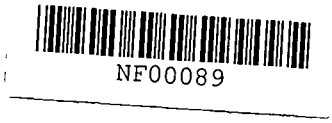
NASA Technical Memorandum 86860

Design Description of a Microprocessor Based Engine Monitoring and Control Unit (EMAC) for Small Turboshaft Engines

Anastacio N. Baez
Lewis Research Center
Cleveland, Ohio

January 1985

NASA



DESGIN DESCRIPTION OF A MICROPROCESSOR BASED ENGINE MONITORING AND CONTROL UNIT (EMAC) FOR SMALL TURBOSHAFT ENGINES

Anastacio N. Baez
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

SUMMARY

Research programs have demonstrated that digital electronic controls are more suitable for advanced aircraft/rotorcraft turbine engine systems than hydromechanical controls. Commercially available microprocessors are believed to have the speed and computational capability required for implementing advanced digital control algorithms. Thus, it is desirable to demonstrate that off-the-shelf microprocessors are indeed capable of performing real-time control of advanced gas turbine engines. The engine monitoring and control (EMAC) unit has been designed and fabricated specifically to meet the requirements of an advanced gas turbine engine control system. The EMAC unit is fully operational in the Army/NASA small turboshaft engine digital controls research program.

INTRODUCTION

As the demand for improved aircraft/rotorcraft turbine engine performance becomes more pronounced, the need for more sophisticated engine control systems becomes more evident. Hydromechanical engine controls, which have been the control mechanism for years, are having difficulty performing some functions considered essential to the operation of modern aircraft/rotorcraft turbine engine systems - e.g., torque matching and overspeed control. Research programs have demonstrated that digital electronic controls can successfully handle such tasks (refs. 1 to 4). Thus, digital electronic controls are recognized to be more suitable for control of future advanced gas turbine engines. Thus far most of the digital controls research programs have implemented the control laws on high-speed minicomputers. These units are typically quite large and not representative of an operational flight-worthy version of digital engine controls. Therefore, future programs will use smaller, denser computer circuitry more representative of flight hardware. This will add more credibility to the results of these programs.

Commercially available 16-bit microprocessors are believed to have the speed and computation capability required for implementing advanced digital electronic engine controls. Thus, it is desirable to demonstrate that off-the-shelf 16-bit microprocessors are indeed capable of performing real-time digital electronic engine control. Several features are necessary in a turbine engine research control system.

The research control system must be portable in order to move it to either a computer simulation laboratory or an engine test facility. In addition, the research unit should (1) have the capability for making software changes easily, (2) be application independent so it can satisfy many research programs, and (3) provide the user with the ability to interrogate the control computer, while it is running in real time, to verify that it is performing

correctly. The engine monitoring and control (EMAC) unit has been designed and fabricated at Lewis Research Center to meet these needs. This report presents a detailed design description of the EMAC unit. The report also describes the methodology needed to effectively use this engine monitoring and control unit in a research environment.

OVERALL FUNCTIONAL DESCRIPTION OF EMAC UNIT

The engine monitoring and control unit (EMAC) consists of two main elements: the monitoring unit and the control unit. These two units are enclosed in a double width relay rack. A photograph of the complete EMAC unit is shown in figure 1. The monitoring unit's main function is to allow the operator to interrogate the control system while it is running in real time to assure proper operation of the control algorithm. This microcomputer controlled unit is capable of displaying, either in volts or engineering units (EU), all the data channels in and out of the EMAC unit. The control unit's main function is to execute in real time the control system algorithm necessary for engine control. This algorithm is designed to maintain the particular research engine under normal operating conditions depending on engine status and engine environment.

When operating, the monitoring unit's microcomputer continuously samples the operator input device, a keyboard located on the front of the monitoring unit. Once the operator selects an actual data channel, the monitoring unit software executes the following steps: (1) the desired channel is routed by the switching matrix multiplexer to the EMAC's microcomputer, and (2) the channel description (which includes channel number, name of variable, scale factor for engineering units display, etc.) is generated and output to the monitoring unit front panel display system. The monitoring unit channel selection software also provides some useful and commonly used special features such as (1) scan up - causes the next sequential channel to be selected, (2) scan down - decrements the channel being displayed by one, (3) previous channel - selects the last channel selected before the presently selected channel, and (4) change units - switches the digital voltmeter display reading from volts to EU or conversely.

The control unit interaction with the user is accomplished by a Decwriter (any RS-232 interface compatible device can be used) as opposed to the monitor unit's simple keyboard. The control unit's microcomputer contains an operating system, CPM-86, which is responsible for all software file management and execution. This operating system, which resides on an 8-in flexible diskette, is loaded into the control microcomputer memory every time a power up or a system reset is acknowledged by the control unit's microprocessor. This loading process, called bootstrapping, is accomplished by a bootstrap loader which resides in the control unit's microcomputer nonvolatile memory (EPROM).

In research programs involving the development of digital electronic control algorithms, it is desirable, and many times necessary, to make software changes to the control algorithm. The control unit's flexible diskette system stores this control algorithm. Since a flexible diskette is a read/write device, changes to the control algorithm can be made easily by using either a compatible microprocessor development system or the control unit's microcomputer operating system file management and support software. The control system algorithm is loaded into the control microcomputer's random access memory

(RAM) using the operating system file management features. The control software will maintain within its safe operating conditions either a real-time hybrid engine simulation or an actual experimental engine.

DETAILED DESCRIPTION OF EMAC HARDWARE

Monitoring Unit Hardware

A block diagram showing the main components of the EMAC monitoring unit is presented in figure 2. The heart of the monitoring unit is a custom-designed microcomputer. A photograph of this microcomputer is shown in figure 3. This microcomputer is based on the Intel Corporation 8085A-2 microprocessor. Some of the reasons for using the Intel 8085A-2 microprocessor are familiarity with the microprocessor software and hardware, reliable vendor with a reputation of guaranteeing software and hardware support, and the existence of an in-house Intel compatible development system.

The 8085A-2 is a single chip, n-channel type, 8-bit parallel central processing unit (CPU) with a 0.8 μsec basic instruction cycle and a 5-MHz internal system clock. Other major specifications include on-chip clock generator (only needs an external crystal or a capacitor), a single 5-V power supply, on-chip system controller, four vectored interrupts, and a serial input/output port. A detailed description of the Intel 8085A-2 microprocessor and its support hardware is provided in references 5 and 6.

The monitoring microcomputer system contains 8K bytes of programmable read only memory (PROM), 1K bytes of random access memory (RAM), six 8-bit parallel input/output ports, and a programmable keyboard/display interface.

The memory map for the monitoring microcomputer is shown in figure 4. The nonvolatile memory (EPROM) is located at the lowest part of the memory map. The monitoring unit software package (the executive and service routines) starts at memory location 0000H.

The proms used are 2716-1, which are ultraviolet, erasable, and electrically programmable read only memory (EPROM) with a 2K by 8-bit organization. The rams are static 2114-2, which have a 1K by 4-bit organization. The input/output interface consist of two 8255A-5 programmable peripheral interface chips. Each of these chips contains three 8-bit parallel ports which can be programmed to be either inputs, outputs, or a combination of both. An 8279-5 programmable keyboard/display controller is used as the interface between the CPU and the keyboard and digital channel display. Commands and channel selection inputs to the monitoring microcomputers are made by means of a 20-key keyboard. This keyboard consists of a 5 by 4 pushbutton switch matrix. The monitoring microcomputer resides in a three-dimensional wirewrap card. Circuit diagrams of the monitoring microcomputer are shown in figure 5.

The front panel of the monitoring unit contains the display system, the 20-key keyboard, the status and warning lights, and the illuminated discrete switches. Figure 6 shows the monitoring unit front panel. The display system components are an alphanumeric display, a digital voltmeter (DVM), a 4-digit 7-segment digital display, and a discrete engineering units/volts indicator. The alphanumeric display is a 20-character programmable random access display,

which interfaces directly to the 8085A-2 microprocessor data bus. This alphanumeric display is used to display messages to the user and to display important information on the channel being displayed by the monitoring unit: for example, "DVM OUTPUT IN VOLTS", "INVALID CHANNEL", "NG GAS GEN SPEED RPM", etc. The digital voltmeter has 4-1/2 digits with a full scale of $\pm$200 mV. A passive network (voltage divider) is used to scale down the input to the DVM. The position of the decimal point in this DVM is externally driven by the monitoring microcomputer. The front panel also contains 16 discrete indicators which are used by the control microcomputer to indicate the status of the engine and control system under investigation. These indicators are also very helpful when an engine operating limit is reached, because they will specifically display the condition of the engine at that point. Examples are a turbine input temperature limit or a gas generator overspeed. The discrete indicators are also used to indicate any algorithm calculation overflow, range checks on the inputs to the control microcomputer, etc. Sixteen discrete switches in the front panel are used by the EMAC operator to select or change engine control modes, to initialize different subroutines in the control algorithm, or to call a specific microcomputer program. In the process of evaluating and debugging a digital engine control algorithm it is desirable to check all possible control loops and control modes. For example, the user may want to execute the accel schedule or the idle schedule, or want to freeze the control system commands to the engine (whether it be real or simulated). The operator can also use these discrete switches to either execute or bypass certain sections of code in the control algorithm.

The monitoring unit switching matrix design is based on analog CMOS switches. Figure 7 shows a block diagram of the switching matrix. These switches are latch proof dielectrically isolated analog switches with overvoltage protection of up to $\pm$25 V above the power supply voltages. The switch matrix consists of a total of 96 differential analog channels which are organized as four groups of 24 channels each. Each group of 24 channels resides on a three-dimensional wire-wrap-type electronic card. A picture of a typical group is shown in figure 8. One 8-bit parallel port from the monitoring microcomputer is used to select the desired channel from the switching matrix. These eight parallel lines are binary encoded, thus necessitating a binary decoder. The output of the final stage of the switching matrix is routed to a scaling circuit. This scaling circuit consists of a multiplying digital to analog converter (MDAC) and some discrete circuitry to change the output of this MDAC from a current output to a voltage output. The output of this scaling circuit is in the format:

$$V_{out} = SF \times V_{in}$$

where V_{in} is the analog voltage in the data channel and SF is the scaling factor which is an 8-bit digital word. This scale factor is either one, for data readings in volts, or a fraction of one for an engineering units reading. This digital word is provided by the monitoring unit software and depends on what channel is being selected. The circuit diagram for this scaling circuit is shown in figure 9.

The monitoring microcomputer, the switching matrix, the scaling circuit, and the interface circuitry for the front panel display system all reside in a three-dimensional horizontal chassis with a wire-wrap-type ground plane and a rear input/output frame. This backplane permits the interconnection between different wire wrap cards in the chassis.

