
1294  88DB  85  4E               STA $4E
1295  88DD  A9  0C               LDA #$0C
1296  88DF  85  4F               STA $4F
1297  88E1  85  71               STA $71
1298  88E3  85  4B               STA $4B
1299  88E5  A9  9F               LDA #$9F
1300  88E7  85  7A               STA $7A
1301  88E9  A9  10               LDA #$10
1302  88EB  85  76               STA $76
1303  88ED  A9  40               LDA #$40
1304  88EF  85  4A               STA $4A
1305  88F1  A9  9F               LDA #$9F
1306  88F3  85  70               STA $70
1307  88F5  A9  0E               LDA #$0E
1308  88F7  85  77               STA $77
1309  88F9  85  79               STA $79
1310  88FB  60                   RTS
1311

                          ;;;
```

```
1332  B91F  A9  9F          LDA #$9F
1333  B921  85  7A          STA $7A
1334  B923  A9  90          LDA #$90
1335  B925  85  7C          STA $7C
1336  B927  A9  A0          LDA #$A0
1337  B929  85  7E          STA $7E
1338  B92B  A9  40          LDA #$40
1339  B92D  85  4A          STA $4A
1340  B92F  A9  01          LDA #$01
1341  B931  85  81          STA $81
1342
1343  B933  A4  81    SP2:  LDY $81
1344  B935  B1  4A          LDA ($4A),Y   ; FETCH OL  DISN-STATUS
1345  B937  20  D4  BB      JSR CMPBT     ; COMPUTE PARITY BIT
1346  B93A  85  82          STA $82       ; STORE STATUS IN TMP RG
1347
1348  B93C  98              TYA           ; TRANSFER P-BIT TO ACCU
1349  B93D  18              CLC
1350  B93E  26  82          ROL $82
1351  B940  6A              ROR A
1352  B941  26  82          ROL $82       ; FETCH P-BIT
1353  B943  38              SEC
1354  B944  26  82          ROL $82       ; FETCH STOP BIT '1'
1355  B946  A4  81          LDY $81
1356  B948  A5  82          LDA $82
1357  B94A  91  7E          STA ($7E),Y   ; STORE STAT. IN WRK REG.
1358  B94C  E6  81          INC $81
1359  B94E  A5  81          LDA $81
1360  B950  C9  19          CMP #$19      ; ALL VOTERS DONE ?
1361  B952  D0  DF          BNE SP2       ; NEXT VOTER IF NOT
1362
1363  B954  A9  05          LDA #$05      ; LOAD NUMBER OF VOTERS
1364  B956  85  A5          STA $A5       ; STORE IN COUNTER
1365  B958  A9  02          LDA #$02      ; LOAD 2 BYTES TO BE SENT
1366  B95A  85  60          STA $60       ; STORE IN COUNTER
1367  B95C  20  C4  B9      JSR SCRN      ; CHCK VOTERS PRESENCE
1368  B95F  60              RTS
1369
1370
1371  B960  A9  00    RLYON: LDA #$00     ; ROUTINE TO TURN ON RELAY
1372  B962  85  63          STA $63
1373  B964  A9  DB    RL1:  LDA #$DB      ; FETCH DATA CHIP2 ADDR.
1374  B966  85  41          STA $41       ; STORE IN TX REGISTER
1375  B968  A9  3B          LDA #$3B      ; NEXT BYTE
1376  B96A  85  42          STA $42       ; STORE IN TX REGISTER
1377  B96C  A4  63          LDY $63
1378  B96E  B1  7C          LDA ($7C),Y   ; FETCH PRESENCE STATUS
1379  B970  85  43          STA $43       ; STORE IN TX REGISTER
1380
1381  B972  A9  03          LDA #$03      ; LOAD 3 BYTES TO BE SENT
1382  B974  85  60          STA $60
1383  B976  20  78  BD      JSR NOTIFY    ; DISPLAY PRESENCE STATUS
1384  B979  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1385  B97C  20  78  BD      JSR NOTIFY    ; SEND AGAIN
1386  B97F  C6  60          DEC $60
1387  B981  A9  C7          LDA #$C7      ; FETCH CMD CHIP 2 ADDR.
1388  B983  85  41          STA $41       ; STORE IN TX REGISTER
1389  B985  A9  2B          LDA #$2B      ; NEXT BYTE
1390  B987  85  42          STA $42
1391  B989  20  78  BD      JSR NOTIFY    ; ADDRESS COMMAND RX.
1392  B98C  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1393  B98F  A9  2B          LDA #$2B
1394  B991  85  42          STA $42
1395  B993  A9  A1          LDA #$A1      ; FETCH CLOCK SIGNAL
1396  B995  85  43          STA $43       ; STORE IN TX REGISTER
1397  B997  E6  60          INC $60
1398  B999  20  78  BD      JSR NOTIFY    ; SHIFT DATA 1 POSITION
1399  B99C  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1400  B99F  A9  83          LDA #$83      ; FETCH CLK DISABLE SIGNAL
1401  B9A1  85  43          STA $43       ; STORE IN TX REGISTER
1402  B9A3  20  78  BD      JSR NOTIFY    ; DISABLE CLOCK SIGNAL
1403  B9A6  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1404  B9A9  20  78  BD      JSR NOTIFY    ; DISABLE AGAIN
1405
1406  B9AC  E6  7C          INC $7C       ; NEXT STATUS
1407  B9AE  E6  63          INC $63
1408  B9B0  A5  63          LDA $63
1409  B9B2  C9  05          CMP #$05      ; ALL PRESENCE STATUS SENT?
1410  B9B4  D0  AE          BNE RL1       ; FETCH NEXT P-STATUS IF NOT
1411
```

```
1412  B9B6  A9  91          LDA #$91      ; FETCH RELAY ACTI. SIG.
1413  B9B8  85  43          STA $43       ; STORE IN TX REGISTER
1414  B9BA  20  78  BD      JSR NOTIFY    ; DISPLAY PRESENC E STATE
1415  B9BD  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1416  B9C0  20  78  BD      JSR NOTIFY    ; MAKE SURE RELAY IS ON
1417  B9C3  60              RTS
1418
1419
1420  B9C4            SCRN:             ; CHK ROUT. OF EA VTR (IN 5)
1421  B9C4  A0  01    NXT5: LDY #$01
1422  B9C6  B1  4E          LDA ($4E),Y   ; FETCH VOTER'S (TX1) ADDR.
1423  B9C8  85  41          STA $41       ; STORE IN TX REGISTER
1424  B9CA  C8              INY
1425  B9CB  B1  4E          LDA ($4E),Y   ; NEXT BYTE
1426  B9CD  85  42          STA $42
1427  B9CF  20  78  BD      JSR NOTIFY    ; ADDRESS VOTER
1428  B9D2  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1429
1430  B9D5  A0  01          LDY #$01
1431  B9D7  B1  7A          LDA ($7A),Y   ; FETCH TX2 ADDRESS
1432  B9D9  85  41          STA $41       ; STORE IN TX REGISTER
1433  B9DB  B1  7E          LDA ($7E),Y   ; FETCH OLD P-STATUS
1434  B9DD  85  43          STA $43       ; STORE IN TX REGISTER
1435  B9DF  C8              INY
1436  B9E0  B1  7A          LDA ($7A),Y   ; FETCH NEXT ADDRESS BYTE
1437  B9E2  85  42          STA $42
1438
1439  B9E4  E6  60          INC $60       ; LOAD 3 BYTES TO BE SENT
1440  B9E6  20  A7  BD      JSR TXRX      ; TX AND CHK FOR RESPOND
1441  B9E9  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1442  B9EC  A5  51          LDA $51       ; LOAD PRESENT REG. CONTENT
