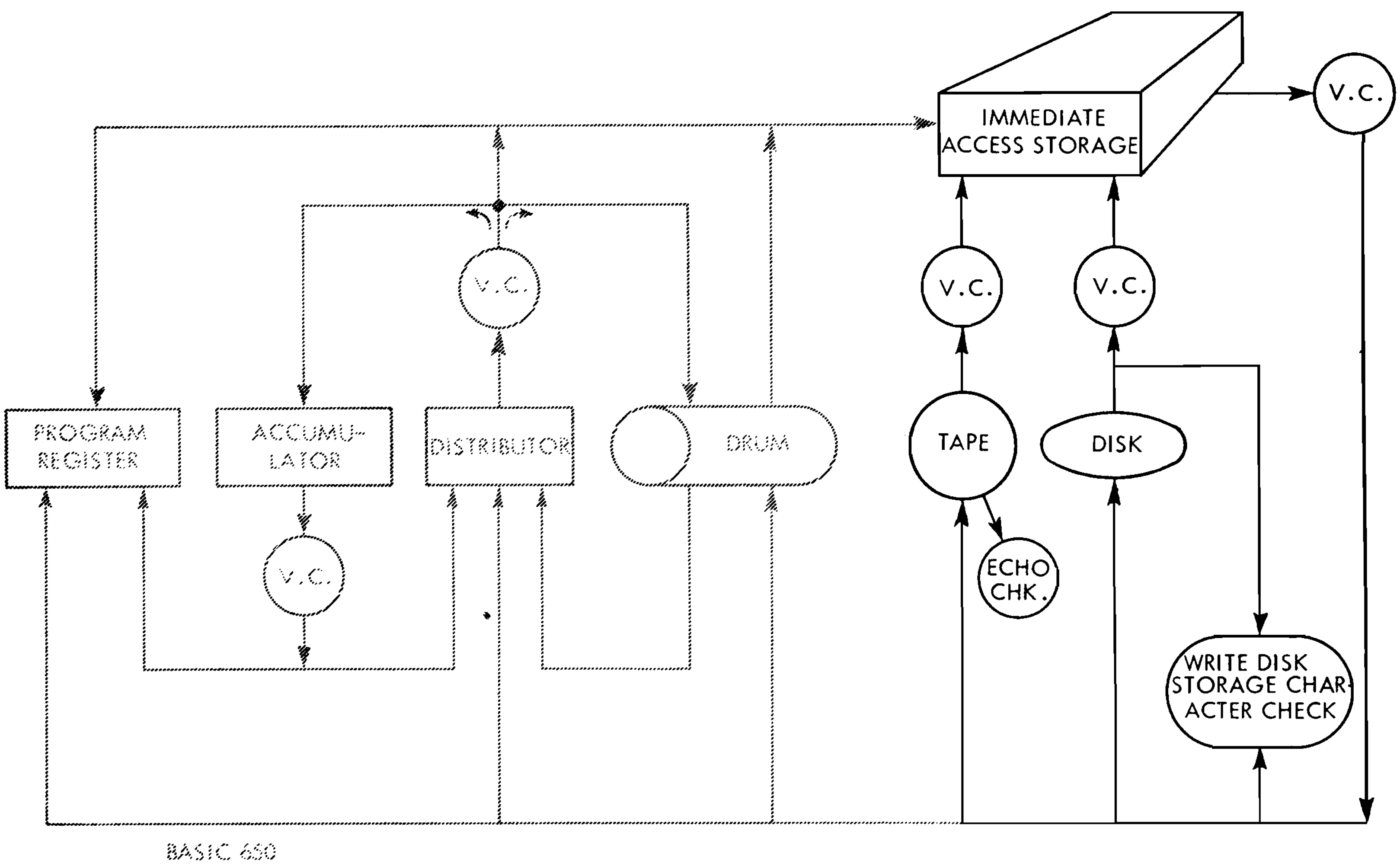


FIGURE 9. 650 SYSTEM DATA FLOW AND CHECKING

## 652 Control Unit

The 355 is connected to the 650 through the 652 Control Unit and the 653 Storage Unit.

The 652 Control Unit combines the circuitry for control of magnetic tape units, disk storage units and inquiry stations. It accepts operation codes and addresses from the 650 to direct the functions of the specified unit.

Figure 10 shows the arrangement of the lights and switches on the 652 Console. These are used to visually indicate the functions taking place in the various units controlled by the 652.

## 653 Storage Unit

The 653 Storage Unit contains the 600 positions of immediate access storage. Whenever a disk record is consulted, the entire track of data is read to or written from immediate access storage. Immediate access storage serves as a link between disk storage and the 650. The capacity of immediate access storage is identical to that of a disk track because each holds 600 digits of data as 60 words.

Figure 11 illustrates the light panel on the 653. It indicates when the units affected by the 653 are functioning in the system.

# TIMING CONSIDERATIONS

## Seek Access Time

In a previous section describing operation codes, the upper and lower limits of seek access time were stated. This section describes the seek time in more detail. Several charts show pictorially the range of access time for address changes involving a move of an access arm. These charts depict the action of any one access arm in a file. They do not show the net effect on timing that results from three access-arm operations. Obviously when the 650 program integrates the operation of the three access arms, the processing unit of the system will have record data from disk storage constantly available.