Control Unit Hardware

The control unit hardware consists of a commercially available single board microcomputer, an analog to digital input multiplexer (MUX) board, two digital to analog output boards (DACS), a discrete input board, and a flexible diskette hardware system. A Decwriter is used to communicate with the control microcomputer via the microcomputer board RS-232 interface. This microcomputer board and its peripheral boards all reside in an Intel Corp. system chassis. A picture of the Intel iSBC-660 system chassis is shown in figure 10. This system chassis is an eight-slot cardcage and backplane for Multibus compatible microcomputer and peripheral boards. It has a heavy duty power supply with all standard Multibus voltages and horizontal board mounting for compactness.

The same reasons for the selection of an Intel 8085A-2 in the design of the monitoring unit microcomputer hold true in the selection of an Intel 8086 based microcomputer as the control unit microcomputer. In addition to these reasons, the 8086 microprocessor has the speed and the computational capability necessary for digital electronic control systems for gas turbine engines. The control microcomputer is an Intel Corporation iSBC 86-12A single board computer, which is based on the 8086 16-bit HMOS microprocessor central processing unit (CPU).

A picture of the 86/12A single board computer is shown in figure 11. The 8086 CPU has a basic instruction cycle of 400 nsec, and a 5-MHz internal system clock. Ths iSBC 86/12A single board microcomputer contains 32K bytes of dynamic RAM (expandable onboard to 64K bytes), sockets for up to 16K bytes of ROM (expandable onboard to 32K bytes), 24 programmable parallel I/O lines, programmable synchronous/asynchronous RS232C compatible serial interface with software selectable baud rates, two programmable 16-bit BCD or binary timers/event counters, and nine levels of vectored interrupt control (expandable to 65 levels). The operating system used with this single board computer requires 64K of RAM memory; thus, an expansion memory board was incorporated to the iSBC 86/12A board. This expansion board sits on the iSBC 86/12A board in a piggyback fashion and provides 32K of additional RAM memory making 64K bytes total onboard RAM memory. Another special feature added to the computer board is a high-speed multimodule numeric data processor. This is an iSBC 337, which functions as a high-speed fixed and floating point coprocessor. This coprocessor is used primarily with the control unit's data collection software. Reference 7 describes the iSBC 86/12A single board computer in detail.

The multiplexer I/O board provides up to 16 differential or 32 single ended analog input channels and two digital to analog converters (DACS). The multiplexer board is Multibus compatible and has 12-bit binary analog to digital conversion capability, a programmable gain amplifier, end of scan and end of conversion interrupts, and an onboard dc/dc power converter as special features. One of the DAC boards has four 12-bit digital to analog converters and four digital output channels. These DAC's are software driven via doubled buffered registers, and they can be individually set to any of five possible output ranges. An input code to each channel can be configured as any of the following codes: natural binary, offset binary, or two's complement. The digital outputs from this board are high-current, high-voltage logic drivers which can be used for on/off (true/false) control of various system functions. The logic drivers have open collector outputs with 30 V and 30 mA rating. The other DAC board provides eight additional 12-bit digital to analog converters with the same specifications as the first one. The discrete input board provides for 24 optically isolated discrete input channels. This board is

Multibus compatible, and its inputs can be configured to operate with voltage or contact closures. The flexible diskette system components are a single board Multibus compatible disk controller and two 8-in disk drives. The disk drives and their associated interface electronics are enclosed in a 19-in rack mountable chassis/cabinet. The controller board, which is compatible with Intel single and double density formats, can support up to four diskette drives.

Interface Hardware

The EMAC unit signal flow diagram is shown in figure 12. All analog signals sent through the EMAC unit are three wires shielded. One hundred data channels are available at the EMAC input. A picture of the inside (back) of the EMAC unit is shown in figure 13. Ten base connectors, each consisting of ten channels, provide the inputs to the EMAC unit. Eighty of these channels are used as the unit trunk lines. Each trunk line is split and fed to the switching matrix input and to a 24 by 34 patch panel, which is located in the front panel of the monitoring unit. The remaining 20 channels are fed directly to the patch panel (without going through the switching matrix). These channels are dedicated to brush recorders and to other analog data recorders. Also, the multiplexer board inputs, the DAC boards outputs, and 16 general purpose channels are available at the patch panel. The general purpose channels are connected directly to the switching matrix and are intended to be used as test points to check and track any signal available at the patch panel. The EMAC unit patch panel is the heart of the EMAC interface hardware because it permits the selection of many different configurations of inputs and outputs which could change with different applications of the EMAC unit. A layout of the EMAC patch panel is shown in figure 14.

Finally, a set of 20 buffer amplifiers completes the EMAC interface hardware. These buffers are high-accuracy, unity gain differential amplifiers which have a low gain error, low nonlinearity, high common mode rejection, and do not require external trim. Since the amplifier's inputs and outputs are available at the patch panel, any signal available at the EMAC patch panel could be buffered (inputs, outputs, etc.).

DETAILED DESCRIPTION OF EMAC SOFTWARE

Monitoring Unit Software

The monitoring unit software package consists of an executive or main program and the following service subroutines: initialization, input, output, computation, table lookup, and data. Flowcharts of the main program and subroutines are shown in figures 15 to 20. These programs are programmed in Intel 8085 assembly language and reside in nonvolatile memory (EPROM).

The function of the main program is simply to call the service subroutines in a sequential order. The initialization subroutine initializes the EMAC monitoring unit computer peripherals which include two 8255-5 parallel port interface chips, one 8279-5 keyboard/display interface chip, and a random access alphanumeric message display. The input routine tests for inputs from the keyboard, through the 8279 decoder interface chip, and sets a flag if any key has been pressed, and it also stores which of the keys in the keyboard have been pressed. The output routine handles all the output functions of the monitoring unit computer. It checks a flag to determine what kind of output has

been called for: output to the alphanumeric display, to the channel selector and display, or to the scaling circuit (for engineering units display). The calculation routine performs all the computation needed between input and output routines. This includes determining which key on the keyboard was pressed and taking appropriate action and other calculations (e.g., the calculation of the scale factor for engineering unit display, scan up, previous channel, etc.). The table lookup routine (which is called by the computation routine) utilizes table lookup techniques to calculate scaling factor, decimal point position, and a name (with units) for the channel being displayed. And, finally, the data routine contains the messages to the alphanumeric display and all the data tables for channel information.

Control Unit Support Software

The control unit support software includes an operating system, a bootstrap loader, a discrete input/discrete output diagnostic, and an analog input/output calibration program. The calibration program adjusts the full scale and zero offsets on the analog input/output boards. The discrete input/output diagnostic tests each discrete input and output individually. These two programs and the bootstrap loader are programmed in assembly language and were developed in-house by NASA. (The bootstrap loader operation was explained in the overall description of the control unit.)

The control unit operating system is CP/M-86. This is a single user, general purpose operating system marketed by the Digital Research corporation. CP/M-86 provides facilities to do program load and unload, console (Decwriter for the EMAC) communications, disk file management, and some rudimentary memory management. When configured to be used with the 64K iSBC 86/12A single board computer, CP/M-86 occupies the lowest 12K of memory. The remaining 52K of memory are used as a transient program area for applications programs and for other scratch pad operating system functions. CP/M-86 contains three major options - the command control processor (CCP), the basic disk operating system (BDOS), and the basic input/output system (BIOS). The CCP accepts commands from the communications console, interprets them, and takes the appropriate action. The BDOS contains all the facilities to open, format, read, and organize in a logical manner files that are stored on a flexible diskette. In addition, it contains routines to do memory management (i.e., memory allocation and deallocation) on the transient program area. The BDOS has an external entry point or 'hook' such that the majority of the file and memory management functions can be accessed by applications programs. For example, a Lewis designed data collection program (ref. 8) will make extensive use of these "hooks." Finally, the BIOS contains all the hardware dependent information necessary to allow CP/M-86 to operate in a particular computer configuration. This includes defining the disk layout (i.e., number of tracks, number of sectors, number of bytes per sector, etc.), the address and input/output format of the communication console device, the memory configuration of the transient program area, etc. Because all the hardware dependent information is concentrated in one particular area, CP/M-86 can be reconfigured easily for a wide variety of hardware environments. When a command is received from the communications console, the CCP processes it and determines if an external applications program must be loaded from a disk. In such case, the program is loaded from the specified disk and placed in the top part of the iSBC 86/12A memory (i.e., highest memory

locations). The free memory above the operating system and below the applications program is still available to be allocated by CP/M-86 for the applications program's own use. A more detailed description of CP/M-86 and its applications is given in references 9 and 10.

EMAC Testing and Verification

All the EMAC monitoring unit's 8085-based software and hardware have been fully tested and verified. A microprocessor development system consisting of the necessary hardware and software support to develop a microprocessor-based system was used to test and verify the monitoring unit. This system contains its own microprocessor and peripherals, a diskette-based operating system, and development software needed to design and debug the target system. It also provides an in-circuit emulator (ICE 85), which makes it possible to debug the user hardware with software debugging aids. The ICE-85/development system combination allows us to perform the following diagnostics on the custom designed microcomputer: (1) set breakpoints in the software to be able to look at the CPU registers and status flags, (2) check the timing of critical signals in the hardware, (3) execute the software step by step for both software and hardware debugging, and (4) perform other tests as required.

The EMAC control unit has also been successfully checked out. The 8086-based control microcomputer has been tested using the 64K configured version of CP/M-86 operating system. Proper disk file management and onboard program execution has been accomplished. The analog and discrete input/output boards have been calibrated to specifications using diagnostic programs specifically written for these boards.

CONCLUDING REMARKS AND FUTURE PLANS

The flexibility of the EMAC's design makes it a useful research tool for evaluating a wide variety of advanced digital electronic control modes. The EMAC unit has been used to evaluate a bill of materials (BOM) control law for a GE YT700 helicopter engine. A real-time hybrid computer simulation of the engine and the real engine were used to evaluate the BOM algorithm. Baseline data have been generated for future controls research programs. This research effort is part of a cooperative Army/NASA program to conduct digital electronic controls research for small turboshaft engines. The main objective of this program is the engine test evaluation of modern control algorithms using a flexible microprocessor-based digital electronic control system. The EMAC will be the mechanism for implementing a modern multivariable control law applied to rotorcraft engines. A detailed description of the controls program is given in reference 11. A description of the GE YT700 BOM engine control system is provided in reference 12, and the real-time engine simulation is described in reference 13.

Upgrading the EMAC unit would include the use of floating point arithmetic and high level language programming of the engine control algorithm. Presently, the BOM control algorithm is programmed in fixed-point assembly language, and only the data collection software is done in a floating point high level language. Also, a dynamic data collection system to store transient data and uplink this data online to a main frame computer for data reduction and plotting is under consideration. This upgrading might need a faster microprocessor

and floating point coprocessor with more memory (larger than the present 64K). Another solution would be the use of multiprocessing technology which is possible with Multibus compatible systems.