1443  B9EE  C9  00          CMP #$00      ; VOTER PRESENT ?
1444  B9F0  F0  7C          BEQ ABSNT     ; NO, HE'S ABSENT
1445  B9F2  38              SEC           ; YES, MARK PRESENT
1446  B9F3  26  A0          ROL $A0
1447
1448  B9F5  E6  4E    NXTV: INC $4E       ; NEXT VOTER
1449  B9F7  E6  4E          INC $4E
1450  B9F9  E6  7A          INC $7A
1451  B9FB  E6  7A          INC $7A
1452  B9FD  E6  7E          INC $7E
1453  B9FF  C6  A5          DEC $A5
1454  BA01  A5  A5          LDA $A5
1455  BA03  C9  00          CMP #$00      ; FIVE VOTERS DONE ?
1456  BA05  F0  07          BEQ LODMM     ; IF YES, STORE P-STAT. TBL
1457  BA07  A9  02          LDA #$02      ; LOAD 2 BYTES TO BE SENT
1458  BA09  85  60          STA $60
1459  BA0B  4C  C4  B9      JMP NXT5      ; CHECK THE NEXT FIVE VOTER
1460
1461
1462  BA0E  A6  61    LODMM: LDX $61      ; STORE ROU. FOR 5 VTR STAT
1463  BA10  A5  A0          LDA $A0       ; LOAD P-REGISTER CONTENT
1464  BA12  95  A1          STA $00A1,X   ; STORE P-STAT. IN WRK REG.
1465  BA14  E6  61          INC $61
1466  BA16  A5  61          LDA $61
1467  BA18  C9  05          CMP #$05      ; ALL VOTERS DONE ?
1468  BA1A  F0  0F          BEQ UPDAT     ; IF YES, UPDATE P-TABLE
1469  BA1C  A9  00          LDA #$00      ; CLEAR P-WRK-REGISTER
1470  BA1E  85  A0          STA $A0
1471  BA20  A9  05          LDA #$05      ; LOAD NUMBRE OF VOTERS
1472  BA22  85  A5          STA $A5       ; STORE IN COUNTER
1473  BA24  A9  02          LDA #$02      ; LOAD 2 BYTES TO BE SENT
1474  BA26  85  60          STA $60
1475  BA28  4C  C4  B9      JMP NXT5       ; CHK NEXT FIVE
1476
1477
1478  BA2B  A2  00    UPDAT: LDX #$00     ; ROU. TO LOAD P-STAT. TABL
1479  BA2D  BD  90  0E UP1:  LDA $0E90,X   ; ARRG. STAT. IN MSB POSN.
1480  BA30  4A              LSR A
1481  BA31  4A              LSR A
1482  BA32  4A              LSR A
1483  BA33  95  A6          STA $00A6,X   ; STORE IN WRK-REGISTER
1484  BA35  E8              INX
1485  BA36  E0  05          CPX #$05      ; ALL ARRANGEMENT DONE ?
1486  BA38  D0  F3          BNE UP1       ; IF NO, CONTINUE
1487
1488  BA3A  A2  00          LDX #$00
1489  BA3C  B5  A1    UP2:  LDA $00A1,X   ; FETCH NEW P-STATUS
1490  BA3E  D5  A6          CMP $00A6,X   ; CHANGE IN P-STATUS ?
1491  BA40  D0  06          BNE STRBL     ; IF YES, UPDATE P-STAT-TBL
```

