

Y-1870

MACHINE TOOL CONTROL  
VIA A MINICOMPUTER

G. L. Bowers

The logo for Union Carbide, featuring the company name in a bold, sans-serif font inside a white hexagonal border.

OAK RIDGE Y-12 PLANT  
OAK RIDGE, TENNESSEE

*prepared for the U.S. ATOMIC ENERGY COMMISSION  
under U.S. GOVERNMENT Contract W-7405 eng 26*

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

## **DISCLAIMER**

**This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.**

## **DISCLAIMER**

**Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.**

Printed in the United States of America. Available from  
National Technical Information Service  
U.S. Department of Commerce  
5285 Port Royal Road, Springfield, Virginia 22151  
Price: Printed Copy \$3.00; Microfiche \$0.95

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

Reference to a company or product name does not imply approval or recommendation of the product by Union Carbide Corporation or the U S Atomic Energy Commission to the exclusion of others that may meet specifications.

UC-38

Y-1870

## MACHINE TOOL CONTROL VIA A MINICOMPUTER

G. L. Bowers

**Oak Ridge Y-12 Plant**

P.O. Box Y, Oak Ridge, Tennessee 37830

Date Issued - April 18, 1973

**NOTICE**

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

Prepared for the U.S. Atomic Energy Commission  
Under U.S. Government Contract W-7405-eng-26

**MASTER**

**DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED**

## ABSTRACT

A minicomputer control system was developed for a two-axis machine tool lathe. One of the principal objectives of this project was to drive the axes' motors via open-loop control with a positioning resolution of 25 microinches.

Two methods for driving the axes' motors were developed. Initially, a combination hardware-software method was used for generating command pulses. Aluminum hemishells and tensile specimens were machined to specifications with this control technique. However, a rapid, traverse speed limitation due to the software addition time of the hardware-software adder was discovered. In order to increase the maximum slide velocity, a pure hardware adder was developed. The maximum rapid traverse rate was increased to 15 inches per minute (ipm). Both techniques offer equivalent machining quality.

By eliminating the software addition required by the hardware-software adder technique, the amount of computer time consumed in the control process is greatly reduced. As a result of the increase in duty-free computer time, a multimachine control system seems to be very feasible.

## CONTENTS

SUMMARY . . . . .	4
INTRODUCTION . . . . .	6
LATHE MACHINE TOOL CONTROL . . . . .	7
System Description . . . . .	7
Hardware/Software Adder . . . . .	7
Hardware Adder . . . . .	12
Manual Control Functions . . . . .	13
Results . . . . .	14
Conclusions . . . . .	14
ACKNOWLEDGEMENTS . . . . .	17
APPENDIX A . . . . .	18
Software for Converting an EIA RS-274-B Tape to a Minicomputer Binary Format . . . . .	18
Introduction . . . . .	18
Operation . . . . .	18
Subsection Descriptions . . . . .	19
APPENDIX B . . . . .	22
Cartridge Tape Unit with Read-Only-Memory (ROM) Load Program . . . . .	22
APPENDIX C . . . . .	23
Engineering Drawings . . . . .	23
APPENDIX D . . . . .	42
Operate Program Description and Flow Charts for Lathe Machine Tool Control . . . . .	42
Introduction . . . . .	42
Operation . . . . .	42
Subsection Descriptions . . . . .	44

## SUMMARY

An open-loop control system using a minicomputer was installed on a two-axis tool lathe. The objectives of this project were: (1) to attain an optimum marriage between control hardware and software; (2) to use the standard EIA RS-274-B numerical control tape service available; (3) to provide the normal numerical control functions while machining from data stored in core memory rather than on tape, and (4) to drive the axes' motors via open-loop control with a positioning resolution of 25 microinches.

Two methods for driving the axes' motors were developed. Initially, a combination hardware-software method was used for generating command pulses. Aluminum hemishells and tensile specimens were machined to specifications with this control technique. However, a rapid, traverse speed limitation due to the software addition time of the hardware-software adder was discovered. In order to increase the maximum slide velocity, a pure hardware adder was developed. The maximum rapid traverse rate was increased to 15 inches per minute (ipm). Both techniques offer equivalent machining quality.

Gear modifications were made to obtain a positioning resolution of 25 microinches. The drive mechanism consisted of an electrohydraulic pulse motor for both the X and Y axes. A digital differential analyzer (DDA) was used to generate a pulse train which was fed to the pulse motors. Each input pulse resulted in a 25-microinch movement on the X or Y slide.

Standard EIA RS-274-B numerical control tape service was provided for the system. Each tape is converted from the EIA format into a binary data table format which is more easily utilized by the minicomputer. The binary part description data are stored on a magnetic-tape cartridge unit and can be loaded into core memory when needed. The manual control features developed for this system are equivalent to those found on most industrial numerically controlled machines.

The following is an itemized listing of the findings and recommendations for this project:

1. A hardware-software DDA was proven to be feasible for machine control as borne out by production results.
2. Rapid traverse speed limitations were found due to the software addition time associated with hardware-software DDA.
3. A pure hardware DDA was designed to improve maximum slide velocity.
4. Software was written for the conversion of an EIA data tape to a binary format thus providing standard EIA RS-274-B numerically controlled tape service.
5. A magnetic tape cartridge unit was installed and found to be practical in a production environment for storage of part description data and the control program.
6. Based on the required central processor unit (CPU) time for one tool, multimachine control using the hardware DDA seems very feasible.

7. Specimens of aluminum, steel, stainless steel, and tungsten to be used for material properties determination were machined to specifications. Four-inch-diameter hemispheres were machined with a true-radius deviation of 50 microinches.

## INTRODUCTION

A machine-tool control system employing a small computer as the control element was installed on a two-axis lathe. Due to the success of this development, another two-axis lathe was chosen as the next candidate to incorporate an open-loop control system using a minicomputer. The concept of open-loop control for a machining process is fairly new in the United States. One of the main objectives of this development project was to determine the practicality of using open-loop control as opposed to the conventional numerically controlled (NC), closed-loop control scheme. Electrohydraulic pulse motors driven with a pulse train generated by a digital differential analyzer (DDA) provide the means for axis movement. The small computer interfaced to the machine tool serves as the system controller.

Initially, the DDA consisted of a combination hardware/software design. With the installation of this system, a limitation was found in the time required to generate successive command pulses. Approximately 120 microseconds, minimum, is required by the computer software for addition. As a worst case, two additions are required to generate one command pulse, resulting in a maximum slide velocity of six inches per minute (ipm).

To eliminate this pulse-rate restriction, a pure hardware DDA was designed. The time per addition was reduced from 120 to 10 microseconds. The maximum ipm rate is now determined by the response time of the electrohydraulic pulse motors.

Standard EIA RS-274-B NC part description tape data are converted to a binary format by the minicomputer software and stored in core memory. Control software acts upon the stored-part description data to perform the desired machine functions and to generate the defined X and Y axes movements.

## LATHE MACHINE TOOL CONTROL

### SYSTEM DESCRIPTION

A block diagram of the computer-controlled two-axis lathe is given in Figure 1, and a pictorial of the system in Figure 2. Axis-movement pulses can be generated manually from the incremental feed logic or from the manual position logic. The minicomputer controls the rate and number of movement pulses generated by the DDA. Both the manual and DDA command pulses are fed into the accelerator/decelerator circuitry to prevent extreme transitions in the pulse rate from reaching the electrohydraulic pulse motors. A view of the pulse motors is provided in Figure 3. For sharp discontinuities in the pulse rate, pulses are held or temporarily stored in a 12-bit counter. As the pulse rate stabilizes, the pulses withheld are transmitted to the motors. Additional computer-controlled functions consist of spindle-speed control, turret select, and the execution of miscellaneous and preparatory code control functions.

To initiate the fabrication cycle, a part description tape is automatically loaded into the computer via a cartridge magnetic tape unit, shown in Figure 4, prior to a machining operation. The procedure for loading and storing tape data in core memory from the cartridge tape unit is explained in Appendix A. Standard NC tape service is provided for the system. The conversion process involved in changing an EIA tape to the minicomputer binary format is described in Appendix B.

Part description data are stacked in a block seen in Figure 5, with a variable length since N, S, M, G, and T are not always present. If both M and G are not required, the block length is only six instead of seven words. By storing binary data instead of the EIA formatted data, the core memory required is cut by one half. The CLKSTR word defines the amount of time allowed for tool travel corresponding to the given X and Y distance values.

### HARDWARE/SOFTWARE ADDER

Initially, the design concept for the DDA consisted of a combination hardware/software approach. An adder-technique block diagram which combines computer software and external hardware is given in Figure 6. Initially, X and Y distance values are loaded into Software Registers C and D. The CLKSTR value is then loaded into the Hardware Register A which is automatically loaded into Register B. The clock pulses count the B register into an overflow condition which initiates an add interrupt to the computer.

The X and Y pulse trains are generated by the hybrid DDA approach, with the add rate controlled by external hardware and the digital integration performed by software. Registers C and D are shifted left to minimize the number of additions and simplify detection of the end of a block. With each add interrupt that is generated by the overflow condition of Register B, Register D is added to Sum Y, and Register C to Sum X. After each addition, an X and Y overflow condition is checked. If the previous add resulted in an overflow for X or Y, a motor pulse is generated. The addition process is complete for a specific block when the marker bit is in the left-most position followed by all zeros in the Sum X and Sum Y

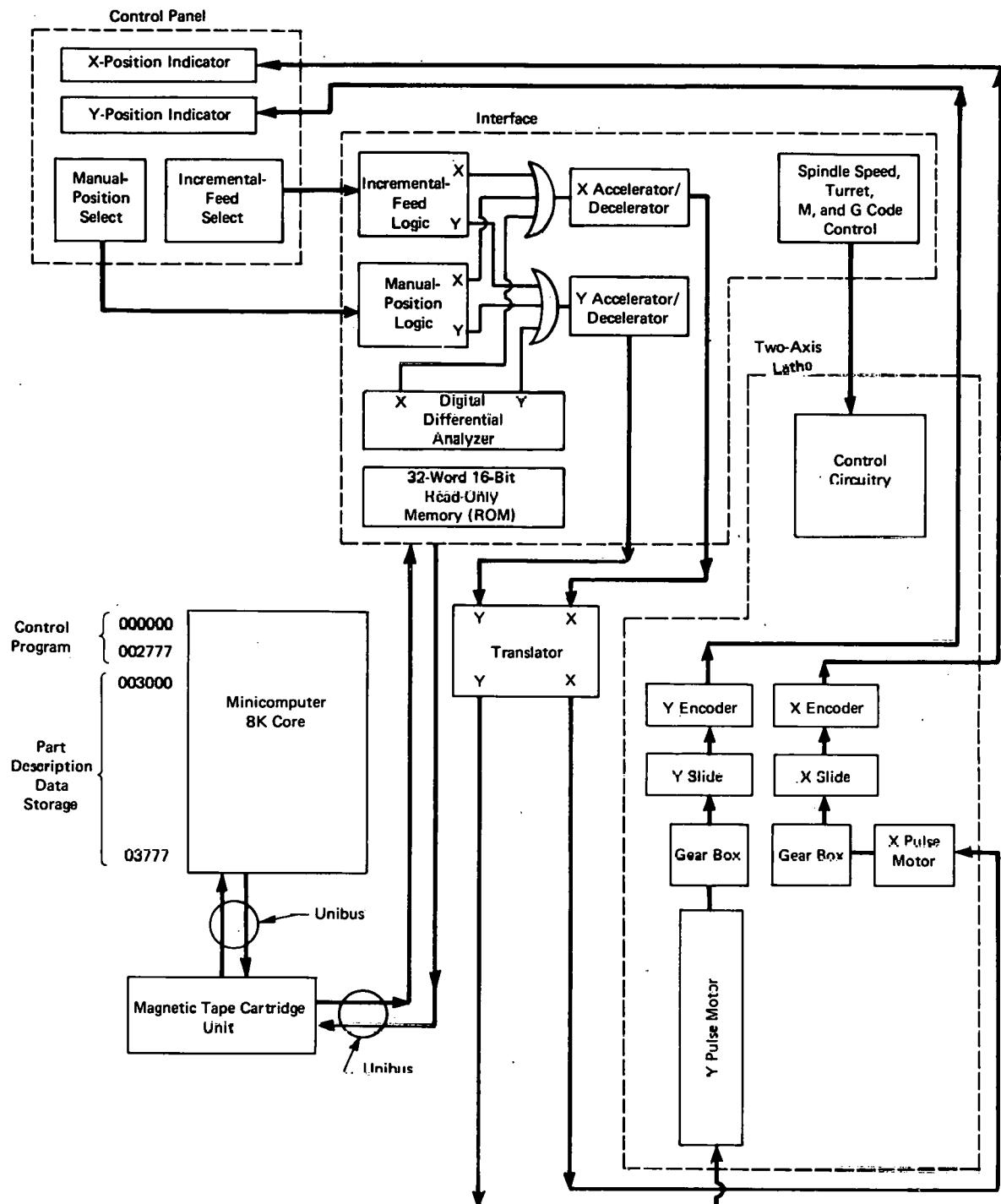


Figure 1. COMPONENTS OF THE COMPUTER-CONTROLLED TWO-AXIS LATHE.

registers. The number of X and Y movement pulses generated correspond directly to the number loaded into the C and D registers.

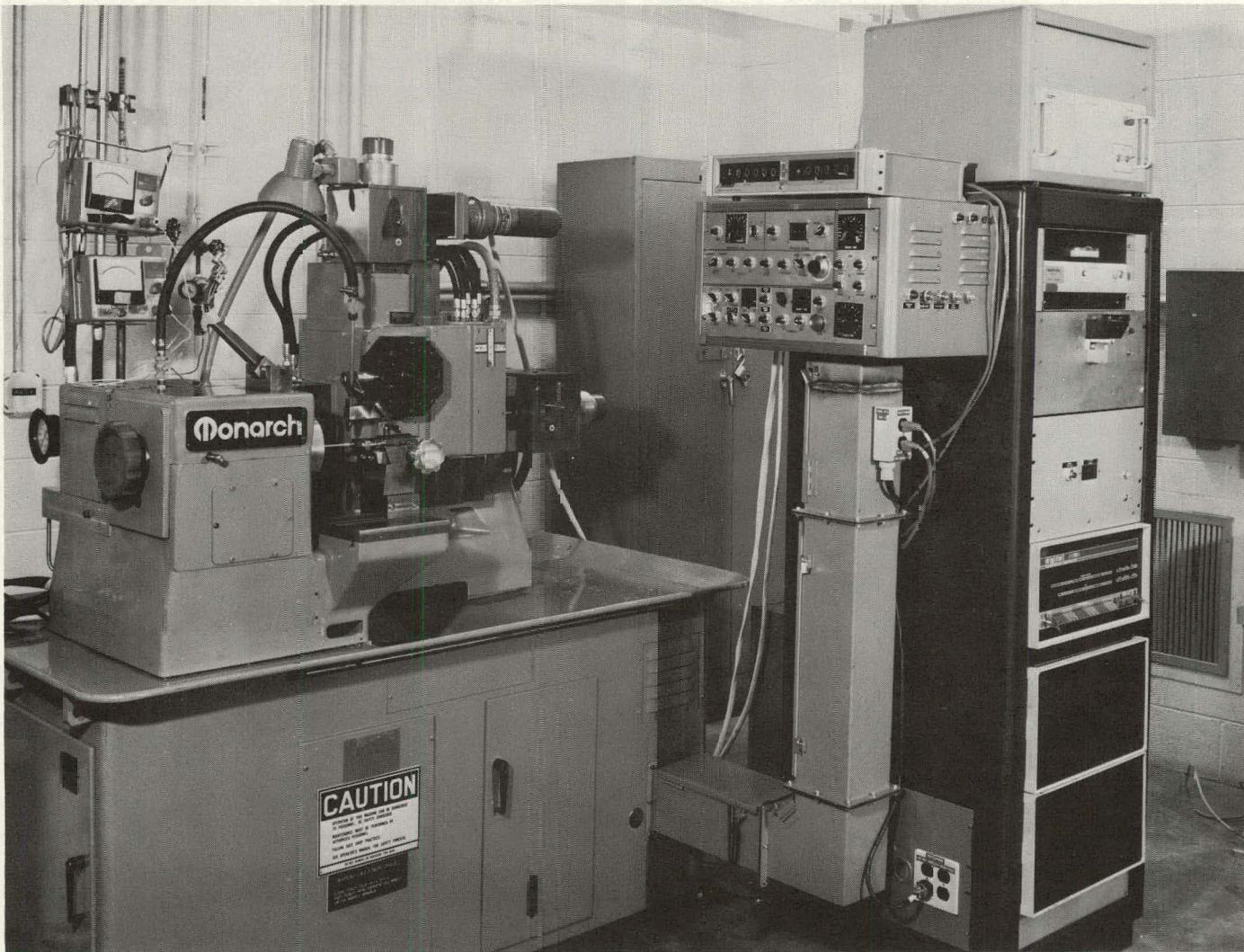
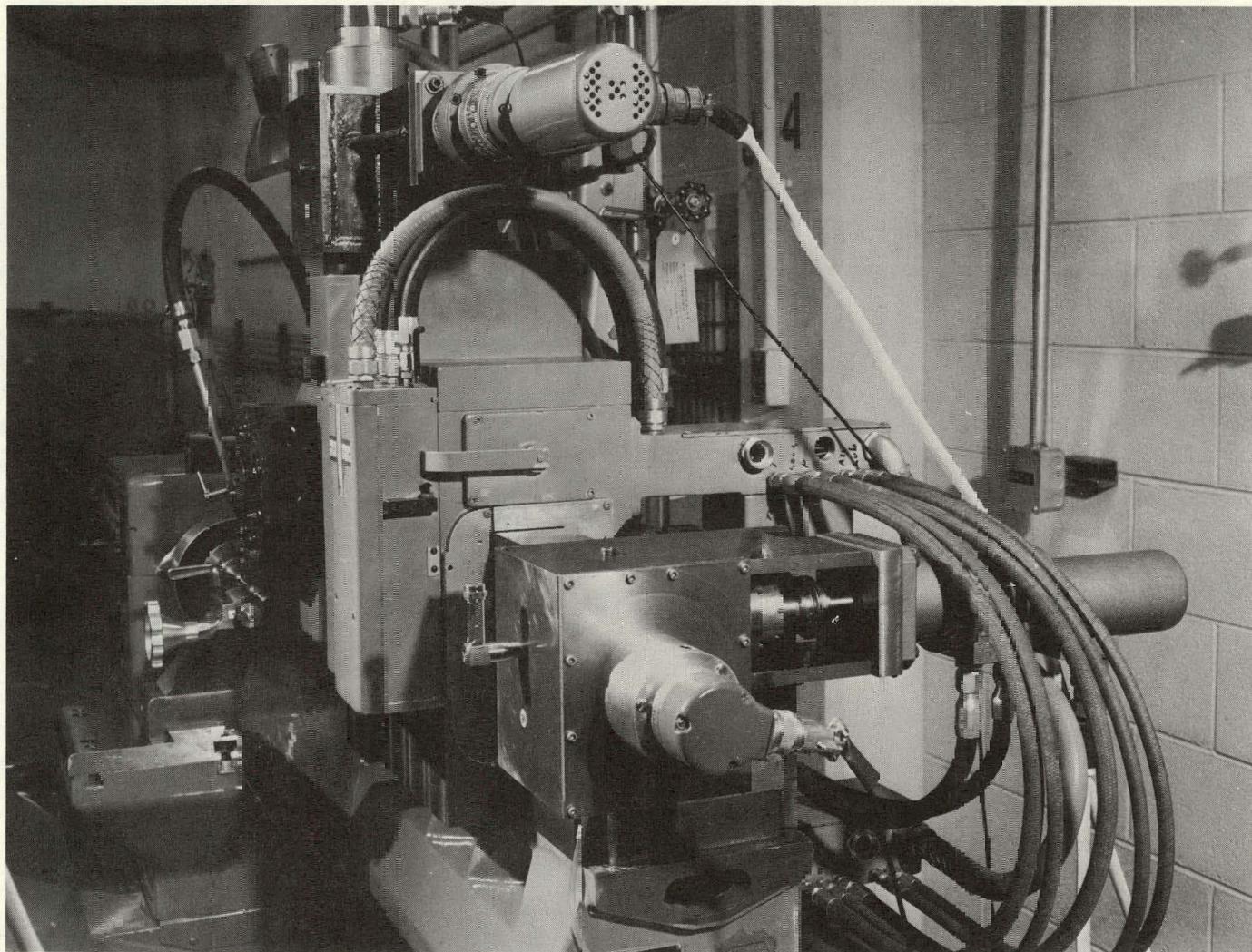