The engine monitoring and control unit has been built to meet all design requirements and is fully operational in the Army/NASA small turboshaft engine research program.

REFERENCES

1. Soeder, J.F.: F100 Multivariable Control Synthesis Program. Computer Implementation of the F100 Multivariable Control Algorithm. NASA TP-2231, 1983.

2. Lehtinen, B.; DeHoff, R.L.; and Hackney, R.D.: Multivariable Control Altitude Demonstration on the F-100 Turbofan Engine. AIAA Paper 79-1204, June 1979.

3. Szuch, John R., et al.: F100 Multivariable Control Synthesis Program Evaluation of a Multivariable Control Using a Real-Time Engine Simulation. NASA TP-1056, 1977.

4. Arpasi, Dale J.; Zeller, John R.; and Batterton, Peter G.: A General Purpose Digital System for On-Line Control of Airbreathing Propulsion Systems. NASA TM X-2168, 1971.

5. MCS-80/85 Family User's Manual. Manual Order Number 21506-001, Intel Corporation, October 1979.

6. Component Data Catalog. Intel Corporation, January 1981.

7. iSBC 86/12A Single Board Computer Hardware Reference Manual. Manual Order Number 9803074-01, Intel Corporation, 1979.

8. Soeder, J.F.: MINDS - A Microcomputer Interactive Data System for 8086-Based Controllers. NASA TP-2378, 1984.

9. Cortesi, David E.: Inside CP/M; A Guide for Users and Programmers with CP/M-86 and MP/M 2. Holt Rinehart and Winston, 1982.

10. CP/m-86 Operating System User's Guide. Digital Research Corporation, 1981.

11. Sellers, J.F.; Baez, A.N.; and Bobula, G.A.: ARMY/NASA Small Turboshaft Engine Digital Controls Research Program. NASA TM 82979, 1982.

12. Curran, J.J.: T700 Fuel and Control System. American Helicopter Society, Annual National Forum, AHS Preprint 771, 1973.

13. Hart, Clint E.; and Wenzel, Leon M.: Real-Time Hybrid Computer Simulation of a Small Turboshaft Engine and Control System. NASA TM-83579, 1984.

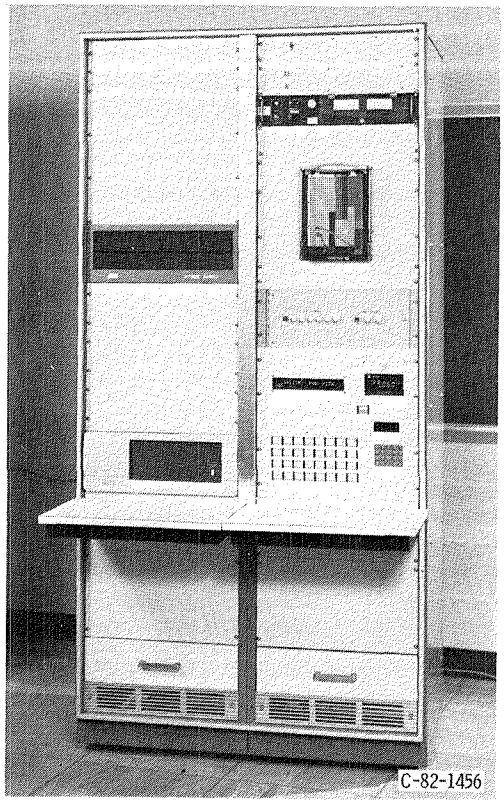


Figure 1. - Complete EMAC unit.

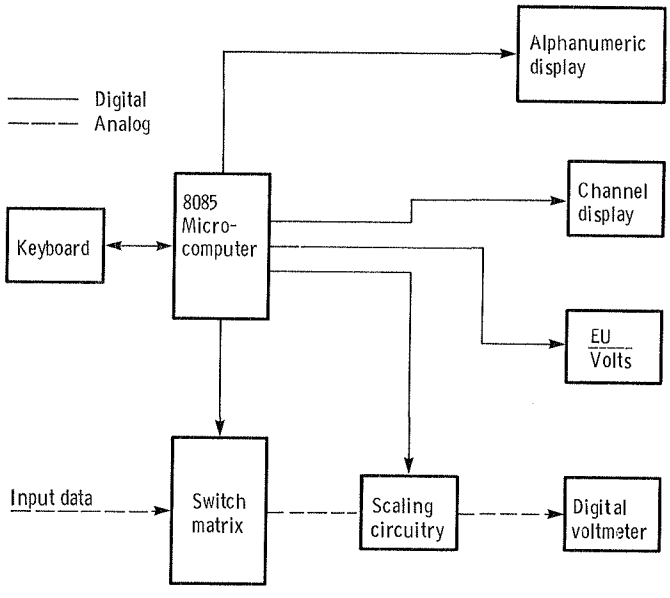


Figure 2. - EMAC monitoring unit.

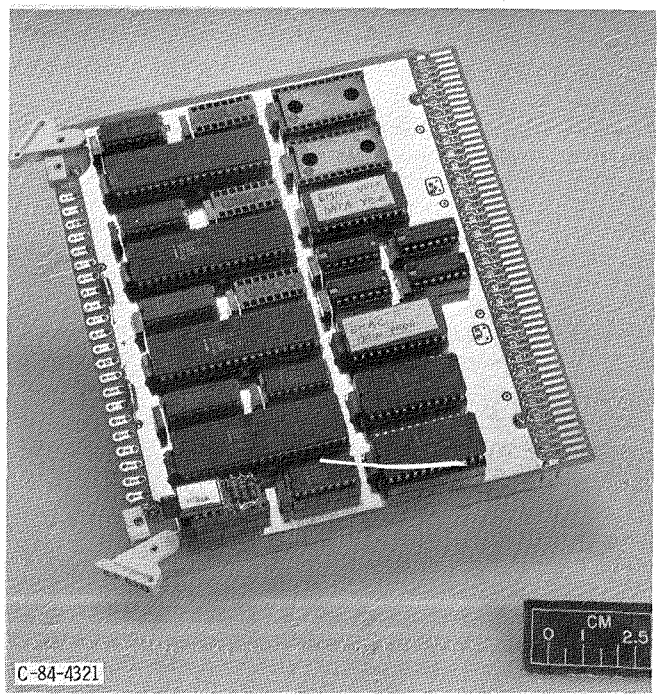


Figure 3. - Monitoring unit microcomputer.

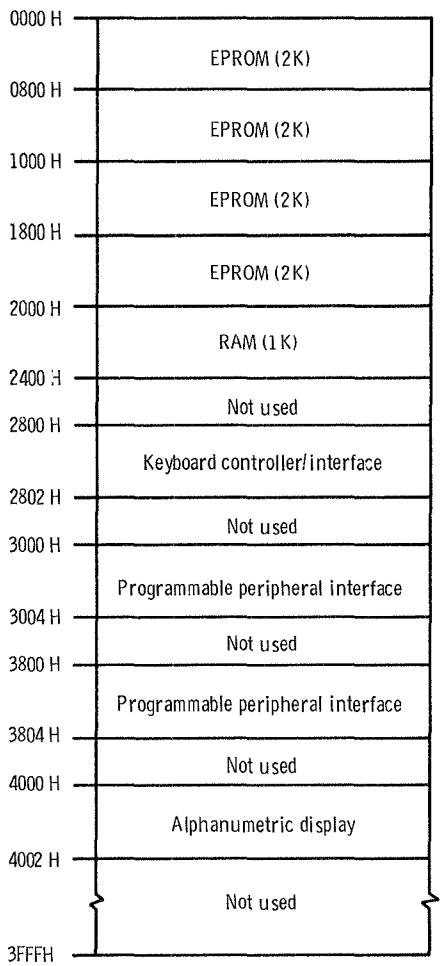
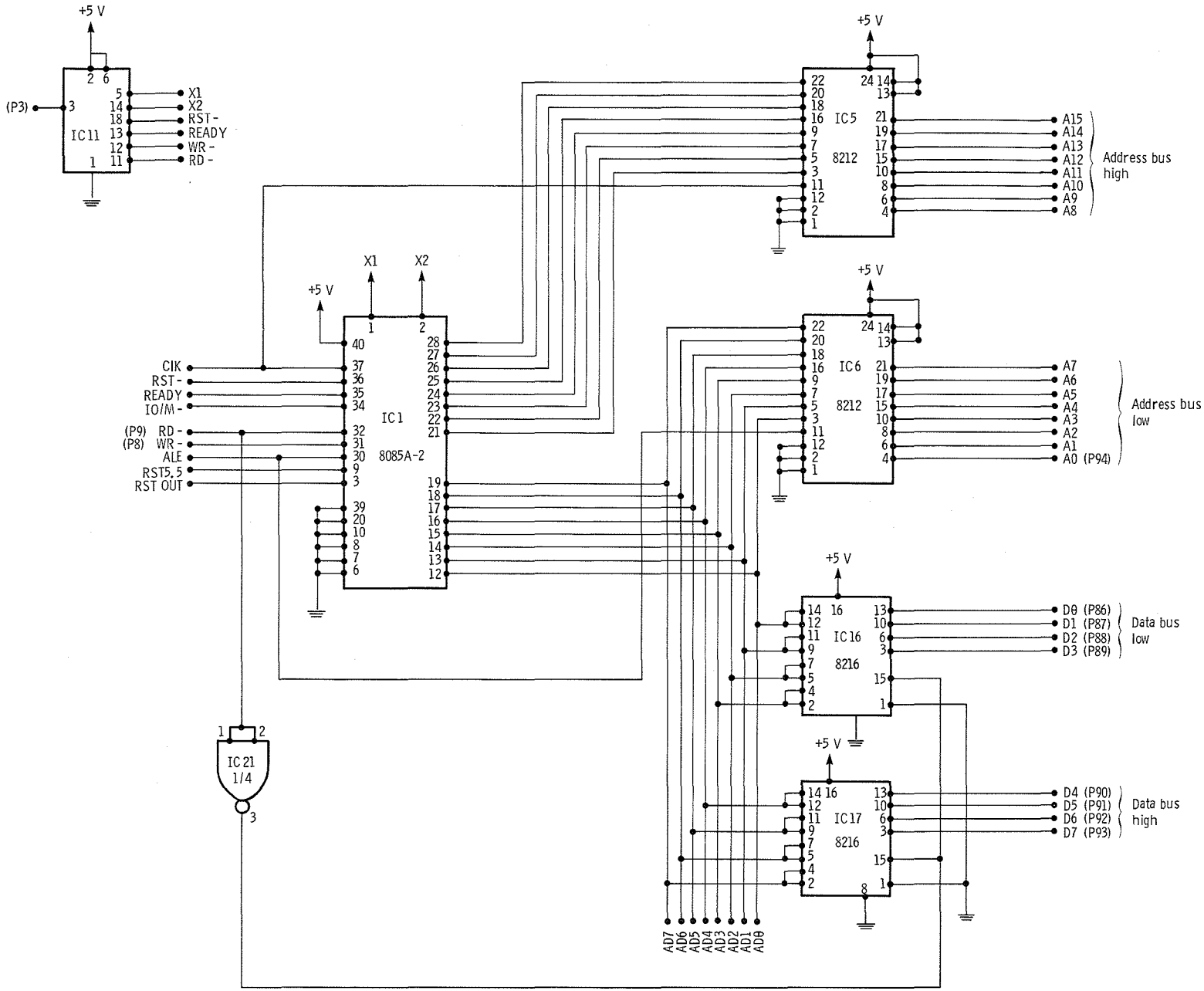
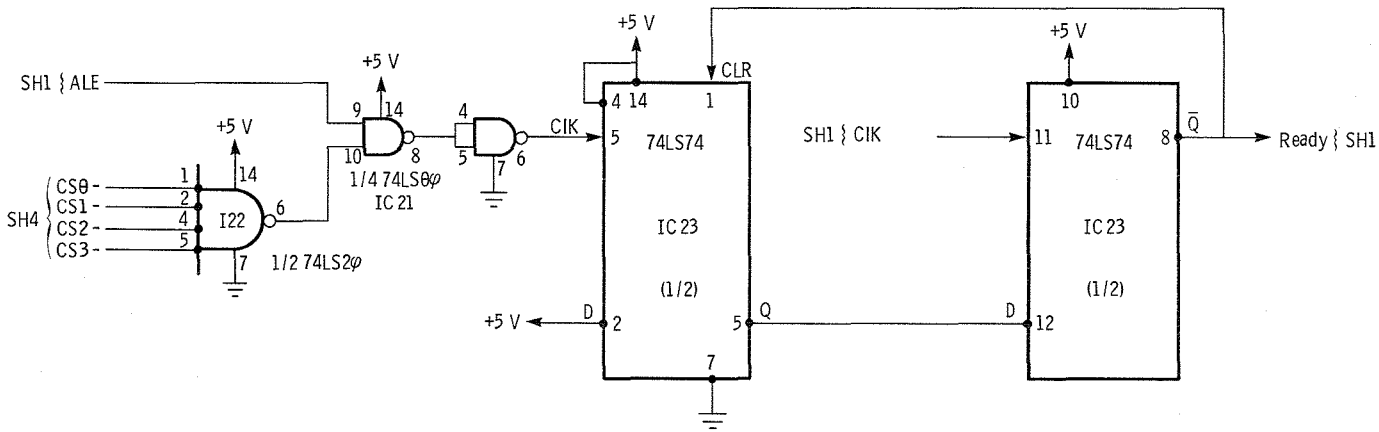


