10.13 GPSS Block Diagrams

The development of a simulation model in GPSS is a block-by-block construction. A set of standard block is arranged in the form of a block diagram that represents the flow of entities through the various paths of the system. Each block represents a step in the action of the system and links, joining the blocks, represent the sequence of events that can occur.

To build a block diagram, it is essential to have a complete description of the system. The meanings of the blocks used in the system must be clearly defined. Each block must be assigned the *block time*, *i.e.*, the time which the execution of the block will take.

In total, a set of 25 specific block types have been designed, which can be used in the construction of a block diagram. Each block type can be used any number of times in a block diagram, but the total number of blocks should not exceed 2047. On the completion of the block diagram, each block is assigned a number, between 1 and 2047, called the *block number*.

10.14 Characteristics of the Blocks

10.14.1 *Block Time*

The block time also called action time is an integer giving the number of units required to execute the action represented by the block. The unit of time is decided by the programmer and is not entered in the block. But all block times throughout the block diagram are in same units of time. The block time is often a random variable, and is specified by the numbers giving *mean* and *spread* about the mean. In case the block time is constant, the spread is zero. In a simple rectangular distributions of time, the value of the block time will lie between the range, mean ± spread, with equal probability given to each integer in the range.

In many situations, the block time is not defined by the rectangular distribution. It may be some other probability density function like exponential, Poisson or normal etc. In such a case a number specifying the distribution is also inserted in the block.

10.14.2 *Alternative Actions*

At many points in a system, alternative actions are possible. More than one line may be leading into a block and more than one line may be leaving out of the block. The convention regarding alternative actions is that except the BRANCH block, not more than two lines may lead from the block and any number of lines may lead into the block. The BRANCH block type' allows upto 127 alternative paths.

10.14.3 *Selection Factor*

All block types other than the terminating block, can have two exits, referred to as *exit 1* and *exit 2*, which lead to exit blocks 1 and 2. The choice of which exit is to be followed is given by a number called the selection factor, S $(0 < S < 1)$ which is entered in the block. S is the probability of selecting exit 2 and $1 - S$ is the probability of selecting exit 1.

The choice of exit may also depend upon the availability of the blocks at exit 1 or exit 2, at the point of time when decision is to be made. The convention is that if the exit block 1 is available, then exit 1 is chosen, otherwise exit 2 is chosen. This mode of selection is indicated by setting selection factor, $S = 1$.

10.14.4 *Transactions*

In each system represented by a block diagram, some entities pass through the system. In a petrol pump system they may be vehicles, in a production system they may be work pieces and in a supermarket they may be customers etc. Those entities are referred to as Transactions. In a simulation, the transactions are created, which move through the block diagram in the same way, as the entities pass through the actual system being simulated.

10.14.5 *Items of Equipment*

In each system worth the name, there are physical equipments, which perform some operations on the transactions. Machine tools in a production shop perform machining operations on work pieces (transactions). A toll booth on a road is an equipment, when vehicles are transactions. A worker at an assembly line workstation is an item of equipment, while the components being assembled are transactions. The items of equipment may operate upon transactions individually or may handle groups. The item of equipment generally has a limited capacity and can cause congestion or waiting lines of transactions.

10.14.6 *Stores and Facilities*

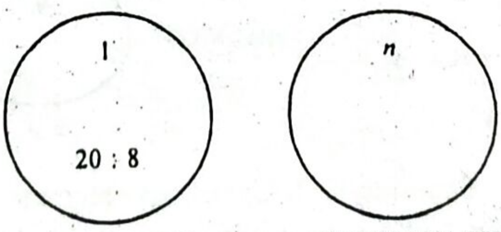
Items of equipment can be *classified* into *stores* and *facilities*, depending upon the capacity for handling the transactions. An item of equipment that can handle only one transaction at a time is called a *facility* while the items of equipment, which can handle a large number of transaction at the same time is called a *store*. There is always a defined capacity of a store. In a production line, a workstation handling one work piece at a time is a facility, while the transfer line between the workstations, is a store. In a library the book issue counter is a facility, while the books stacking area is a store. In a traffic system a toll booth is a facility, while road is a store. In a block diagram, there can be 511 facilities and 511 stores.