Figure 12 shows the range of access time for an address change made to another track on the same

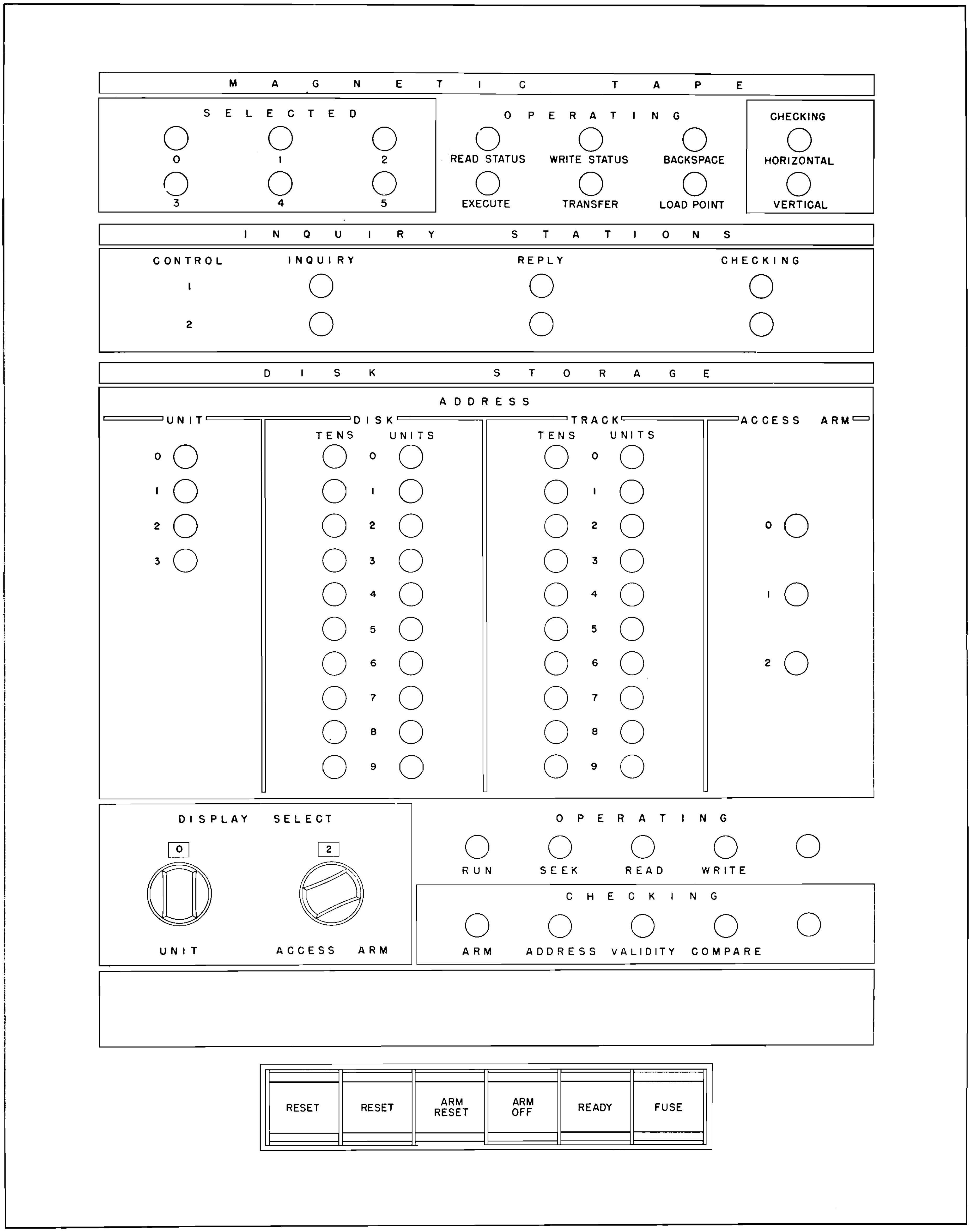


FIGURE 10. 652 CONSOLE

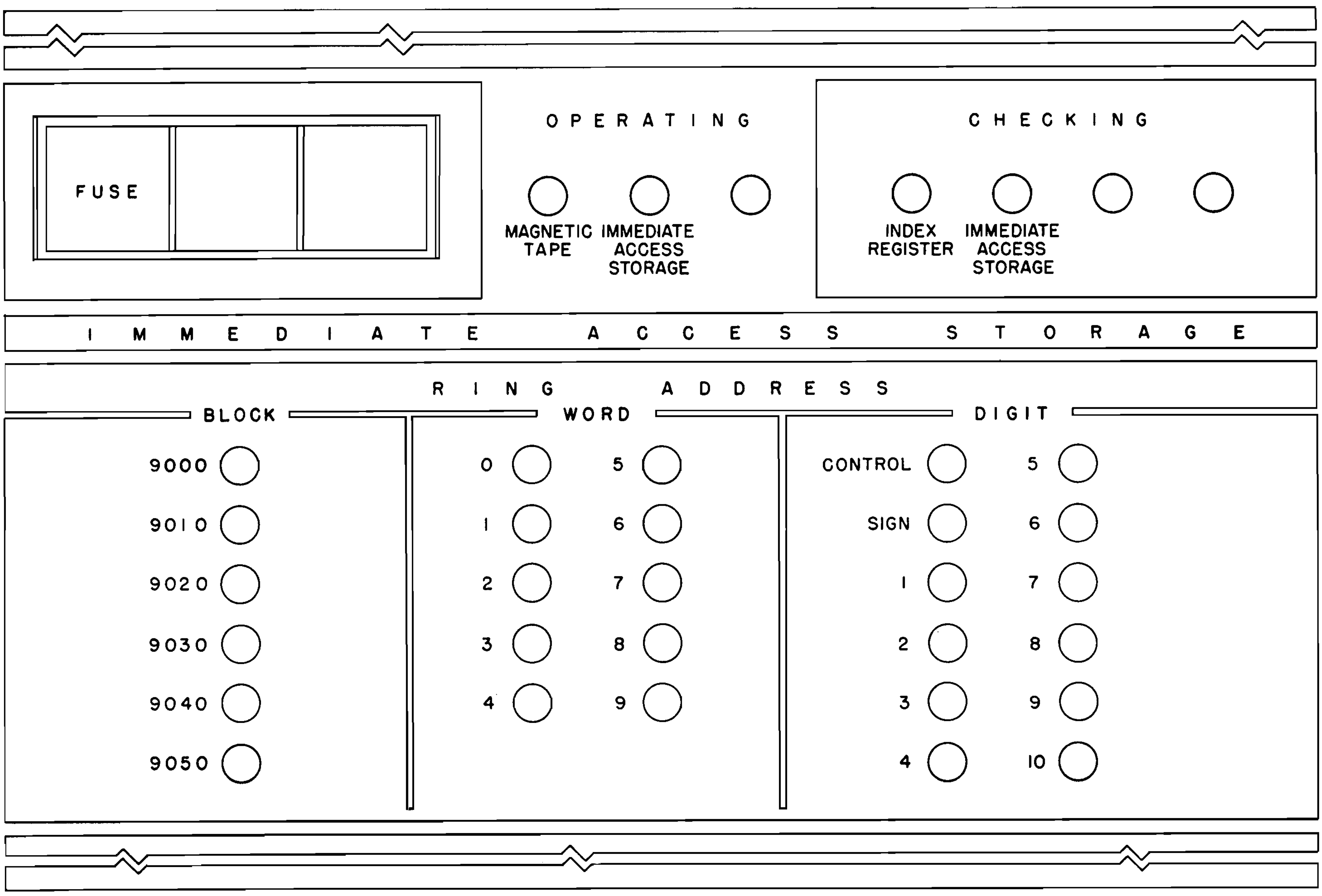


FIGURE 11. 653 CONSOLE

disk. Note that a change of 50 tracks takes about 200 to 250 milliseconds. A change of 25 tracks takes about 175 to 225 milliseconds.

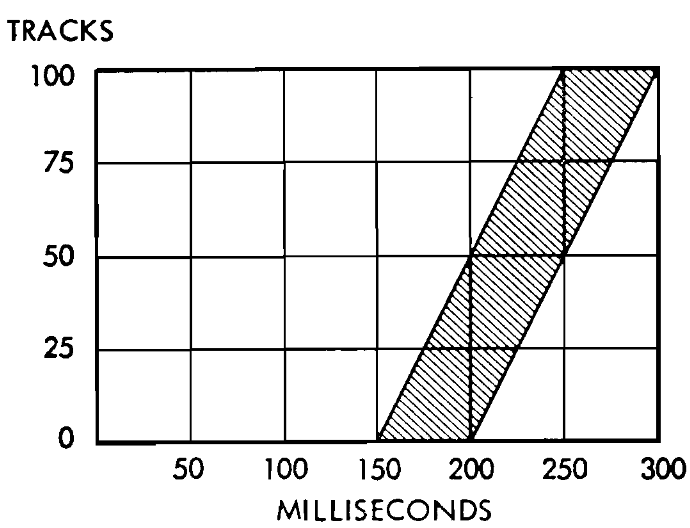


FIGURE 12. SEEK TIME FOR ACCESS ARM MOVING FROM TRACK TO TRACK ON SAME DISK