```
1492
1493  BA42  E8                        INX
1494  BA43  E0  05                    CPX #$05        ; ALL VOTERS P-STAT. CHCK?
1495  BA45  D0  F5                    BNE UP2         ; IF NO, CONTINUE CHECK
1496  BA47  60                        RTS
1497
1498
1499  BA48  A9  00          STRBL:    LDA #$00        ; UPDATE P-STATUS ROUTINE
1500  BA4A  85  63                    STA $63
1501  BA4C  A6  63          ST1:      LDX $63
1502  BA4E  B5  A1                    LDA $00A1,X     ; FETCH NEW P-STATUS
1503  BA50  20  D4  BB                JSR CMPBT       ; COMPUTE PARITY BIT
1504  BA53  18                        CLC
1505  BA54  2A                        ROL A
1506  BA55  A6  63                    LDX $63
1507  BA57  9D  90  0E                STA $0E90,X     ; STORE P-STATUS IN TBL
1508
1509  BA5A  98                        TYA
1510  BA5B  6A                        ROR A
1511  BA5C  3E  90  0E                ROL $0E90,X     ; FETCH PARITY BIT
1512  BA5F  38                        SEC             ; FETCH STOP BIT '1'
1513  BA60  3E  90  0E                ROL $0E90,X     ; STORE IN P-STATUS TABLE
1514  BA63  E6  63                    INC $63
1515  BA65  A5  63                    LDA $63
1516  BA67  C9  05                    CMP #$05        ; UPDATING DONE ?
1517  BA69  D0  E1                    BNE ST1         ; IF NO, UPDATE NEXT 5
1518  BA6B  4C  60  B9                JMP RLYON       ; PROCEED TO TURN ON DISP.
1519
1520
1521  BA6E                  ABSNT:                    ; INACTIVE VOTER ROUTINE
1522  BA6E  18                        CLC
1523  BA6F  26  A0                    ROL $A0         ; MARK VOTER ABSENT
1524  BA71  4C  F5  B9                JMP NXTV        ; PROCEED TO NEXT VOTER
1525
1526
1527  BA74                  INGN1:                    ; ROUT. TO FETCH TX1 ADDR.
1528  BA74  20  92  BA                JSR GTADD       ; FETCH TX1 ADDRESS
1529  BA77  20  78  BD                JSR NOTIFY      ; ADDRESS VOTER
1530  BA7A  20  2C  ED                JSR $ED2C       ; DELAY 5 MS
1531  BA7D  20  87  BA                JSR GRN2        ; FETCH TX2 ADDRESS
1532  BA80  A9  01                    LDA #$01        ; FETCH 3RD BYTE
1533  BA82  85  43                    STA $43         ; STORE IN TX REGISTER
1534  BA84  E6  60                    INC $60         ; LOAD 3 BYTES TO BE SENT
1535  BA86  60                        RTS             ; RETURN TO MAIN ROUTINE
1536
1537
1538  BA87                  GRN2:                     ; TX2 ADDR. FETCH ROUTINE
1539  BA87  A0  02                    LDY #$02
1540  BA89  B1  7A          GR1:      LDA ($7A),Y     ; LOAD ADDR. BYTES
1541  BA8B  99  40  00                STA $0040,Y     ; STORE IN TX REGISTER
1542  BA8E  88                        DEY
1543  BA8F  D0  F8                    BNE GR1         ; NXT BYTE IF NOT DONE
1544  BA91  60                        RTS
1545
1546
1547  BA92                  GTADD:                    ; TX1 ADDR. FETCH ROUTINE
1548  BA92  A0  02                    LDY #$02
1549  BA94  B1  4E          GT1:      LDA ($4E),Y     ; LOAD ADDRESS BYTE
1550  BA96  99  40  00                STA $0040,Y     ; STORE IN TX REGISTER
1551  BA99  88                        DEY
1552  BA9A  D0  F8                    BNE GT1         ; NEXT BYTE IF NOT DONE
1553  BA9C  A9  02                    LDA #$02        ; LOAD 2 BYTES TO BE SENT
1554  BA9E  85  60                    STA $60
1555  BAA0  60                        RTS
1556
1557
1558  BAA1                  PCHK1:                    ; PARITY TEST ROUTINE
1559  BAA1  A5  51                    LDA $51
1560  BAA3  C9  00                    CMP #$00        ; IS VOTER PRESENT?
1561  BAA5  F0  56                    BEQ NXSTN       ; NO, PROCEED TO NXT VOTER
1562
1563  BAA7  A5  50                    LDA $50         ; YES, LOAD DATA BYTE 1
1564  BAA9  20  D4  BB                JSR CMPBT       ; COMPUTE PARITY BIT
1565  BAAC  98                        TYA             ; TRANSFER P-BIT TO ACCU.
1566  BAAD  18                        CLC
1567  BAAE  6A                        ROR A
1568  BAAF  A9  00                    LDA #$00
1569  BAB1  2A                        ROL A
1570  BAB2  85  5A                    STA $5A         ; STORE P-BIT IN WRK REG.
1571  BAB4  60                        RTS
```

```
1572
1573
1574  BA85                          PCHK3:              ; VALIDITY TEST ROUTINE
1575
1576  BA85  A5  51                        LDA $51       ; FETCH TRANSMITTED P-BIT
1577  BA87  18                            CLC
1578  BA88  4A                            LSR A
1579  BA89  A9  00                        LDA #$00
1580  BA8B  2A                            ROL A
1581  BA8C  C5  5A                        CMP $5A       ; PARITY BIT VALID ?
1582  BA8E  D0  40                        BNE BERR3     ; IF NO, DSP ERR, HALT OPR
1583
1584  BAC0  A5  52                        LDA $52       ; FETCH NXT DATA BYTE
1585  BAC2  20  D4  BB                    JSR CMPBT     ; COMPUTE PARITY BIT
1586  BAC5  98                            TYA           ; TRANSFER P-BIT TO ACCU.
1587  BAC6  18                            CLC
1588  BAC7  6A                            ROR A
1589  BAC8  A9  00                        LDA #$00
1590  BACA  2A                            ROL A
1591  BACB  85  5B                        STA $5B       ; STORE P-BIT IN WRK REG.
1592
1593  BACD  A5  53                        LDA $53       ; FETCH TRANSMITTED P-BIT
1594  BACF  18                            CLC
1595  BAD0  4A                            LSR A
1596  BAD1  A9  00                        LDA #$00
1597  BAD3  2A                            ROL A
1598  BAD4  C5  5B                        CMP $5B       ; P-BIT TEST VALID?
1599  BAD6  D0  42                        BNE BERR4     ; IF NOT, DSP ERR,HALT OPR
1600
1601  BAD8  A5  52                        LDA $52       ; LOAD VOTER'S DIS-STATUS
1602  BADA  85  49                        STA $49       ; STORE IN WRK-REGISTER
1603  BADC  46  49                        LSR $49       ; DISCARD CI BIT
1604  BADE  18                            CLC
1605  BADF  A9  00                        LDA #$00
1606  BAE1  A2  03                        LDX #$03
1607  BAE3  66  49              PC1:      ROR $49       ; POSITION D-STAT. BIT LSB
1608  BAE5  2A                            ROL A
1609  BAE6  CA                            DEX
1610  BAE7  D0  FA                        BNE PC1
1611  BAE9  85  49                        STA $49       ; STORE IN WRK-REGISTER
1612
1613  BAEB  20  D4  BB                    JSR CMPBT     ; TEST FOR VALID BUTTON
1614  BAEE  C0  01                        CPY #$01      ; 1 BUTTON PUSHED ?
1615  BAF0  D0  59                        BNE NVALD     ; IF NO, GO TO INVALID ROU.
1616  BAF2  18                            CLC
1617  BAF3  6A                            ROR A         ; REQ. BUTTON PUSHED ?
1618  BAF4  B0  64                        BCS RQST      ; IF YES, UPDATE D-STATUS
1619  BAF6  6A                  BK4:      ROR A         ; CANCEL BUTTON PUSHED ?
1620  BAF7  B0  70                        BCS CANCL     ; IF YES, UPDATE D-STATUS
1621  BAF9  6A                  BK3:      ROR A         ; QUES. BUTTON PUSHED ?
1622  BAFA  B0  7E                        BCS QSTN      ; IF YES, UPDATE D-STATUS
1623  BAFC  60                  BK1:      RTS           ; RETURN TO MAIN ROUTINE
1624
1625
1626
1627  BAFD                      NXSTN:                  ; LINK ROUTINE
1628  BAFD  4C  2B  B8                    JMP NXT1      ; PROCEED TO NXT STATION
1629
1630
1631  BB00                      BERR3:                  ; ERR MSAGE AND HALT ROUT.
1632
1633  BB00  20  13  EA                    JSR $EA13     ; CLR DISPLAY
1634  BB03  A9  45                        LDA #$45
1635  BB05  20  05  EF                    JSR $EF05     ; DISPALY ASCII 'E'
1636  BB08  A9  52                        LDA #$52
1637  BB0A  20  05  EF                    JSR $EF05     ; DISPLAY ASCII 'R'
1638  BB0D  A9  52                        LDA #$52
1639  BB0F  20  05  EF                    JSR $EF05     ; DISPLAY ASCII 'R'
1640  BB12  A9  33                        LDA #$33
1641  BB14  20  05  EF                    JSR $EF05     ; DISPLAY ASCII '3'
1642  BB17  4C  17  BB            ERR3:   JMP ERR3      ; HALT OPERATION
1643
1644
1645  BB1A                      BERR4:                  ; ERR MSAGE, HALT OPRN ROU
1646  BB1A  20  13  EA                    JSR $EA13     ; CLEAR DISPLAY
1647  BB1D  A9  45                        LDA #$45
1648  BB1F  20  05  EF                    JSR $EF05     ; DISPLAY ASCII 'E'
1649  BB22  A9  52                        LDA #$52
1650  BB24  20  05  EF                    JSR $EF05     ; DISPLAY ASCII 'R'
1651  BB27  A9  52                        LDA #$52
```

```
1652  8829  20  05  EF          JSR $EF05     ; DISPLAY ASCII 'R'
1653  882C  A9  34             LDA #$34
1654  882E  20  05  EF          JSR $EF05     ; DISPLAY ASCII '4'
1655  8831  4C  31  88  ERR4:   JMP ERR4      ; HALT OPERATION
1656
1657  8834                    PCHK2:          ; VTR PRESENT TSTS ROUTINE
1658  8834  A5  51              LDA $51       ; LOAD P-REGISTER CONTENT
1659  8836  C9  00              CMP #$00      ; ACTIVE STATION ?
1660  8838  F0  0E              BEQ NXSTM     ; IF NOT, CHK NEXT STATION
1661
1662  883A  A5  50              LDA $50       ; LOAD RECEIVED DATA BYTE
1663  883C  20  D4  88          JSR CMPBT     ; COMPUTE PARITY BIT
1664  883F  98                  TYA           ; TRANSFER P-BIT TO ACCU.
1665  8840  18                  CLC
1666  8841  6A                  ROR A
1667  8842  A9  00              LDA #$00
1668  8844  2A                  ROL A
1669  8845  85  5A              STA $5A       ; STORE P-BIT IN WRK-REG
1670  8847  60                  RTS
1671
1672
1673  8848                    NXSTM:          ; LINK ROUTINE
1674  8848  4C  79  88          JMP NXT2      ; PROCEED TO NEXT VOTER
1675
1676
1677  884B                    NVALD:          ; INVALID BUTTON SEL ROUT.
1678  884B  C0  00              CPY #$00      ; NO BUTTON PUSHED ?
1679  884D  F0  3A              BEQ ORAMM     ; IF YES, GO TO CHK ROUTIN
1680  884F  A0  01              LDY #$01      ; NO, 2 OR MOR BUT. PUSHED
1681  8851  A9  00              LDA #$00
1682  8853  91  4C              STA ($4C),Y   ; CLR OLD D-STATUS
1683  8855  91  4A              STA ($4A),Y
1684  8857  4C  FC  8A          JMP BK1       ; RETURN TO MAIN ROUTINE
1685
1686
1687  885A                    RQST:           ; REQ. STATUS UPDATE ROUT.