10.15 Block Types

The detailed description of all the block types is beyond the scope of this book. Here, only a brief description of some block types is given, to help understand the concept of GPSS block diagram.

ORIGINATE: The ORIGINATE block type is concerned with the creations of transactions and feeding them into the simulation. Generation of arrival of customers at a general store, generation of passengers arrival at a railway ticket window, generation of inter-arrival times of vehicles entering a service station, are the exemption of creating transactions. Transactions are created according to the given block time. Fig. 10.2 (*a*) shows an ORIGINATE block numbered 1 with block time of 20 ± 8 seconds. The time interval between transactions will be an integer between 12 and 28, with all integers having equal probability.

TERMINATE: The TERMINATE block type is concerned with the removal of the transaction from the simulation or the destruction of the transactions. At this block the entity represented by the transaction moves out from the system. TERMINATE block has no block time and only the number of the block is given as in Fig. 10.2 (*b*).

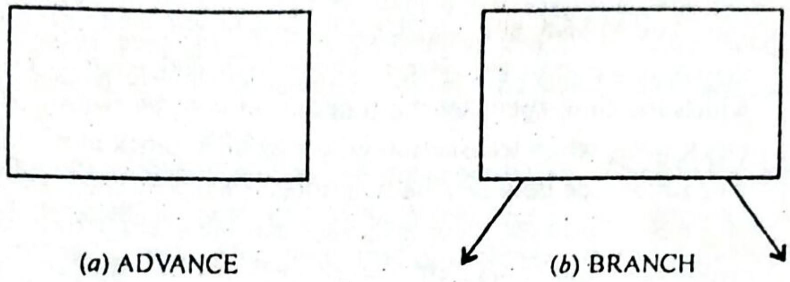


(*a*) ORIGINATE (*b*) TERMINATE

Fig. 10.2

ADVANCE: The ADVANCE block is concerned with the flow of transactions. It represents any action requiring time but does not involve any equipment, and hence cannot cause any congestion or waiting line. The ADVANCE block, provided with a zero block time, is used as a buffer at the exit of a block using some equipment. The symbol of ADVANCE block is shown in Fig. 10.3 (*a*).

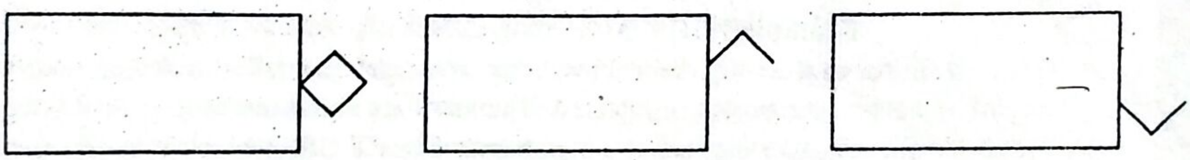
BRANCH: The BRANCH block is also concerned with the movement of transactions and 127 exits are possible from a BRANCH block. The selection of exit is governed by the selection factor, which must be either 0 or 1. When $S = 0$, the exit is selected randomly, with all exits having equal probability of selection. When $S = 1$, the lowest number exit gets first priority. If it is not available the next higher in order, is attempted and so on. If all exits are busy the transaction waits for one of the exits to become available. The symbol for the BRANCH block is shown in Fig. 10.3 (*b*).



(*a*) ADVANCE (*b*) BRANCH

Fig. 10.3

HOLD, SEIZE and RELEASE: These blocks are concerned with the use of facilities. HOLD block allows a transaction which enters the block to engage the facility for the prescribed action time. The flag on the side of the block as shown in Fig. 10.4 (*a*), carries the number of the facility associated with the block. The SEIZE block allows a transaction to engage a facility if it is available. The RELEASE block allows the transaction to disengage the facility. In between the SEIZE and RELEASE blocks any number of blocks can be inserted depending upon the actions to be performed on the transactions. Symbols for the HOLD, SEIZE and RELEASE block are given in Figs. 10.4 (*a*), (*b*) and (*c*) respectively.