Figure 13 depicts an address change involving a disk change; that is, an address change where the ac-

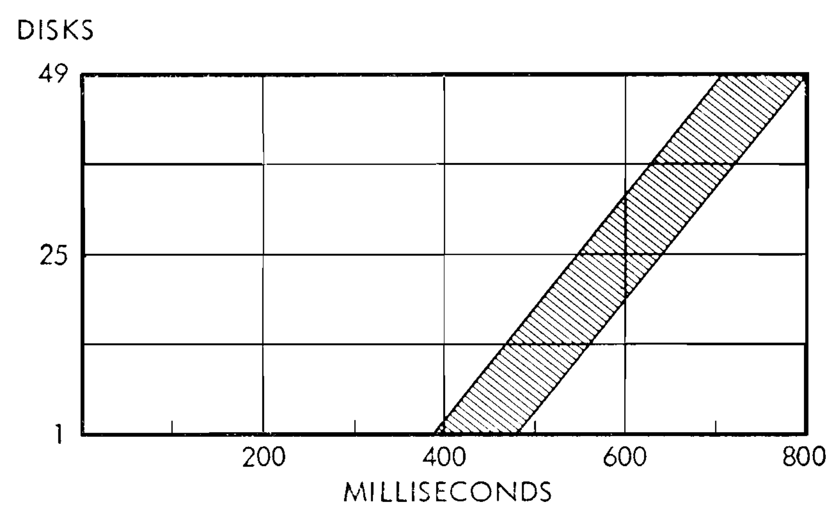


FIGURE 13. SEEK TIME FOR ACCESS ARM MOVING FROM TRACK 00 ON ONE DISK TO TRACK 00 ON ANOTHER DISK

cess arm leaves a specific track on one disk, moves up or down to another disk, and moves in to a specific track on the new disk. Note that the minimum move is about 400 milliseconds. A move of 25 disks takes approximately 600 milliseconds. A change of 37 disks takes approximately 650 milliseconds.

