X-324-67-461

NASA TM X-55991

# A HIGH-SPEED SELECTIVE RECORDING SYSTEM (HSSRS)

SEPTEMBER 1967

GPO PRICE $ _____

CFSTI PRICE(S) $ _____

Hard copy (HC) 3.00

Microfiche (MF) .65

ff 653 July 65


GODDARD SPACE FLIGHT CENTER

GREENBELT, MARYLAND

X-324-67-461

# A HIGH-SPEED SELECTIVE RECORDING SYSTEM (HSSRS)

September 1967

Prepared by:

Norman S. Babbitt
Electronic Test Branch

Reviewed by:

Phillip Yaffee
Head, Electronic Test Branch

Approved by:

John H. Boeckel
Associate Chief, Test and Evaluation Division

GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland

## PROJECT STATUS

This document is the final report on the high-speed selective recording system initially prepared as a thesis and submitted in February 1967 to the George Washington University.

Authorization

Test and Evaluation Division Charge No.

324-125-23-02-13

# A HIGH-SPEED SELECTIVE RECORDING SYSTEM (HSSRS)

Norman S. Babbitt
Test and Evaluation Division

## SUMMARY

Many environmental test laboratories use a centralized automatic data-handling system for collecting and processing the large volume of specimen and facility data generated during test operations. Use of such a system involves an inherent difficulty in efficiently selecting specific channels of data for processing without using an on-line computer. This restriction was encountered in the data-collection system (DCS) used at the Goddard Space Flight Center for evaluating the performance of spacecraft in simulated environments.

The high-speed selective recording system was designed and installed to overcome this difficulty. It provides the capability of acquiring selected data in a block-gap format from any number of data points available in the DCS. No computer is required. The acquired data are synchronous with the DCS and programmable, and are permanently recorded on magnetic tape for subsequent processing.

CONTENTS

ILLUSTRATIONS

# A HIGH-SPEED SELECTIVE RECORDING SYSTEM (HSSRS)

## INTRODUCTION

The high-speed data collection system (DCS) is capable of recording 1500 different data points at a rate of 7500 per second.

The DCS is arranged in a 20- by 75-matrix format (Figure 1). The rows of the matrix are referred to as frames; the columns are designated as time slots. The first data point is time slot 1, frame 1; the second data point is time slot 2, frame 1. The format proceeds across the 20 time slots (columns) until a frame (row) is completed. The next frame is begun, and when all 20 time slots have been completed, the third frame begins. After time slot 20, frame 75, is completed, an end of record (E/R) pulse is received. The format returns to time slot 1, frame 1, and repeats the procedure continuously. Each time slot is present for 133 $\mu$sec, each frame takes 2.66 milliseconds, and one entire record lasts 200 milliseconds. Figure 2 is a timing diagram of the DCS.

The E/R signal is buffered and sent to the control logic to synchronize the high-speed selective recording system (HSSRS) with the DCS.

Data can be multiplexed into any one of the 1500 data points, making it possible to record signals from 1500 separate data channels. Two methods have been used for acquiring data from a desired data channel after the DCS had assembled them into a format. In the first method, the entire format was fed into an on-line computer which was programmed to strip out the desired information and to transmit these data to a tape recorder. In the second method, the entire format was recorded on magnetic tape and a program written to strip the desired data when the tape was run off-line through a computer.

Neither of these methods is entirely acceptable. In the first one, an on-line computer must be scheduled for the entire test. This is an expensive operation which requires multiple computers if several channels of data are needed simultaneously.

In order to record the entire format, the second method requires about one roll of magnetic tape every 5 minutes for the duration of the test. After the information is recorded, it must still be stripped from the undesired data. This results in a difficult tape-changing operation if a test of more than 5 minutes is to be monitored. In addition, although the basic DCS is designed to service many test areas, its use in the high-speed collection mode (7500 samples per