Figure 4. - Monitoring unit microcomputer memory map.



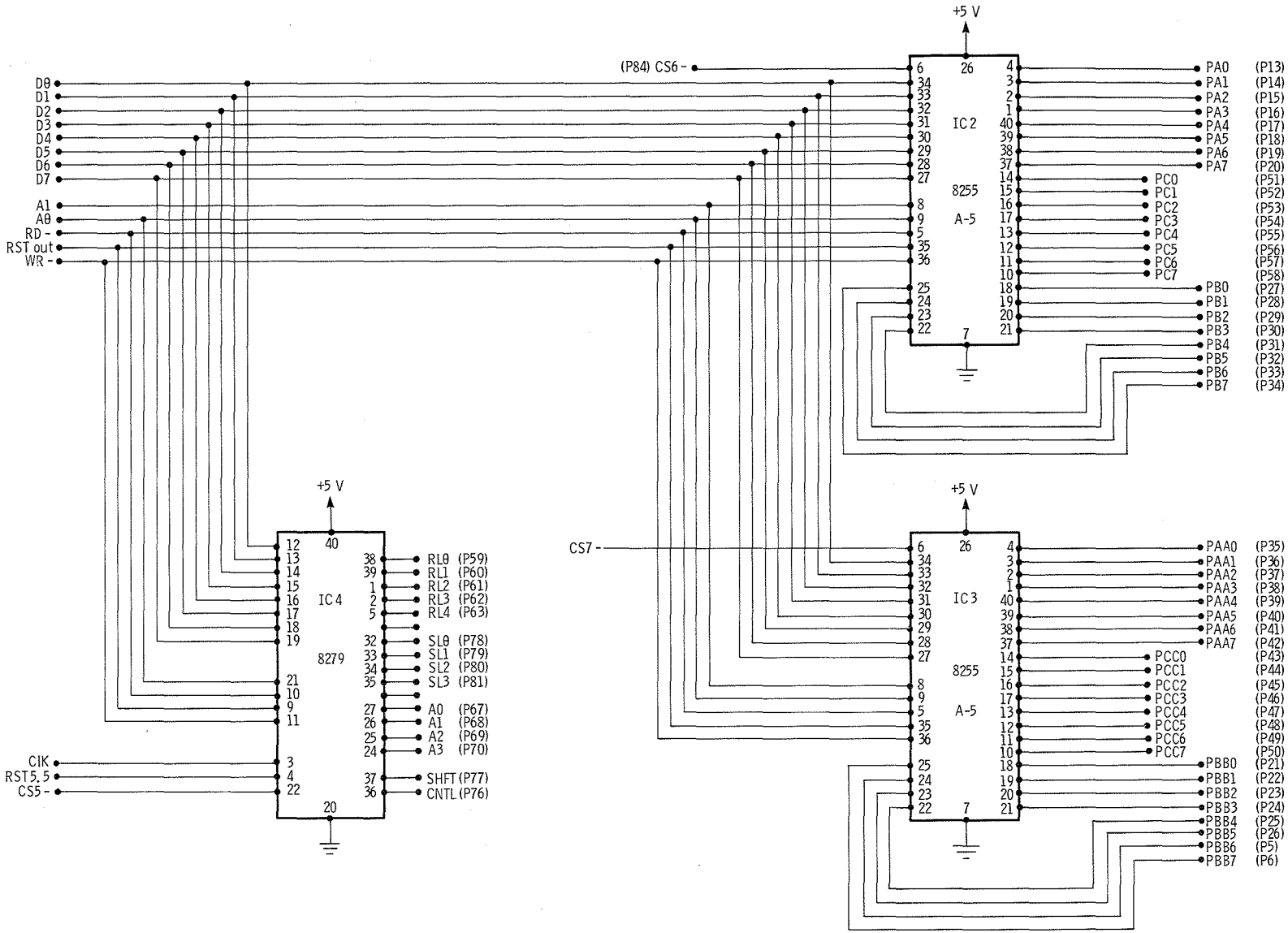
(a) Central processing unit.

Figure 5. - Monitoring unit microcomputer circuit diagram.



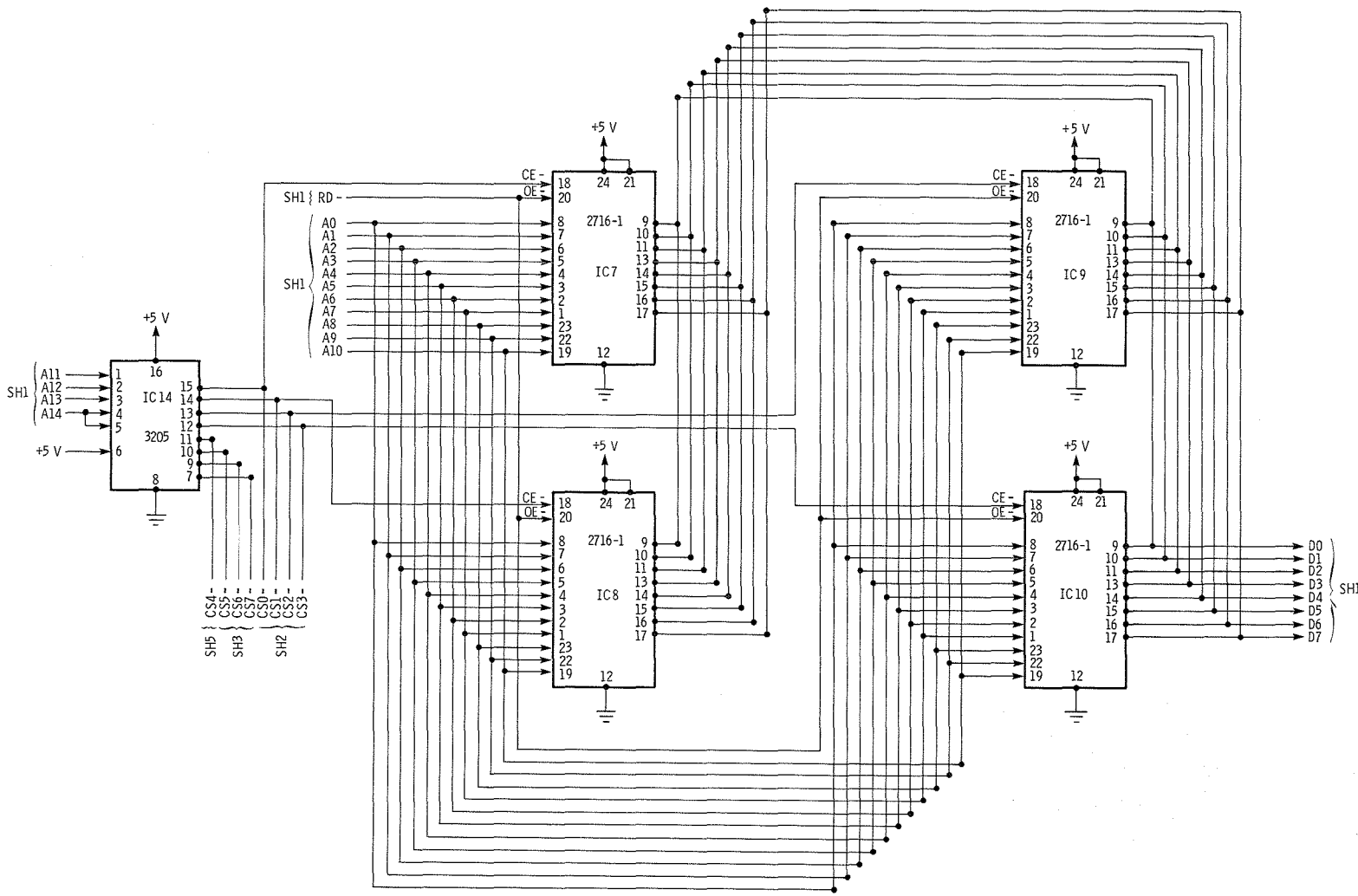
(b) Wait state generator.