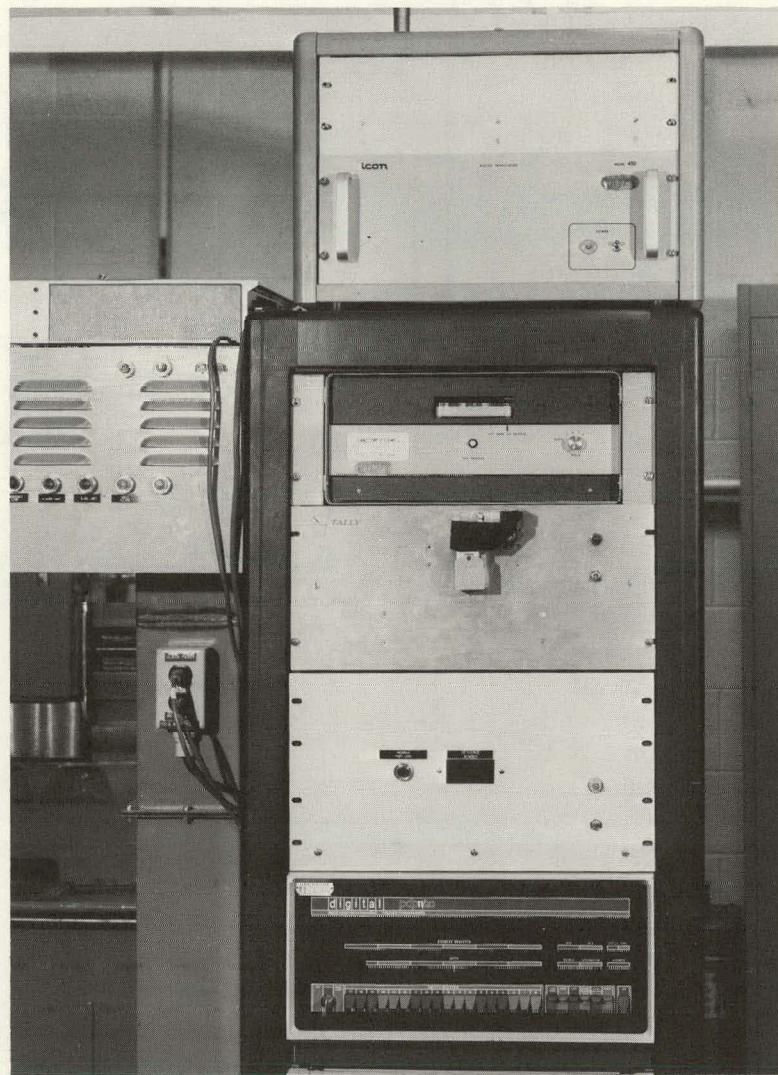
Figure 2. COMPUTER-CONTROLLED TWO-AXIS LATHE.

146609



146607

Figure 3. ELECTROHYDRAULIC PULSE MOTORS FOR THE X AND Y AXES.



146608

Figure 4. SYSTEM CONTROL UNIT.

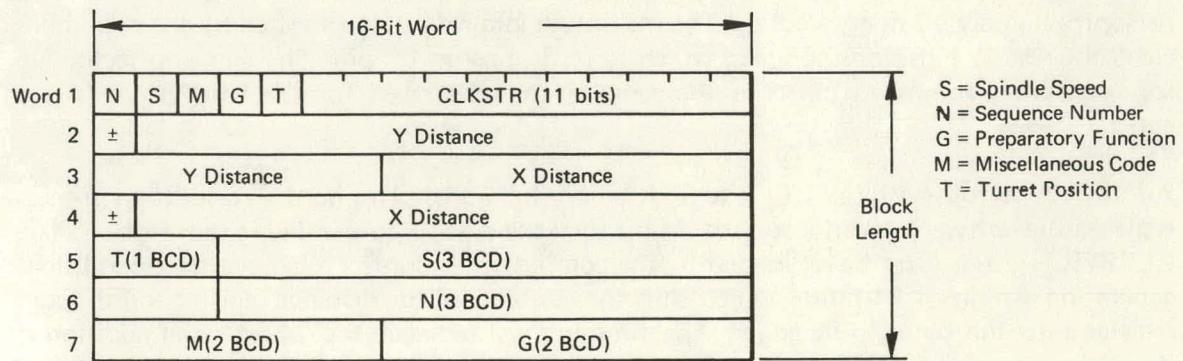


Figure 5. PART DESCRIPTION DATA STORAGE FORMAT.

The magnitude of the CLKSTR number determines the rate at which add-interrupt signals are generated. The time consumed in the software addition routine (see flow diagram in Figure 7) must be less than the time between successive interrupt signals. The software addition time was experimentally determined to be 120 microseconds. Therefore, the smallest absolute CLKSTR value would be 12 (10- $\mu$ sec clock pulse). This time element results in a maximum pulse rate of 8,000 pulses per second or a machining rate of 12 ipm. Initially, this value was thought to be sufficient for all machining requirements and suitable for rapid traverse also. If a command pulse were generated for each software addition, then the maximum machining rate would be 12 ipm. As shown in Figure 8, a command pulse is generated for every addition except the first for a distance value of (0101). For a distance value of (0101), a command pulse requires 240 microseconds or two software additions before being generated. As a result, the maximum contouring rate is reduced by a factor of two, to 6 ipm. This speed imposes a limitation on the maximum traverse rate and possibly restricts some machine-tool cutting speeds.

## HARDWARE ADDER

In an effort to eliminate the restrictions imposed by the hardware/software adder, the pure hardware DDA, shown in Figure 9, was designed. The time per addition is reduced from 120 to approximately 10 microseconds. The maximum ipm rate is now limited by the maximum electrohydraulic pulse motor speed which is 10.3 kpps or 15 ipm. The addition technique for generating command pulses is the same as that described for the hardware/software adder.

Addition is performed serially by a two-bit binary full adder. This hardware-addition process replaces the software addition of the 24-bit distance register to the 24-bit sum register. The CLKSTR register, after being loaded by the computer, is counted to an overflow condition, generating a train of 24 pulses which shift the contents of the distance register and the sum register into the binary full adder. The time interval between the 24-bit serial addition is determined by the CLKSTR value. The adder output is serially loaded into the sum register and also is used to drive the command pulses in conjunction with the T24 pulse.

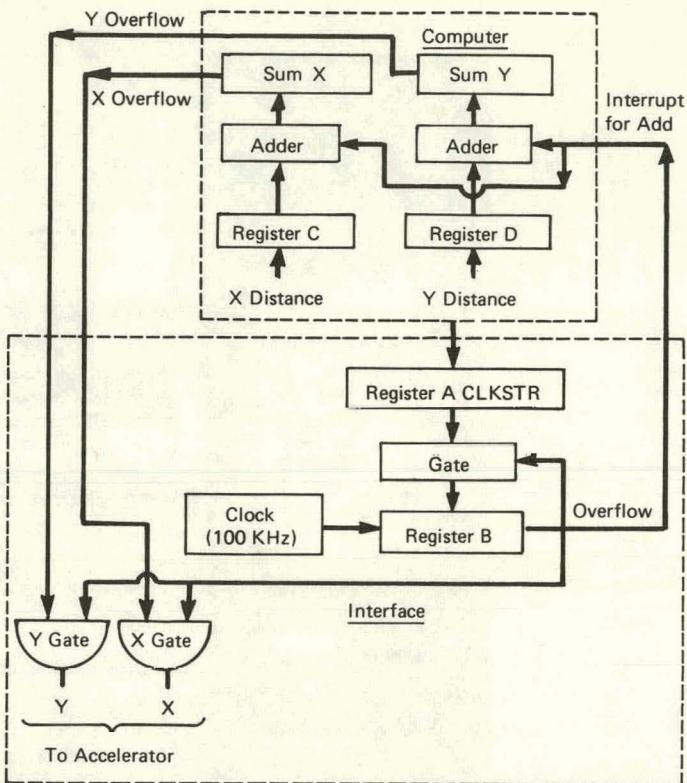


Figure 6. HARDWARE/SOFTWARE DIGITAL DIFFERENTIAL ANALYZER.

## MANUAL CONTROL FUNCTIONS

The hardware and software requirements for the system can be divided into two parts: one is the manual machine control package and the other is the on-line computer control. The manual machine control features are:

1. Feedrate Override - This feature provided a manual control over the system clock. X and Y movement pulses are generated at a rate mutually dependent upon the clock frequency. Thus, by varying the clock frequency, the tool velocity can be changed during machining, but not the tool path.
2. Sequence Number - Various segments in the machining process are given a sequence number. If the operator desires to begin cutting at a particular location, the correct sequence number is loaded into a set of thumbwheel switches, and an "initiate" button is actuated. The computer then resumes the cutting cycle at the proper location.
3. Incremental Feed - A specific number of X or Y movement pulses can be generated from the control panel, thus providing a method for manually positioning the X or Y slides. These pulses are generated at a constant clock frequency of 1.8 KHz.
4. Single-Block Option - This feature enables the opera-

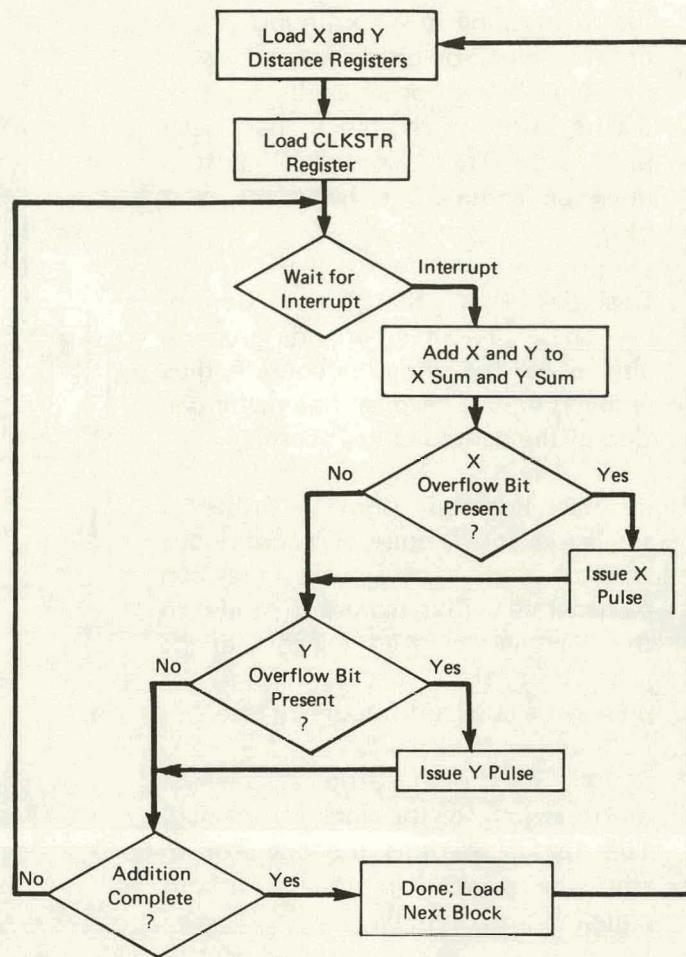


Figure 7. FLOW DIAGRAM FOR THE DIGITAL DIFFERENTIAL ANALYZER SOFTWARE.

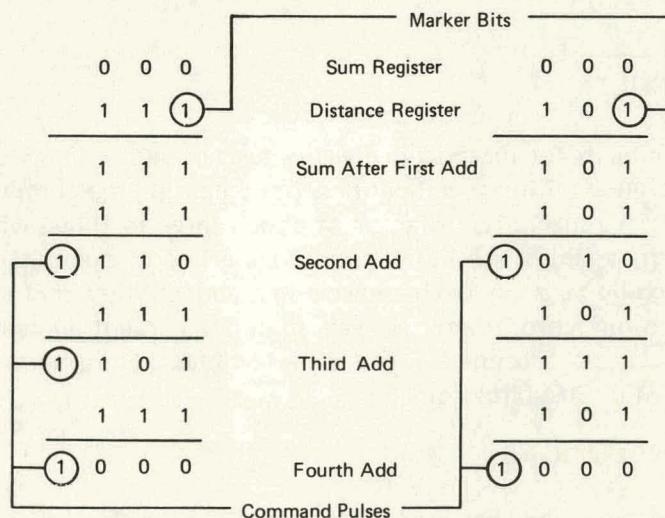


Figure 8. DIGITAL DIFFERENTIAL ANALYZER BINARY ADDITION.

tor to machine in a continuous mode or in a block-by-block fashion. With the single-block option enabled, a halt occurs after each block has been machined. The "continue" button must be actuated to begin the next block.

5. Load Address, Start, and Continue - These features provide remote control for the computer console, thus eliminating the need for any manipulation of the computer keyboard.
6. Manual Position Control - Either a single-movement pulse or a continuous train of X or Y movement pulses can be generated. This feature is similar to the incremental feed except for its generation of a specific number of pulses at a constant frequency rate.
7. Clock Start and Stop - A manual on/off switch for the clock is provided. This feature permits the operator to stop the machining process without stopping the computer.
8. Manual Auto Switch - Pulses generated by the computer and those generated from the manual control package cannot be simultaneously applied to the hydraulic pulse motors.

These manual control features are shown in Figure 10.

## RESULTS

Specimens for material-properties tests of aluminum, steel, stainless steel, and tungsten were machined within specifications. In comparing the computerized machine tool with manually operated machines, a factor of about three in time savings is realized. Due to the toughness of the stainless steel and tungsten specimens, one cutting tool was used for machining the roughing cuts on twelve specimens and another tool was used for the finishing cuts. This machining procedure resulted in a 50 percent increase in tool life. In addition to the tensile-test specimens, four-inch-diameter hemispheres were machined with a true radius deviation of 50 microinches.

## CONCLUSIONS

Replacing the hardware/software DDA with the pure hardware DDA resulted in a considerable increase in unused computer CPU time. After the distance and CLKSTR

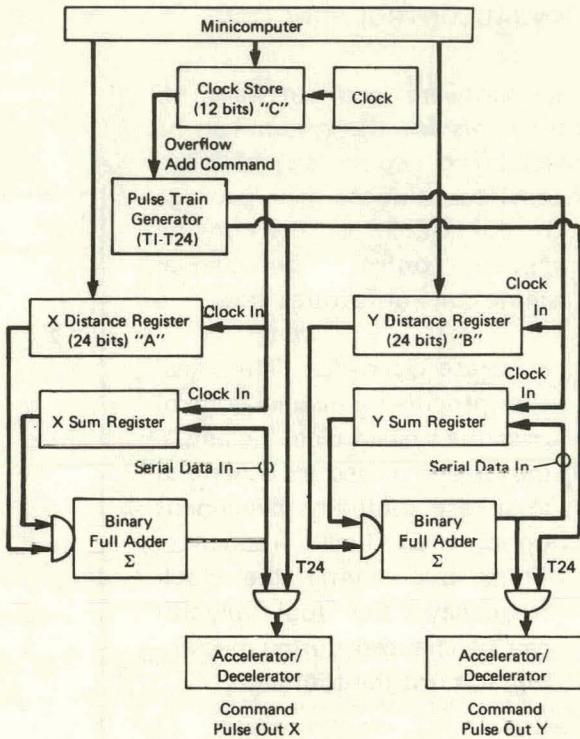
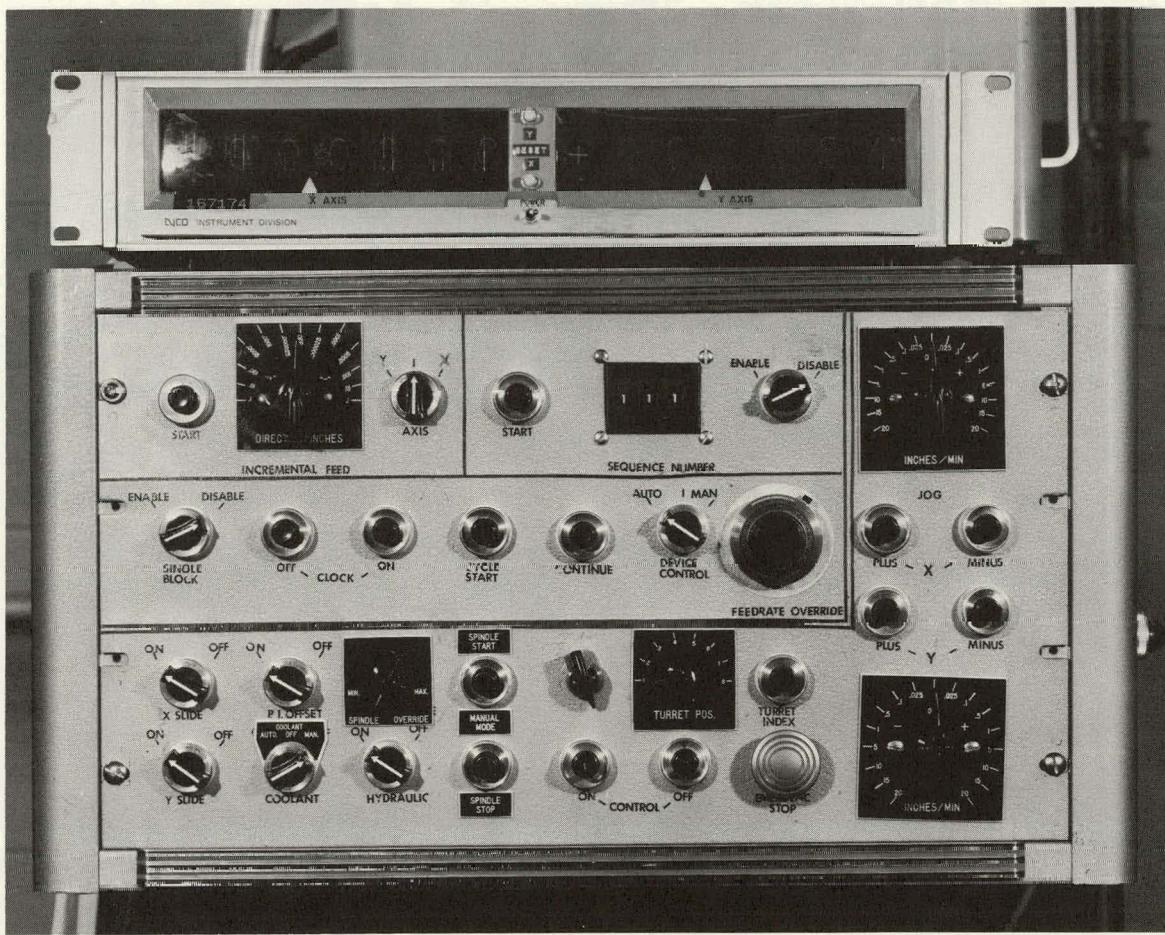


Figure 9. HARDWARE DIGITAL DIFFERENTIAL ANALYZER.