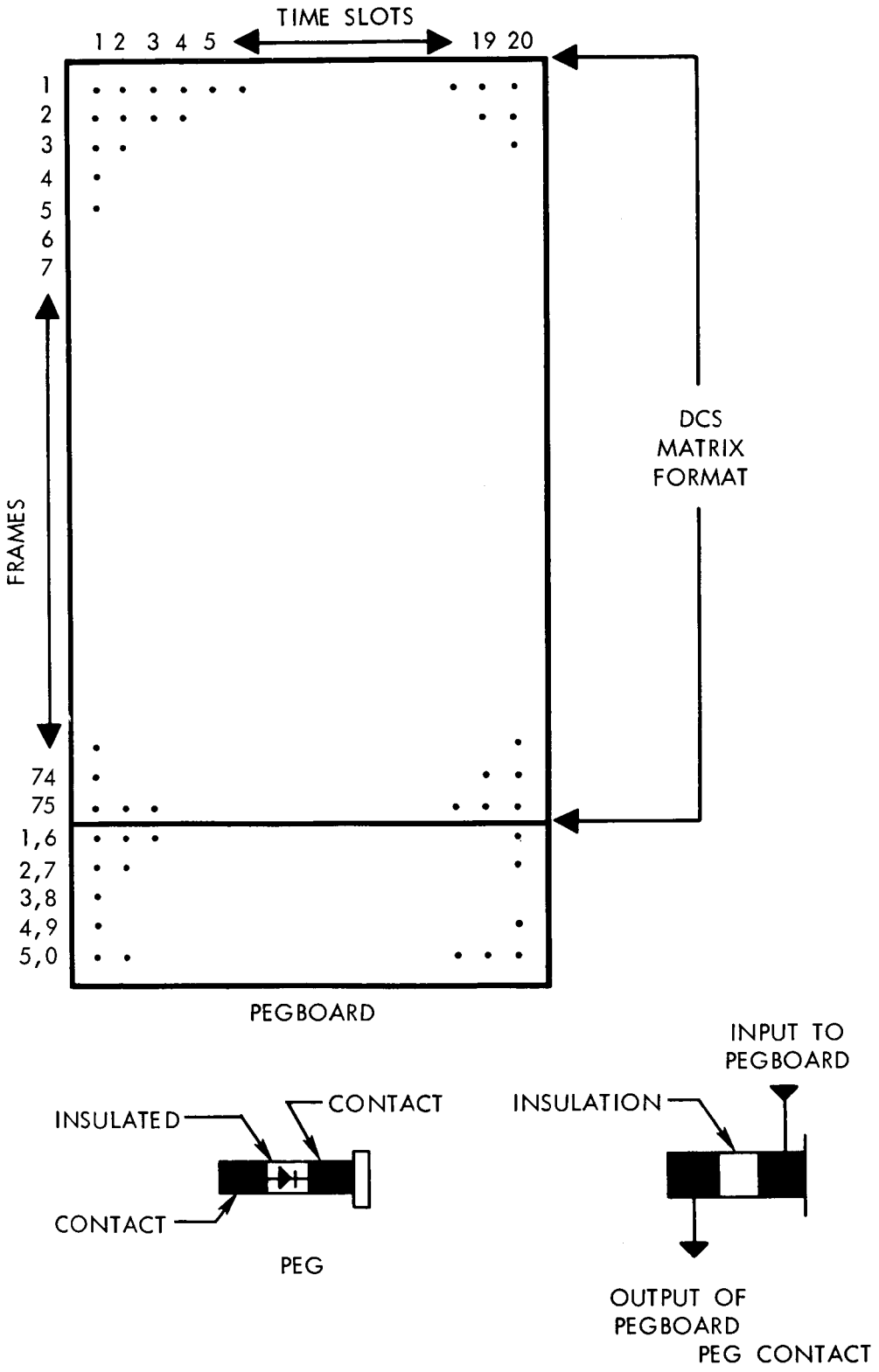


Figure 1. Pegboard and Peg Layout

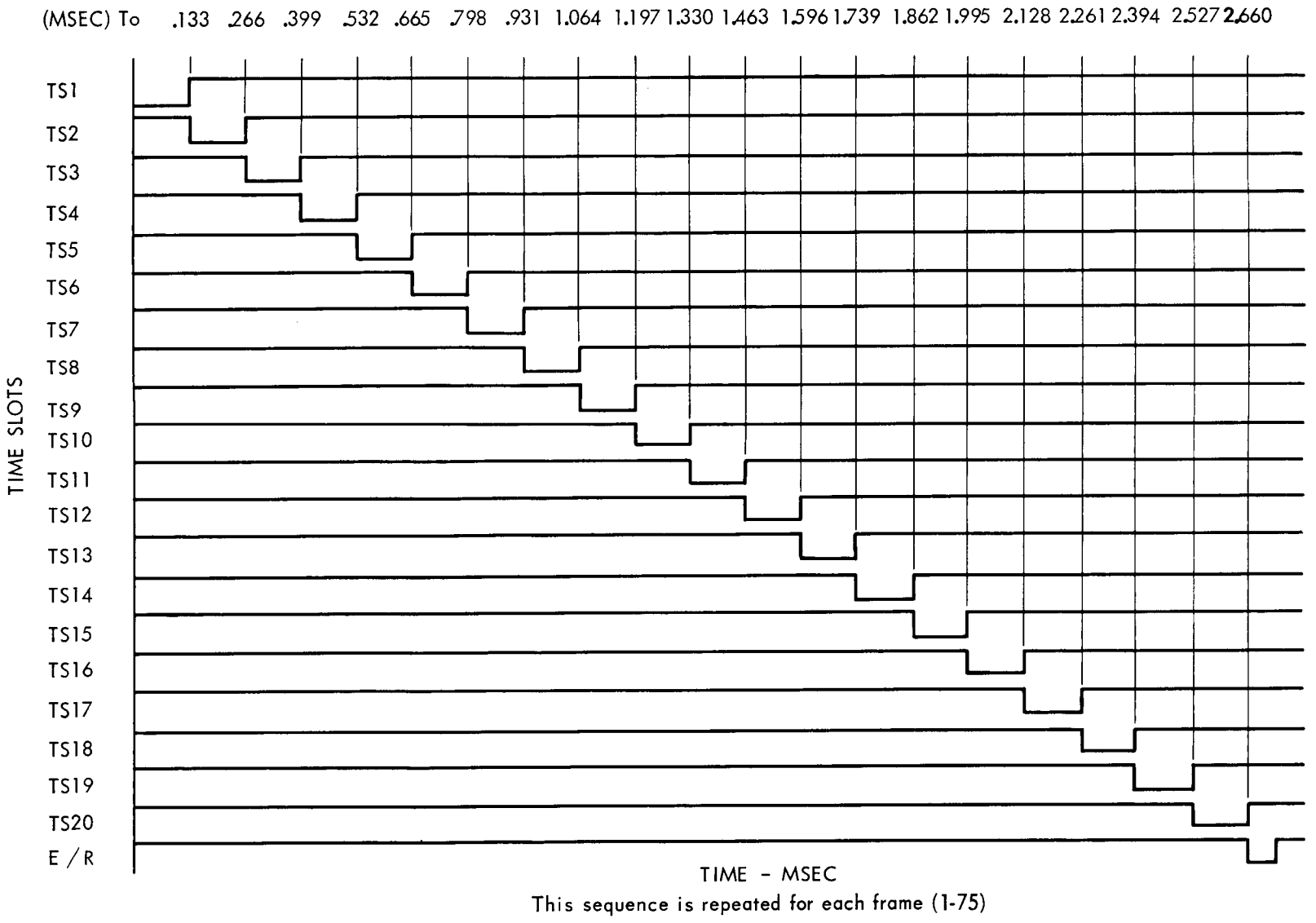


Figure 2. Data-Collection System, Timing Diagram

second) is usually limited to a single test at any given time. Thus, only a small percentage of the 1500 data points in the fixed format are occupied by meaningful data. The remainder are occupied by meaningless noise.

To alleviate some of these problems, an adaptable selective system was designed to select any combination of data points in the DCS format for recording in a dense, tape-saving manner.

## HSSRS GENERAL SYSTEM

Data from the DCS are formatted in the assembler (Figure 3). From the assembler data are fed in 12-bit words to the prememory buffer and the control-logic buffer. The HSSRS receives 21 timing signals from the DCS. These signals are time slots (TS) 1 through 20 and end of record (E/R). In the control-logic section, TS1 to TS20 are buffered and sent to a matrix where they are combined with the internally generated frames to form the 1500 points in the format. An additional 100 signals are generated in this matrix. These signals, shown below frame 75 on the pegboard in Figure 1, are identified as (1,6), (2,7), (3,8), (4,9), and (5,0). In any chosen time slot, these signals are formed with all frames that have either of the numbers in parentheses as their least significant digit. For example, (2,7) in TS5 would have a signal present whenever TS5 occurs in frames 2, 7, 12, 17, 22, ...., 67, and 72. This allows signals which require more frequent sampling to be inserted 15 times per record with one connection. Frames are generated by advancing a counter once each time TS20 occurs. After the time slots and frames are combined in the matrix, the 1600 points are connected to the pegboard so that a timing signal is present at the pegboard which is simultaneous with the high-speed data format.

The selective recording system provides a means for recording individually selected data points. Pegs are inserted into a pegboard in the holes representing the desired data points. These points are recorded on tape in a block-gap format. No programming is necessary to add or change channels; changes are made by simply rearranging pegs and setting two thumbwheel switches. Because this system dumps blocks of information onto tape only after an entire block has been accumulated in memory, the tape runs for short periods only and a relatively long test may require only one reel of tape. Any combination of data points in the high-speed format can be recorded, making it possible to record information for one particular spacecraft test or test area, no matter where these data are located in the high-speed format.

To select a particular data point for recording, a peg is placed in the corresponding peg contact (Figure 1), connecting that signal with the output line.

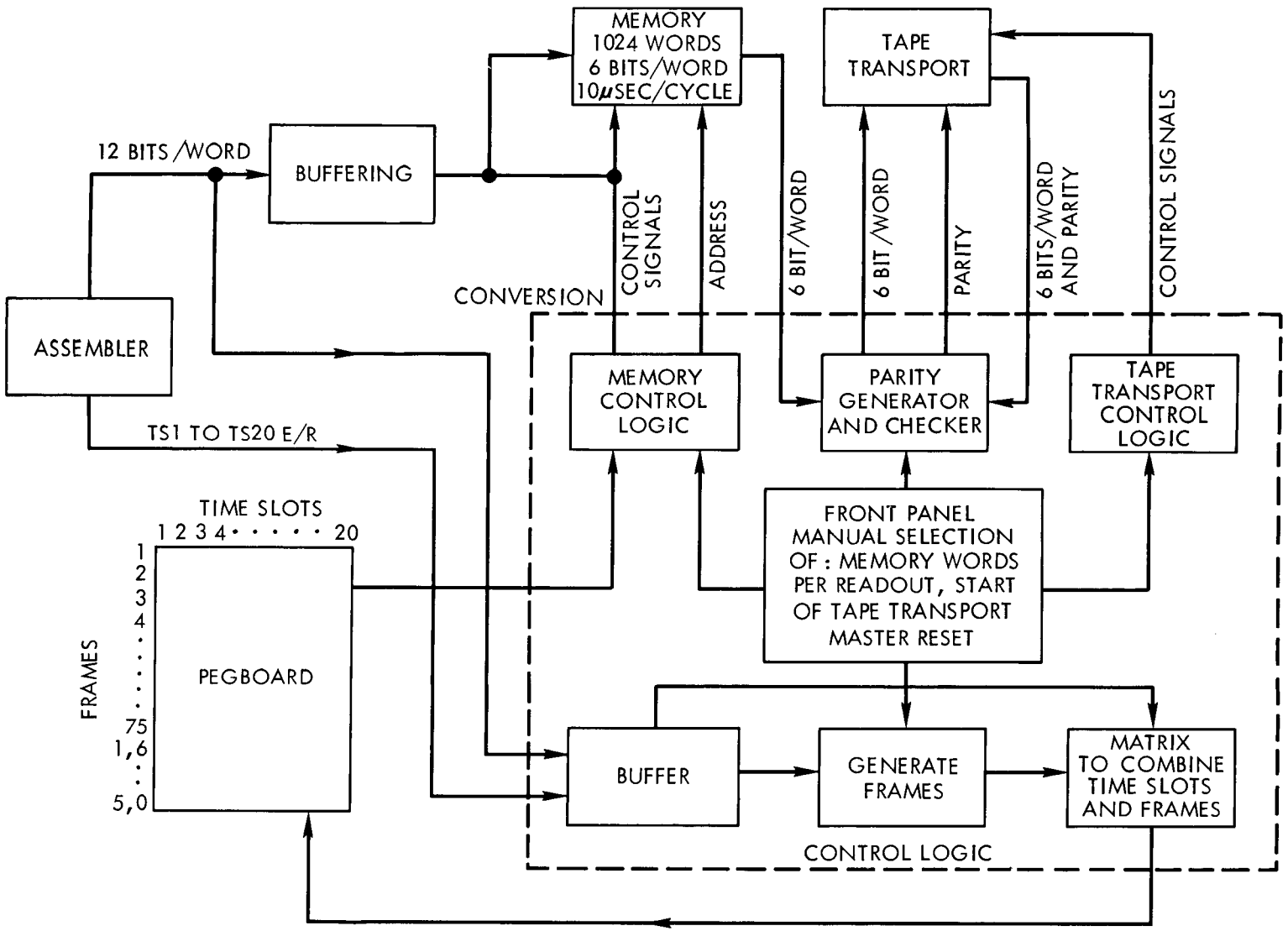


Figure 3. HSSRS, Block Diagram

The output line from the pegboard is an input to the memory control logic (Figure 3). Every time a timing pulse is received from the pegboard, the data word present at the assembler is strobed into the memory and the memory address is advanced. Because the tape format calls for a 6-bit data word with odd parity, the 12-bit word from the assembler must be separated into two 6-bit words. The six most significant bits are stored first, then the six least significant bits. Therefore, each data word requires two memory addresses.

Two 4-digit binary-coded-decimal thumbwheel switches are located on the front control panel.

The first switch provides manual selection of the memory bits per readout (the address in memory at which one desires to begin dumping the memory onto tape). To determine this setting, the number of points being monitored by the pegboard is multiplied by 2 (two memory bits per data word). This number represents the addresses in memory required to store the data received from one record of the high-speed format. To obtain the number of records which can be stored in memory, the foregoing number is divided into 1024 (capacity of memory). The resulting whole number is the number of complete records which the memory can store.