Figure 5. - Continued.



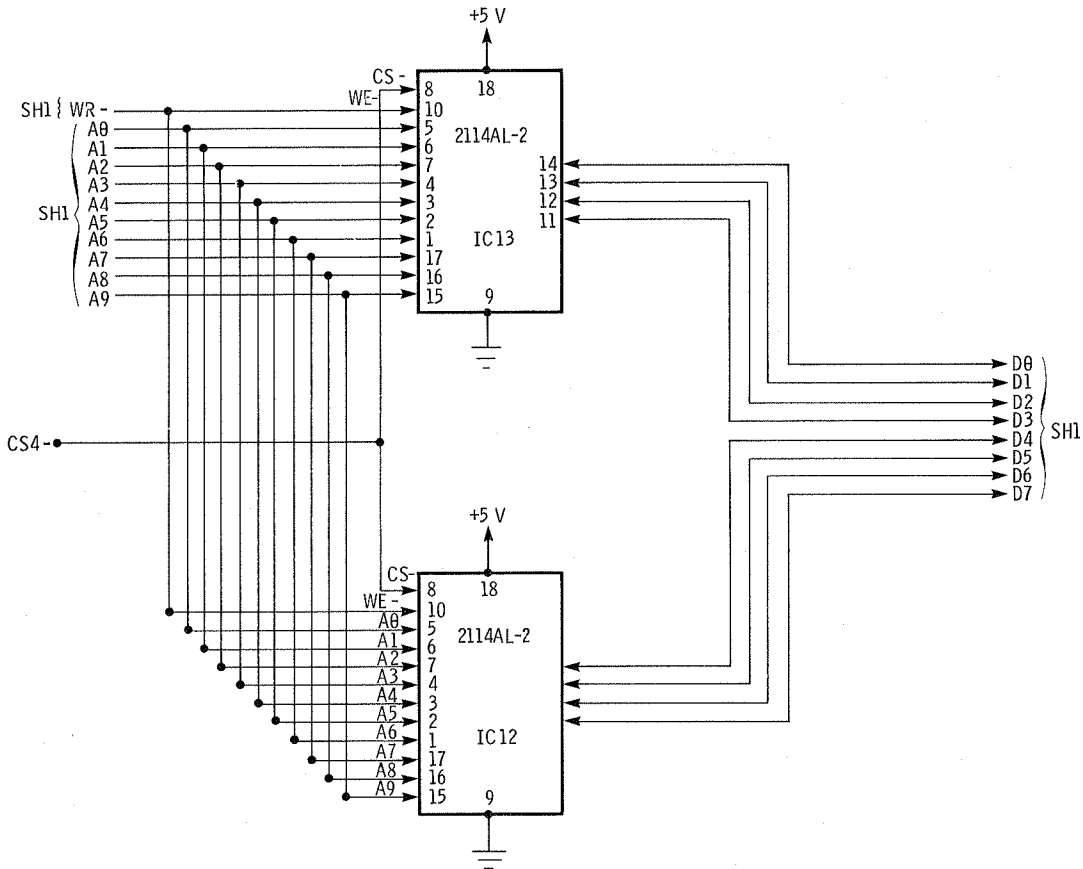
(c) Input/output ports.

Figure 5. - Continued.

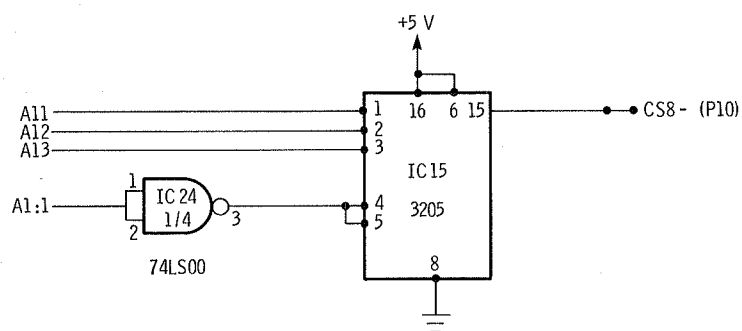


(d) Memory (EPROM).

Figure 5. - Continued.



(e) Memory (RAM).



(f) Decoder.

Figure 5. - Concluded.

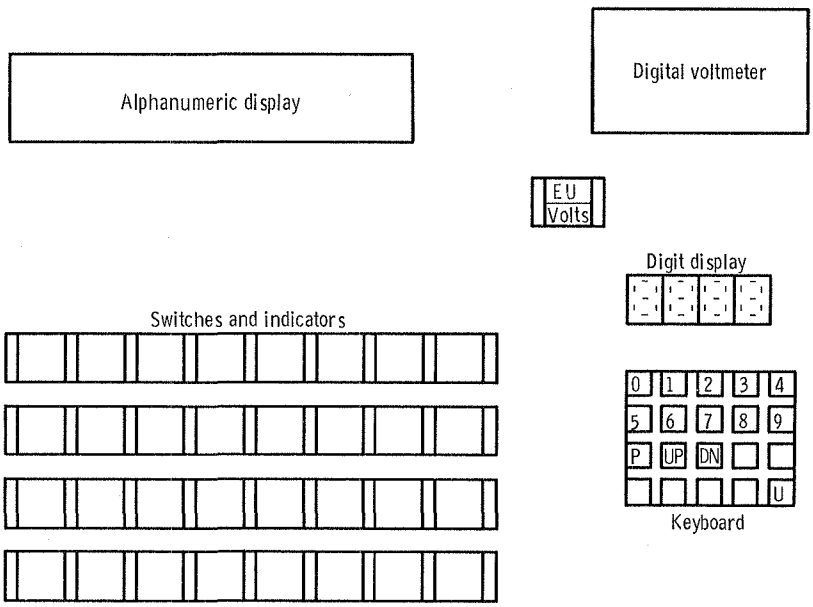


Figure 6. - EMAC front panel layout.

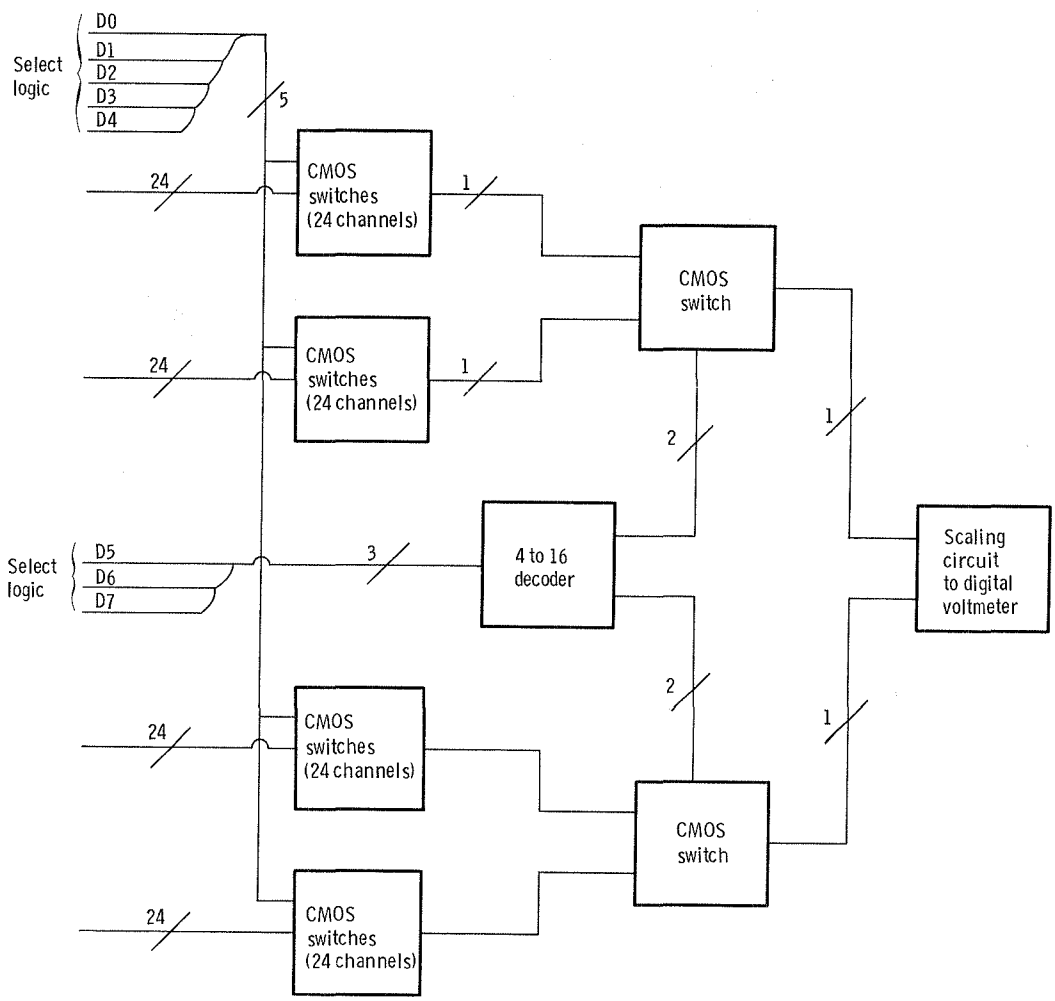


Figure 7. - Switching matrix block diagram.

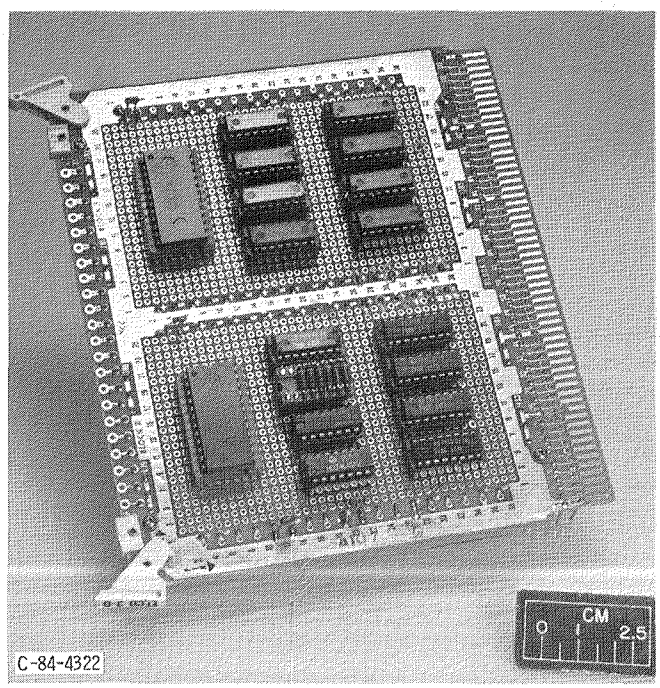


Figure 8. - Switch matrix circuit card (one group of 24 channels).