1688  885A  A0  01              LDY #$01
1689  885C  A5  49              LDA $49       ; FETCH REQUEST STATUS
1690  885E  91  4C              STA ($4C),Y   ; STORE IN TEMP. WRK-SPACE
1691  8860  11  4A              ORA ($4A),Y   ; PERFORM 'OR' OPERATION
1692  8862  D1  4A              CMP ($4A),Y   ; CHANGE IN NEW D-STATUS ?
1693  8864  D0  30              BNE UDAT1     ; IF YES, UPDATE D-STATUS
1694  8866  4C  F6  8A          JMP BK4       ; RETURN TO MAIN ROUTINE
1695
1696
1697  8869                    CANCL:          ; ROUT. TO CANCL Q/R SIGNL
1698  8869  A0  01              LDY #$01
1699  886B  A9  00              LDA #$00
1700  886D  91  4C              STA ($4C),Y   ; CLR TEMP. WRK-SPACE
1701  886F  A5  49              LDA $49       ; LOAD NEW D-STATUS
1702  8871  11  4A              ORA ($4A),Y   ; PERFORM 'OR' OPERATION
1703  8873  D1  4A              CMP ($4A),Y   ; CHANGE IN NEW D-STATUS ?
1704  8875  D0  2C              BNE UDAT2     ; IF YES, UPDATE NEW STAT.
1705  8877  4C  F9  8A          JMP BK3       ; RETURN TO MAIN ROUTINE
1706
1707
1708  887A                    QSTN:           ; QUES. STAT. UPDATE ROUT.
1709  887A  A0  01              LDY #$01
1710  887C  A5  49              LDA $49       ; LOAD NEW D-STATUS
1711  887E  91  4C              STA ($4C),Y   ; STORE IN TEMP WRK-SPACE
1712  8880  11  4A              ORA ($4A),Y   ; PERFORM "OR' OPERATION
1713  8882  D1  4A              CMP ($4A),Y   ; CHANGE IN NEW D-STATUS ?
1714  8884  D0  26              BNE UDAT3     ; IF YES, UPDATE NEW STAT.
1715  8886  4C  FC  8A          JMP BK1       ; RETURN TO MAIN ROUTINE
1716
1717
1718  8889                    ORAMM:          ; INACTI. BUTTON UPDATE ROU.
1719  8889  A0  01              LDY #$01
1720  888B  A5  49              LDA $49       ; LOAD D-STATUS
1721  888D  11  4A              ORA ($4A),Y   ; PERFORM 'OR' OPERATION
1722  888F  D1  4A              CMP ($4A),Y   ; CHANGE IN NEW D-STATUS
1723  8891  D0  03              BNE UDAT1     ; IF YES, UPDATE D-STATUS
1724  8893  4C  FC  8A          JMP BK1       ; RETURN TO MAIN ROUTINE
1725
1726
1727  8896                    UDAT1:          ; REQ. STAT.UPDATE ROUTINE
1728  8896  A0  01              LDY #$01
1729  8898  A5  49              LDA $49       ; LOAD RQST STATUS
1730  889A  91  4A              STA ($4A),Y   ; STORE IN D-STATUS TABLE
1731  889C  91  4C              STA ($4C),Y   ; STORE IN TEMP (DSP) SPACE
```