This number of records constitutes a block of data. In the last record of a block, the address of the last data point is set in the "memory addresses per cycle" switch. When the memory reaches this address, a signal is generated to initiate the dumping of memory onto the tape in the same order as it was received. The loading and unloading of memory at the beginning of a block always starts at address 0. Dumping of the stored data onto tape continues sequentially until a pulse from the pegboard indicates that new data are to be stored.

The memory stops unloading, goes back to address 0, and stores this word of data. It will then automatically go to the address in the unloading sequence where it had been previous to the signal from the pegboard, and continue the dumping of the block. If more signals are received from the pegboard, the memory will load these data in addresses 2 and 3 (first piece of data was stored in addresses 0 and 1), and again revert to the last address in the unloading sequence. In this manner, the same point in successive blocks of data is stored in the same address in memory, and there will be no skipping over of data. This process will continue until the entire block is recorded on tape. The memory will again fill up to the designated point, at which time the foregoing procedure will be repeated.

The second switch provides for the start of the tape transport. The tape transport used in this system requires 3 milliseconds to attain its maximum

speed. Thus, the transport should be started 3 milliseconds before the memory address reaches its final address to signal the dumping of the block of data it contains. This point precedes the final address by 23 or more time slots (133 $\mu$sec per time slot). The number set into the "memory addresses start of tape" switch is the address of the data point which is 23 or more time slots before the final data point in the block. Data may be fed into memory at a maximum rate of 15 kc from the data-collection system. Data may be dumped out of the memory at the rate of 100 kc; however, the constraining condition for dumping the memory onto tape will be the recording speed of the tape transport. This holds, of course, only if the tape transport records at less than 100 kc, which was true of the original design. The Control Data Corporation (CDC) tape transport was chosen because of its availability. This transport has a maximum recording rate of 41.7 kc. To remain synchronized with the DCS, the optimum obtainable rate less than or equal to 41.7 kc would be 37.5 kc, with blanks occurring each time a new data word is received during a memory dump.

Figure 4 shows the HSSRS, except for the tape transport, mounted in temporary racks during the system checkout. Figure 5 is a closeup view of the matrix pegboard which simulates the high-speed format, and the system-control panel.

## MAJOR EQUIPMENT

Programming of the system is accomplished by means of a 22-1/4 by 12 inch pegboard (Figure 1) into which diode pegs are inserted. A pair of normally open contacts is mounted inside each pegboard hole. Each peg contains a diode whose anode and cathode are terminated in contacts. When the peg is inserted into the hole, the diode contacts mate with the fixed pegboard contacts. The cathode of the diode is thus connected to the timing signal which represents that peg position in the format. The anode of the diode connects to a common point in an OR circuit (for negative logic). The anodes of all inserted pegs connect to this point. Thus, if any timing signal in the high-speed format is present at an inserted peg position, the signal will go through the diode to the control logic and that word of data will be stored. Data points corresponding to open peg positions will not be stored.

A 1024-word (6 bits per word) Rese Engineering Company memory stores the data before it is written on the magnetic tape. This memory is a random access, magnetic, digital data storage device. It has an access time of 4 $\mu$sec, and a full-cycle time of 10 $\mu$sec, and it operates in a random-parallel mode of access. The memory is addressed with parallel pure binary signals, with two lines per bit. The input data are stored in a parallel mode and can be presented