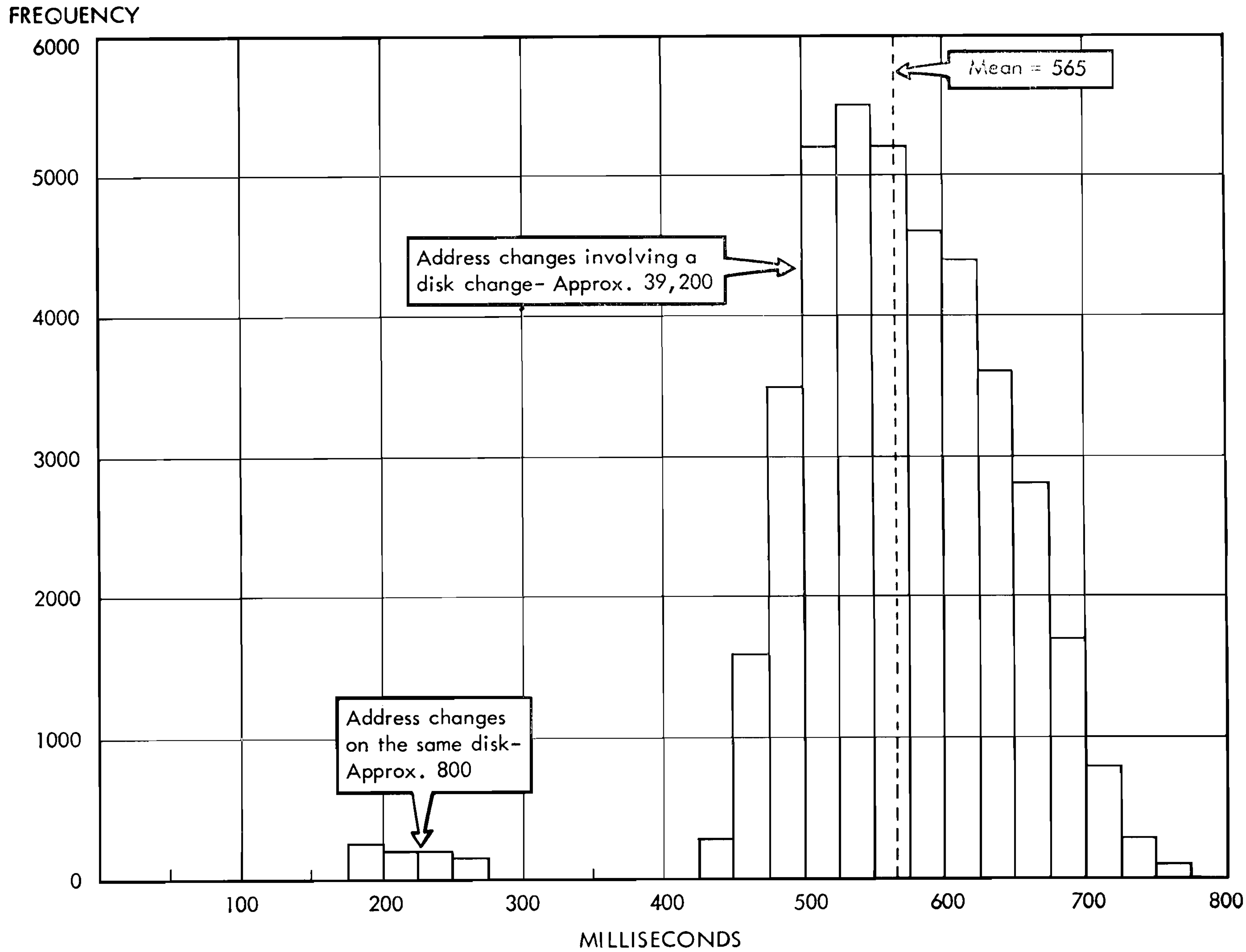


FIGURE 14. FREQUENCY DISTRIBUTION OF ACCESS TIME FOR 40,000 DISK STORAGE CHANGES

## Estimating the Overlap of Access Time with 650 Processing

One of the objectives in programming the 650 RAMAC is to completely overlap seek time with processing of data. Where the processing time is longer than the net seek time (3-access arm operation) the seek time is immaterial. Where the processing cycle is short, it is advantageous to know the average seek time. This, however, is dependent on several factors, such as how the data in the file are organized; the method used to convert the record identification code to a disk address, and the possibly unknown pattern of address changes requested of the file. If the access arm is continually being asked to move to extreme ends of the disk file, the average access time will be higher than if the access arm moves short distances for successive address changes. Therefore, an average access time to data in the file will vary from one application to another.

Statistical studies have been made of access time for address changes resulting from random number addresses. Figure 14 is a bar graph showing the frequency distribution for 40,000 random address changes. Note that 800 of the 40,000 changes were to an address on the same disk. Address changes of one to 49 disks totaled 39,200. This averaged out to about 565 milliseconds for a random change in address. Again, keep in mind that the bar graph is plotted for one access arm operation only and is based on random seeks to the entire file of 100 disk faces.

Figure 15 shows a probability curve for access time involving a disk change. This is plotted against milliseconds. To interpret the curve assume that the programmer allows 565 milliseconds for access time to a record in the file. The probability curve shows that in approximately fifty per cent of the cases this will be sufficient time for one access arm to move anywhere in the file to find the record. When 680 milliseconds is allowed, the probability curve indicates approximately 90 per cent.