146613

Figure 10. MANUAL CONTROL PANEL.

registers have been loaded, the computer enters a WAIT cycle and remains in the WAIT cycle until the block-completion signal is generated. The possibility of controlling more than one machine tool using the hardware DDA approach seems favorable.

Converting EIA part description data to a binary format results in approximately a 50 percent savings in core-storage requirements. Machining from data stored in core memory rather than on Mylar tape also eliminates the possibility of tape-reader error. Using the minicomputer as the system controller, permits certain adaptive control features to be incorporated into the system in the future. If the backlash for the slides can be determined and remains constant, it can be eliminated by means of software techniques. Detecting a sign change for any axis and inserting the required number of movement pulses would reduce the effective backlash to within the pulse resolution of the system. Manual-control functions provided for the system are basically the same as those found on commercially available NC systems, thus reducing the amount of time required to train an operator.

One of the primary objectives of this project was to determine the feasibility of controlling a machine tool with open-loop control compared with the conventional closed-loop NC

system. Basing a conclusion on the overall performance of this system and the accuracy of the parts machined, open-loop control is at least as good as closed-loop control.

## ACKNOWLEDGEMENTS

The author wishes to thank T. L. Williams of the Development Division for his part in this project. Several of the manual control features and the basic accelerator/decelerator circuitry were provided by him. The author also wishes to thank C. M. Lay of the Fabrication Division for his contribution to the project, specifically for modifying the accelerator/decelerator circuitry and providing assistance in the design of the hardware DDA.

A word of thanks goes to O. Smelcher of the Fabrication Division for his involvement in the mechanical phase of this project, which included a complete mechanical retrofit of the lathe. The original drive gears were replaced by precision gearing making possible a 25-microinch resolution. He was also instrumental in obtaining EIA data tapes for the various parts to be machined and in supervising the machining operations. The author also wishes to thank Lavoy Lane of the Maintenance Division for his contribution in marrying the control system to the lathe machine tool. Appreciation is also expressed to A. E. Stephens for his technical guidance and supervision during the project.

## APPENDIX A

**SOFTWARE FOR CONVERTING AN EIA RS-274-B TAPE TO A MINICOMPUTER BINARY FORMAT****Introduction**

This software performs the conversion of an EIA RS-274-B tape to a binary data table format suitable for the minicomputer. Each data block of the EIA tape is converted into a variable-length (4 to 7 words, as shown in Figure 1) 16-bit word data block.

**Operation**

Operation procedure for generating a binary data tape is as follows:

1. Insert Cartridge Tape 2 into the read station of the cartridge-tape unit.
2. Select line number one.
3. Press the "load-tape" button.
4. Place the EIA tape in the tape reader of the teletypewriter. Set the teletypewriter control to ON LINE and the tape punch control to ON.
5. After the EIA conversion program has been loaded from the cartridge tape, set the computer console switch register to 000200 and depress the "load add" and "start" buttons. The converted EIA tape will then be punched onto paper tape by the teletypewriter.

Table A-1 lists the memory allocations of software for converting an EIA RS-274-B tape to a minicomputer binary format.

Table A-1  
MEMORY ALLOCATION OF SOFTWARE FOR  
CONVERTING AN EIA RS-274-B  
TAPE TO A MINICOMPUTER  
BINARY FORMAT

Location	Function
000200-001440	Initialization and Character Store Routine
001442-001658 002654-005422	Data Convert Routine
001660-002652	Data Output Routine

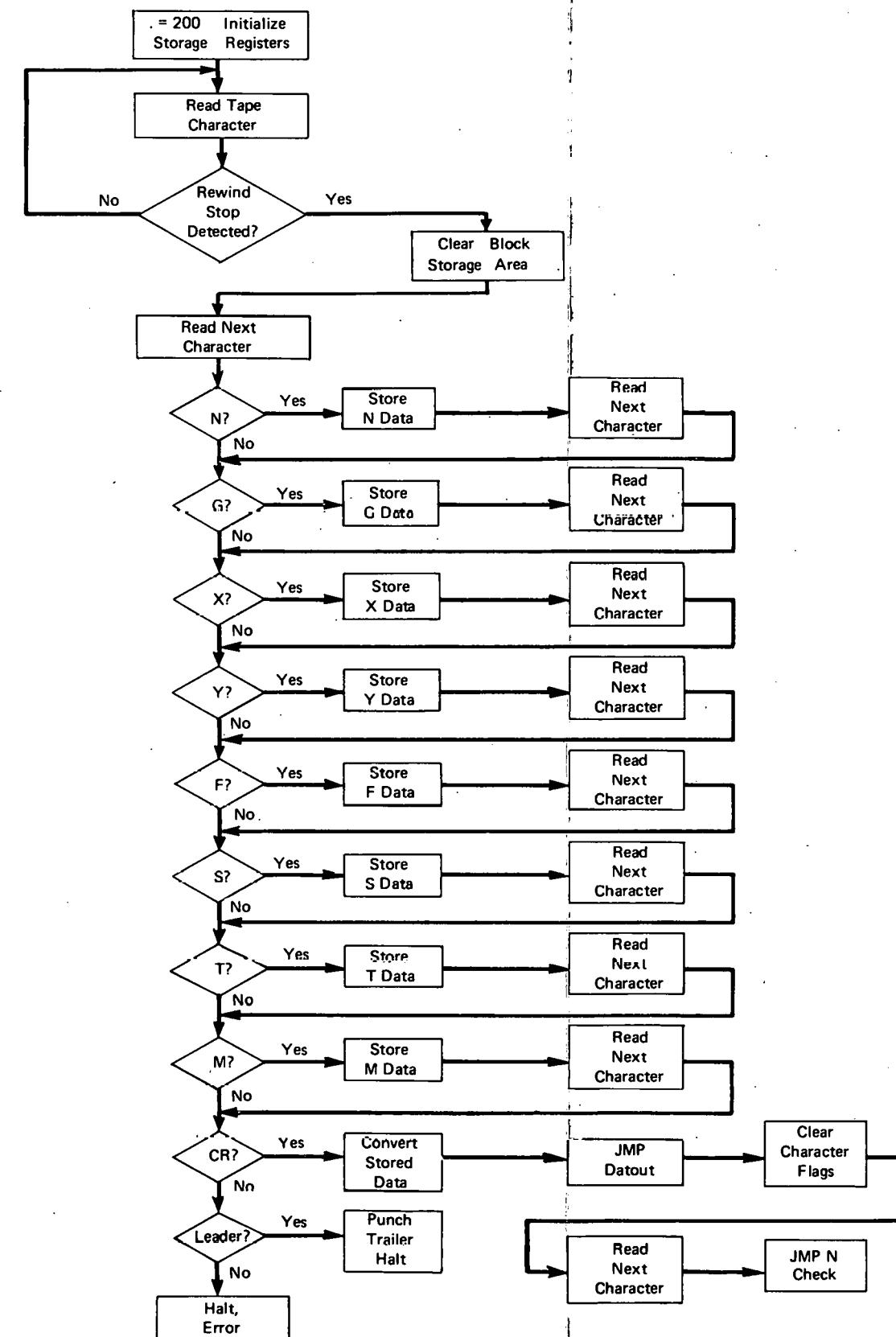
### Subsection Descriptions

**Title - Initialization and Character Store Routine**

**Location - 000200-001440**

**Purpose -** EIA characters are stored in block format as read by the teletypewriter

**Description -** The EIA tape is loaded into the reader station of the high-speed tape reader. The leader preceding the EIA data should be placed over the read head. No data are stored until a rewind stop code is detected, after which the first data block is stored. When an end-of-block code is detected, the data just stored is converted to the binary format for the minicomputer and outputted via the teletypewriter punch.



## Title - Data Convert Routine

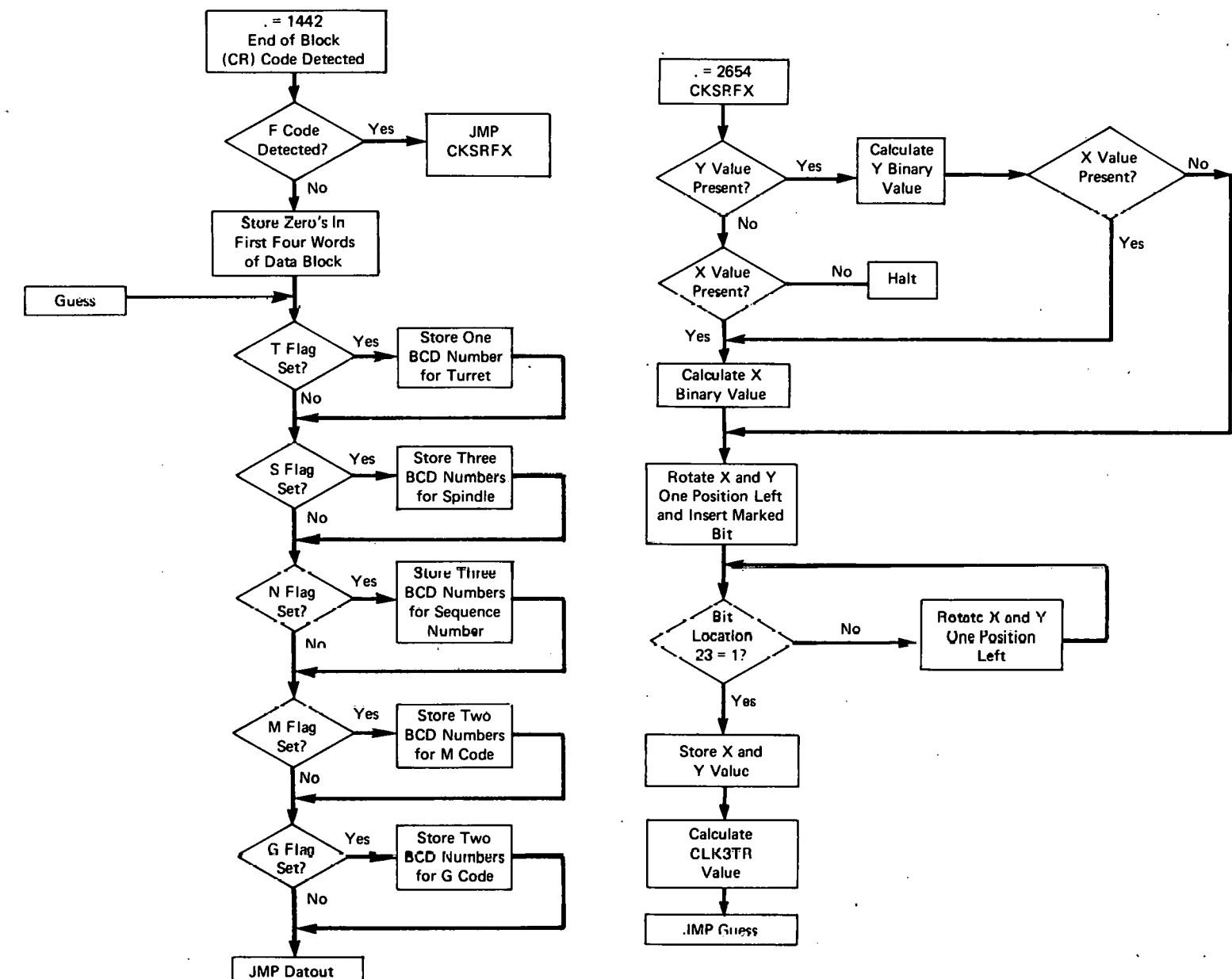
Location - (1442-1656), (2654-5422)

**Purpose** - Data stored from the previous EIA block are converted to a binary format and stored in a variable-length block starting in Location 3000.

**Description** - T, S, N, M, and G functions are of fixed length and can be converted to a BCD value and stored without much difficulty. X, Y, and F functions are of variable length and require certain arithmetic operations before the binary values can be stored. X and Y distance values must be divided by 25 to obtain the number of 25-microinch pulses needed for a specified movement. The CLKSTR value is calculated from the feedrate number, F, and the longest distance moved on the X or Y slide.

$$\text{CLKSTR} = \left[ \left( \frac{6 \times 10^8}{F} \right) \div (2^N) \right] \div [8]$$

N is the number of bits of the larger X or Y binary value. The constant "8" is derived from the 8-microsecond clock period of the hardware adder.



**Title - Data Output Routine**

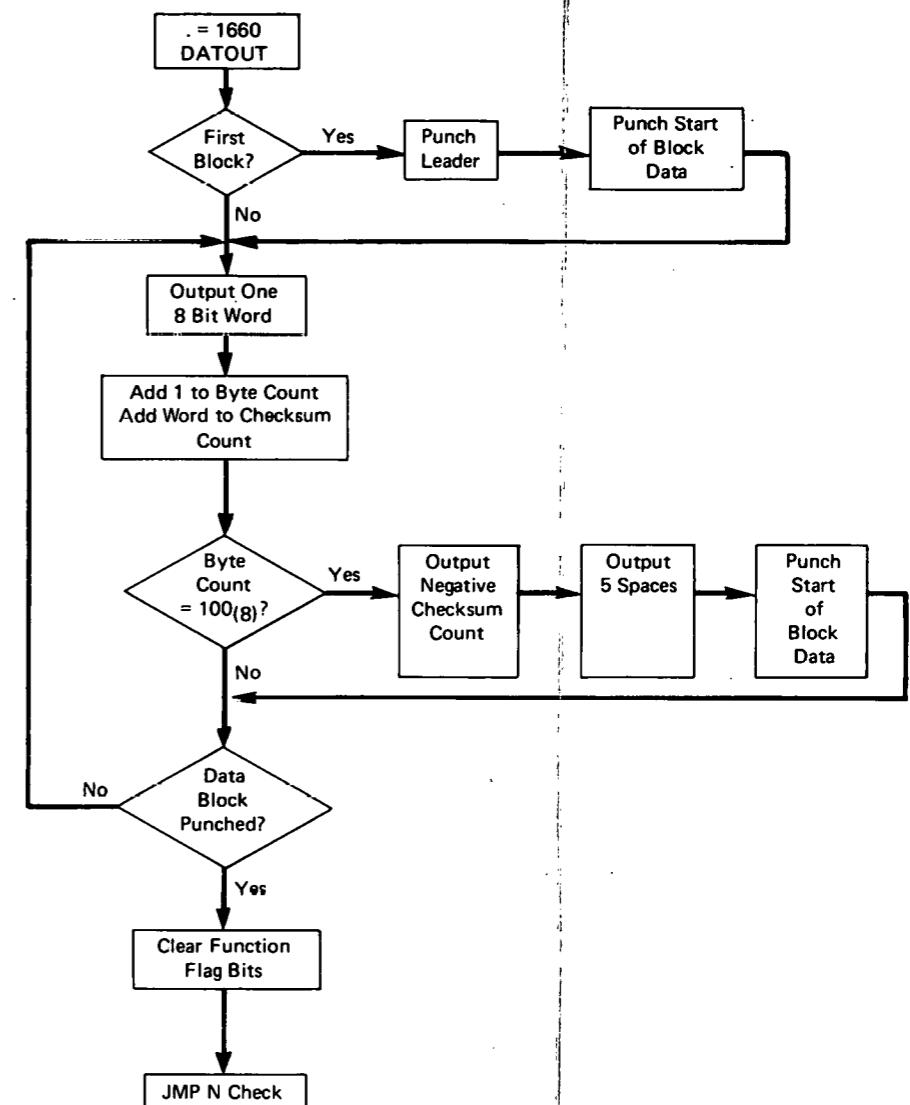
**Location - 1660-2652**

**Purpose** - As each EIA block is converted to a binary block, the data are outputted via the teletypewriter punch.

**Description** - After the rewind stop code is detected at the beginning of the EIA tape, one block of data is stored. When the end of block code (CR) is detected, indicating an end of the current EIA block, the data are converted to a binary format. Upon completion of the conversion process, the data are outputted by the teletypewriter punch. Care must be taken to format the outgoing data correctly. A block of data punched on paper tape in absolute binary format has the following form:

FRAME 1	001	Start Frame
2	000	Null Frame
3	xxx	Byte Count (low 8 bits)
4	xxx	Byte Count (high 8 bits)
5	yyy	Load Address (low 8 bits)
6	yyy	Load Address (high 8 bits)
		Data are
		placed
		here.
zzz		Last frame contains a block check sum.

A program on paper tape may consist of one or more blocks of data. Each block having a byte count (Frames 3 and 4) greater than six will cause subsequent data to be loaded into core starting at the address specified in Frames 5 and 6. The byte count is a positive integer containing the total number of bytes in the block, excluding the check sum. This tape is formatted so that each block contains 100<sub>(8)</sub> bytes. The check sum is accumulated for each block, negated, and punched at the end of every block.