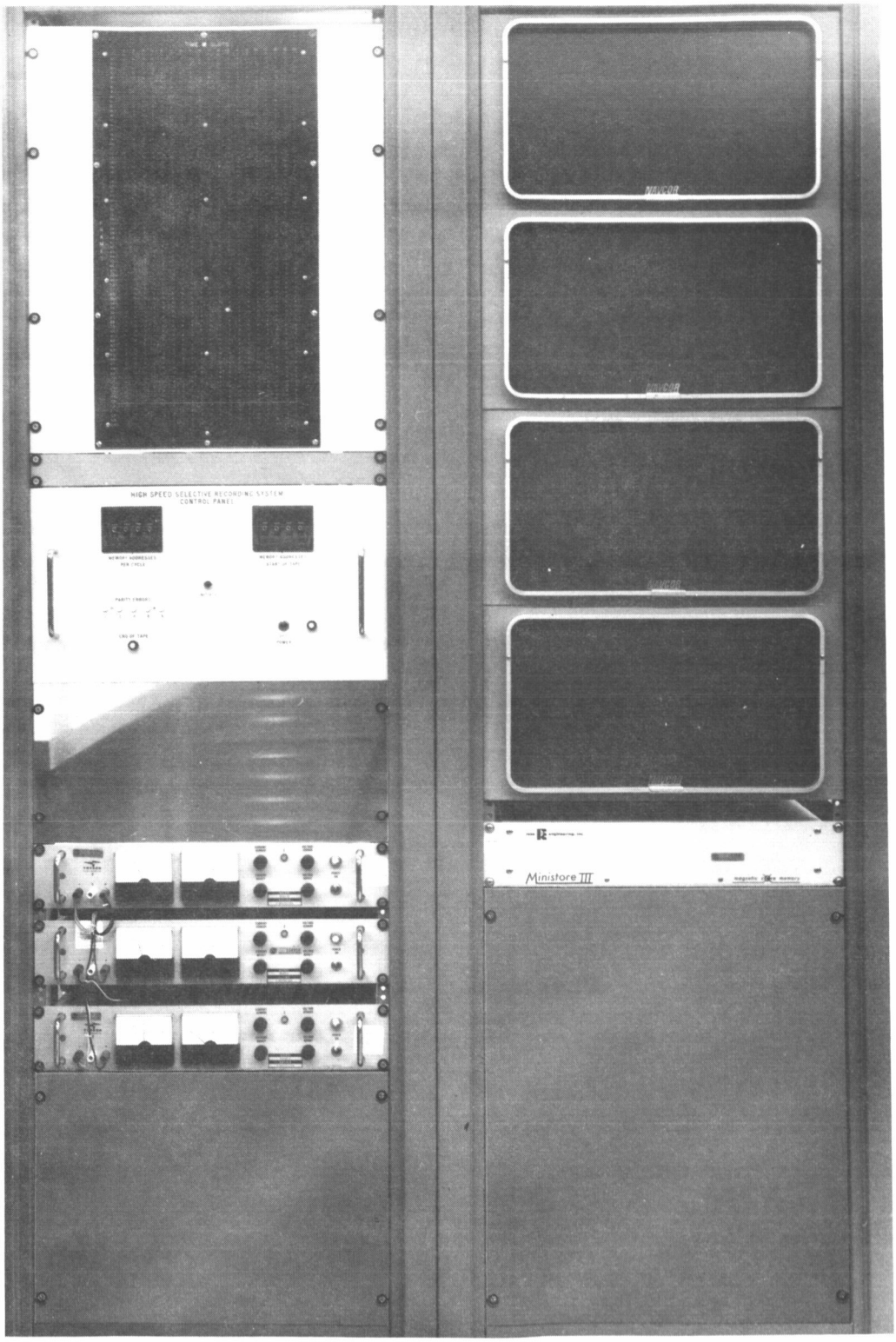


Figure 4. Test Setup for System Without Tape Transport

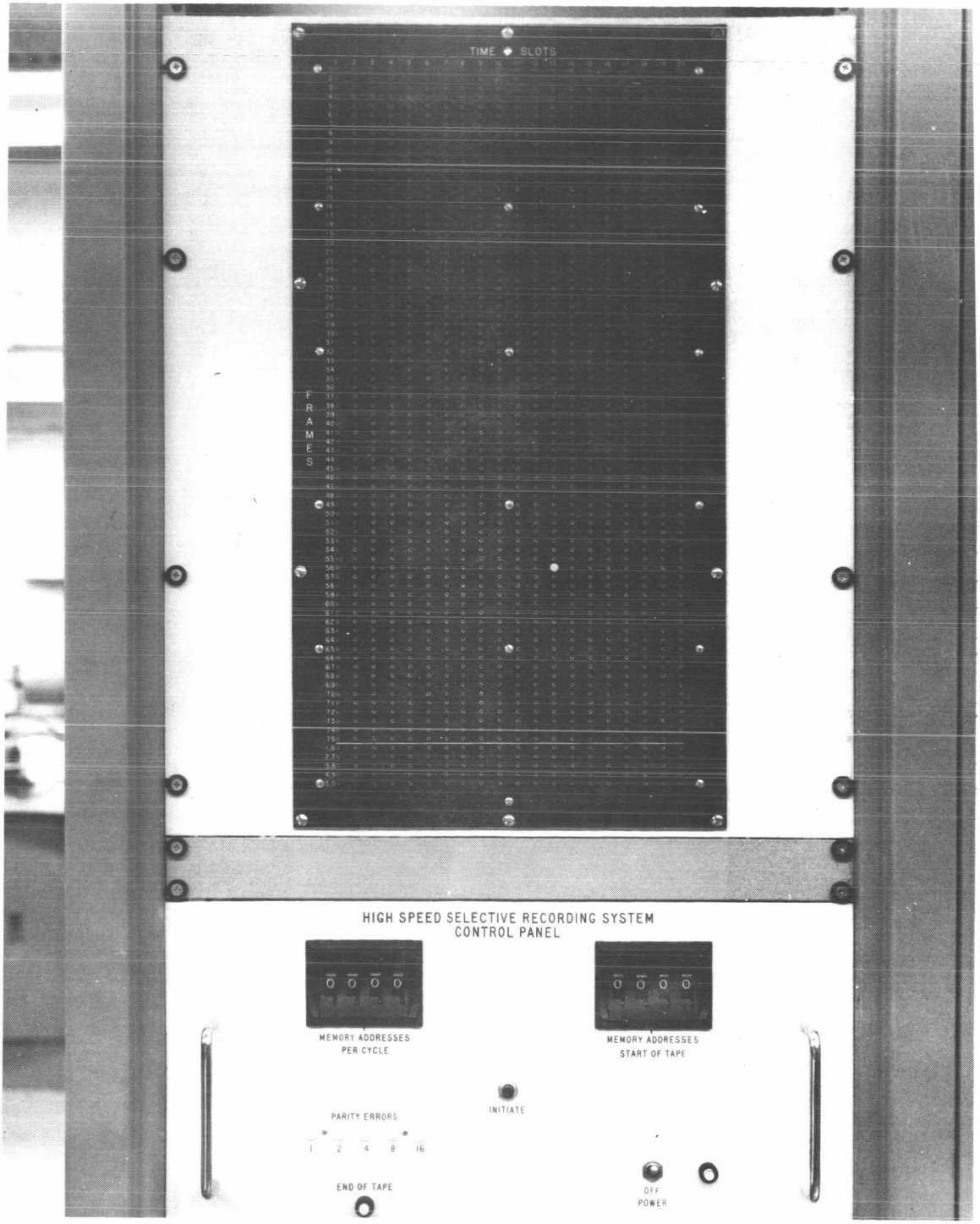


Figure 5. Pegboard and Control Panel

to the memory as either a level, which is clocked in, or a positive-going pulse. The output operates in a parallel mode with two lines per bit and presents the data as a dc level. Signals required for loading and unloading operations are:

- Load operation signals:
  1. Address information
  2. Load/unload level in negative condition
  3. Start sync pulse
  4. Data to be written
- Unload operation signals:
  1. Address information
  2. Load/unload level at 0-volt condition
  3. Start sync pulse

Figure 6 shows the timing signals for load and unload operations.

Any number of different magnetic tape transports can be used in the HSSRS. The system was originally designed to interface with a CDC 603 tape transport which uses a 7-track nonreturn-to-zero (change-on-ones) recording scheme. Six of the tracks represent the digital word; the seventh track is used for the parity bit. A longitudinal parity bit is required at the end of each record (even parity) by the data-processing equipment. Recording can be done at a density of either 556 frames per inch or 200 frames per inch, at a tape velocity of 75 inches per second. Characters can be recorded at either 41,700 lines per second or 15,000 lines per second. This unit requires 3 milliseconds to start and 2 milliseconds to stop. Ground is a logical 1 and -16 volts is a logical 0 on the input-output cards. The input signals used are:

1. Data (NRZ1)
2. Parity (ODD)
3. Write sprocket
4. Forward level
5. Write select level

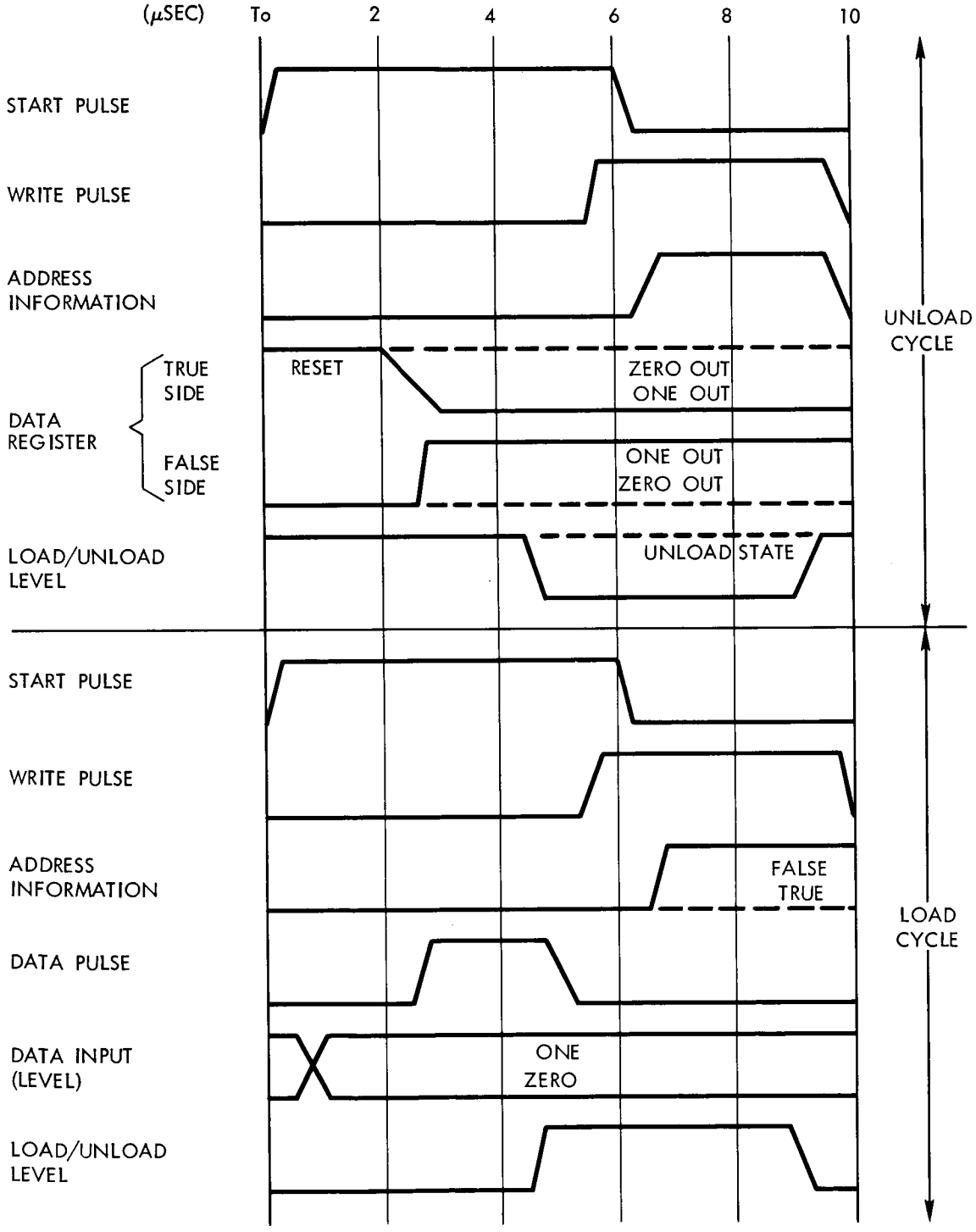


Figure 6. Load and Unload Cycles, Timing Diagrams

## DIGITAL CIRCUITRY

Printed-circuit card logic modules comprise the entire logic system. The main logical building blocks are Navigation Computer Corporation (NAVCOR) 400 series modules. They are 4- by 6-inch transistorized printed-circuit cards, rated at a maximum operating frequency of 300 kc. For all special circuits or applications, such as a diode matrix or inverters, comparable modules designed at the Goddard Space Flight Center are used. Five 60-card drawers of logic are used in the HSSRS.

The logic levels are: 1 equals -6.6 to -7.3 volts, and 0 equals 0 to 0.2 volt. The collector supply voltage is -12 volts with a clamping diode connected to a -6.8-volt Zener voltage.

The most-used printed-circuit board in the system is a special matrix (22 by 10) which combines the time slots and frames to form the 1600-signal lines which serve as inputs to the pegboard. The configuration of this printed-circuit board is designed so that it can be used to form all pegboard signals. Each of these boards has four time slots (TS X to TS X+3) and three frames (FY, FY+5, FY+10) as inputs. The twelve combinations of the four time slots and three frames result in twelve outputs which also go to OR diodes. These diodes will be connected with similar diodes on the other printed-circuit boards to form the signal lines designated as (1,6), (2,7), (3,8), (4,9), and (5,0) on the pegboard. Five of these boards are required to form (1,6) or any of the same type of multiple signals. Five times five or 25 boards are required to form all five of the multiple signals for the four time slots used. For all 20 time slots, 5 times 25 or 125 printed-circuit boards of the same configuration are required.

Other main building blocks in the system are:

1. NAVCOR Model 410—A five-bit parallel binary counter with a settling time of 1 microsecond. These units may be cascaded to form any length of binary counter desired. It takes n microseconds for n cascaded modules to settle.

2. NAVCOR Model 412—A parallel binary coded decimal (BCD) counter with the same settling time as the 410 module.

3. NAVCOR Model 446—A parity module which contains logic to check or generate parity for six bits of binary data.

4. NAVCOR Model 450 (X)—A variable frequency astable multivibrator. This unit normally has frequencies up to 100 kc; however, crystal-controlled clocks can be ordered with frequencies up to 300 kc.