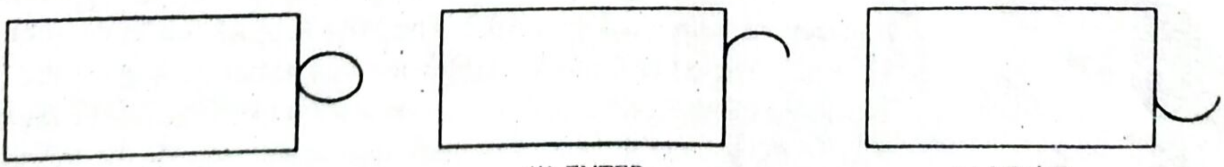


(*a*) HOLD (*b*) SEIZE (*c*) RELEASE

Fig. 10.4

STORE, ENTER and LEAVE: These three blocks are associated with stores. The symbols for these blocks are given in Fig. 10.5. The flag on the side of each block is for indicating the number of store. The STORE block allows a transaction to occupy space in the store. A transaction can remain in STORE for as much time as associated with the block.

Like SEIZE and RELEASE blocks allow the transaction to engage and disengage a facility, the ENTER and LEAVE blocks allow a transaction to occupy a space or vacate a space in a store. ENTER will allow a transaction to enter the store, if space is available. The LEAVE block will remove a transaction from the store. While SEIZE and RELEASE are associated with the same transaction, the ENTER and LEAVE may be assigned with different transaction.



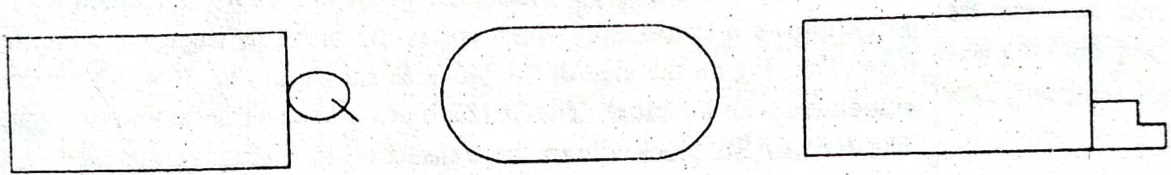
(*a*) STORE (*b*) ENTER (*c*) LEAVE

Fig. 10.5

Blocks for Gathering Statistics: In the GPSS block diagram, some blocks are provided for gathering statistics about the estimated performance of the system. The QUEUE, MARK and TABULATE are such blocks.

QUEUE: QUEUE block is placed at a position, where congestion in flow of transactions is expected. The blocks involving equipment can cause congestion that is waiting lines. QUEUE block collects statistics, like the maximum queue length, average queue length, and also samples the queue at uniform intervals to determine the probability distribution of the queue length. As many as 511 QUEUE blocks can be included in the system.

The MARK and TABULATE blocks are used to measure the time spent by a transaction in the system or a part of the system. The MARK and TABULATE blocks are inserted at the points between which the time spent by the transaction is to be measured. The program makes note of the current clock time, when transaction enters MARK block and again when it arrives the TABULATE block. The difference between the two times is the time spent by the transaction in the system and is entered in a table. From the records in the table, the average time, maximum time, the probability distribution with mean and spread etc. can be computed by the program. The symbols for the QUEUE, MARK and TABULATE blocks are shown in Fig. 10.6. The flag on the side of a block carries the number associated with the block.



(*a*) QUEUE (*b*) MARK (*c*) TABULATE

Fig. 10.6

Example 10.1: A Machining Center is producing parts at the rate of 10 ± 2 minutes. The parts move to a quality control inspector, who spends 9 ± 4 minutes on inspection of a part. At inspection 10% of the parts are rejected. The statistics about the time spent by the parts in the inspection, the queue length etc. are to be gathered. Draw a GPSS block diagram for the production system.

Solution: In the given production system, the transactions are created at the machining center. The time between the transactions is variable with equal probability of 8, 9, 10, 11 and 12 minutes. These parts are inspected by the quality control inspector, the inspection time again a variable with equal probability of 5, 6, 7, 8, 9, 10, 11, 12 and 13 minutes. Since there is only one inspector, this will be represented by a facility and HOLD block which represent the inspection counter. Because of different arrival and inspection rates congestion will occur before the inspection counter and a QUEUE will build up.