## APPENDIX B

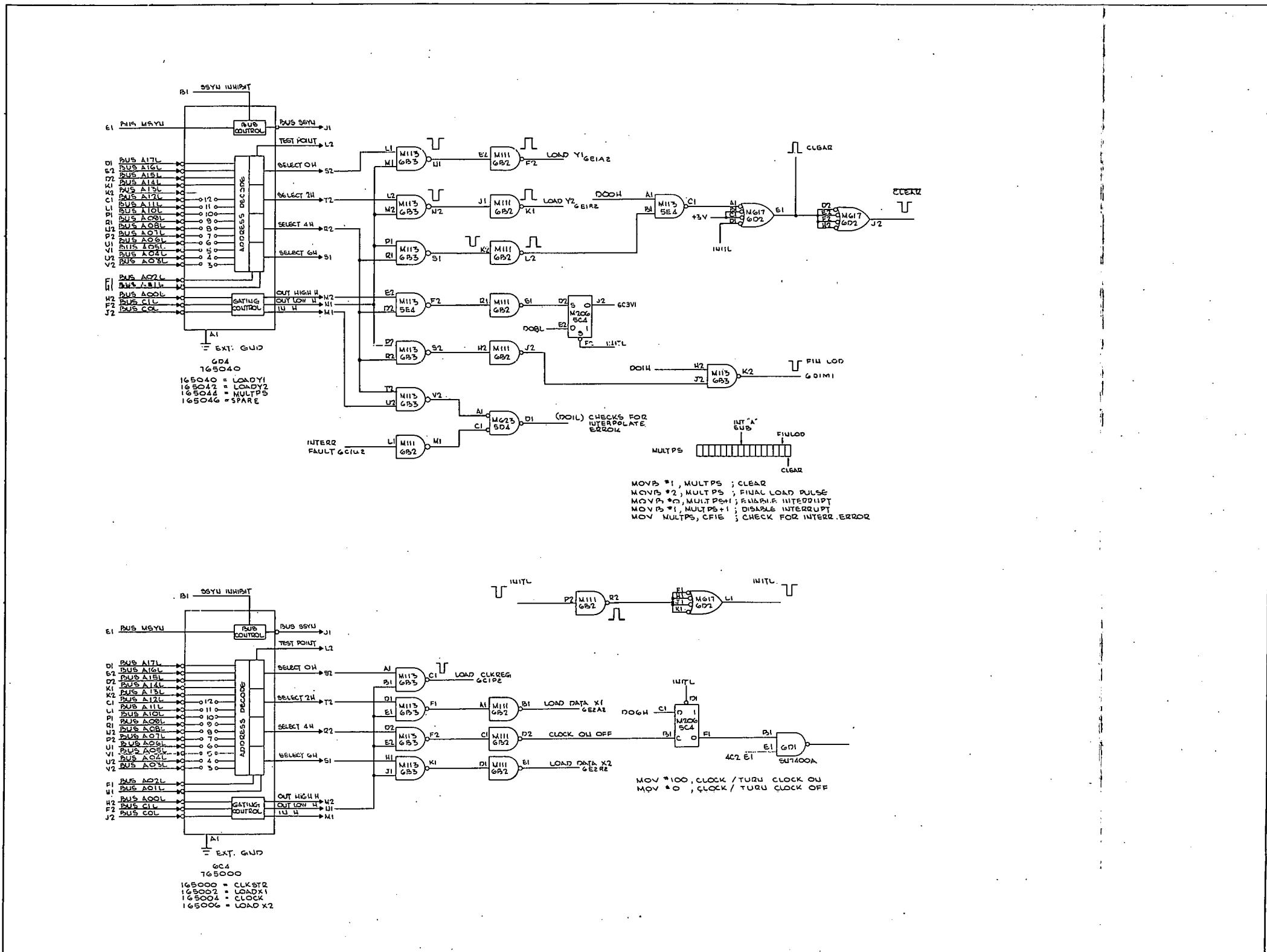
**CARTRIDGE TAPE UNIT WITH READ-ONLY-MEMORY (ROM) LOAD PROGRAM**

Standard NC tape data are converted to a binary data table format, as explained in Appendix A. When the binary conversion is complete, the data can be stored onto the cartridge magnetic tape. The binary tape generated from the conversion process is read into core memory using the high-speed reader. After the tape has been read in correctly, the desired cartridge is placed in the magnetic tape unit, and Track 1, 2, 3, or 4 is selected. The switch register is then set to 035000, and the "load address" and "start" keys are depressed.

The magnetic tape unit dump program was altered so that the minicomputer's entire 8K of core memory is loaded onto the selected 8K track. The cartridge unit load program was also altered and reduced in length to 31 words. A 32-word (16-bit length) ROM was programmed with the 31-word load program. A remote "load address and start" feature was designed for the minicomputer with a hardwired starting address of 037000, which is the starting address of the loader program. Therefore, in order to load the operate program and the part description data for any part to be machined, the operator selects the correct cartridge and the proper track, loads the cartridge, and depresses the "load program" push button located on the chassis above the computer console. The red light of the push button will remain on until 8K of core has been loaded. This feature makes it possible for the machine operator to load the desired part description data without touching the computer console.

## APPENDIX C

## ENGINEERING DRAWINGS



**Figure C-1. MONARCH MACHINE TOOL CONTROL II. (M105 Control Commands)**

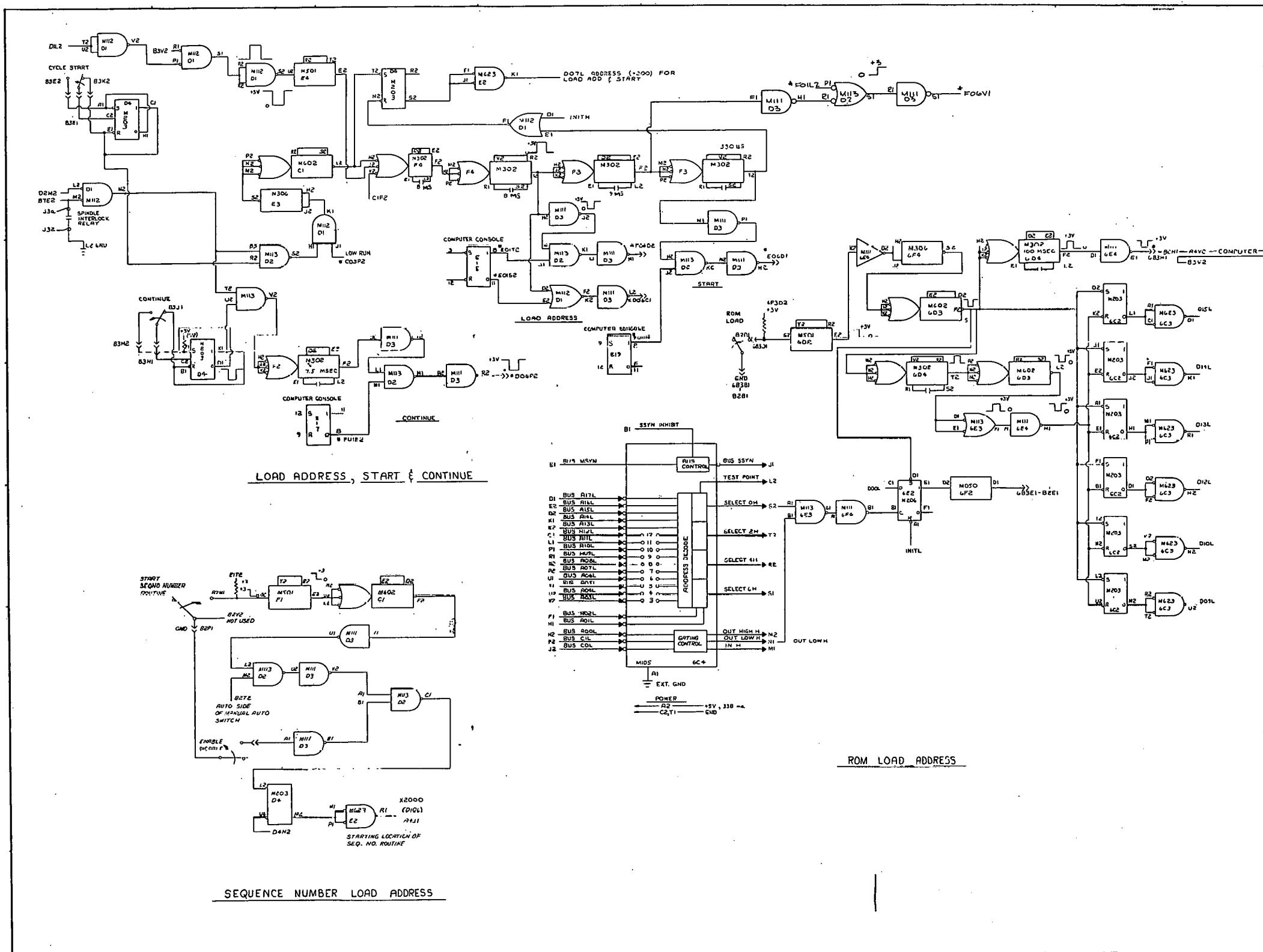


Figure C-2. MONARCH MACHINE TOOL CONTROL II. (Remote Load Add, Start, and Continue)

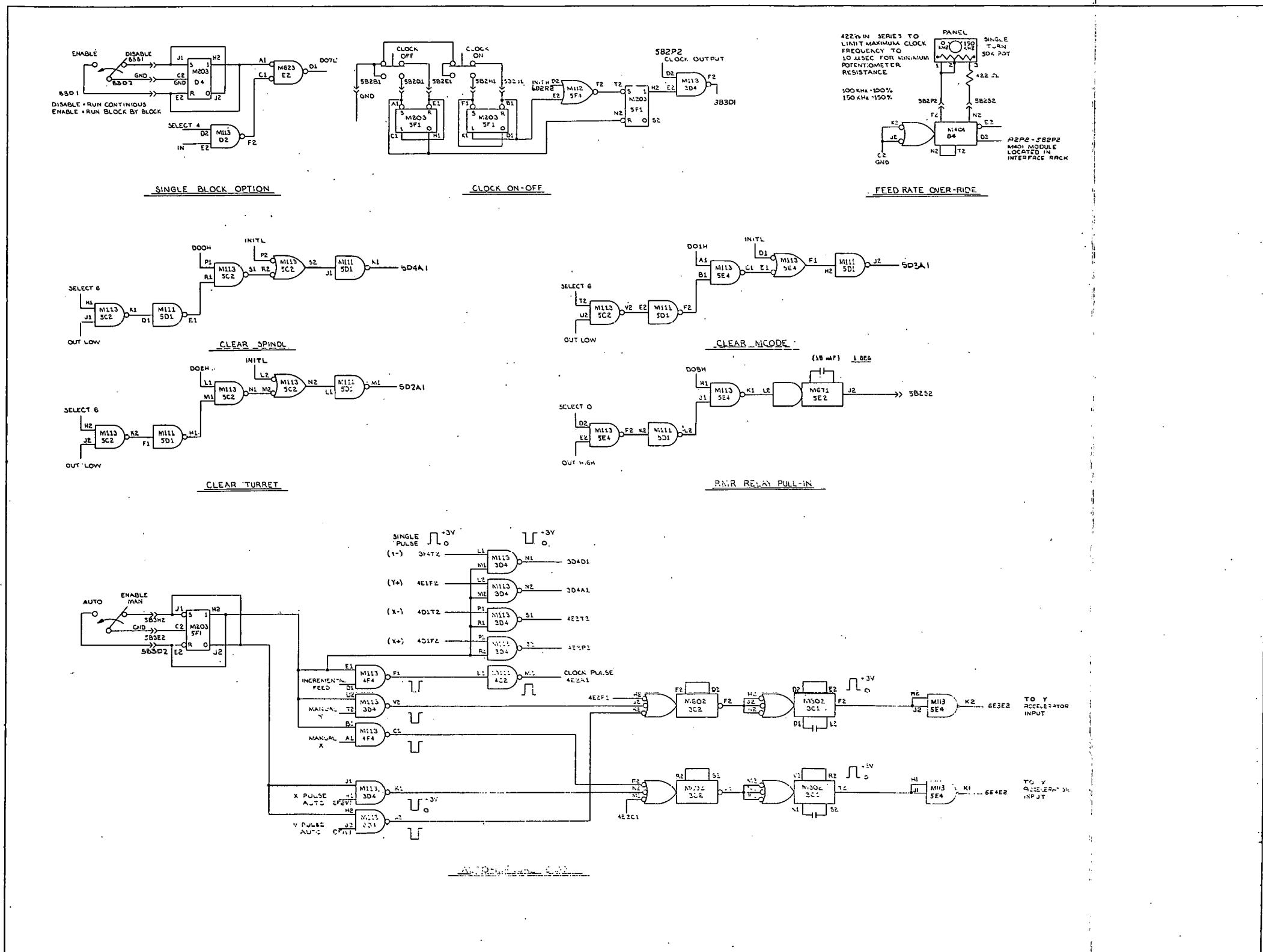


Figure C-3. MONARCH MACHINE TOOL CONTROL II. (Miscellaneous Functions)

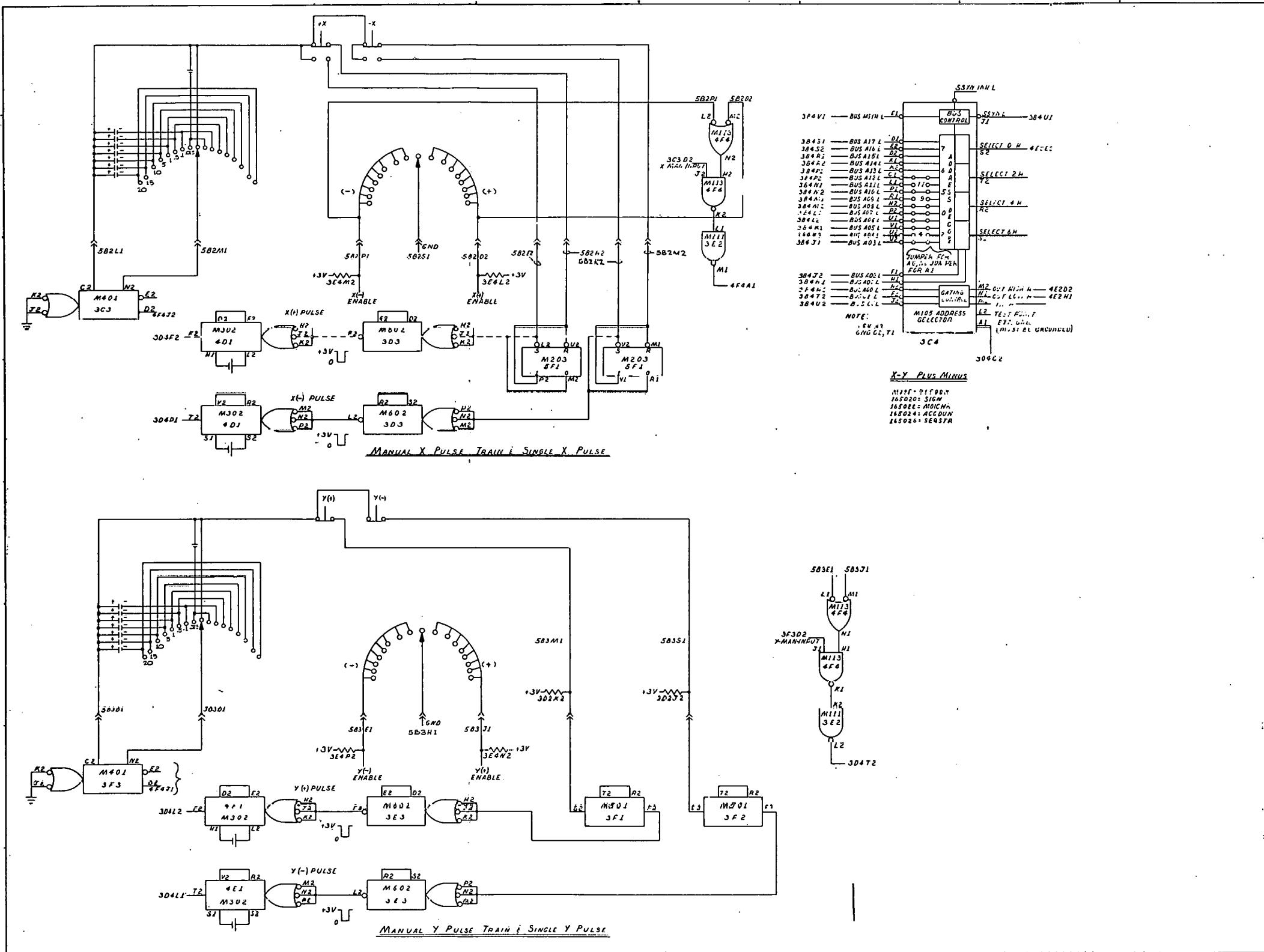


Figure C-4. MONARCH MACHINE TOOL CONTROL II. (Manual X-Y Pulse Train)

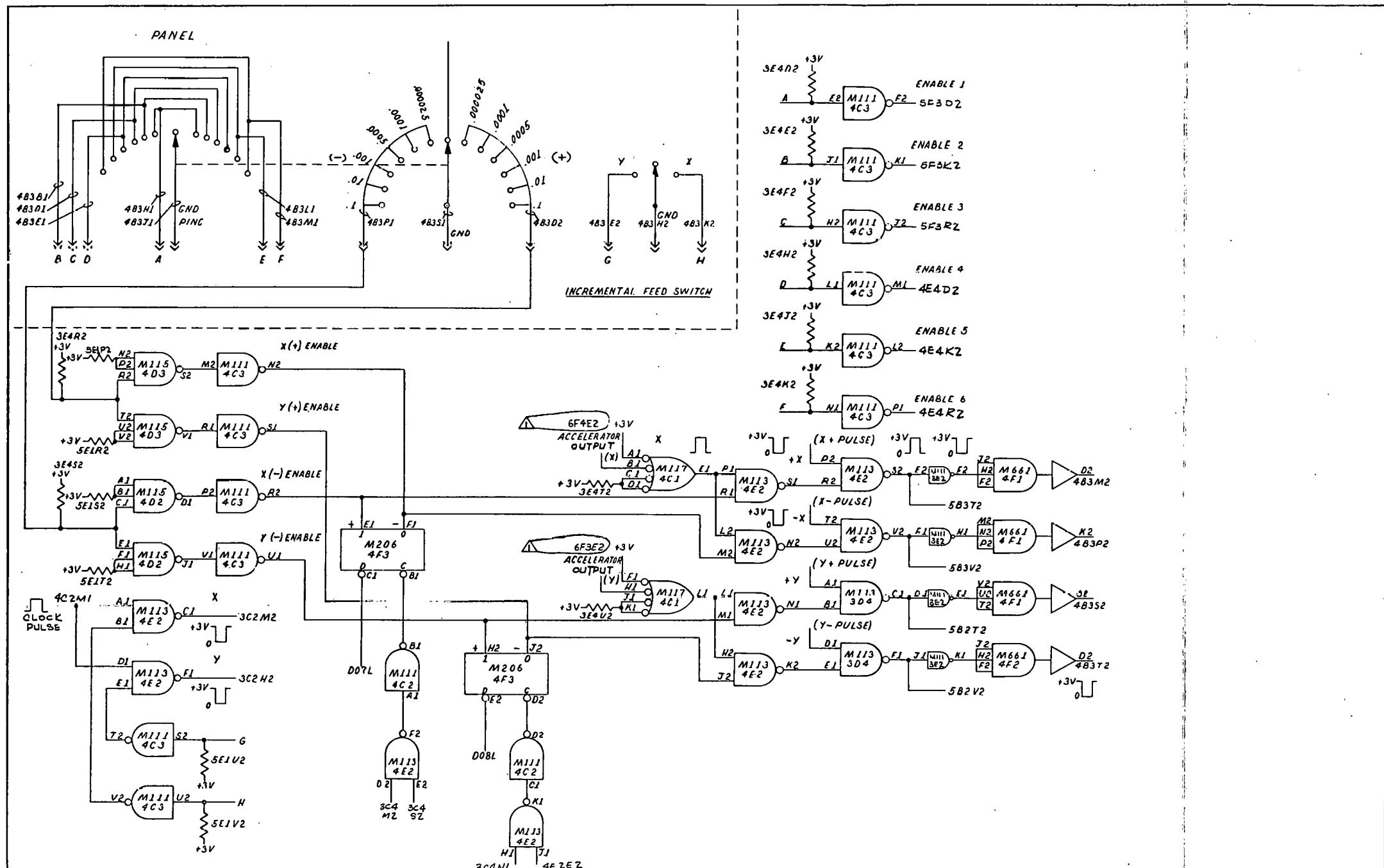


Figure C-5. MONARCH MACHINE TOOL CONTROL II. (Manual-Computer Signal)