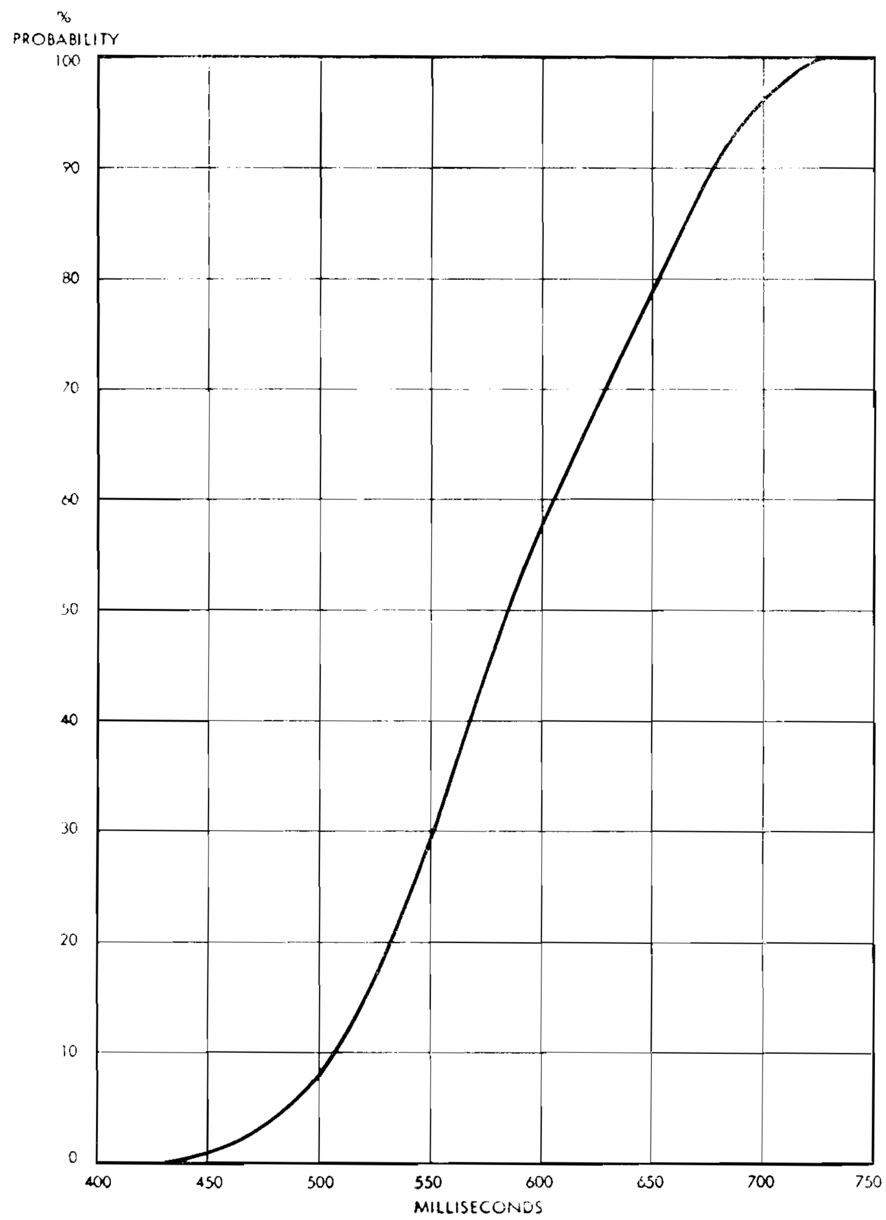


FIGURE 15. ACCESS TIME PROBABILITY INVOLVING A CHANGE IN DISKS

## OVER-ALL JOB TIMING CONSIDERATIONS

THE ESTIMATING of the total time required for a data-processing system to handle an application is based on several factors: the input and output document volume, the internal processing time, and the frequency (daily or weekly, etc.) of output. All these factors make up the total time. These factors are directly affected by the speed and timing considerations of the individual components of the data-processing system. For example, the maximum output printing speed of the 407 is 150 lines per minute, and, where the 407 is the output device, this is the maximum output production. Each system contains a variety of components whose operation needs consideration; however, the three basic elements are input volume, processing time, and output volume. The most efficient operation will be determined by overlapping the operational time of the various components as much as possible. No estimated input or output speed should be assumed until plotted as described herein.

This section of the manual describes a way to layout the operational time of the 650 RAMAC components. The application illustrated in Figure 2 is used as the example. The first chart shows the layout for processing an order card when only one 355 access arm is utilized. The second chart shows how the punch feed of the 533 Card Read Punch can be kept in continuous operation and two access arms are used. The third chart shows the 407 printer operating at maximum speed of 150 lines per minute and three access arms are utilized. The timing considerations for including manual inquiries are described in another section which follows the description of these units.

The timing charts show a separate horizontal line for each functional component. The length of each horizontal line segment is expressed in milliseconds and labeled to indicate the function performed in that segment of time. A millisecond (ms) is 1/1000 of a second and provides a convenient time scale, because all functions are very easily expressed in this medium. For example: ½ second = 500 milliseconds; 1/10 second is 100 milliseconds, etc. The 533 Card Read Punch operates at 100 punch cycles per minute; therefore, each punch cycle takes 600 milliseconds for the punch to complete the cycle. The 650 program is delayed for only a portion of this time and can be preparing data for the next punch cycle. In applications where this 650 processing is completed within the time required for the 533 cycle, the punch can be operated at its maximum speed.