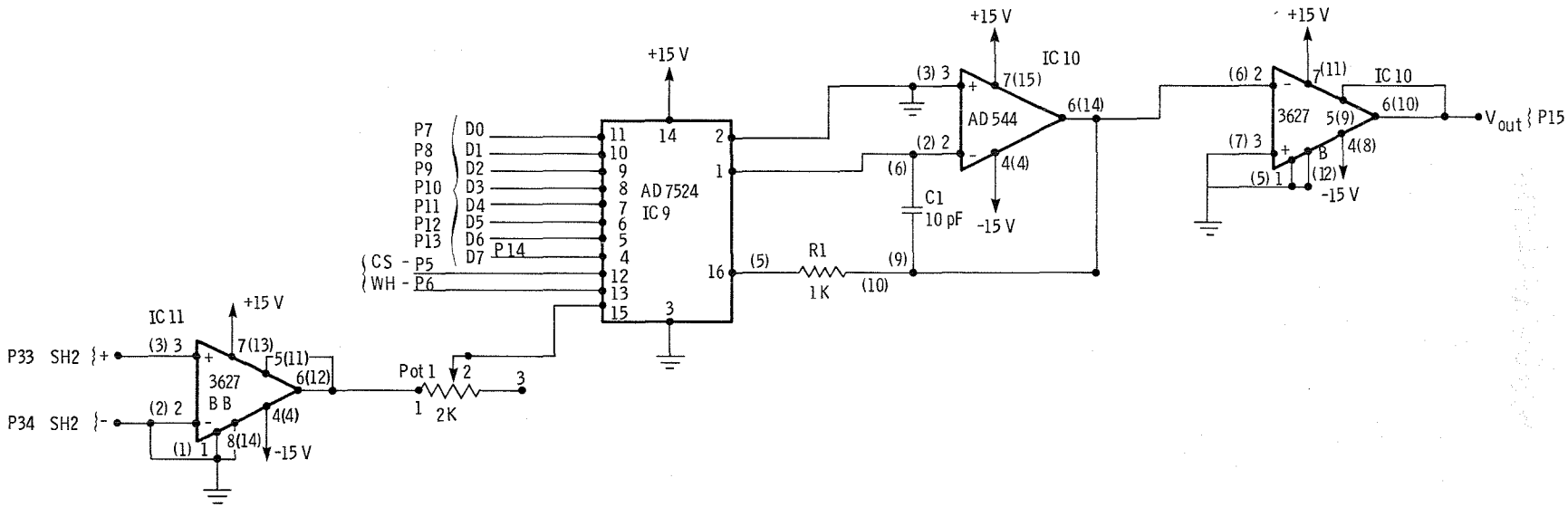


Figure 9. - Scaling circuit diagram.

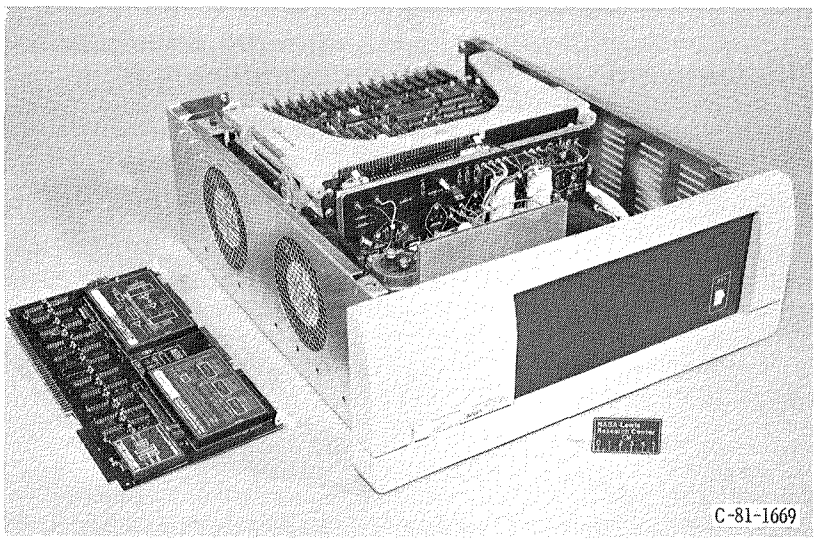


Figure 10. - Control unit system chassis (iSBC-660).

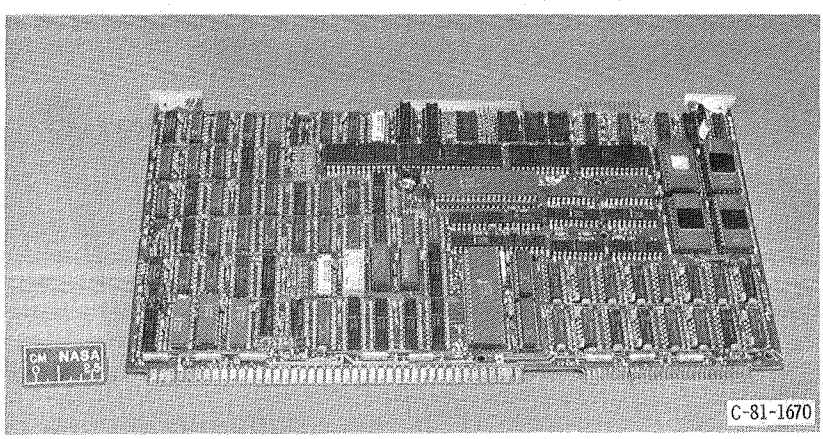


Figure 11. - Intel 86/12A single board computer.

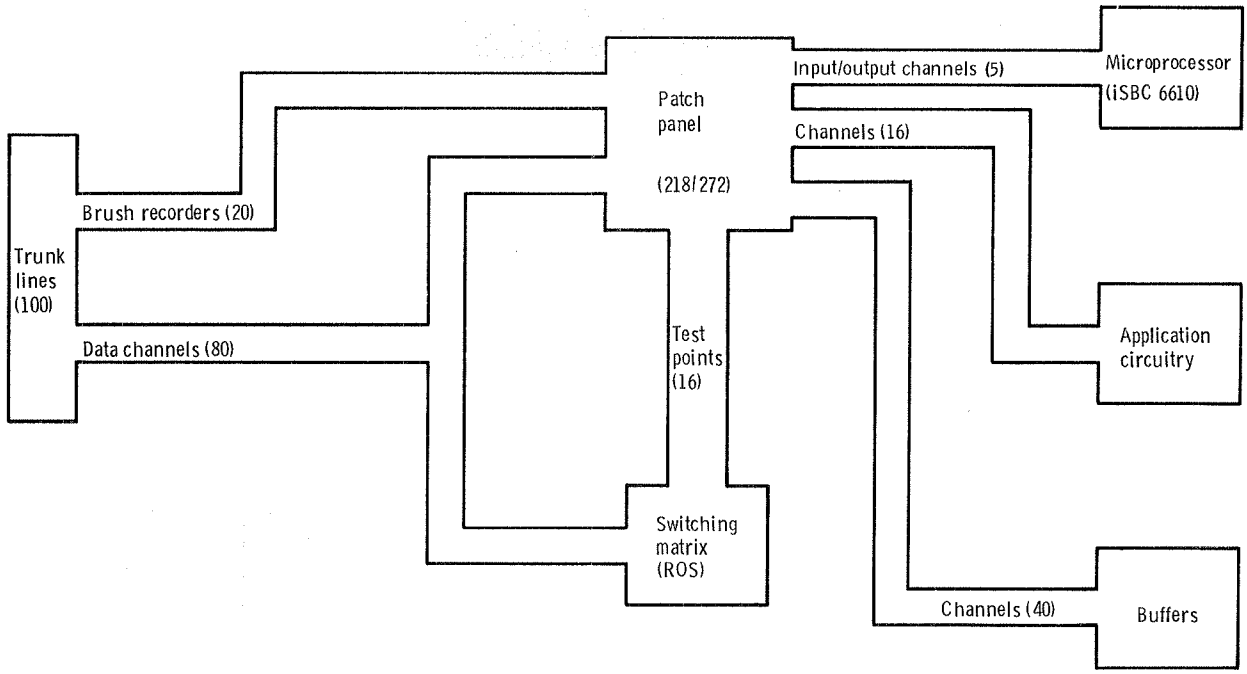


Figure 12. - EMAC unit signal flow.

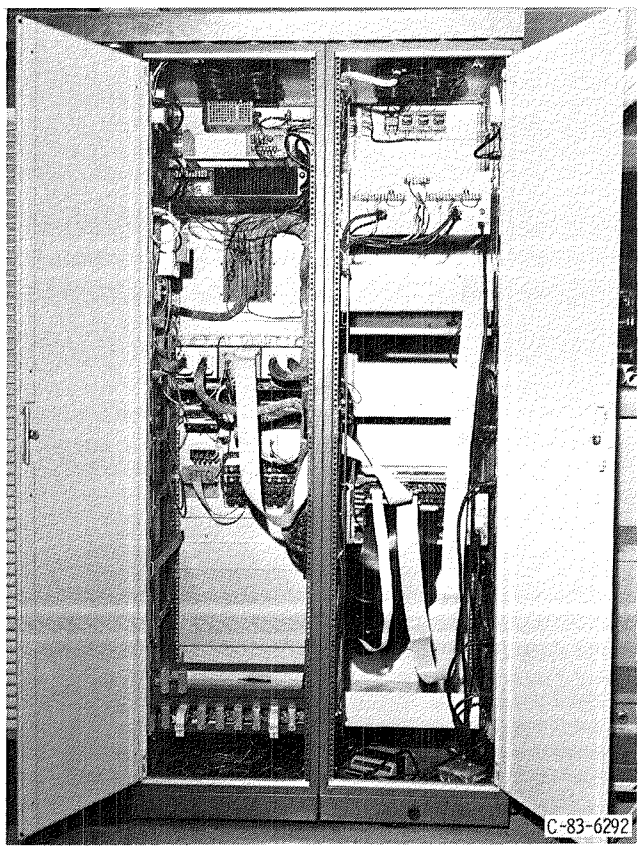


Figure 13. - EMAC unit (inside).