```
1732  BB9E  91  78          STA ($78),Y   ; STORE IN TEMP (PRTR) SPACE
1733  BBA0  4C  FC  BA      JMP BK1
1734
1735
1736  BBA3                  UDAT2:        ; CANCL STATUS UPDATE ROUT
1737  BBA3  A0  01          LDY #$01
1738  BBA5  A5  49          LDA $49       ; LOAD D-STATUS
1739  BBA7  91  4A          STA ($4A),Y   ; STORE IN D-STATUS TABLE
1740  BBA9  4C  FC  BA      JMP BK1
1741
1742
1743  BBAC                  UDAT3:        ; QUES. STAT. UPDATE ROUT.
1744  BBAC  A0  01          LDY #$01
1745  BBAE  A5  49          LDA $49       ; LOAD D-STATUS
1746  BBB0  91  4A          STA ($4A),Y   ; STORE IN D-STATUS TABLE
1747  BBB2  91  4C          STA ($4C),Y   ; STORE IN TEMP (DSP) SPACE
1748  BBB4  4C  FC  BA      JMP BK1
1749
1750
1751  BBB7                  CLRLY:        ; ROUT. TO BLINK DSP LAMPS
1752  BBB7  A9  D3          LDA #$D3      ; FETCH TX1 CMD CHIP 1 ADDR
1753  BBB9  85  41          STA $41       ; STORE IN TX REGISTER
1754  BBBB  A9  2B          LDA #$2B      ; NEXT BYTE
1755  BBBD  85  42          STA $42
1756  BBBF  A9  01          LDA #$01      ; FETCH DEACTIVATION SIGNL
1757  BBC1  85  43          STA $43       ; STORE IN TX REGISTER
1758  BBC3  A9  03          LDA #$03      ; LOAD 3 BYTES TO BE SENT
1759  BBC5  85  60          STA $60
1760  BBC7  20  78  BD      JSR NOTIFY    ; DEACTI. RELAY (LAMPS-OFF)
1761  BBCA  20  2C  ED      JSR $ED2C     ; DELAY 5 MS
1762  BBCD  20  78  BD      JSR NOTIFY    ; MAKE SURE
1763  BBD0  20  2C  ED      JSR $ED2C
1764  BBD3  60              RTS
1765
1766
1767  BBD4                  CMPBT:        ; ROUT. TO COMPUTE PAR-BIT
1768  BBD4  A0  00          LDY #$00      ; CLR PARITY BIT COUNTER
1769  BBD6  A2  08          LDX #$08      ; LOAD BIT COUNTER VALUE
1770  BBD8  2A              CKBT0: ROL A  ; BIT '0' ?
1771  BBD9  90  01          BCC CKBT1     ; IF YES, TEST NEXT BIT
1772  BBDB  C8              INY           ; BIT IS '1', INCR. COUNT
1773  BBDC  CA              CKBT1: DEX    ; DECR. CNTR. ALL BITS DONE?
1774  BBDD  D0  F9          BNE CKBT0     ; IF NOT, CONTINUE COMPUT.
1775  BBDF  2A              ROL A
1776  BBE0  60              RTS           ; RETURN TO MAIN ROUTINE
1777
1778
1779  BBE1                  GDAT:         ; D-STAT N ADDR. ARRGM. ROU.
1780  BBE1  A0  01          LDY #$01
1781  BBE3  B1  7A          LDA ($7A),Y   ; FETCH TX2 ADDR BYTE
1782  BBE5  85  41          STA $41       ; STORE IN TX REGISTER
1783  BBE7  B1  4A          LDA ($4A),Y   ; FETCH VOTER D-STATUS
1784  BBE9  85  5C          STA $5C       ; TORE IN WRK-REGISTER
1785  BBEB  C8              INY
1786  BBEC  B1  7A          LDA ($7A),Y   ; FETCH NXT ADDR. BYTE
1787  BBEE  85  42          STA $42       ; STORE IN TX REGISTER
1788  BBF0  A5  5C          LDA $5C       ; FETCH D-STATUS
1789  BBF2  20  D4  BB      JSR CMPBT     ; COMPUTE PARITY BIT
1790  BBF5  98              TYA           ; TRANSFER P-BIT TO ACCU.
1791  BBF6  6A              ROR A         ; ODD OR EVEN P-BIT?
1792  BBF7  B0  0D          BCS PBODD     ; IF ODD, P-BIT IS '1'
1793  BBF9  18              CLC           ; P-BIT IS '0'
1794  BBFA  26  5C          ROL $5C       ; FETCH CI BIT
1795  BBFC  26  5C          ROL $5C       ; FETCH P-BIT
1796  BBFE  38              SEC           ; FETCH STOP BIT '1'
1797  BBFF  26  5C          ROL $5C
1798  BC01  A5  5C          RTN:  LDA $5C ; LOAD D-STATUS
1799  BC03  85  43          STA $43       ; STORE IN TX REGISTER
1800  BC05  60              RTS
1801
1802
1803  BC06                  PBODD:        ; P-BIT '1' SORT ROUTINE
1804  BC06  18              CLC
1805  BC07  26  5C          ROL $5C       ; FETCH CI BIT
1806  BC09  38              SEC           ; FETCH P-BIT '1'
1807  BC0A  26  5C          ROL $5C
1808  BC0C  38              SEC           ; FETCH STOP BIT '1'
1809  BC0D  26  5C          ROL $5C
1810  BC0F  4C  01  BC      JMP RTN
1811
```

```
1812
1813  BC12                        PRINT:              ; REQ-STAT. PRINT ROUTINE
1814  BC12  A9  0E                       LDA #$0E      ; LOAD INDIR. PTRS VALUE
1815  BC14  85  77                       STA $77       ; STORE IN WORK REGISTER
1816  BC16  85  79                       STA $79
1817  BC18  A9  10                       LDA #$10      ; NEXT VALUE
1818  BC1A  85  76                       STA $76
1819  BC1C  A9  70                       LDA #$70
1820  BC1E  85  78                       STA $78
1821  BC20  A9  0D                       LDA #$0D
1822  BC22  85  4D                       STA $4D
1823  BC24  A9  D0                       LDA #$D0
1824  BC26  85  4C                       STA $4C
1825
1826  BC28  20  44  EB                    JSR $EB44     ; CLR PRINTER POINTER
1827
1828  BC2B  A9  18                       LDA #$18      ; LOAD VOTERS COUNTER
1829  BC2D  85  64                       STA $64
1830
1831  BC2F  A0  01              NXPRT:   LDY #$01
1832  BC31  B1  78                       LDA ($78),Y   ; FETCH VTRS REG. STATUS
1833  BC33  6A                          ROR A         ; REQUEST MADE ?
1834  BC34  90  2B                       BCC PASS      ; IF NOT, CHCK NEXT VOTER
1835  BC36  20  01  E9                   JSR $E901     ; ACTIVATE PRINTER
1836  BC39  A9  52                       LDA #$52      ; FETCH ASCII 'R'
1837  BC3B  20  00  F0                   JSR $F000     ; STORE IN PRINTER BUFFER
1838  BC3E  20  3E  E8                   JSR $E83E     ; SKIP A SPACE
1839  BC41  A2  03                       LDX #$03
1840  BC43  A0  01                       LDY #$01
1841  BC45  B1  76              PRT1:    LDA ($76),Y   ; FETCH VOTER'S INITIALS
1842  BC47  20  00  F0                   JSR $F000     ; STORE IN PRINTER BUFFER
1843  BC4A  C8                          INY           ; NEXT INITIAL
1844  BC4B  CA                          DEX
1845  BC4C  D0  F7                       BNE PRT1      ; IF NOT DONE, NEXT INITIAL
1846
1847  BC4E  A9  0D                       LDA #$0D      ; FETCH PRINTER ACTI. SIG.
1848  BC50  20  00  F0                   JSR $F000     ; OUTPUT BUFFER CONTENT
1849  BC53  20  7C  BC                   JSR CDIS      ; UPDATE PREVIOUS DSP STAT.
1850  BC56  20  44  EB                   JSR $EB44     ; CLR PRINTER POINTER
1851
1852  BC59  A2  64                       LDX #$64
1853  BC5B  20  2C  ED          PDLY:    JSR $ED2C     ; DELAY 0.5 SECOND
1854  BC5E  CA                          DEX           ; NEXT COUNT
1855  BC5F  D0  FA                       BNE PDLY      ; DELAY IF NOT TIME-UP
1856  BC61  E6  78              PASS:    INC $78       ; NEXT VOTER
1857  BC63  E6  76                       INC $76
1858  BC65  E6  76                       INC $76
1859  BC67  E6  76                       INC $76
1860  BC69  C6  64                       DEC $64
1861  BC6B  A5  64                       LDA $64
1862  BC6D  C9  00                       CMP #$00      ; ALL VOTERS DONE ?
1863  BC6F  D0  BE                       BNE NXPRT     ; IF NOT, CHCK NEXT VOTER
1864
1865  BC71  A2  18                       LDX #$18
1866  BC73  A9  00                       LDA #$00
1867  BC75  9D  70  0E          PRT2:    STA $0E70,X   ; CLEAR OLD RQST-STATUS
1868  BC78  CA                          DEX
1869  BC79  D0  FA                       BNE PRT2      ; CONTINUE IF NOT DONE
1870  BC7B  60                          RTS
1871
1872
1873  BC7C                        CDIS:               ; DISPLAY REFRESH ROUTINE
1874  BC7C  20  44  EB                    JSR $EB44     ; CLEAR DISPLAY POINTER
1875  BC7F  A9  44                       LDA #$44
1876  BC81  20  05  EF                   JSR $EF05     ; DISPLAY ASCII 'D'
1877  BC84  A9  2D                       LDA #$2D
1878  BC86  20  05  EF                   JSR $EF05     ; DISPLAY ASCII '-'
1879  BC89  A9  4D                       LDA #$4D
1880  BC8B  20  05  EF                   JSR $EF05     ; DISPLAY ASCII 'M'
1881  BC8E  A9  4F                       LDA #$4F
1882  BC90  20  05  EF                   JSR $EF05     ; DISPLAY ASCII 'O'
1883  BC93  A9  44                       LDA #$44
1884  BC95  20  05  EF                   JSR $EF05     ; DISPLAY ASCII 'D'
1885  BC98  A9  45                       LDA #$45
1886  BC9A  20  05  EF                   JSR $EF05     ; DISPLAY ASCII 'E'
1887  BC9D  A9  21                       LDA #$21
1888  BC9F  20  05  EF                   JSR $EF05     ; DISPLAY ASCII '!'
1889  BCA2  A9  56                       LDA #$56
1890  BCA4  20  05  EF                   JSR $EF05     ; DISPLAY ASCII 'V'
1891  BCA7  A9  3F                       LDA #$3F
```