After the inspection, parts will branch into two paths as 10% of the parts will be rejected and 90% will be accepted. To gather the desired statistics, MARK, QUEUE and TABULATE blocks are to be inserted at appropriate locations.

The block diagram for the system is shown in Fig. 10.7. The ORIGINATE bock creates transactions with block time 10:2. The MARK block marks the time of a transaction as it joins the QUEUE. The SEIZE block enables the transaction to engage the facility represented by HOLD block. RELEASE block enables the transaction to disengage the facility. Than before the branching, TABULATE block is placed to mark the current time of the transaction and to compute the time spent in inspection system. Entries are made in table 1. The BRANCH block divides the transactions into two parts, 10% parts are rejected and 90% accepted. The two exits lead to the two TERMINATE blocks.

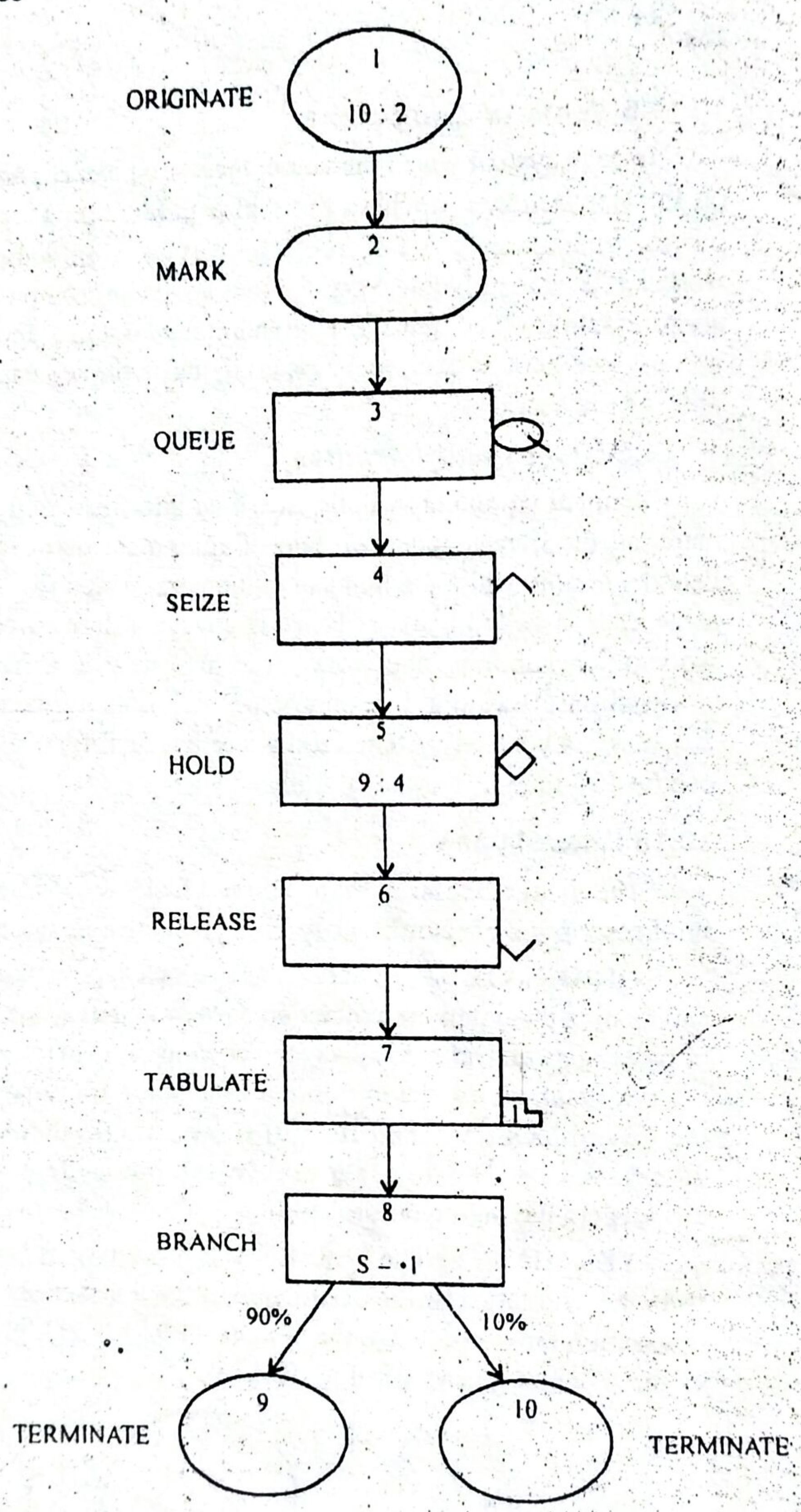


Fig. 10.7

Example 10.2: Customers enter a supermarket at an arrive rate of 50 ± 20 seconds. They have a choice of getting a basket or a trolley. While only 100 trolleys are there, there is no limit on the baskets. 30% of the shoppers go for basket and 70% for trolleys. If a trolley is not available, customer leaves the supermarket without shopping. The customers with basket spend 1200 ± 500 seconds in the market, while those with trolleys spend 1800 ± 800 seconds. There is one check out counter for customers with baskets and three check out counters for customer with trolleys. The working of the supermarket is to be simulated to measure the flow of traffic through the market. Construct a GPSS block diagram for the system.

Solution: The GPSS block diagram for the given supermarket operation is shown in Fig. 10.8. The ORIGINATE block creates transactions, each representing one customer. In other words, creating arrivals of customers at the supermarket with inter-arrival time of 50 ± 20 seconds, that is with mean of 50 seconds and spread of 20 seconds. The unit of time used in this example is one second.