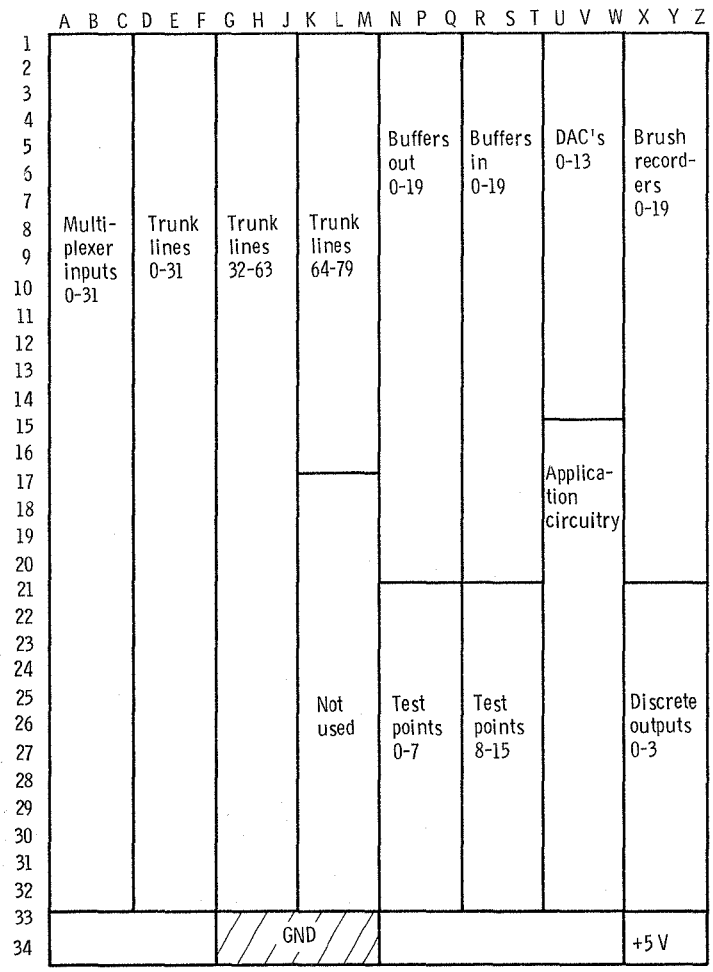


Figure 14. - Patch panel layout.

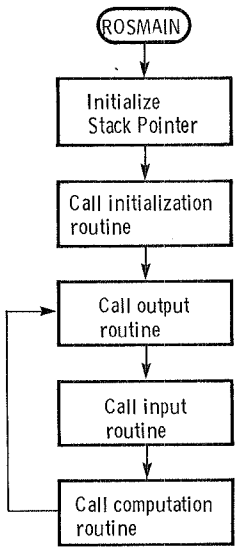


Figure 15. - Main program flowchart.

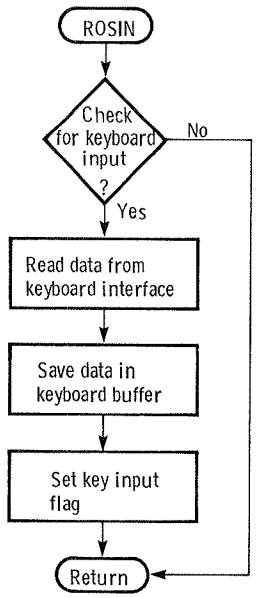


Figure 16. - Input routine flowchart.

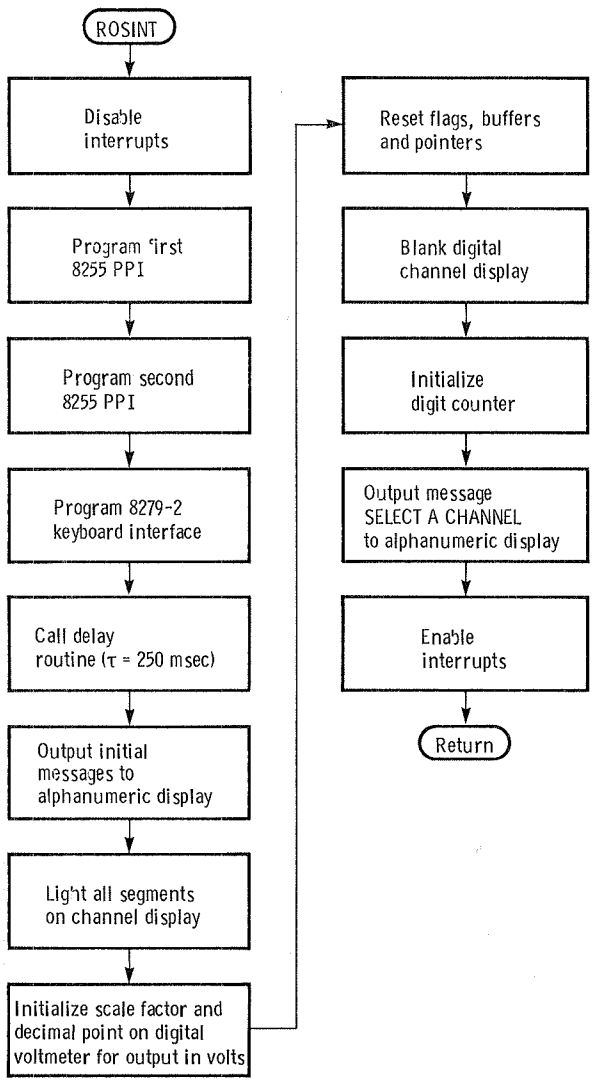
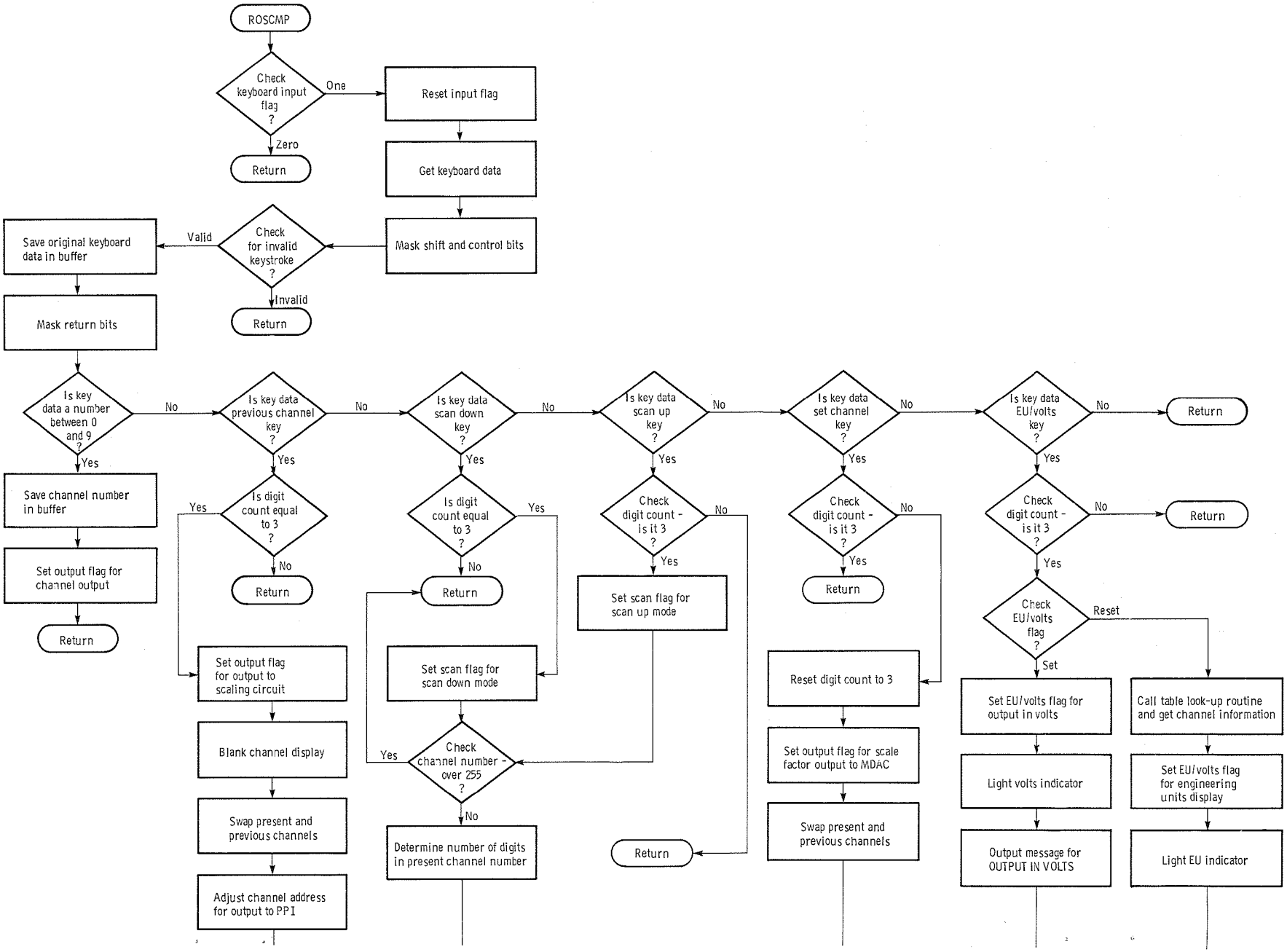


Figure 17. - Initialization routine flowchart.