```
1892  BCA9  20  05  EF          JSR $EF05      ; DISPLAY ASCII '?'
1893  BCAC  60                  RTS
1894
1895  BCAD                BCHK1:             ; G/R STAT-PAIR SORT ROUT.
1896  BCAD  A9  0D              LDA #$0D       ; LOAD INDIR. PTR VALUE
1897  BCAF  85  4D              STA $4D        ; STORE IN WRK-REGISTER
1898  BCB1  A9  D0              LDA #$D0       ; NEXT VALUE
1899  BCB3  85  4C              STA $4C
1900  BCB5  A9  0C              LDA #$0C
1901  BCB7  85  71              STA $71
1902  BCB9  85  64              STA $64
1903  BCBB  A9  9F              LDA #$9F
1904  BCBD  85  70              STA $70
1905  BCBF  A0  01        BT1:  LDY #$01
1906  BCC1  B1  4C              LDA ($4C),Y    ; FETCH D-STATUS
1907  BCC3  85  65              STA $65        ; STORE IN WRK-REGISTER
1908  BCC5  C8                  INY
1909  BCC6  B1  4C              LDA ($4C),Y    ; FETCH NXT VOTER'S D-STAT.
1910  BCC8  18                  CLC
1911  BCC9  66  65              ROR $65
1912  BCCB  A2  03              LDX #$03
1913  BCCD  2A            BT2:  ROL A          ; SHIFT D-STAT. INTO 1 BYTE
1914  BCCE  66  65              ROR $65
1915  BCD0  CA                  DEX
1916  BCD1  D0  FA              BNE BT2        ; CONTINUE IF NOT DONE
1917
1918  BCD3  88                  DEY
1919  BCD4  91  70              STA ($70),Y    ; STORE STAT-PAIR IN TABLE
1920  BCD6  E6  4C              INC $4C        ; NEXT PAIR
1921  BCD8  E6  4C              INC $4C
1922  BCDA  E6  70              INC $70
1923  BCDC  C6  64              DEC $64
1924  BCDE  A5  64              LDA $64
1925  BCE0  C9  00              CMP #$00       ; OPERATION DONE ?
1926  BCE2  D0  DB              BNE BT1        ; FETCH NXT PAIR IF NOT
1927  BCE4  60                  RTS
1928
1929
1930  BCE5                BCHK2:             ; G/R STAT. DSP SORT ROUT.
1931  BCE5  A9  0C              LDA #$0C       ; LOAD INDIR. PTRS VALUE
1932  BCE7  85  71              STA $71        ; STORE IN WRK-REGISTER
1933  BCE9  A9  9F              LDA #$9F       ; NEXT VALUE
1934  BCEB  85  70              STA $70
1935  BCED  A9  0C              LDA #$0C
1936  BCEF  85  64              STA $64
1937  BCF1  A0  01        DSP1: LDY #$01
1938  BCF3  B1  70              LDA ($70),Y    ; FETCH G/R STATUS PAIR
1939  BCF5  20  D4  BB          JSR CMPBT      ; COMPUTE PARITY BIT
1940  BCF8  85  66              STA $66        ; STORE STAT. IN WRK REG.
1941  BCFA  98                  TYA            ; TRANSFER P-BIT TO ACCU.
1942  BCFB  18                  CLC
1943  BCFC  26  66              ROL $66        ; FETCH CI BIT
1944  BCFE  4A                  LSR A
1945  BCFF  26  66              ROL $66         ; FETCH P-BIT
1946  BD01  A0  02              LDY #$02
1947  BD03  B9  90  0C          LDA $0C90,Y    ; FETCH TX1 DATA CHP1 ADDR.
1948  BD06  4A                  LSR A
1949  BD07  38                  SEC            ; FETCH STOP BIT '1'
1950  BD08  26  66              ROL $66
1951  BD0A  2A                  ROL A          ; SHIFT BIT S6 INTO 2ND BYTE
1952  BD0B  85  42              STA $42        ; STORE IN TX REGISTER
1953
1954  BD0D  A9  03              LDA #$03       ; LOAD 3 BYTES TO BE SENT
1955  BD0F  85  60              STA $60
1956  BD11  A0  01              LDY #$01
1957  BD13  B9  90  0C          LDA $0C90,Y    ; FETCH DATA CHIP 1 ADDR.
1958  BD16  85  41              STA $41        ; STORE IN TX REGISTER
1959  BD18  A5  66              LDA $66        ; FETCH G/R STATUS
1960  BD1A  85  43              STA $43        ; STORE IN TX REGISTER
1961  BD1C  20  78  BD          JSR NOTIFY     ; DISPLAY G/R STATUS
1962  BD1F  20  2C  ED          JSR $ED2C      ; DELAY 5 MS
1963  BD22  20  78  BD          JSR NOTIFY     ; SEND AGAIN
1964
1965  BD25  A0  01              LDY #$01
1966  BD27  B9  30  0C          LDA $0C30,Y    ; FETCH CMD CHP1 ADDR.(TX1)
1967  BD2A  85  41              STA $41        ; STORE IN TX REGISTER
1968  BD2C  C8                  INY
1969  BD2D  B9  30  0C          LDA $0C30,Y    ; NEXT BYTE
1970  BD30  85  42              STA $42
1971  BD32  C6  60              DEC $60        ; LOAD 2 BYTES TO BE SENT
```

```
1972  BD34  20  78  BD          JSR NOTIFY     ; ADDRESS CMD CHIP1
1973  BD37  20  2C  ED          JSR $ED2C      ; DELAY 5 MS
1974  BD3A  A0  03               LDY #$03
1975  BD3C  B9  90  0C          LDA $0C90,Y    ; FETCH TX2 CMD CHIP1 ADDR.
1976  BD3F  85  41               STA $41        ; STORE IN TX REGISTER
1977  BD41  C8                   INY
1978  BD42  B9  90  0C          LDA $0C90,Y    ; NEXT BYTE
1979  BD45  85  42               STA $42
1980  BD47  A9  0B               LDA #$0B       ; FETCH CLOCK SIGNAL
1981  BD49  85  43               STA $43        ; STORE IN TX REGISTER
1982  BD4B  E6  60               INC $60        ; LOAD 3 BYTES TO BE SENT
1983  BD4D  20  78  BD           JSR NOTIFY     ; SHIFT DSP DATA 1 POSITION
1984  BD50  20  2C  ED           JSR $ED2C      ; DELAY 5 MS
1985  BD53  A9  01               LDA #$01       ; FETCH CLOCK DISABLE SIG.
1986  BD55  85  43               STA $43        ; STORE IN TX REGISTER
1987
1988  BD57  20  78  BD           JSR NOTIFY     ; DISABLE CLOCK SIGNAL
1989  BD5A  20  2C  ED           JSR $ED2C      ; DELAY 5 MS
1990  BD5D  20  78  BD           JSR NOTIFY     ; DISABLE AGAIN
1991
1992  BD60  E6  70               INC $70        ; NEXT PAIR OF STATUS
1993  BD62  C6  64               DEC $64
1994  BD64  A5  64               LDA $64
1995  BD66  C9  00               CMP #$00       ; ALL VOTERS STATUS FETCH ?
1996  BD68  D0  87               BNE DSP1       ; CONTINUE IF NOT
1997
1998  BD6A  A9  83               LDA #$83       ; FETCH RELAY ACTI. SIGNAL
1999  BD6C  85  43               STA $43        ; STORE IN SEND REGISTER
2000  BD6E  20  78  BD           JSR NOTIFY     ; TURN ON DISPLAY
2001  BD71  20  2C  ED           JSR $ED2C      ; DELAY 5 MS
2002  BD74  20  78  BD           JSR NOTIFY     ; MAKE SURE
2003  BD77  60                   RTS
2004
2005
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ;
                                ; TRANSMISSION ROUTINE
                                ; (TWO (TX1) OR THREE (TX2) ADDRESS BYTES)
                                ;