5. NAVCOR Model 451—Three monostable multivibrators with variable ON period duration. These units have a duty cycle of 90 percent with a minimum recovery time of 3 microseconds.

6. Specially made modules—Fifteen inverting amplifiers per module. These units provide most of the drive and buffering for the system.

## DECIMAL-TO-BINARY CONVERSION

Two methods were considered for decimal- (BCD) to-binary conversion: Figure 7 shows the first, basically a parallel method, whereas the second method used essentially serial conversion. The basic algorithm for this method is:

1. Determine the length of the BCD number.

2. Set up gating to sample these bits.

3. Shift the number down by one bit (toward LSB).

4. If any digit is now 8 (binary 1000) or greater, subtract 3.

5. Repeat steps 3 and 4 for each bit.

This conversion requires a maximum of 4 (n-1) clock pulses, where n represents the number of BCD digits. The truth table, input, and output equations for one section of the configuration in Figure 7 are:

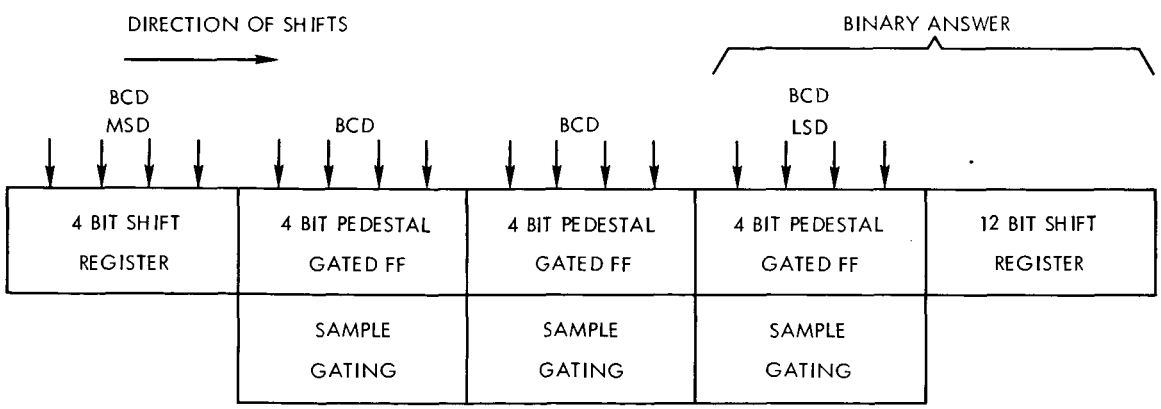


Figure 7. Decimal-to-Binary Conversion, Parallel

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 0101 | 0101 | 0101 | 0101 | 0101 | 0101 | 0101 | 0101 |
| 2 | 0011 | 0011 | 0011 | 0011 | 0011 | 0011 | 0011 | 0011 |
| 4 | 0000 | 1111 | 0000 | 1111 | 0000 | 1111 | 0000 | 1111 |
| 8 | 0000 | 0000 | 1111 | 1111 | 0000 | 0000 | 1111 | 1111 |
| $C_{in}$ | 0000 | 0000 | 0000 | 0000 | 1111 | 1111 | 1111 | 1111 |
| $1_S$ | 0011 | 0011 | 0110 | 0110 | 0011 | 0011 | 0110 | 0110 |
| $2_S$ | 0000 | 1111 | 1110 | 0001 | 0000 | 1111 | 1110 | 0001 |
| $4_S$ | 0000 | 0000 | 0001 | 1111 | 0000 | 0000 | 0001 | 1111 |
| $8_S$ | 0000 | 0000 | 0000 | 0000 | 1111 | 1111 | 1111 | 1111 |
| $C_{out}$ | 0101 | 0101 | 1010 | 1010 | 0101 | 0101 | 1010 | 1010 |

$1_S = \bar{8}2 + 8\bar{2}1 + 82\bar{1}$ $\qquad 1_R = \bar{1}_S$

$2_S = \bar{8}4 + 8\bar{4}\bar{2} + 8\bar{4}\bar{1} + 8421$ $\qquad 2_R = \bar{2}_S$

$4_S = 84 + 821$ $\qquad 4_R = \bar{4}_S$

$8_S = C$ $\qquad 8_R = \bar{8}_S$

$C_{out} = \bar{8}1 + 8\bar{1}$

Note: A logic diagram of one section is shown in Figure 8a.

The highest order BCD digit (thousands) is loaded into the first shift register (far left in Figure 7). The remaining three digits are loaded into the three storage registers in descending order from left to right. The 12-bit shift register will contain the 12 least significant bits (LSB) of the binary number at completion of the conversion. The last 4-bit storage register will contain the most significant bits (MSB) of the binary word. Again, the final result will be in descending order as read from left to right.

Figure 8a shows one set of a storage register and the corresponding inspection gating. $C_{in}$ is the carry (LSB) from the next higher order digit. $C_{out}$ is

the carry (LSB) to the next lower order digit. Figure 8b shows a typical pedestal gate which is enabled when X is in a 0 state and Y is in transition from a 0 to a 1 state.

The operation begins when the BCD digits are placed in their respective registers. Instead of being placed directly into the registers, they are all shifted by one bit toward the least significant bit. This is done because the entire operation is performed on the basis that when the LSB of a digit (1 bit) is shifted to the MSB of the next lower order digit (8 bit), if it were a 1 it would have represented 10 times the value of the lower order's 1 bit (or a ratio of 10 to 8 of the value of the MSB of the lower order digit). The value of a true binary number would be divided in half by shifting one bit to the right and the result would be 5. However, the shift makes the 10 into an 8 and 3 must be subtracted to attain the correct value.

When first loading in our number, the 8 bit of the LSB is a true binary 8 and should not be altered. Because it need only be shifted to the right by one, it can be initially loaded in one bit toward the LSB, making all future operations adaptable to the preceding truth table, input, and output equations.

Figure 9 shows the second method for BCD conversion. This method uses the principle that pulses are equal in number no matter in which system the count is represented. The BCD number is loaded into a BCD counter and a pulse source is used to count down the BCD counter and simultaneously count up a binary counter. When the BCD counter reaches the count of zero, the pulse source is disconnected and the binary counter will then contain the equivalent binary number.

The implementation shown in Figure 9 is necessary because of the limitations of the hardware. The BCD counter will count in only one direction—up; the binary counter will count in either direction—up or down. The BCD number is loaded into gating which will perform a 9's complement operation. The result is loaded into the BCD counter and, when a convert signal is given, a clock (astable multivibrator) is activated which counts up both the BCD and the binary counters. When the BCD counters contain 9999, a signal is generated to stop the clock and the equivalent binary number is then contained in the binary counter. The truth table and input equations for the gating of the three MSD of the decimal number are:

| | | | | |
|---|---|---|---|---|
| 1 | 0101 | 0101 | 0101 | 0101 |
| 2 | 0011 | 0011 | 0011 | 0011 |

| | | | | |
|---|---|---|---|---|
| 4 | 0000 | 1111 | 0000 | 1111 |
| 8 | 0000 | 0000 | 1111 | 1111 |
| $1_S$ | 1010 | 1010 | 10ØØ | ØØØØ |
| $2_S$ | 0011 | 0011 | 00ØØ | ØØØØ |
| $4_S$ | 0011 | 1100 | 00ØØ | ØØØØ |
| $8_S$ | 1100 | 0000 | 00ØØ | ØØØØ |

$1_S = \overline{1}$ 9999

$2_S = 2$ -XXXX (decimal number)

$4_S = 2\overline{4} + 4\overline{2}$ 9999-XXXX (load into BCD counters)

$8_S = \overline{2}\overline{4}\overline{8}$ Count to 9999 which will require XXXX counts

In this case, the MSD can be only a 1 or a 0 (decimal) because the highest memory location is 1023.

Comparison of the two methods of operation shows that the first method requires far less time to convert a number (assuming that the number is greater than 4). However, the first method requires more sophisticated gating and therefore costs more and takes up more space. If the conversion is to be done during the course of other operations and in a minimum of time, the first method would be preferred. However, if the conversion is to be done only once and before any other function is performed, as in this system, speed is not of prime concern and the slower, more economical approach would be more appropriate. For this reason, the second method is the best suited to the requirements of the HSSRS.

## MAJOR SEQUENCE CONTROL SIGNALS

As previously described, two 4-digit BCD switches are mounted on the control panel. One of these switches controls the number of memory locations used per memory block; the other controls the starting of the tape transport.