Thus, a method of planning applications efficiently is to lay out the estimated timing considerations with as much overlapping of functions as possible within the programming structure. The layout may show a good balance between input, processing, and output times, or it may indicate that the operation is not functioning at its maximum speed, in which case further planning or a new approach may improve the efficiency.

The timing chart for one access arm operation (Figure 16) shows a separate line for the 355 access arm, the 650 programming, the 533 punch, and the 533 read operation. Each of the 355 and 533 functions is highlighted in the block diagram to show the

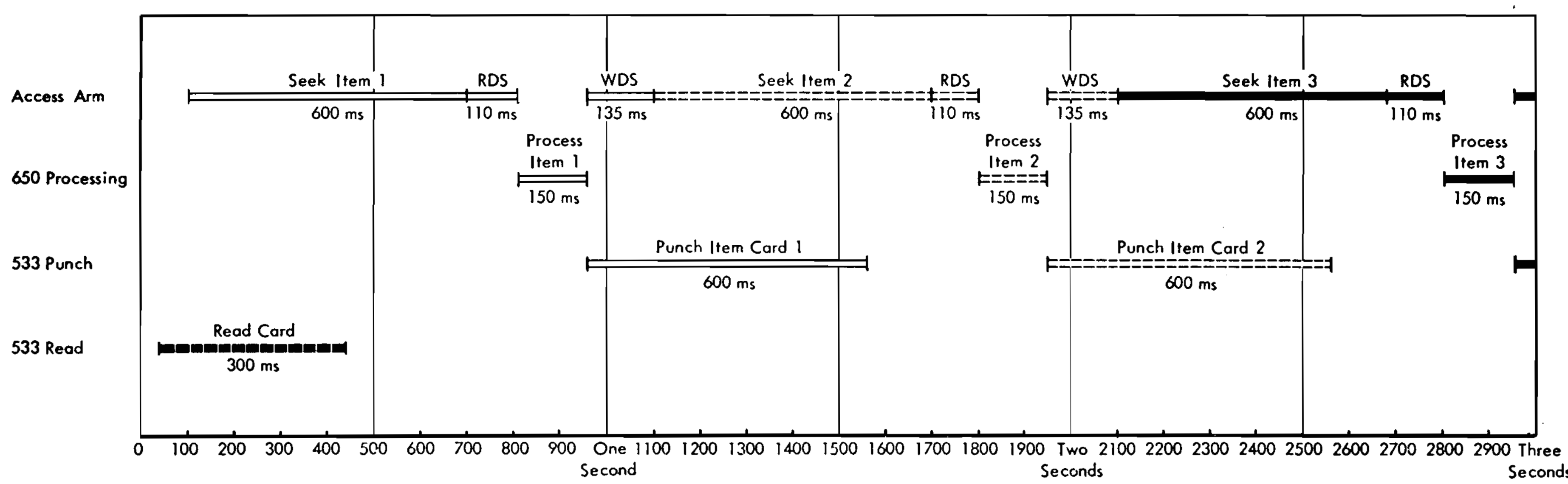


FIGURE 16. TIMING CHART FOR ORDER CARD PROCESSING USING ONE ACCESS ARM

relation of the timing chart to the over-all programming. The timing charts assume a seek time of 600 ms and a disk storage read time of 100 ms. The chart begins by reading an order card and programming a seek for the first-ordered item. The timing chart shows that the next card is being read by the 533 while the seek is executed. At the completion of the seek, the item record is read into immediate access storage.

The 650 processing time includes the programming required to update the inventory record for this item, calculate the price and cost, perform the distributions by class, etc. The write-disk storage instruction is then given to record the updated item record, an item card is punched, and the seek for the second item on the order card is begun. Note the overlap of disk storage write and seek time with card punching time. A complete processing cycle for one item totals 995 milliseconds. This is composed of a 600 seek, a 110 read, a 650 processing time of 150, and disk storage write of 135. The punching time is completely overlapped by RAMAC operations.

A better approach would be to use two access arms and keep the 533 punching output cards continuously. Figure 17 shows that access arm 1 seeks, reads, and writes item one, three, five, etc., of the order card. Access arm 2 seeks, reads, and writes item two, four, six, etc., of the order card. The 650 processing and immediate access storage time is short enough to be overlapped by a card punch cycle: therefore, the punch runs continuously at its maximum rate for all the items on the order card.

The complete series of operations for each given item are identified by the same shading. This shading permits following one item from start to completion. Note that the pattern for a given item follows a diagonal line from the top to the bottom of the chart. By a vertical inspection at any point on the chart, all the units in operation at that time can be seen. By following each horoizontal line the operating time of each component can be determined. A study of the timing chart from these various angles can be utilized to evaluate the machine approach that is planned for a given application.

As an alternate method the application output may be directly printed on a 407 attached to the 650; the timing chart in Figure 18 applies. The maximum operating rate of 150 lpm of the 407 is faster than the punching rate of 533. Therefore, the 650 processing and disk operations must produce an output every 400 milliseconds in order to keep the 407 running at maximum speed (150 lpm). This can be accomplished in this application study by using three access arms. Applications that require large amounts of 650 processing time obviously will reduce the printed output below the maximum rate.