ROSCMP
Check keyboard input flag ?
One
Zero
Reset input flag
Return
Get keyboard data
Mask shift and control bits
Check for invalid keystroke ?
Valid
Invalid
Save original keyboard data in buffer
Return
Mask return bits
Is key data a number between 0 and 9 ?
No
Yes
Is key data previous channel key ?
Is key data scan down key ?
Is key data scan up key ?
Is key data set channel key ?
Is key data EU/volts key ?
Return
Save channel number in buffer
Set output flag for channel output
Return
Is digit count equal to 3 ?
Return
Set output flag for output to scaling circuit
Blank channel display
Swap present and previous channels
Adjust channel address for output to PPI
Is digit count equal to 3 ?
Return
Set scan flag for scan down mode
Check channel number - over 255 ?
Determine number of digits in present channel number
Check digit count - is it 3 ?
Set scan flag for scan up mode
Return
Check digit count - is it 3 ?
Return
Reset digit count to 3
Set output flag for scale factor output to MDAC
Swap present and previous channels
Check digit count - is it 3 ?
Return
Check EU/volts flag ?
Reset
Set
Set EU/volts flag for output in volts
Light volts indicator
Output message for OUTPUT IN VOLTS
Call table look-up routine and get channel information
Set EU/volts flag for engineering units display
Light EU indicator

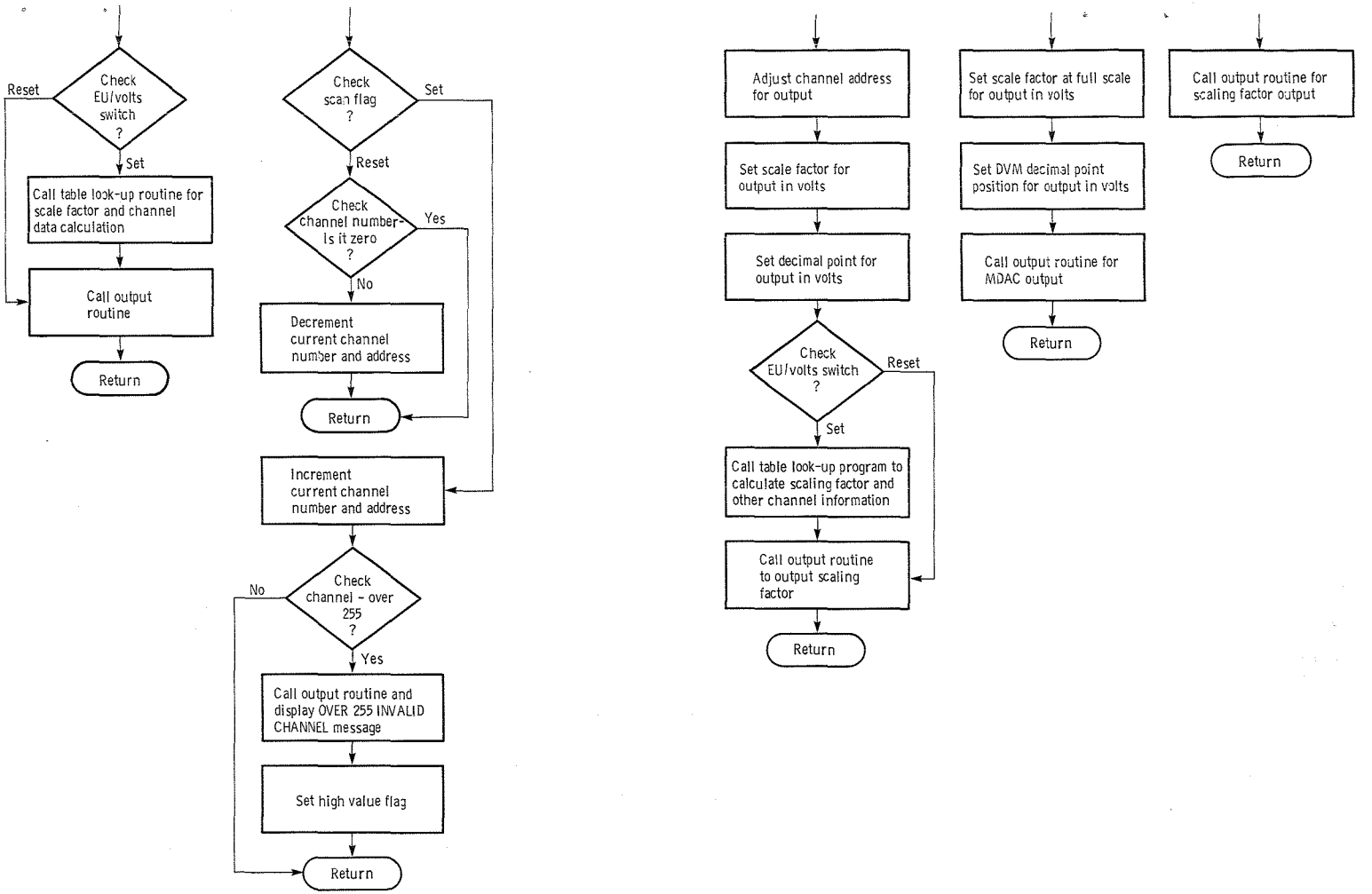


Figure 18. - Computation routine flowchart.

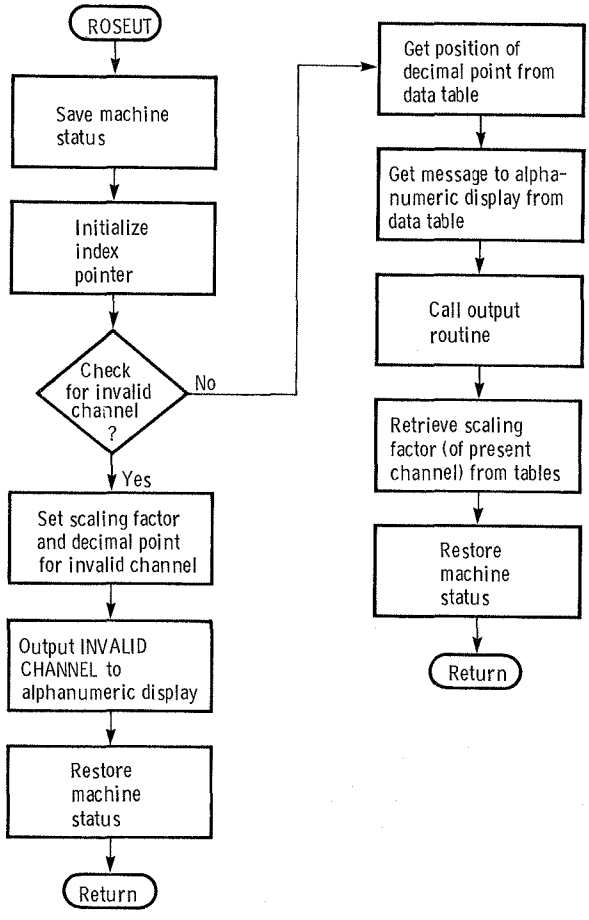


Figure 19. - Table look-up routine flowchart.

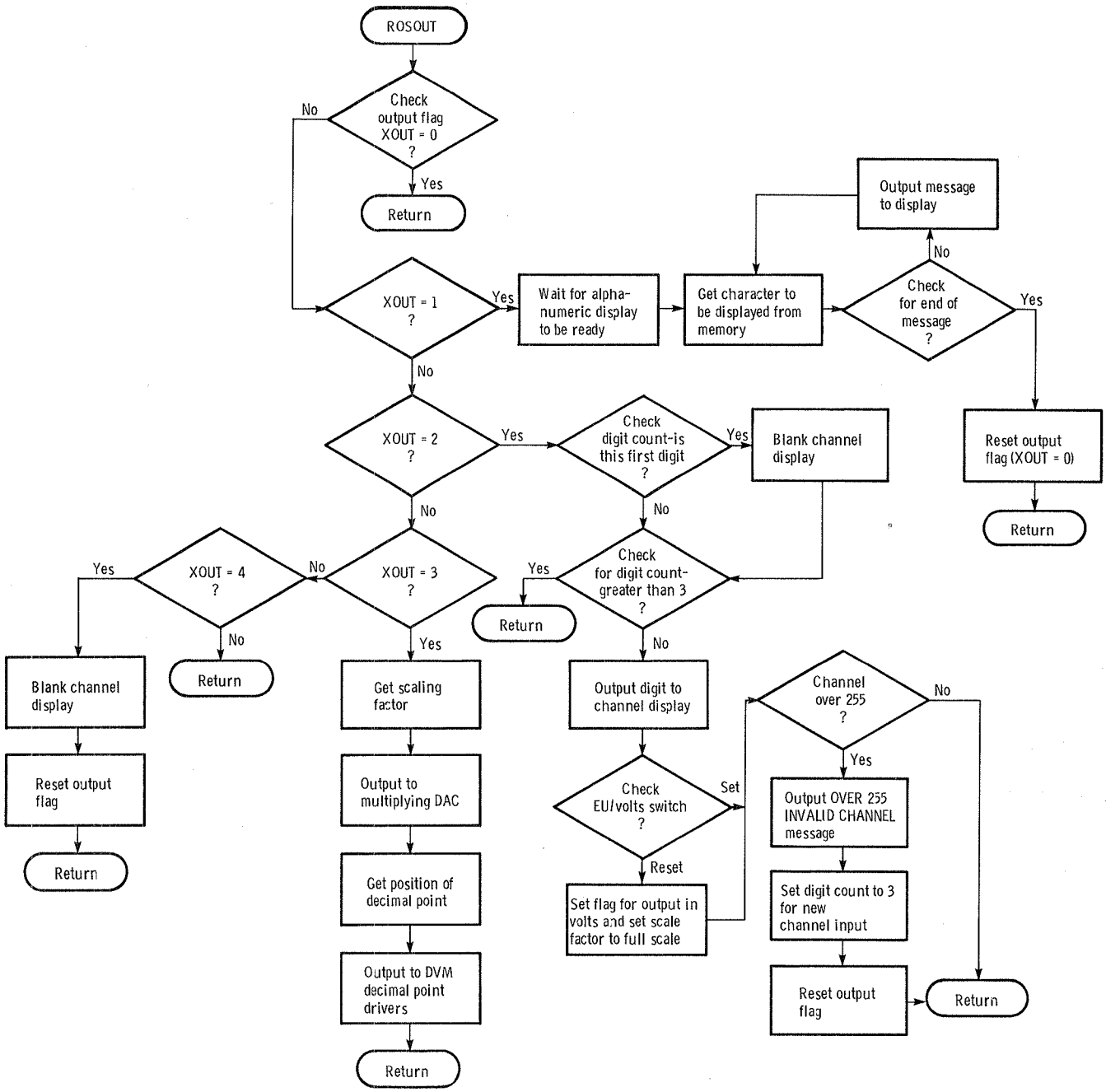


Figure 20. - Output routine flowchart.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM-86860		

4. Title and Subtitle	5. Report Date
Design Description of a Microprocessor Based Engine Monitoring and Control Unit (EMAC) for Small Turboshaft Engines	January 1985
	6. Performing Organization Code 505-40-5B
7. Author(s) Anastacio N. Baez	8. Performing Organization Report No. E-2324
	10. Work Unit No.
9. Performing Organization Name and Address National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135	11. Contract or Grant No.
	13. Type of Report and Period Covered Technical Memorandum
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

Research programs have demonstrated that digital electronic controls are more suitable for advanced aircraft/rotorcraft turbine engine systems than hydromechanical controls. Commercially available microprocessors are believed to have the speed and computational capability required for implementing advanced digital control algorithms. Thus, it is desirable to demonstrate that off-the-shelf microprocessors are indeed capable of performing real-time control of advanced gas turbine engines. The engine monitoring and control (EMAC) unit has been designed and fabricated specifically to meet the requirements of an advanced gas turbine engine control system. The EMAC unit is fully operational in the Army/NASA small turboshaft engine digital controls research program.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Digital control; Microprocessor; Small turboshaft engines	Unclassified - unlimited STAR Category 01

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price*
Unclassified	Unclassified		A02

*For sale by the National Technical Information Service, Springfield, Virginia 22161

End of Document