2006  BD78                      NOTIFY:                ; 2/3 BYT TX ROUT. W/O RCPT
2008  BD78  78                         SEI            ; DISABLE INTERRUPT
2009  BD79  A6  60                      LDX $60        ; LOAD NUMBER BYTES FOR TX
2010  BD7B  A0  01                      LDY #$01       ; LOAD BYTE INDIR. POINTER
2011  BD7D  A9  00                      LDA #$00       ; DISABLE SHIFT REGISTER
2012  BD7F  8D  0B  A0                  STA $A00B
2013  BD82  A9  14               NFY1:  LDA #$14       ; SET SHIFT-OUT BY TIMER T2
2014  BD84  8D  0B  A0                  STA $A00B
2015
2016  BD87  A9  44                      LDA #$44       ; LOAD 140 US
2017  BD89  8D  08  A0                  STA $A008      ; STORE IN TIMER T2
2018  BD8C  A9  00                      LDA #$00       ; CLR T2 FLG.
2019  BD8E  8D  09  A0                  STA $A009      ; START OPERATION
2020
2021  BD91  B9  40  00                  LDA $0040,Y    ; FETCH TX DATA FROM REG.
2022  BD94  8D  0A  A0                  STA $A00A      ; SHIFT-OUT VIA SHIFT REG.
2023  BD97  AD  0D  A0           NFY2:  LDA $A00D      ; READ INTERRUPT FLG.
2024  BD9A  29  04                      AND #$04       ; 8 BITS DONE (INT. FLG UP)?
2025  BD9C  F0  F9                      BEQ NFY2       ; IF NOT, READ AGAIN
2026  BD9E  CA                          DEX            ; 2 BYTES TRANSMITTED ?
2027  BD9F  F0  05                      BEQ RT1        ; IF YES, RETURN TO M-ROUT.
2028  BDA1  C8                          INY
2029  BDA2  C0  04                      CPY #$04       ; 3 BYTES TRANSMITTED ?
2030  BDA4  D0  DC                      BNE NFY1       ; IF NOT, FETCH NXT BYTE
2031  BDA6  60                   RT1:   RTS            ; RETURN TO MAIN ROUTINE
2032
2033
                                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                                ;
                                ; TRANSMISSION (3 BYTES) AND RECEPTION ROUTINE
                                ;
2034  BDA7                      TXRX:                  ; TRANS./RCPT ROUTINE
2035  BDA7  78                         SEI            ; DISABLE INTERRUPT
2036  BDA8  A0  01                      LDY #$01
2037  BDAA  A9  00                      LDA #$00       ; DISABLE SHIFT REGISTER
2038  BDAC  8D  0B  A0                  STA $A00B
2039  BDAF  A9  14               TR1:   LDA #$14       ; SET SHIFT-OUT BY TIMER T2
2040  BDB1  8D  0B  A0                  STA $A00B
2041  BDB4  A9  44                      LDA #$44       ; LOAD 1 BIT TIME (140 US)
2042  BDB6  8D  08  A0                  STA $A008      ; STORE IN TIMER T2
2043  BDB9  A9  00                      LDA #$00       ; CLR TIMER FLG.
```