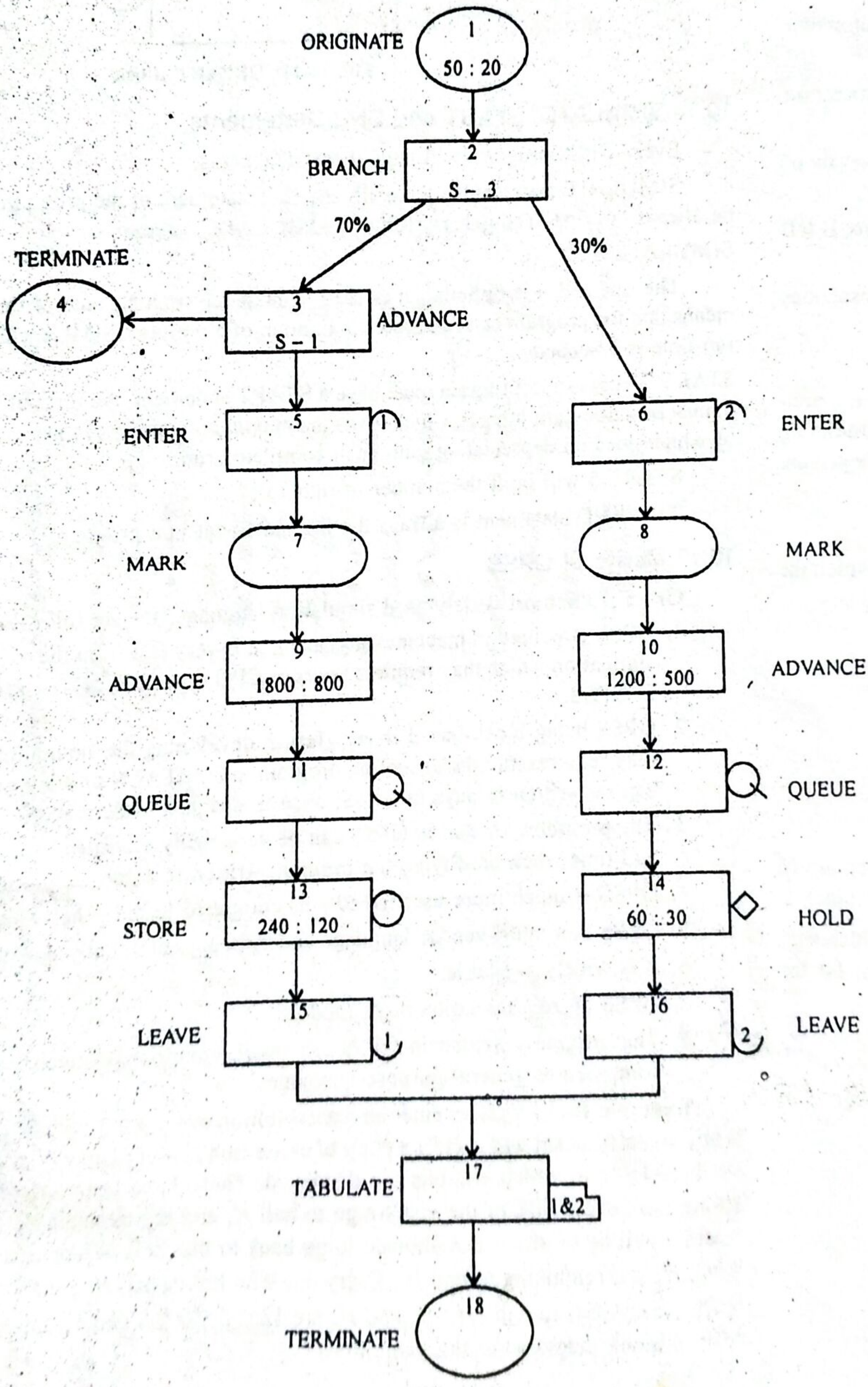


Fig. 10.8

The customers are then branched into two streams, those requiring basket to shop are 30% and remaining 70% require trolleys. The BRANCH mode with selection factor S-.3 represents the divisions of customers. The transaction representing customers requiring trolleys will move into the ENTER block numbered 5 associated with store 1, and the attempt to enter will be made through an ADVANCE block, number 3, with zero block time and selection factor $S = 1$. Thus the transaction will move into ENTER block, if there is space available, that is a trolley is available, otherwise it will move to TERMINATE block numbered 4, that is the customer leaving the supermarket without shopping, as the trolley was not available.

After getting the trolley the customer moves into the supermarket. The MARK block numbered 7 is provided here to mark the current clock time of the transaction. The shopping is represented by the ADVANCE block numbered 9 with block time, that is shopping time of 1800:800 that is mean 1800 and spread 800. After completing the shopping, customers moves to join the QUEUE (block No. 11) before the cheek out counter. Since there are three checkout counters for the customers with trolleys, these counters will be represented by a STORE block numbered 13. After leaving the check out counter, the customer, returns the trolley by entering the LEAVE block.