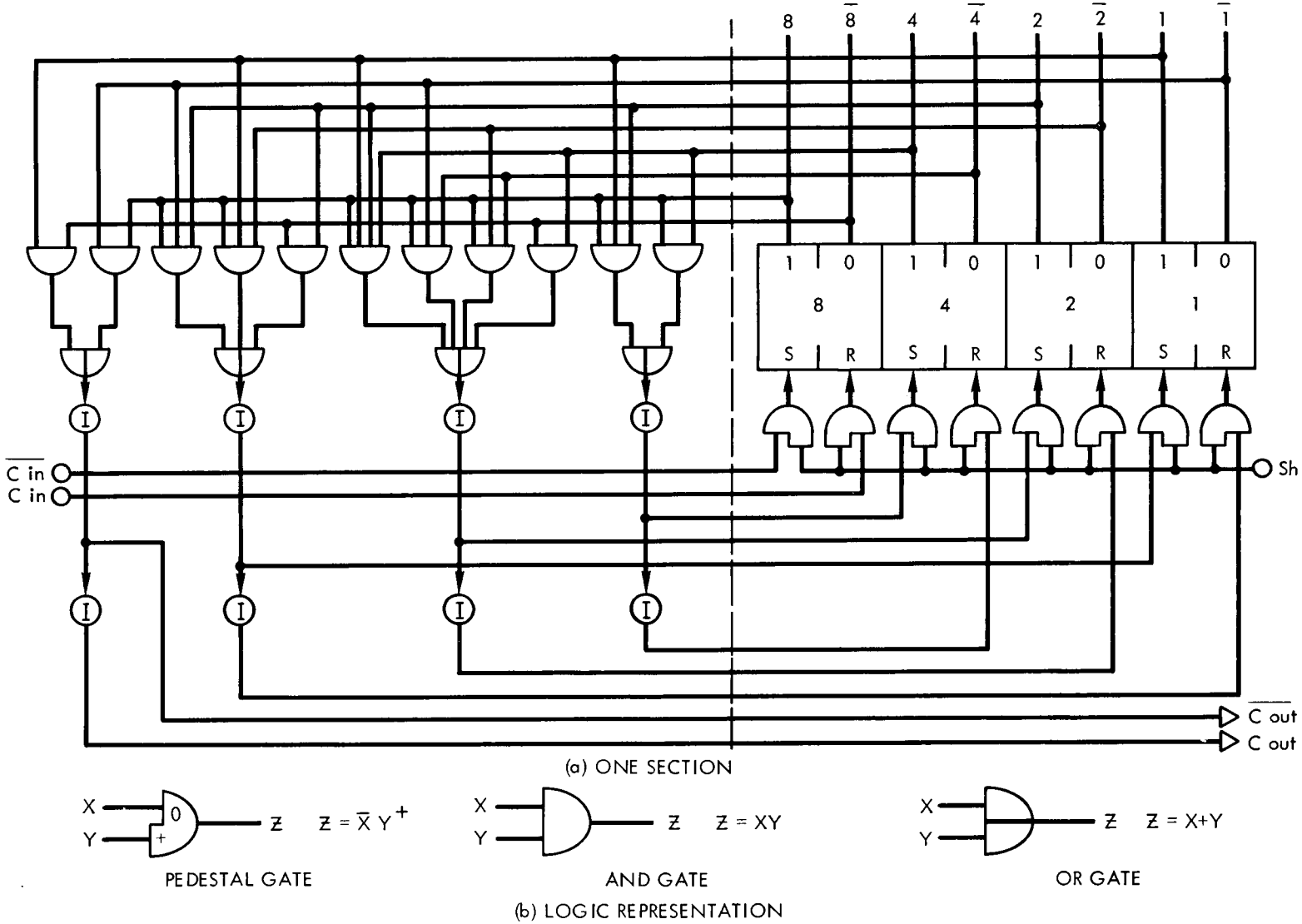


Figure 8. Four-Bit Pedestal Gated Register with Inspection Gating

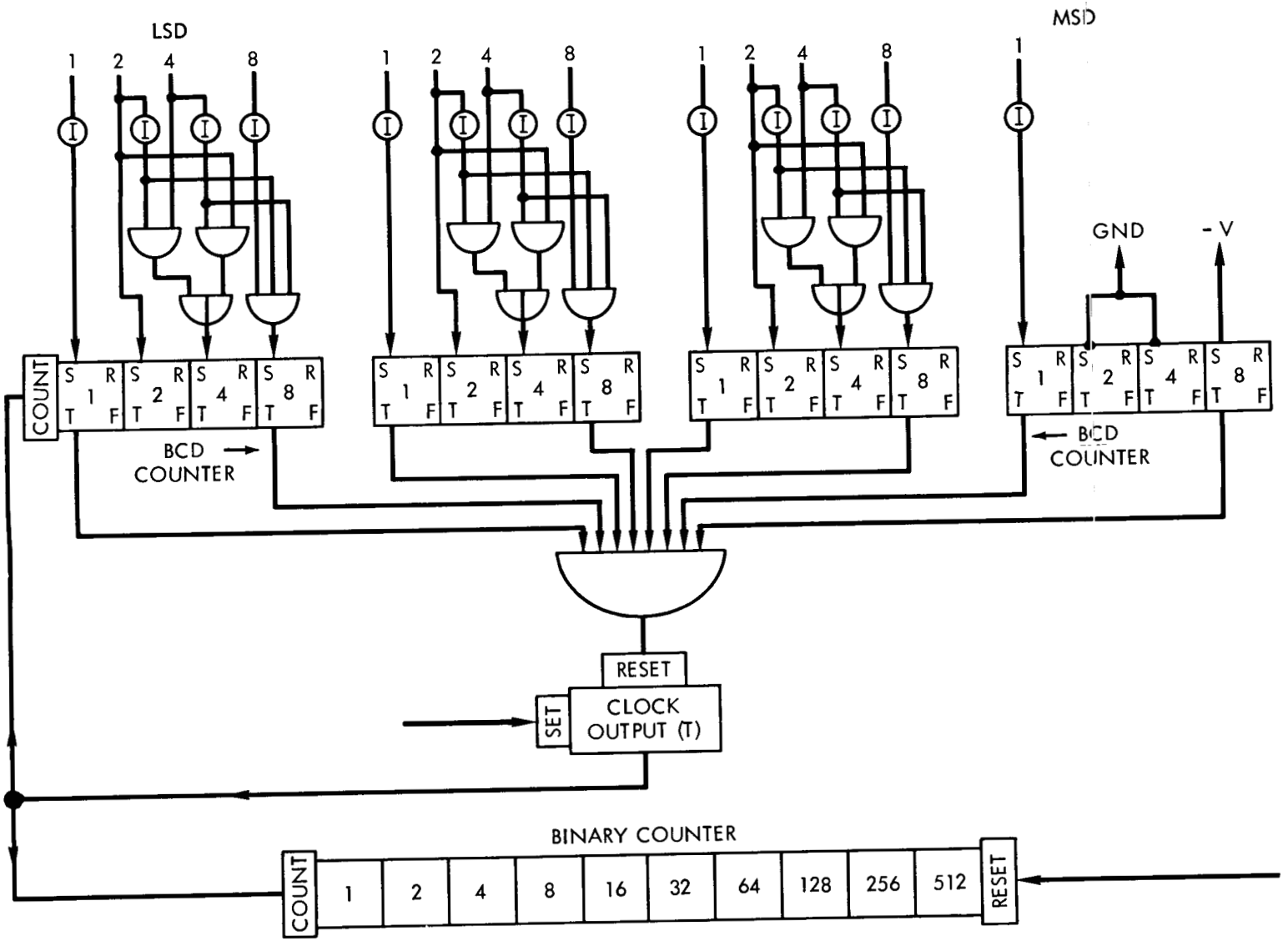


Figure 9. Decimal-to-Binary Conversion, Serial

The three major control signals derived from these two switches are:

1. Tape start

2. Unload cycle start

3. Unload cycle complete

The first and second signals are formed by a logical combination of the binary equivalent of the decimal number on their respective switches and a binary counter which contains the memory address at all times during the loading cycle. Figure 10 shows the configuration used.

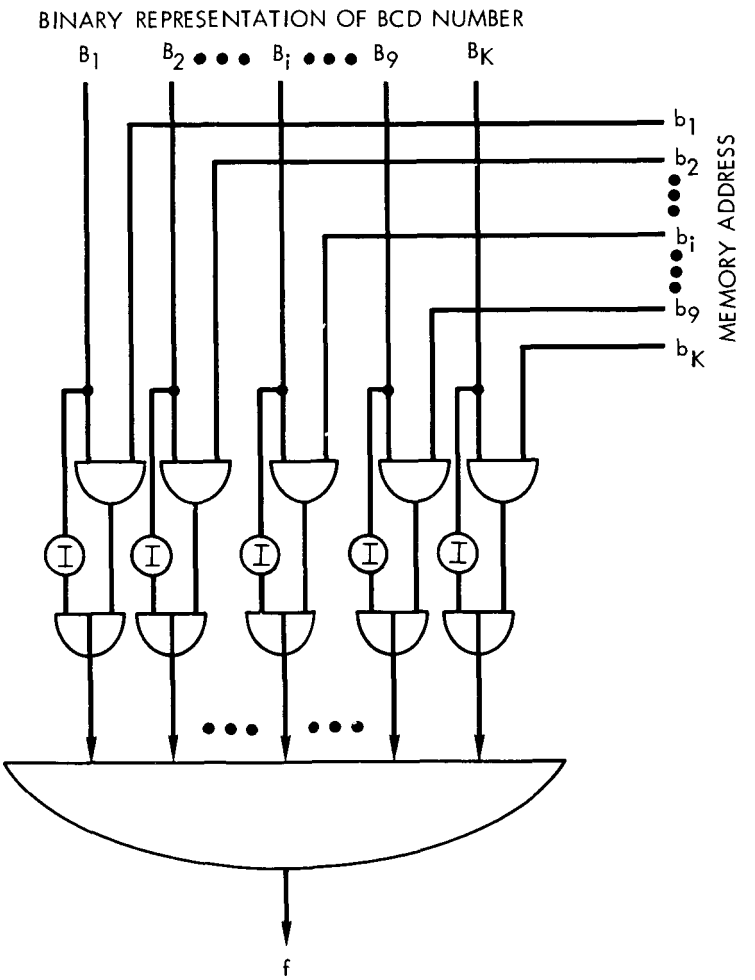


Figure 10. Major Sequence Control Signals

As previously described, the tape transport requires 3 milliseconds for the tape to attain its proper recording speed. Therefore, the "tape start" signal should be activated 3 milliseconds before the beginning of a memory dump. The "unload cycle start" signal triggers a memory dump. Each bit ($b_i$) of the binary counter which contains the memory address during a load cycle is combined with the equivalent bit ($B_i$) of the binary counter which contains the converted BCD number representing the number of memory addresses used per block of data. They are combined as follows:

$$f = \left[(b_i \cdot B_i) + \overline{B}_i\right] \cdot \left[(b_{i+1} \cdot B_{i+1}) + \overline{B}_{i+1}\right] \cdots \left[(b_k \cdot B_k) + \overline{B}_k\right]$$

There are two ways in which each of the expressions in the brackets can become a logical 1, thereby enabling one leg of the k-legged AND gate. First, the $B_i$ bit may not be selected, eliminating the necessity of the corresponding $b_i$ bit being a 1 to achieve an output. Second, the $B_i$ bit may be selected, in which case the $b_i$ bit must also be a 1 in order to enable f. In this manner, when all of the $B_i$ 1 bits are matched by $b_i$ 1 bits, the f AND gate will be enabled. It can be seen that when counting in the binary system, the correct result will be reached by classifying 0 bits ($B_i$) as "don't care" terms and searching only for the 1 bits. A binary count with more 1 bits, including those designated as selected bits, cannot be attained without passing through a state in which the selected bits are the only 1's present.

The "unload-cycle-complete" signal is a combination of the binary equivalent for the decimal number of the switch used for signal 2 and a binary counter which always contains the address of the memory during the unloading cycle. The configuration is the same as that of the two previous signals. In the preceding manner, signals are developed which indicate when to start the tape transport so that it will acquire recording speed, when the loading cycle is complete and it is time to dump the memory onto tape, and when the unloading cycle dump has been completed.

## LOADING CYCLE

Figure 6 is a timing diagram of a load cycle. This timing sequence is implemented with the circuitry shown in Figure 11. When a signal is received from the pegboard, it triggers a monostable multivibrator (O/S) which strobes the six most significant bits (7-12) into the memory. At the completion of this strobe, there is a 3-microsecond delay after which the six least significant bits (1-6) are strobed into the memory. After each word of data is stored in the memory, a binary counter, which keeps track of the current address in the memory during only the load cycle, receives a pulse which advances it to the

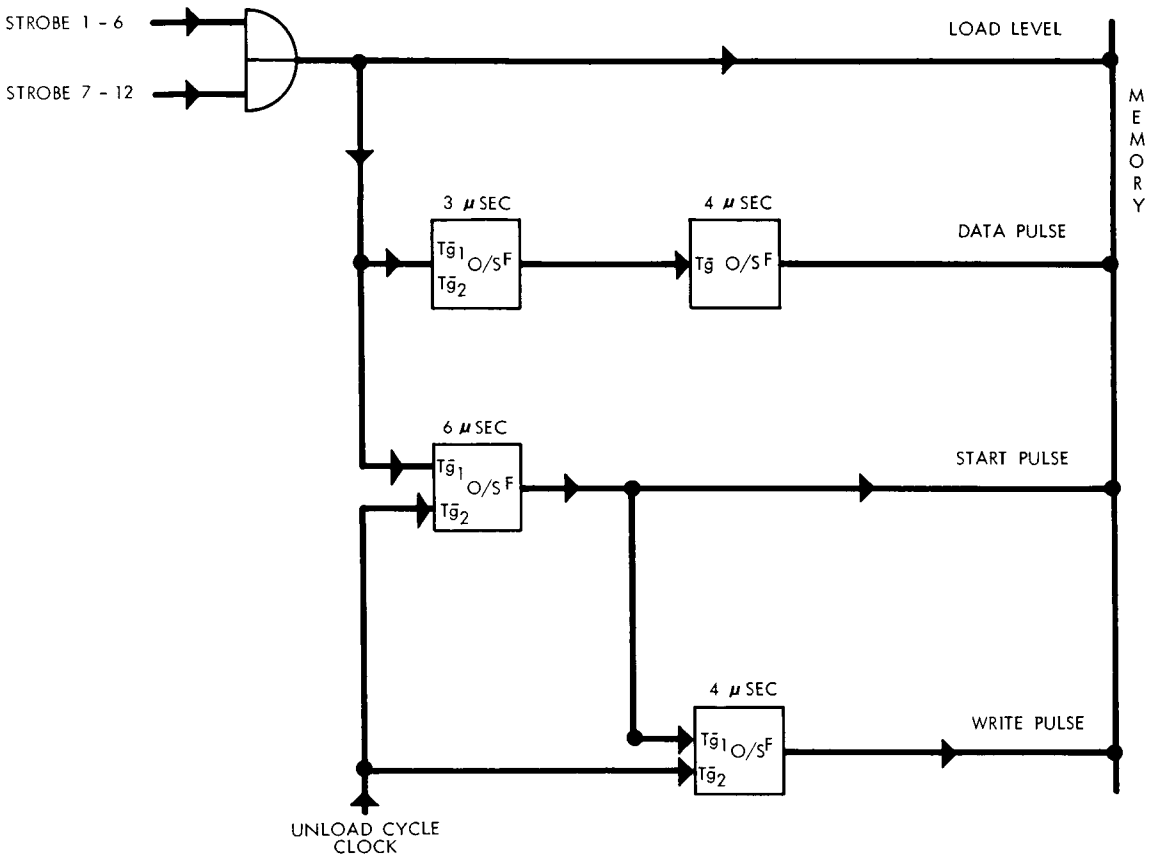


Figure 11. Memory Control Section, Read and Write, Logic Diagram

next address. This counter also addresses the memory. This counter is logically OR'ed with another binary counter (Figure 12) which contains the current address during the unloading cycle. The load-level line serves as the decision line and decides which phase of the operation is being performed. If the load level is a logical 1, the load-cycle address will control the memory; if the load level is a logical 0, the unload-cycle address will control the memory.

## UNLOADING CYCLE

The unloading cycle begins when the memory-address set on the front-panel thumbwheel switch is reached, thereby generating the "unload cycle start" signal. The signals which unload the memory are the same as those required to load the memory except that input data (with or without a strobe pulse) need not be supplied. Therefore, it is possible to use a configuration similar to Figure 11 but with different input triggers.