```
2044  BDBB  8D  09  A0          STA $A009     ; LOAD COUNTER, START TIMING
2045
2046  BDBE  B9  40  00          LDA $0040,Y   ; FETCH DATA
2047  BDC1  8D  0A  A0          STA $A00A     ; START TRANSMISSION
2048  BDC4  AD  0D  A0    TR2:  LDA $A00D     ; READ INTERRUPT FLAG
2049  BDC7  29  04              AND #$04      ; 8 BITS DATA TRANSMITTED ?
2050  BDC9  F0  F9              BEQ TR2       ; IF NOT, READ FLG AGAIN
2051  BDCB  C8                  INY
2052  BDCC  C0  04              CPY #$04      ; 3 BYTES OF DATA SENT ?
2053  BDCE  D0  DF              BNE TR1       ; IF NOT, FETCH NXT BYTE
2054
2055  BDD0  A2  00              LDX #$00      ; LOAD BYTE COUNTER VALUE
2056  BDD2  AD  00  A0    NBYT: LDA $A000     ; RESET CB2 INT. FLAG
2057  BDD5  A9  88              LDA #$88      ; ENABLE CB2 INTERRUPT
2058  BDD7  8D  0E  A0          STA $A00E
2059  BDDA  A9  00              LDA #$00
2060  BDDC  8D  0C  A0          STA $A00C     ; SET FLG. TRAIL EDGE TRIG.
2061  BDDF  A0  FF              LDY #$FF      ; LOAD PRESNT COUNTER VALU.
2062  BDE1  78                  SEI           ; DISABLE INTERRUPT
2063
2064  BDE2  AD  0D  A0    SRCH: LDA $A00D     ; READ INTERRUPT FLAG
2065  BDE5  29  08              AND #$08      ; FLG. UP? (STRT BIT '0'?)
2066  BDE7  F0  39              BEQ DACTR     ; IF NOT, DECR. COUNTER
2067  BDE9  A0  0B              LDY #$0B      ; LOAD DELAY TIME
2068  BDEB  88            DLY1: DEY           ; DELAY 1 1/2 BIT TIME
2069  BDEC  D0  FD              BNE DLY1
2070
2071  BDEE  A9  04              LDA #$04
2072  BDF0  8D  0B  A0          STA $A00B     ; SET SHIFT-IN BY TIMER T2
2073  BDF3  A9  44              LDA #$44      ; LOAD 1 BIT TIME
2074  BDF5  8D  08  A0          STA $A008     ; STORE IN TIMER T2
2075  BDF8  A9  00              LDA #$00      ; CLR T2 INT. FLAG
2076  BDFA  8D  09  A0          STA $A009     ; LOAD COUNTER, START TIMING
2077
2078  BDFD  AD  0A  A0          LDA $A00A     ; START SHIFT-IN DATA
2079  BE00  AD  0D  A0    SFLG: LDA $A00D     ; READ S/R INTERRUPT FLAG
2080  BE03  29  04              AND #$04      ; 8 BITS SHIFTED-IN ?
2081  BE05  F0  F9              BEQ SFLG      ; IF NOT, WAIT
2082
2083  BE07  AD  0A  A0          LDA $A00A     ; YES, LOAD RECEIVED DATA
2084  BE0A  95  50              STA $0050,X   ; STORE DATA IN RX REGISTER
2085  BE0C  E8                  INX
2086  BE0D  A0  1B              LDY #$1B      ; LOAD DELAY TIME
2087  BE0F  88            DLY2: DEY           ; DELAY 1 1/2 BIT TIME
2088  BE10  D0  FD              BNE DLY2
2089
2090  BE12  A9  00              LDA #$00
2091  BE14  8D  0B  A0          STA $A00B     ; DISABLE SHIFT REGISTER
2092  BE17  AD  0A  A0          LDA $A00A     ; LOAD RX DATA (S2-P) BITS
2093  BE1A  95  50              STA $0050,X   ; STORE IN RX REGISTER
2094  BE1C  E8                  INX
2095  BE1D  E0  04              CPX #$04      ; 4 BYTES STORED ?
2096  BE1F  D0  B1              BNE NBYT      ; IF NOT, FETCH NEXT BYTE
2097  BE21  60            RTRX: RTS           ; REURN TO MAIN ROUTINE
2098
2099
2100  BE22                DACTR:              ; PRESNCE STAT. TRACK ROUT.
2101
2102  BE22  88                  DEY           ; DECREMENT COUNTER
2103  BE23  D0  BD              BNE SRCH      ; COUNT DOWN ? SRCH IF NOT
2104  BE25  A9  00              LDA #$00      ; YES, MARK VOTER ABSENT
2105  BE27  85  51              STA $51
2106  BE29  4C  21  BE          JMP RTRX      ; RETURN TO MAIN ROUTINE
2107
2108
2109
2110
2111                            ****************************************

                                ;* ADDRESS TABLES                       *
2112
                                ****************************************
                                ;
                                ; ADDRESSING ADDRESS (TX1) V OR D MODE
                                ;
2113  BE2C  00                  .BYTE
2114  BE2D  D0  2B  C8    TAB1: $D0, $2B, $C8, $2B, $D8, $3B, $C4, $2B, $D4, $3B
      BE30  2B  D8  3B
      BE33  C4  2B  D4
      BE36  3B
```

```
2115  BE37  CC  3B  DC          $CC, $3B, $DC, $2B, $C2, $2B, $D2, $3B, $CA, $3B
      BE3A  2B  C2  2B
      BE3D  D2  3B  CA
      BE40  3B
2116  BE41  DA  2B  C6          $DA, $2B, $C6, $3B, $D6, $2B, $CE, $2B, $DE, $3B
      BE44  3B  D6  2B
      BE47  CE  2B  DE
      BE4A  3B
2117  BE4B  C1  2B  D1          $C1, $2B, $D1, $3B, $C9, $3B, $D9, $2B, $C5, $3B
      BE4E  3B  C9  3B
      BE51  D9  2B  C5
      BE54  3B
2118  BE55  D5  2B  CD          $D5, $2B, $CD, $2B, $DD, $3B, $C3, $3B
      BE58  2B  DD  3B
      BE5B  C3  3B
2119
2120                                    ;
                                        ; VOTERS TX2 ADDRESSES (DISCUSSION MODE)
                                        ;
2121
2122  BE5D  00                    .BYTE
2123  BE5E  D0  2B  C8      TAB2: $D0, $2B, $C8, $2B, $D8, $3B, $C4, $2B, $D4, $3B
      BE61  2B  D8  3B
      BE64  C4  2B  D4
      BE67  3B
2124  BE68  CC  3B  DC          $CC, $3B, $DC, $2B, $C2, $2B, $D2, $3B, $CA, $3B
      BE6B  2B  C2  2B
      BE6E  D2  3B  CA
      BE71  3B
2125  BE72  DA  2B  C6          $DA, $2B, $C6, $3B, $D6, $2B, $CE, $2B, $DE, $3B
      BE75  3B  D6  2B
      BE78  CE  2B  DE
      BE7B  3B
2126  BE7C  C1  2B  D1          $C1, $2B, $D1, $3B, $C9, $3B, $D9, $2B, $C5, $3B
      BE7F  3B  C9  3B
      BE82  D9  2B  C5
      BE85  3B
2127  BE86  D5  2B  CD          $D5, $2B, $CD, $2B, $DD, $3B, $C3, $3B
      BE89  2B  DD  3B
      BE8C  C3  3B
2128
2129                                    ;
                                        ; VOTERS TX2 ADDRESSES (VOTING MODE)
                                        ;
2130
2131  BE8E  00                    .BYTE
2132  BE8F                  TAB3:
2133  BE8F  D0  2A  C8          $D0, $2A, $C8, $2A, $D8, $3A, $C4, $2A, $D4, $3A
      BE92  2A  D8  3A
      BE95  C4  2A  D4
      BE98  3A
2134  BE99  CC  3A  DC          $CC, $3A, $DC, $2A, $C2, $2A, $D2, $3A, $CA, $3A
      BE9C  2A  C2  2A
      BE9F  D2  3A  CA
      BEA2  3A
2135  BEA3  DA  2A  C6          $DA, $2A, $C6, $3A, $D6, $2A, $CE, $2A, $DE, $3A
      BEA6  3A  D6  2A
      BEA9  CE  2A  DE
      BEAC  3A
2136  BEAD  C1  2A  D1          $C1, $2A, $D1, $3A, $C9, $3A, $D9, $2A, $C5, $3A
      BEB0  3A  C9  3A
      BEB3  D9  2A  C5
      BEB6  3A
2137  BEB7  D5  2A  CD          $D5, $2A, $CD, $2A, $DD, $3A, $C3, $3A
      BEBA  2A  DD  3A
      BEBD  C3  3A
2138
2139        0000                .END
```

```
ABSN    B6D8        ABSNT   BAGE        BCHK1   BCAD        BCHK2   BCE5
BERR1   B41B        BERR2   B435        BERR3   BB00        BERR4   BB1A
BK1     BAFC        BK3     BAF9        BK4     BAF6        BLANK   B6B4
BTST    B44F        BT1     BCBF        BT2     BCCD        BUZZ    B6FC
```

BZADD	8747	CANCL	8869	CDIS	8C7C	CKBTO	88D8
CKBT1	88DC	CLKS1	810B	CLKS2	8135	CLR	82AE
CLRLY	8887	CMPBT	88D4	CMPVR	8G7A	CR1	867A
CR2	867E	CR6	869C	CR7	869E	CSTAT	8846
DACTR	8E22	DCUSN	8765	DEL	821A	DELY	8759
DISS	826A	DLAY	8161	DLY1	8DEB	DLY2	8E0F
DL1	821E	DP1	84D9	DP2	84F6	DP3	8502
DP4	8518	DSPVT	8488	DSP1	8CF1	DY	8163
ENAM	817D	ERR1	8432	ERR2	844C	ERR3	8817
ERR4	8831	ETR1	8181	ETR2	81D2	ETR3	81E2
ETR4	81E6	ETR5	8205	GAD1	8389	GAD2	83C4
GD1	838B	GD2	83C6	GTADD	8A92	GT1	8A94
HERE1	8813	HERE2	8894	HERE3	8858	IDPTR	88CB
INGN1	8A74	INIT1	8375	INRX1	82FD	INRX2	8333
INRX3	832D	INTGN1	83A6	INTVL	8755	ITST	8057
KYB	88A7	LOAD	8293	LODMM	8A0E	LP1	87CD
LP3	8808	LP4	882D	LP5	8878	MSELT	822B
NAME	8054	NBYT	8DD2	NFY1	8D82	NFY2	8D97
NO	86C6	NOTIFY	8D78	NVALD	8848	NVBTN	8463
NVTR1	82D4	NXDSP	8524	NXPRT	8C2F	NXSTM	8848
NXSTN	8AFD	NXTV	89F5	NXT1	882B	NXT2	8879
NXT5	89C4	NXVTR	87F0	ORAMM	8889	PASS	8CG1
PBODD	8C06	PBONE	849E	PCHK1	8AA1	PCHK2	8834
PCHK3	8AB5	PC1	8AE3	PDAT	88C4	PDLY	8C5B
PRCD1	8315	PRCD2	8355	PRINT	8C12	PRTER	85AA
PRT1	8C45	PRT2	8C75	PR1	8584	PR2	85EF
PR3	85F1	PR4	8603	PTST	808B	QDAT	88E1
QRN2	8A87	QR1	8A89	QSTN	887A	RCHK1	8392
RCHK2	839C	RELD	8788	REPT	8850	RESET	816A
RLYON	8960	RL1	8964	RQST	885A	RTN	8C01
RTRX	8E21	RT1	8DA6	SCRN	89C4	SFLG	8E00
SFT1	80AD	SFT2	80F7	SLECT	8253	SP1	8900
SP2	8933	SRCH	8DE2	STCHK	88FC	STRBL	8A48
ST1	8A4C	TAB1	8E2D	TAB2	8E5E	TAB3	8E8F
TON1	8046	TON2	8000	TR1	8DAF	TR2	8DC4
TXRX	8DA7	UDATV	846C	UDAT1	8896	UDAT2	88A3
UDAT3	88AC	UD1	8499	UPDAT	8A28	UP1	8A2D
UP2	8A3C	VERFY	83CF	VFY1	840A	VGM1	8267
VMODE	826D	VM1	828A	VM2	82EE	VRAB1	8399
VRAB2	83A3	VRTN1	841A	VSEL	88C7	YES	86EA

. = 8E8F

ERRORS DETECTED: 0

*C/PTP,C=C.KJK
RUN-TIME: 1 2 0 SECONDS
CORE USED: 4K

VITA

Kia Jom Kwang was born in China, August 25 1953. After receiving The City and Guilds of London Institute Full Technological Certificate in Electrical and Electronic in 1978, he came to The University of Tenne- ssee at Martin in fall 1980. In spring 1981, he transferred to The University of Tennessee at Knoxville and received a bachelor of science degree in electrical engineering in winter 1983.

He entered the graduate school in spring 1983. He is a part time employee of The Public Building Authority of The County of Knox and The City of Knoxville, Tennessee. In fall 1983, he received teaching assistantship from The Electrical Engineering Department of The University of Tennessee.

He is a member of Tau Beta Phi National Honor Society, Eta Kappa Nu Honor Society, The Golden Key Honor Society and a member of IEEE.