As in the previous chart all the operations for a given item are identified by a specific shading. The chart assumes six items are ordered on the one order card. The chart carries the processing through the reading of the next card. Note that the 650 process-

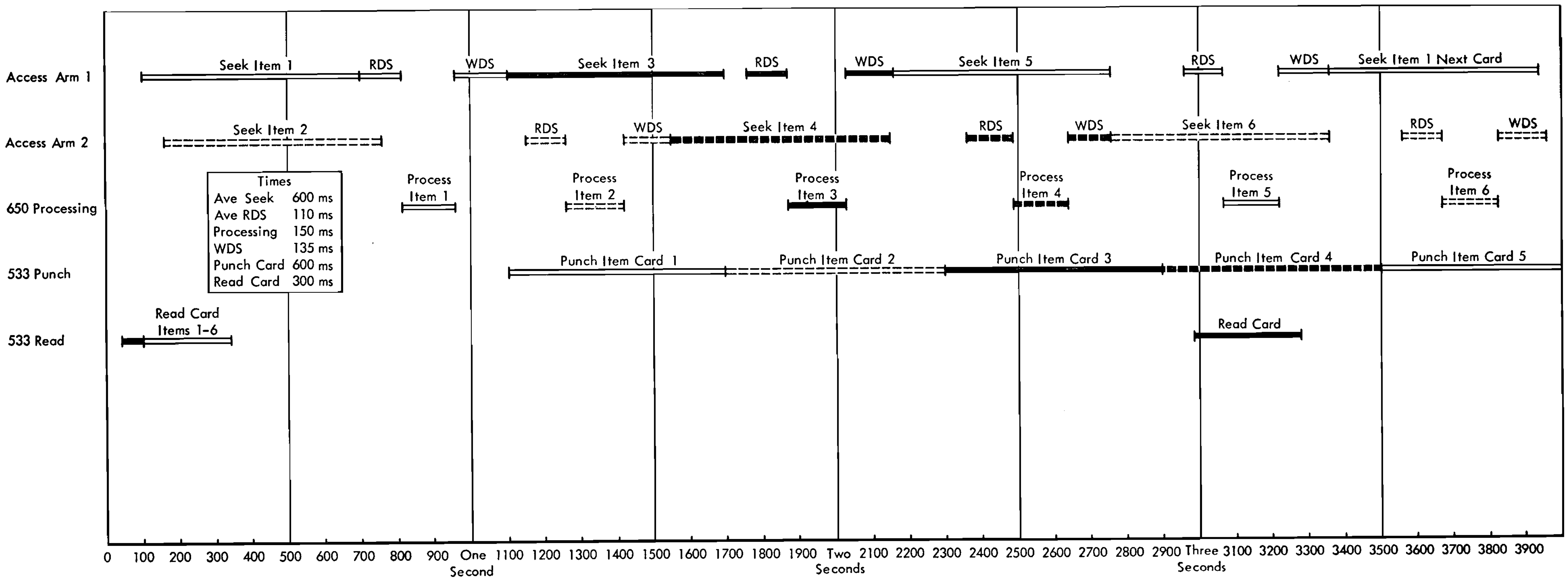


FIGURE 17. TIMING CHART FOR TWO ACCESS ARM OPERATION



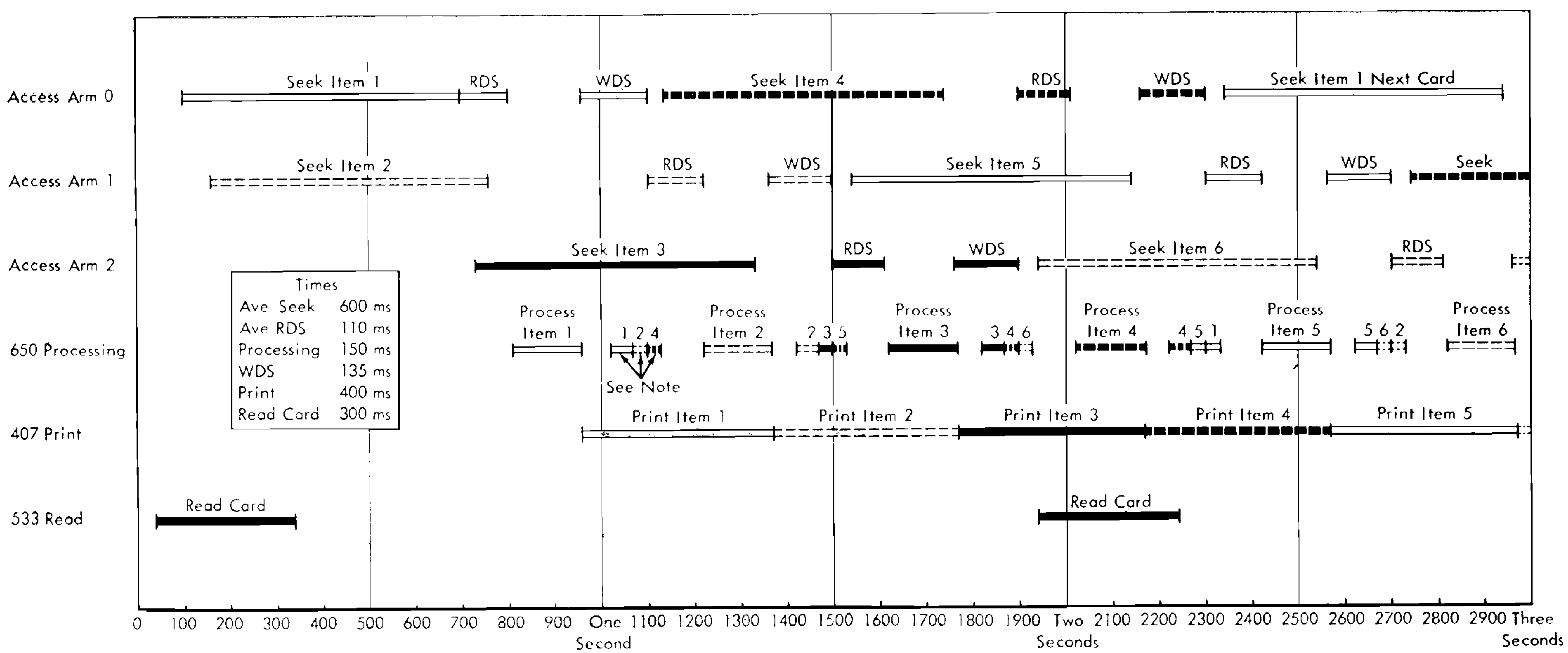


FIGURE 18. TIMING CHART FOR PRINTING 150 LPM AND UTILIZING THREE ACCESS ARMS

ing time is shown as several line segments. The basic processing for an item occurs between the read-disk storage and write-disk storage operations. Other 650 processing can occur during the read and write-disk storage operations.

Note the two-part line segment of 650 processing, which occurs during a write-disk storage operation. The first part is used to make the sales distribution on the drum for the previous item processed. The second part is preparation to accept the next record read from disk storage. During the read-disk storage operation the seek address for the next item is developed. By arranging the 650 processing time properly, the maximum output for an application can be planned and block diagrams of the programming prepared accordingly.

## Unloading the Disk File

The integration of magnetic-tape units with disk files offers considerable flexibility for application planning. One advantage is that periodically it may be desirable to unload the file so that it can be used for another application. It is also desirable to unload the file so that an independent record of the status of data in disk storage at a given time is available. This independent record can be utilized to create periodic reports of inventory, such as a monthly stock-status report and others. The independent record offers a valuable system check on the data in the disk file.

The disk file can be unloaded to punched cards or to magnetic tape. Unloading to magnetic tape is more desirable because it is much faster. A complete file can be unloaded to magnetic tape in about 30 to 40 minutes. Unloading to cards would take approximately 7 hours if two card punches are connected. Figure 19 shows the programming to unload the file