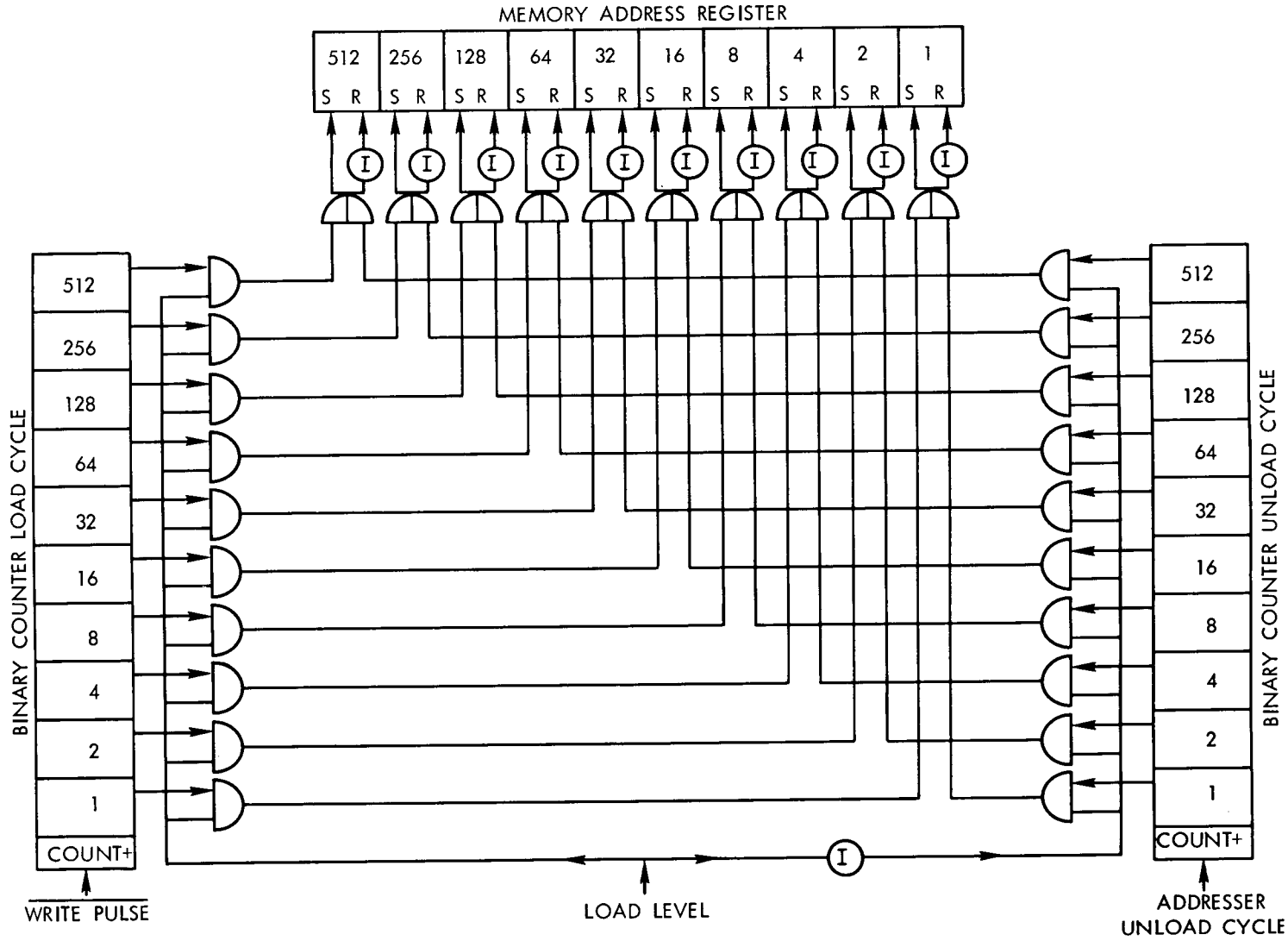


Figure 12. Memory Control Section, Address, Logic Diagram

Upon the initiation of the unloading cycle, an astable multivibrator is set into operation. This clock has a period which is a whole number divisor of the data-collection system time-slot period. The clock is the controlling element in the operation of the HSSRS during the unloading cycle and it replaces the signals from the pegboard used during a loading operation. The clock pulses serve as start sync pulses and also serve to step the unload cycle binary counter which will address the memory during the unload cycle. The load/unload level will remain at 0 volt because there are no signals from the pegboard. The operation continues asynchronously until either the unloading cycle is completed or a signal is received from the pegboard to store data. If a signal is received from the pegboard, the clock is stopped and at the same time a monostable multivibrator, with a period equal to the period of the time-slot signals, is triggered. When the one shot returns to its quiescent state, it retriggers the clock into operation. In this manner, the data and parity are strobed from the memory through a parity generator into a register of complementing flip-flops which change state only on a transition from a logical 0 to a logical 1, thereby generating NRZ on 1's data at the output of this register.

From here the data and parity are strobed onto magnetic tape at a constant rate which is intermittently interrupted by incoming data. The latter phenomenon causes a blank space on the tape which is indistinguishable from changes in the distance between the data caused by variations in the speed of the reels of the tape transport.

Longitudinal parity is accomplished by resetting the complementing register, which contains the NRZ data. This action will cause all the flip-flops in a 0 state to remain in a 0 state and will change all the flip-flops in a 1 state to a 0 state. Therefore, longitudinal parity will contain a 1 on the magnetic tape in every position where the register ended in the 1 state at the end of a memory dump. Because all cells of this register will be initially reset to the 0 condition, any register cell which contains a 1 at the end of a dump will show that there was an odd number of 1's recorded on that line. Therefore, longitudinal parity should also contain a 1 in that position in order to supply even parity.

Figure 13 shows the memory input-output register and the NRZ register.

## CONCLUSIONS

The HSSRS provides the capability for: (1) selecting any number of data points, (2) storing these points in a memory until a maximum number of records are stored, and (3) dumping these data in a block format through a parity generator into a tape transport. The tape transport is automatically started and

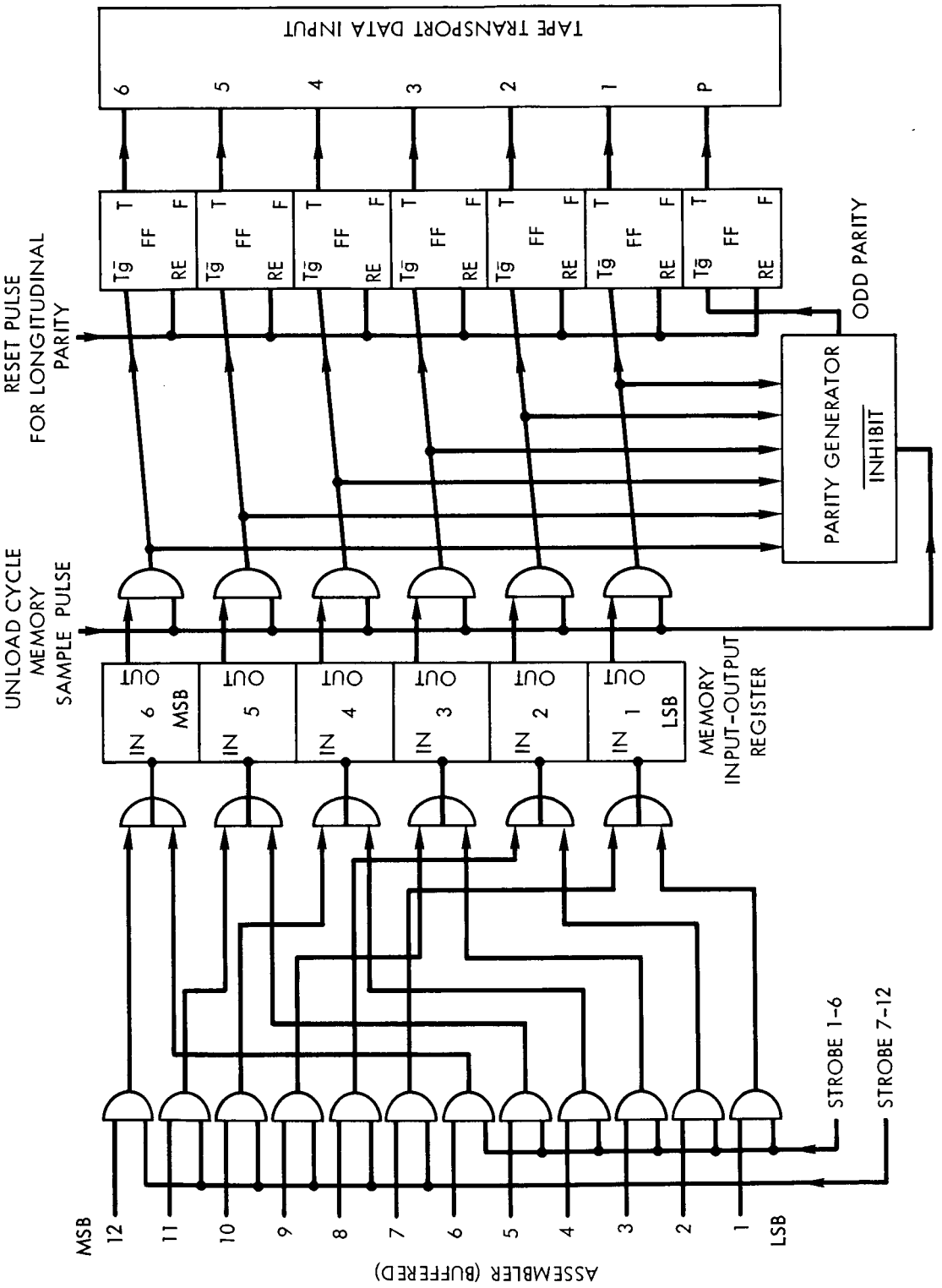


Figure 13. Assembler to Memory to Tape Transport Interface

stopped to provide maximum packing density on the tape. This system can be reprogrammed by simply rearranging the pegs in the pegboard and changing the setting on the two binary coded decimal switches.

Use of the HSSRS pegboard eliminates the need for an on-line computer programmed to strip the desired data from the remaining data or for recording the entire format on magnetic tape.

Also, this sytem is very adaptable in that the block size can be adjusted to any length, making it possible to format any configuration of data inputs. In addition, the start of the tape transport can be controlled to provide maximum tape use while providing a gapped tape. The HSSRS accomplishes the major objectives set forth to enhance the efficiency of high-speed data collection and processing.

## BIBLIOGRAPHY


Flores, Ivan: Computer Logic. McGraw-Hill Book Company, Inc., New York, 1960

Gill, Arthur: Introduction to the Theory of Finite-State Machines. McGraw-Hill Book Company, Inc., New York, 1962

Maley, Gerald A., and Earle, John: The Logic Design of Transistor Digital Computers. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1963

NAVCOR Users Manual. Navigation Computer Corporation, Norristown, Pennsylvania, 1964

Richards, R. K.: Arithmetic Operations in Digital Computers. D. Van Nostrand Company, Inc., Princeton, New Jersey, 1955

Strauss, Leonard: Wave Generation and Shaping. McGraw-Hill Book Company, Inc., New York, 1960