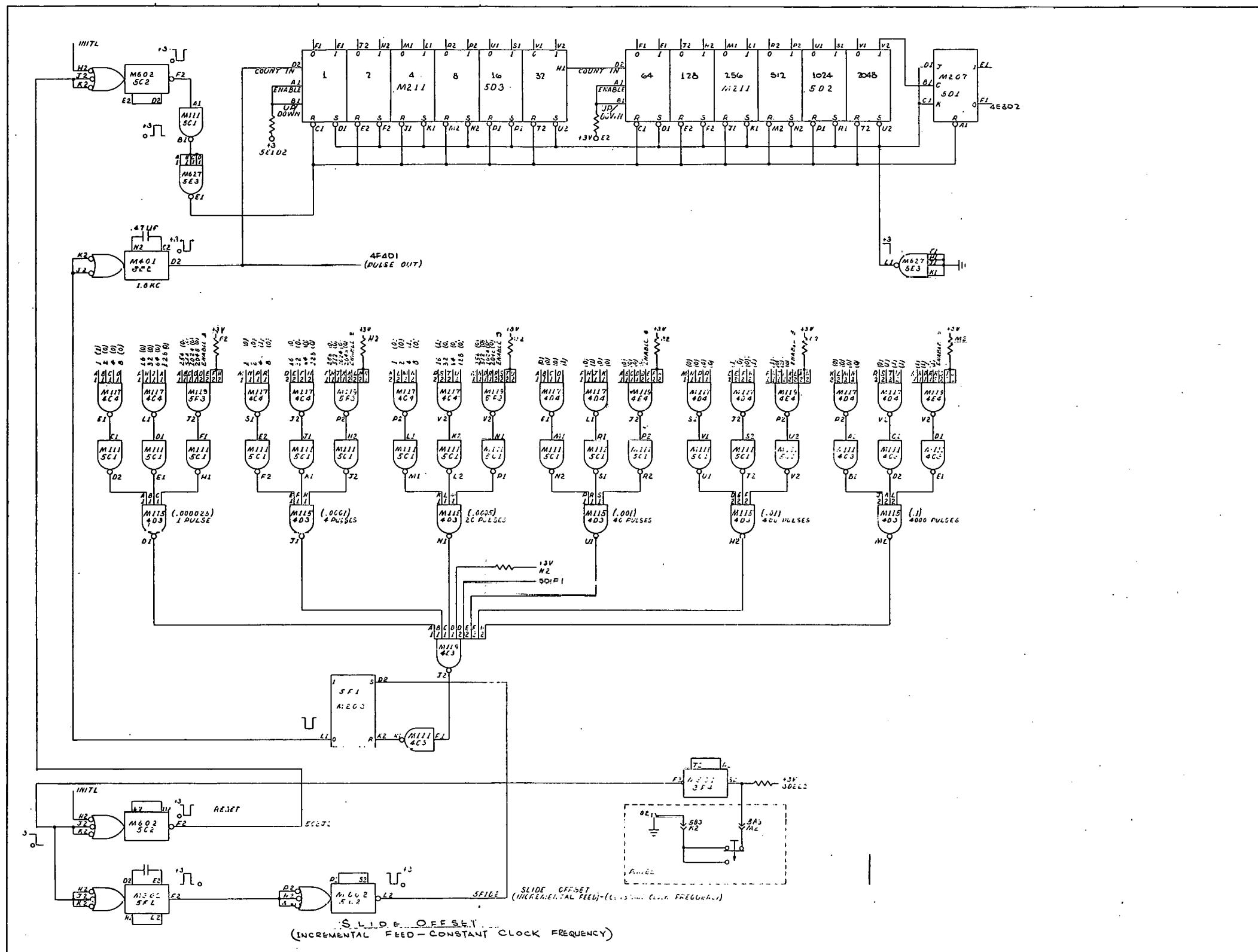


Figure C-6. MONARCH MACHINE TOOL CONTROL II. (Incremental Feed)

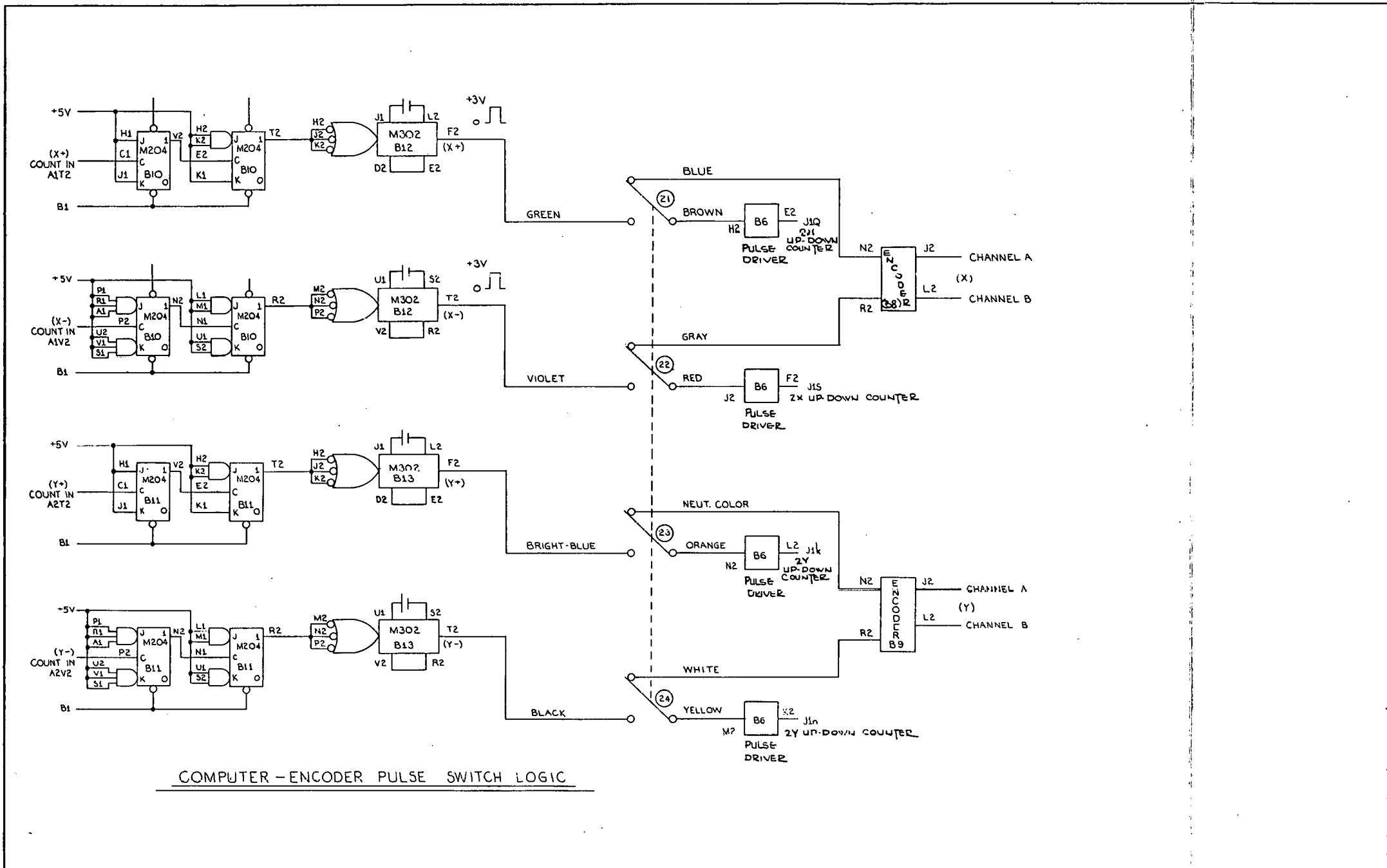


Figure C-7. MONARCH MACHINE TOOL CONTROL II. (Computer-Encoder Pulse)

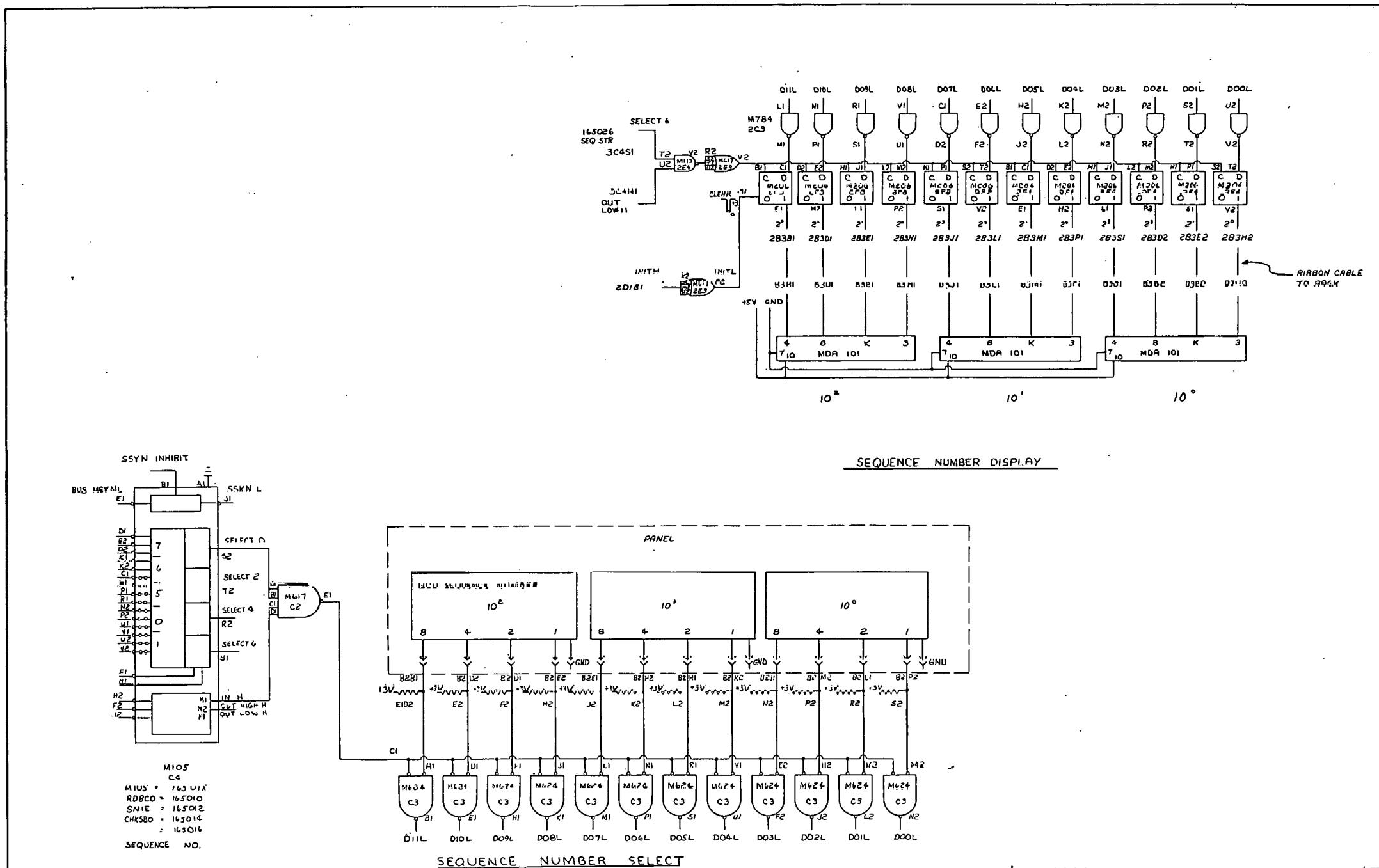
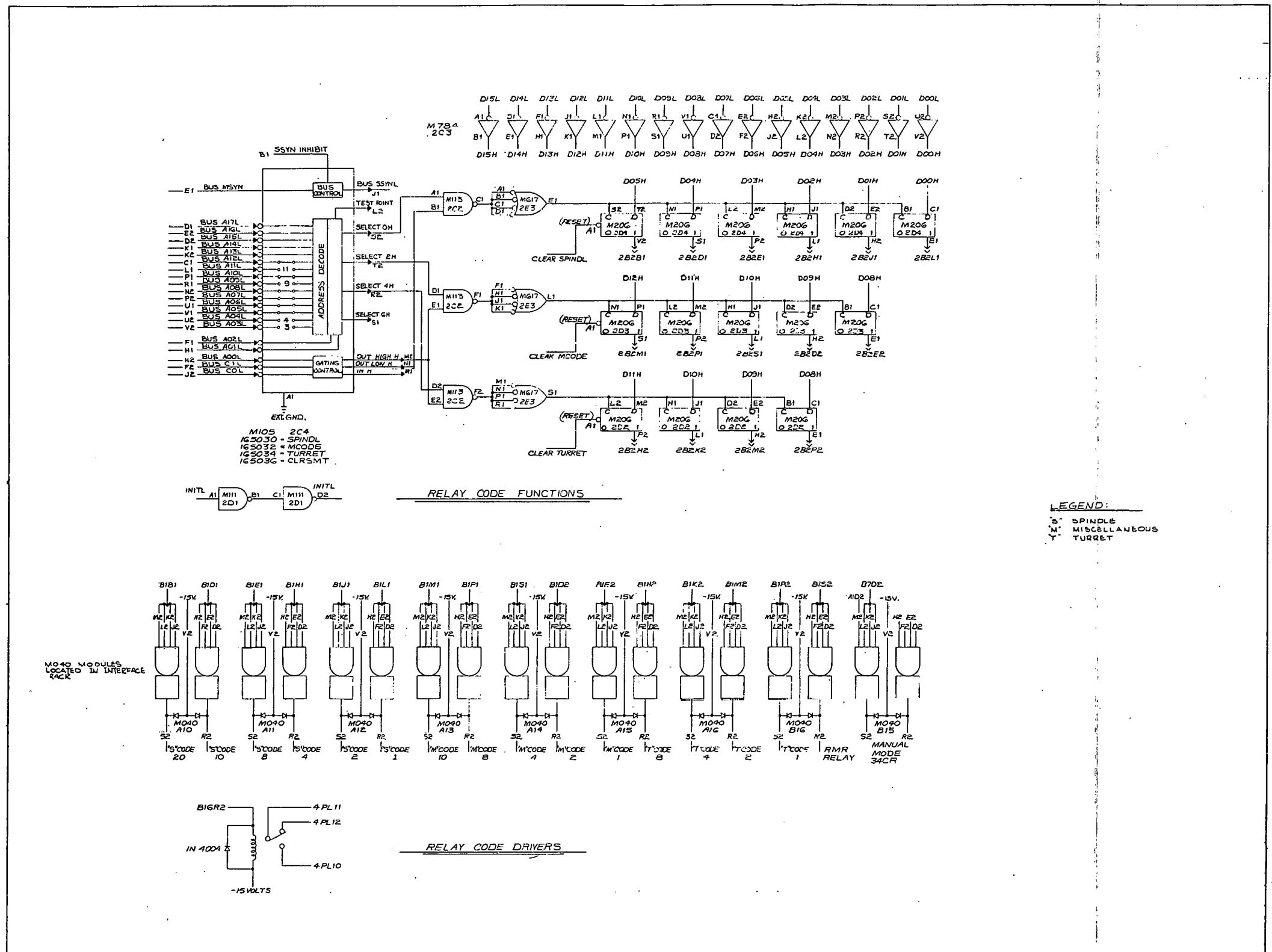
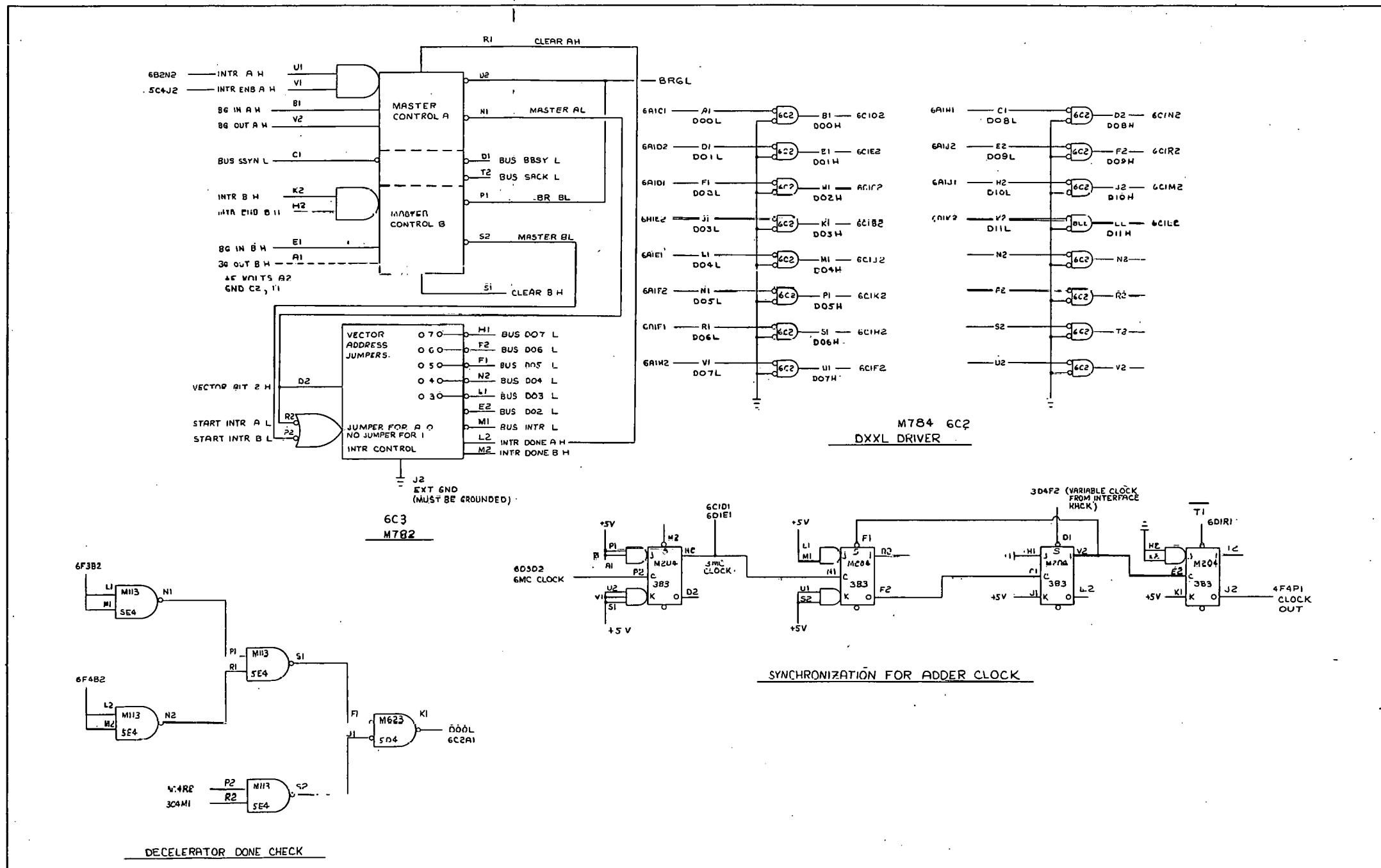