to magnetic tape. Note that it utilizes two access arms which keep immediate access storage operating continuously except when an arm moves to the next disk. A program written for three access arms would completely overlap the disk change seek time. As a general rule when using multiple arms, it is desirable to send an arm seeking a new address as soon as it has completed its functions.

| INSTR NO. | LOCATION OF INSTRUCTION | OPERATION | | ADDRESS | | REMARKS |
|---|---|---|---|---|---|---|
| | | ABBRV. | CODE | DATA | INSTRUCTION | |
| | | | | | | |
| | | Start Routine | | | | |
| | | | | | | |
| | 0051 | RAL | 65 | 0001 | 0052 | Lower Accumulator used for Arm 1 |
| | 0052 | AU | 10 | 0002 | 0053 | Upper Accumulator used for Arm 0 |
| | 0053 | LD | 69 | 8002 | 0054 | |
| | 0054 | SDS | 85 | 9000 | 0055 | Even tracks developed in Lower Accumulator |
| | 0055 | LD | 69 | 8003 | 0056 | |
| | 0056 | SDS | 85 | 9000 | 0057 | Odd tracks developed in Upper Accum. |
| | | | | | | |
| | | | | | | Arm 1 used for even tracks only |
| | | Main Routine | | | | |
| | | | | | | Arm 0 used for odd tracks only |
| | 0057 | LD | 69 | 8002 | 0058 | |
| | 0058 | RDS | 86 | 9000 | 0059 | Overflow indicates file is unloaded. |
| | 0059 | AL | 15 | 0003 | 0060 | |
| | 0060 | LD | 69 | 8002 | 0061 | |
| | 0061 | SDS | 85 | 9000 | 0062 | |
| | 0062 | WT | 06 | 8010 | 0063 | |
| | 0063 | BNTS | 25 | 0065 | 0064 | |
| | 0064 | BNEF | 54 | Error | End File | |
| | 0065 | LD | 69 | 8003 | 0066 | |
| | 0066 | RDS | 86 | 9000 | 0067 | |
| | 0067 | AU | 10 | 0003 | 0068 | |
| | 0068 | LD | 69 | 8003 | 0069 | |
| | 0069 | SDS | 85 | 9000 | 0070 | |
| | 0070 | WT | 06 | 8010 | 0071 | |
| | 0071 | BROV | 47 | End | 0072 | |
| | 0072 | BNTS | 25 | 0057 | 0073 | |
| | 0073 | BNEF | 54 | Error | End File | |
| | | | | | | |
| | | | | | | |
| | | Constants | | | | |
| | | | | | | |
| | 0001 | | 00 | 0000 | 0001 | Selects Arm 1 |
| | 0002 | | 00 | 0100 | 0010 | Selects Arm 0 |
| | 0003 | | 00 | 0200 | 0020 | Up dates tracks for next Seek. |
| | | | | | | High order 2 keeps track of |
| | | | | | | Seeks, and Branch Overflow |
| | | | | | | indicates End of Unload. |

FIGURE 19. UNLOAD COMPLETE DISK FILE ON TAPE