Figure C-8. MONARCH MACHINE TOOL CONTROL II. (Sequence Number Select and Display)



**Figure C 9. MONARCH MACHINE TOOL CONTROL II. (Relay Code Function)**



**Figure C-10. MONARCH MACHINE TOOL CONTROL II. (Miscellaneous Logic)**

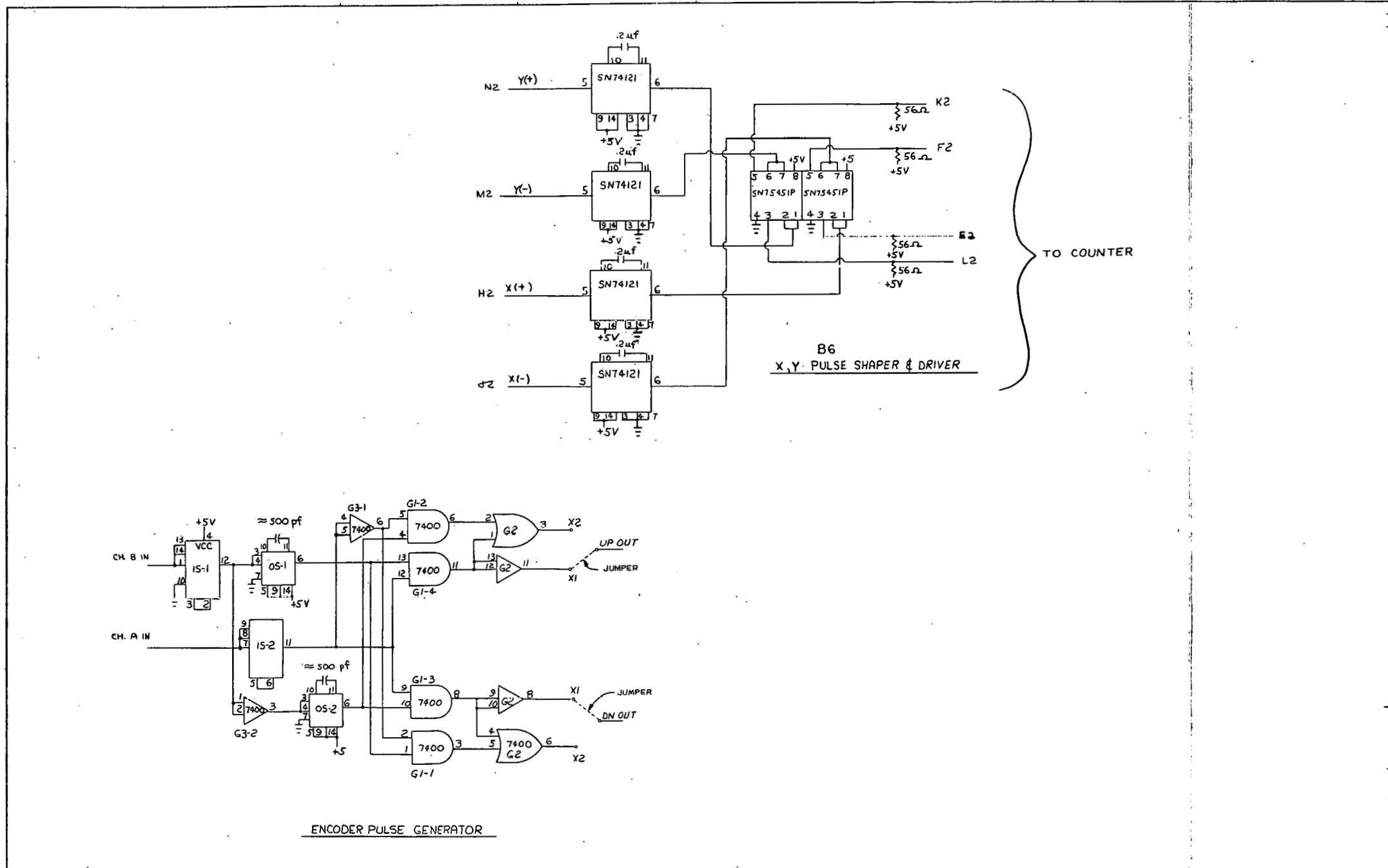
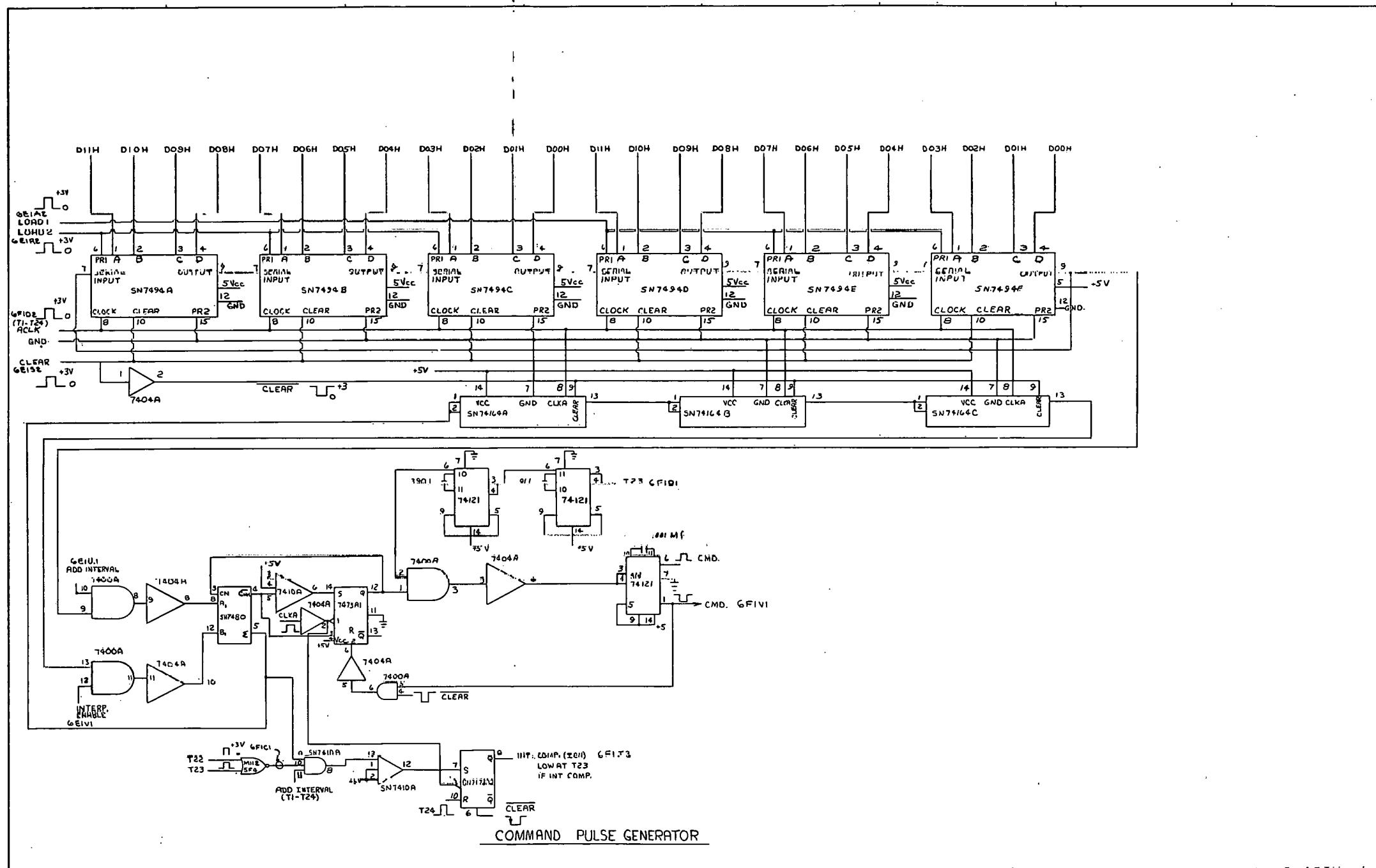


Figure C-11. MONARCH MACHINE TOOL CONTROL II. (X-Y Pulse Drive and Encoder Board)



**Figure C-12. MONARCH MACHINE TOOL CONTROL II. (Command Pulse Generator)**

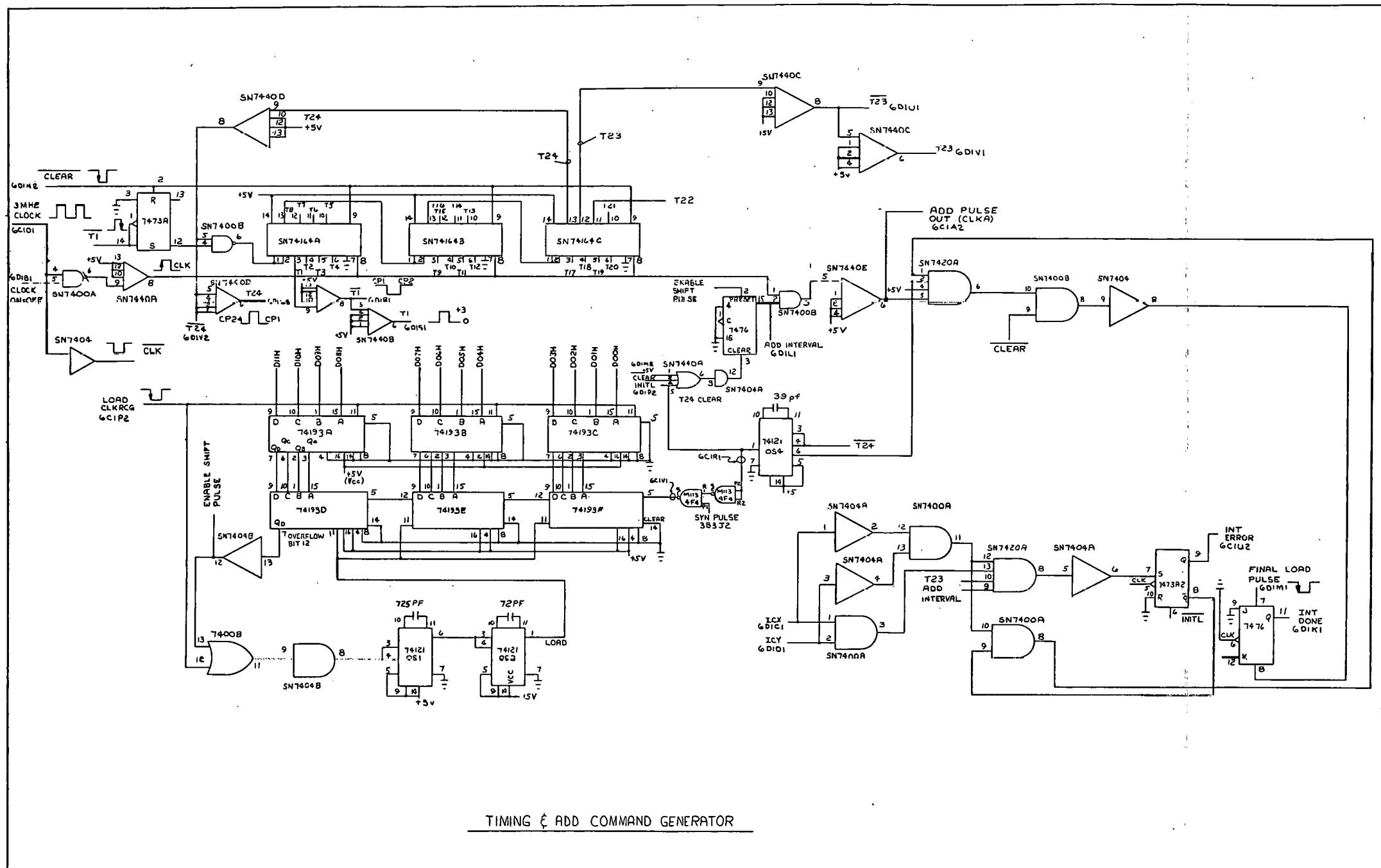


Figure C-13. MONARCH MACHINE TOOL CONTROL II. (Timing and Add Command Generator)

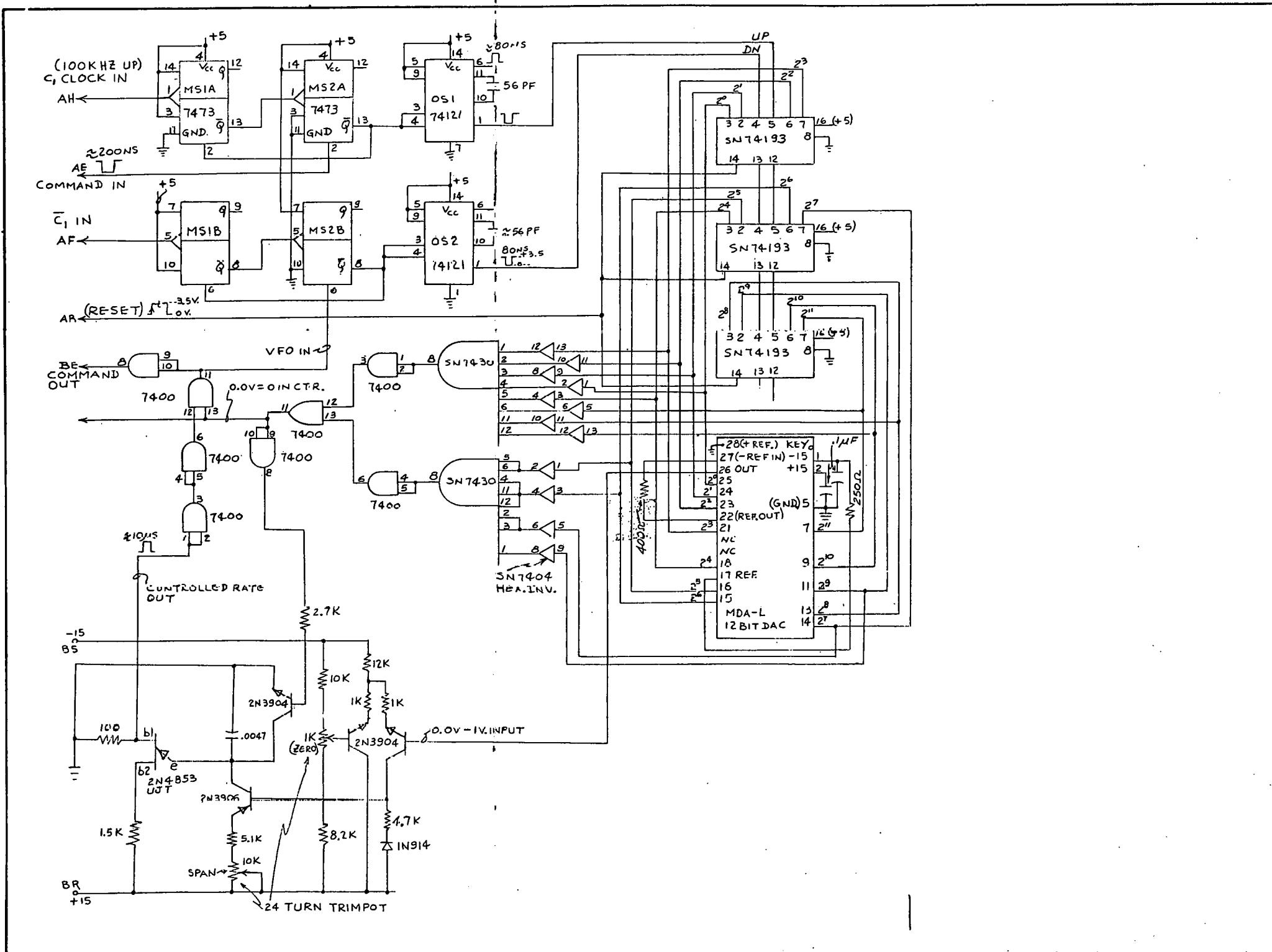


Figure C-14. MONARCH MACHINE TOOL CONTROL II. (Auto Accel/Decel DCC Machine)

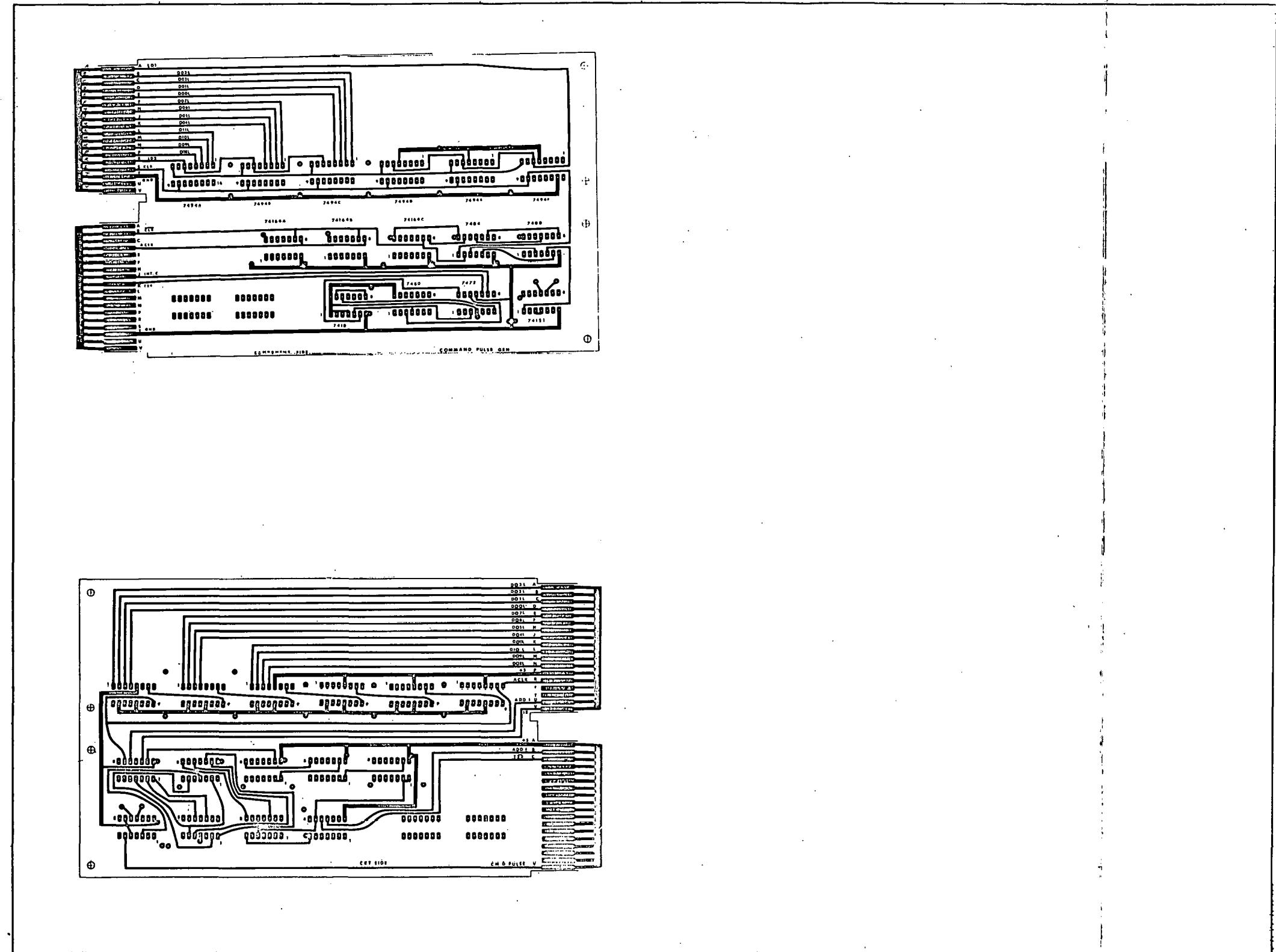
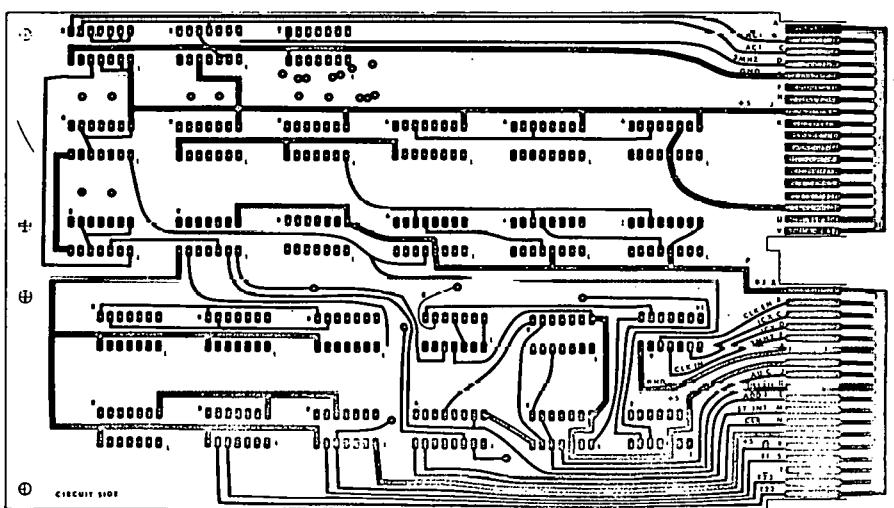
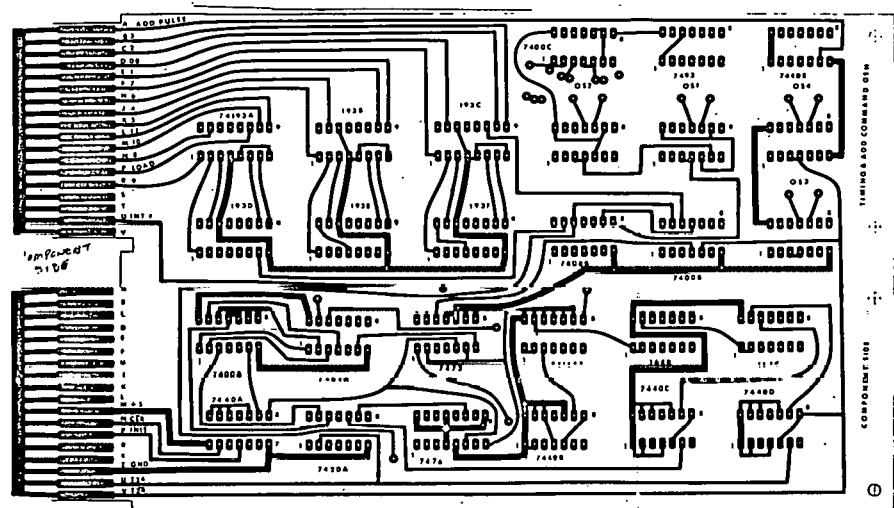


Figure C-15. MONARCH MACHINE TOOL CONTROL II. (Command Pulse Generator P/C Layout)



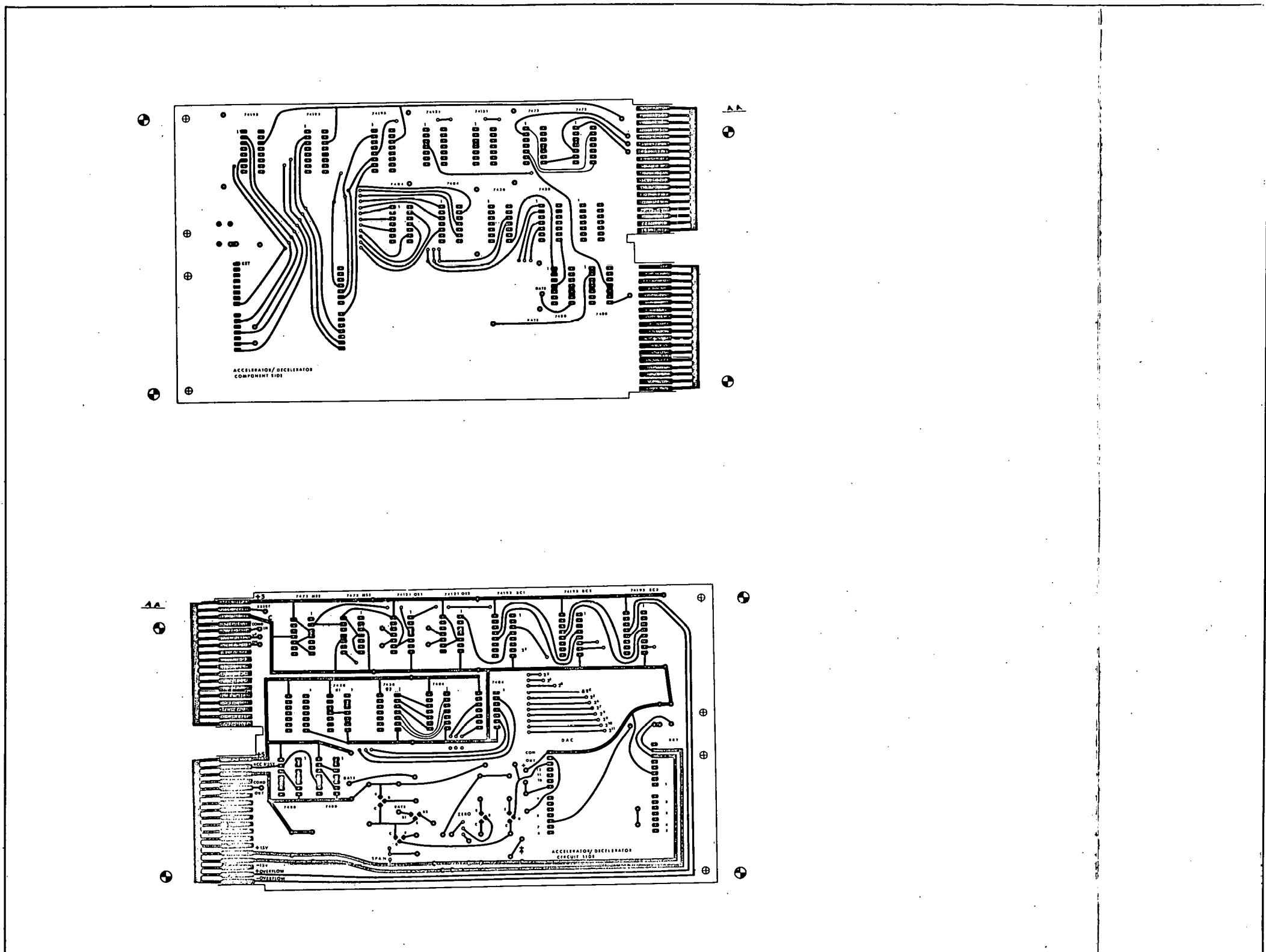
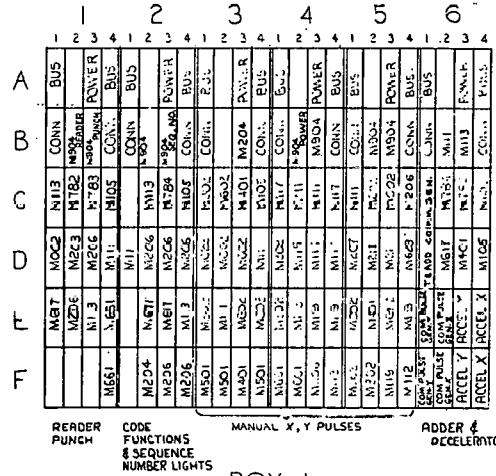
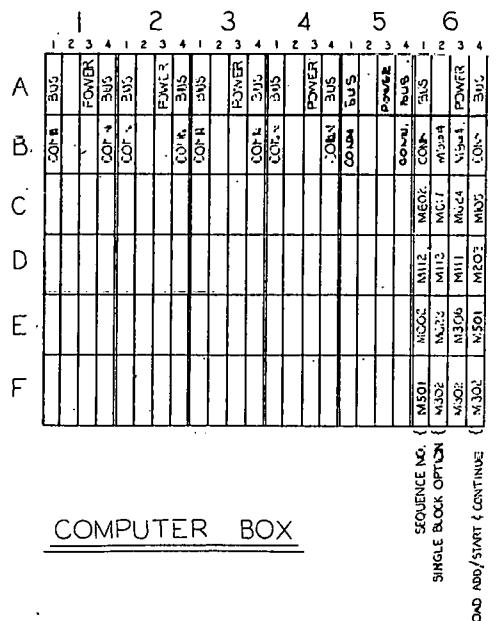


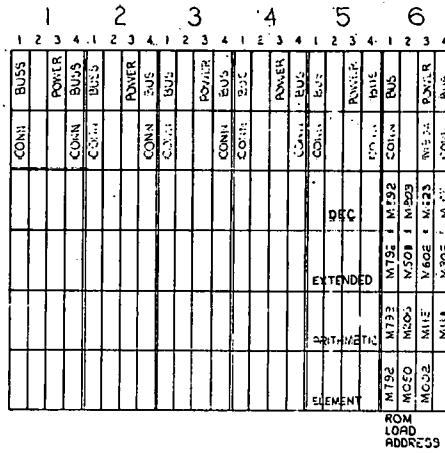
Figure C-17. MONARCH MACHINE TOOL CONTROL II: (Auto Accel/Decel P/C Layout)



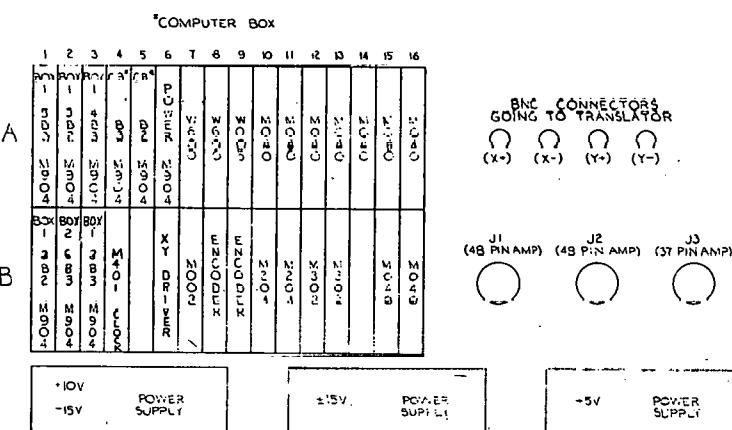
## BOX



## COMPUTER BOX



BOX 2



## INTERFACE RACK

**Figure C-18. MONARCH MACHINE TOOL CONTROL II. (Module Utilization)**

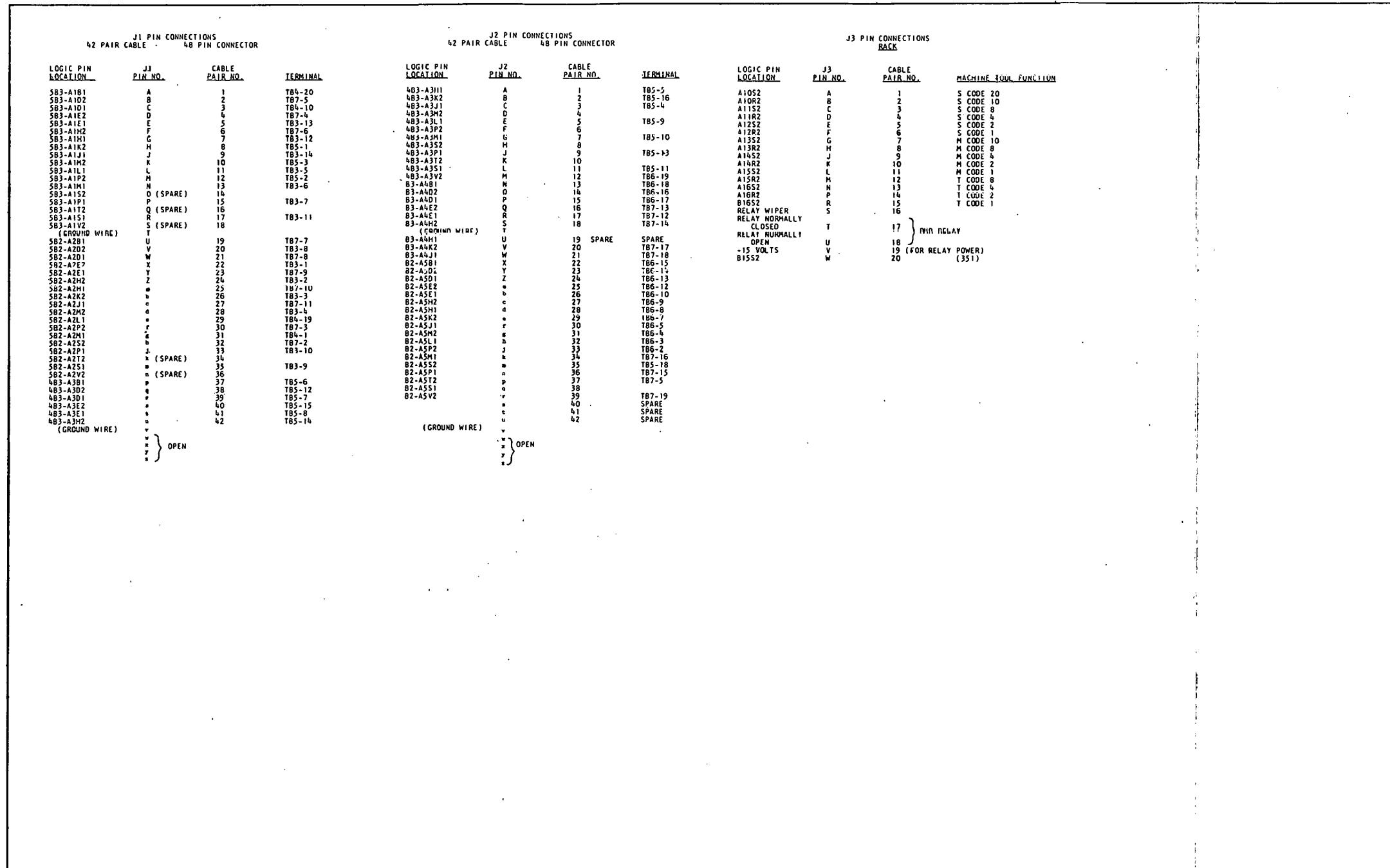


Figure C-19. MONARCH MACHINE TOOL CONTROL II. (Cable Diagram and Module Utilization)

## APPENDIX D

**OPERATE PROGRAM DESCRIPTION AND FLOW CHARTS FOR LATHE MACHINE  
TOOL CONTROL****Introduction**

This software package is for the operate control functions of the machine tool. The data contained in the block, as shown in Figure 1, are decoded, and specific action is initiated. The various function code flags are interrogated first; and, if present, the function code is fed to the machine-tool control circuitry. The sign flags for the X and Y axes are checked, and the proper sign conditions are set. The 24-bit distance values for X and Y are then loaded into Registers "A" and "B", shown in Figure 9. The CLKSTR value is then loaded into Register "C". After all data registers have been loaded, the system clock is turned on. Following this action, the X and Y pulse trains are generated. The control software goes into a WAIT instruction and remains there until all movement pulses have been generated. An interrupt signal indicates the completion of the data block. Several M code functions, if present, are executed at the end of the block. This sequence of events is repeated until an M code stop function is detected.

Software provisions are made to find a specific sequence number and start the machining process from that point. The operator loads the desired sequence number in a three-digit thumbwheel switch and depresses the sequence number "start" push button. The sequence number is found by software, and the corresponding block address is loaded.

An accelerator/decelerator software "done check" is made prior to the completion of each block if the sign of the X or Y distance value changes from the previous block to the current block. As a result of the pulse storage characteristics of the accelerator/decelerator circuitry, it is possible to lose pulses from the last block when a sign change occurs in the slide direction for the next block. In order to avoid this difficulty, a delay is executed until the accelerator/decelerator "done check" signal is generated, indicating that all pulses have been outputted for the last data block.

**Operation**

Several features of the control panel will be explained as follows:

1. The "auto-manual" switch determines the mode of control for the system. All manual features (such as incremental feed, X-Y single pulse, and manual feed) are only enabled in the manual mode. The "cycle-start" push button is enabled by the combination of the auto mode and when the spindle on-off lever, located on the lathe, is turned on.
2. To use the "sequence number search" feature, the "auto-manual" switch must be in the manual mode. Set the desired sequence number in the thumbwheel switch and turn the "enable-disable" switch to "enable". Depress the "sequence number start" push button. The proper sequence number will be found, and the correct block address loaded after which the computer will halt. Return the "auto-manual" switch to "auto". The operator

should then depress the "continue" push button when ready for the machining operation to begin.

3. To load the control program and a specific part description from the cartridge tape, the "auto-manual" switch must be in the manual mode position. With the cartridge tape loaded and the desired track selected, the tape can be read by depressing the "load program" push button. After the tape has been loaded, the operator may start the machining process by using the "cycle start" push button or the "sequence number search" routine.

Table D-1 lists the memory allocations of the operate program for lathe control software.

Table D-1  
MEMORY ALLOCATION OF THE OPERATE  
PROGRAM FOR LATHE MACHINE  
TOOL CONTROL SOFTWARE

Location	Function
000200-000774	Initialization, Function Code Detection, Sign Selection, and Data Loading
000776-001466	Data Block Completion Routine
001470-001760	Sign Repair
002000-002122	Sequence Number Subroutine
002336-002520	Address Fix

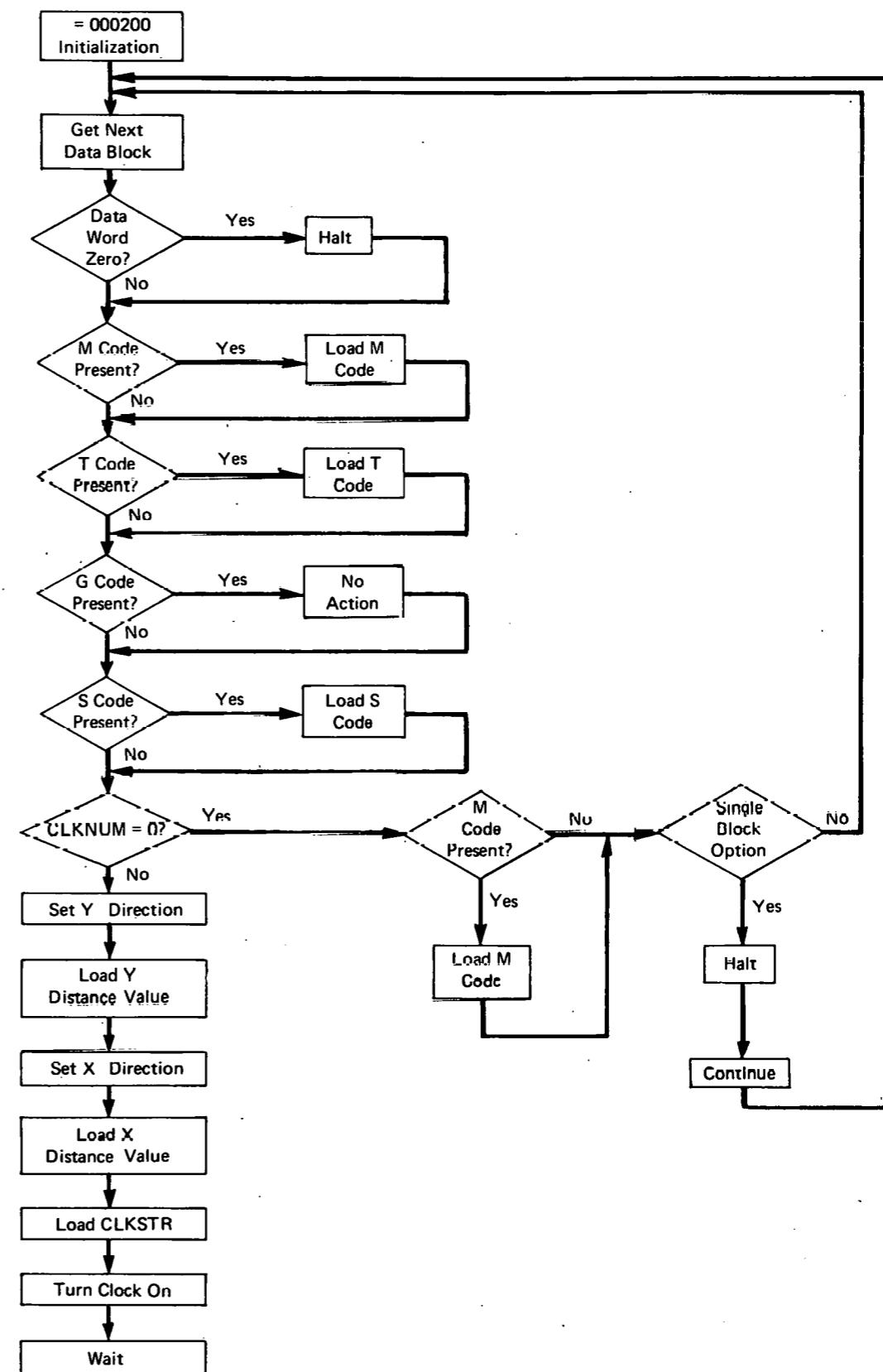
### Subsection Descriptions

**Title** - Initialization, Function Code Detection, Sign Selection, and Data Loading

**Location** - 000200-000774

**Purpose** - The function codes are detected, the correct X and Y axis signs are set, and the distance values are loaded.

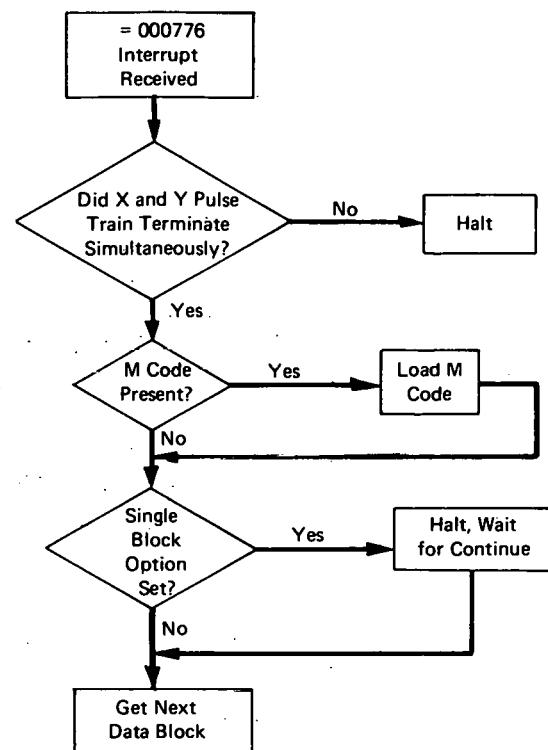
**Description** - The various function code flags are interrogated first; and, if present, the function code is fed to the machine tool control circuitry. Sign flags for the X and Y axes are checked and the proper sign conditions set. The 24-bit distance values, X and Y, are then loaded along with the CLKSTR value.



**Title - Data Block Completion Routine****Location - 000776-001466**

**Purpose** - Logic conditions are interrogated after the "interrupt done" signal is received.

**Description** - After the "interrupt done" signal is received, a check is made to see if both axes finished at the same time. If the pulse train generated for each axis does not terminate simultaneously, an error flag is set, and the computer halts. An M code function check is made; and, if present, the code function is loaded. A single block option check is made; and, if enabled, a halt occurs after the completion of each block. If the single block option is not enabled, the next block is executed.

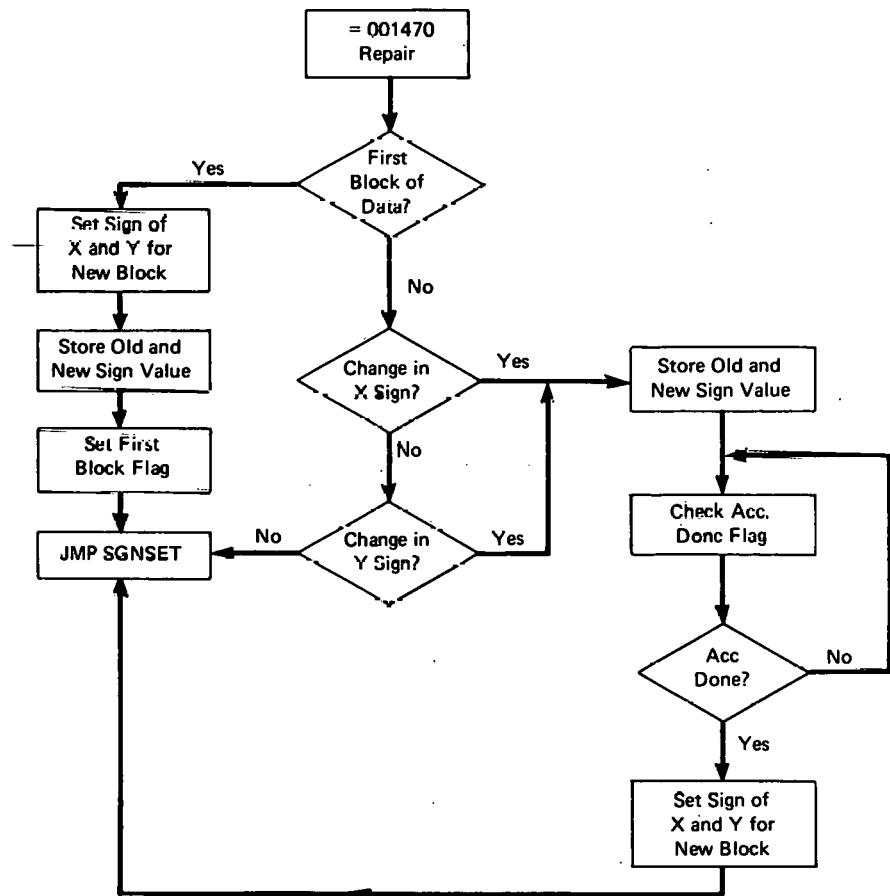


**Title** - Sign Repair

**Location** - 001470-001760

**Purpose** - A sign change from the previous block to the current block is detected.

**Description** - If a sign change occurs on either axis from the previous block to the current block, a delay is executed until all pulses have been outputted from the accelerator/decelerator. After the accelerator/decelerator "done" flag is recognized, the sign flip flops are set for the new block.

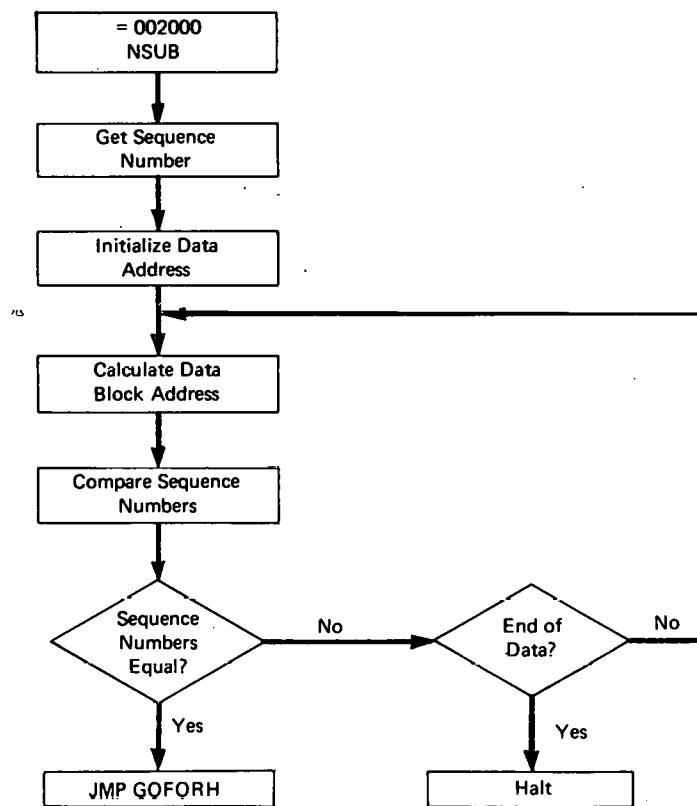


**Title** - Sequence Number Subroutine

**Location** - 002000-002122

**Purpose** - The sequence number location is found in the part description data.

**Description** - The sequence number loaded in the thumbwheel switches is strobed into the computer. The corresponding sequence number is found in the part description data, and the starting address of the next block is determined.

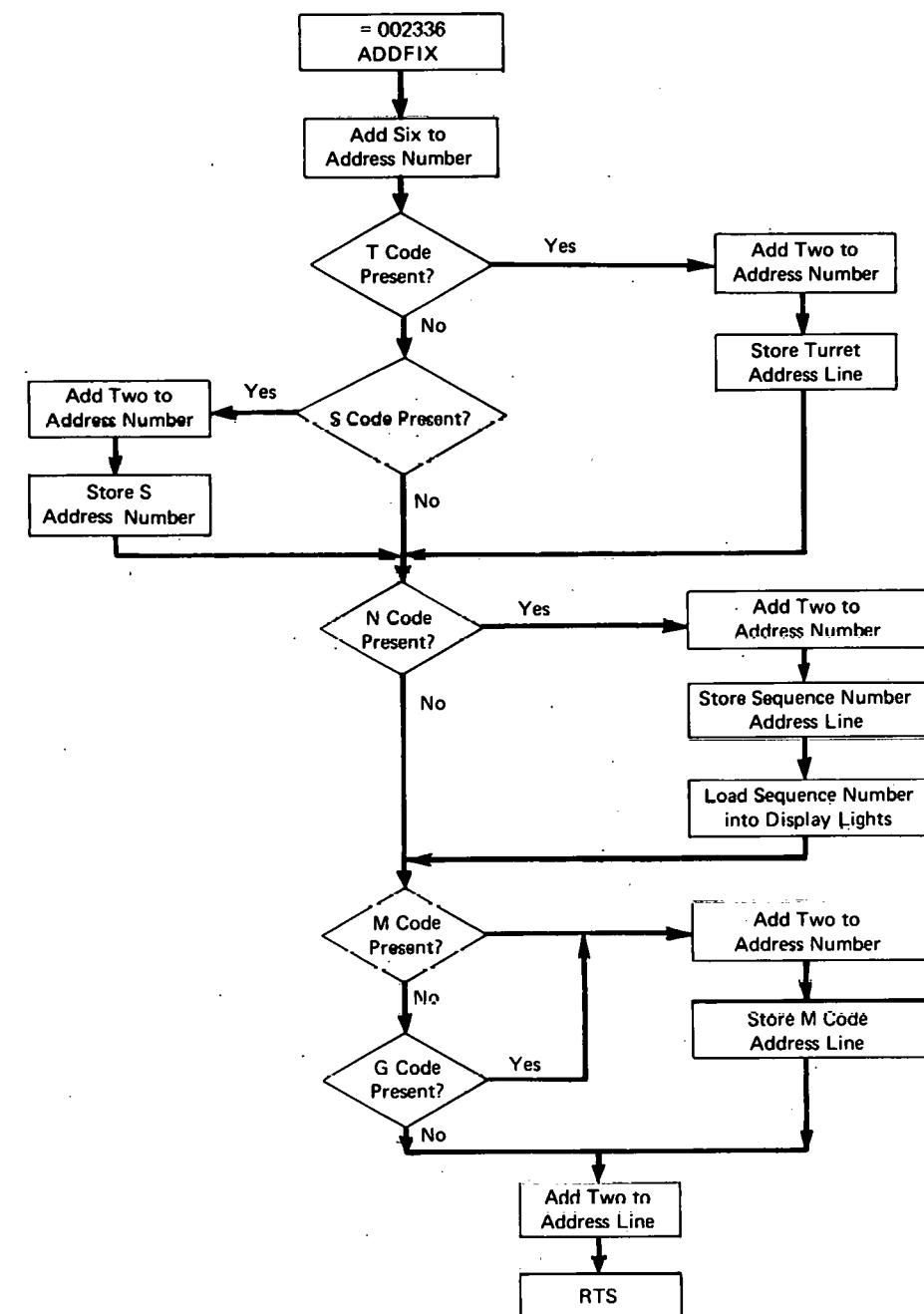


**Title - Address Fix**

**Location - 002336-002520**

**Purpose -** The address of the next data block is determined

**Description -** Due to the fact that the data block is of variable length, an address calculation routine is necessary. The minimum block length is four words, and the maximum is seven words. The function code flags are interrogated to determine the block length.



**DISTRIBUTION****Atomic Energy Commission - ALO**

Vespe, V. C.

**Atomic Energy Commission - ORO**Hickman, H. D.  
Zachry, D. S., Jr**Lawrence Livermore Laboratory**Eckard, R.  
Jenson, J. A.  
Spies, R. J.  
Tech Inf Div**Los Alamos Scientific Laboratory**

Stack, F. E.

**Oak Ridge Gaseous Diffusion Plant**McLaren, R. A.  
Wilcox, W. J., Jr  
Winkel, R. A.**Oak Ridge National Laboratory**Adams, R. K.  
Barkowski, C. J.**Oak Ridge Y-12 Plant**Alvey, H. E.  
Bernander, N. K.  
Bowers, G. L. (5)  
Burditt, R. B.  
Burkhart, L. E.  
Butturini, W. G.  
Conley, C. E.  
Denny, A. (2)  
Foulk, D. L.  
Gritzner, V. B.  
Groppe, W. H.  
Hemphill, L. F.  
Hensley, C. E.Jones, F. W.  
Kahl, K. G.  
Keith, Alvin  
Kite, H. T.  
Lay, C. M. (5)  
Miskell, R. V.  
Mitchel, G. W.  
Noey, J. L.  
Oliphant, G. W.  
Perry, A. E.  
Smelcher, O.  
Smith, D. N.  
Smith, H. F., Jr  
Smith, R. D.  
Stephens, A. E. (5)  
Stephens, W. E.  
Stoner, H. H.  
Thompson, C. H.  
Tilson, F. V.  
Trotter, T. C.  
Tuel, E. F.  
Tunnell, H. A.  
Weathersby, W. E.  
Webber, T. R.  
Wright, C. C.  
Yaggi, W. J.Y-12 Central Files (5)  
Y-12 Central Files (master copy)  
Y-12 Central Files (route)  
Y-12 Central Files (Y-12RC)**Paduoa Gaseous Diffusion Plant**

Millican, R.

**Sandia - Albuquerque**Gardner, W. A.  
Mail Serv Sec**Sandia - Livermore**Davies, L. E.  
Library

In addition, this report is distributed in accordance with the category UC-38, **Engineering and Equipment**, as given in the *USAEC Standard Distribution Lists for Unclassified Scientific and Technical Reports*, TID-4500.