The customer requiring basket, picks up a basket at ENTER block numbered 6 associated with store number 2. There is no constraint on the availability of baskets. The transaction (customer) than moves through the MARK block numbered 8, to the ADVANCE block numbered 10 with block time 1200:500 for shopping. After completing the shopping customer moves to QUEUE block numbered 12 and than to the check out counter which is only one in number and is represented by a facility number 1 and HOLD block with block time 60:30 represents this facility. After checking out the customer returns the basket by moving through LEAVE block numbered 16.

After returning the trolleys and baskets, the transactions moves through TABULATE block where the current times of transaction are marked, time spent in system are entered in Tables 1and 2, for computing the desired statistics. The transactions are than removed at the TERMINATE block.

GENERATE: The generate block is also for the creation of transactions. In case of ORIGINATE block, the transactions are continuously generated according to the given block time (mean ± spread). In case of GENERATE block transactions are created as desired by the program and as long as they can enter the system. The GENERATE block action time is described as under.

GENERATE A, B, C, D, E

The fields A and B represent the mean and spread of the time as in ORIGINATE block.

The operand C is the time at which first transaction is to enter the system.

The operand D is the maximum number of transactions to be created.

The operand E is called the priority, which can be any positive integer.

The symbols for the GENERATE block are shown in Fig. 10.9 (*a*) and (*b*). When only A and B fields are needed, the symbol as in Fig. 10.9 (*a*) is used, otherwise the symbol of Fig. 10.9 (*b*) is used.

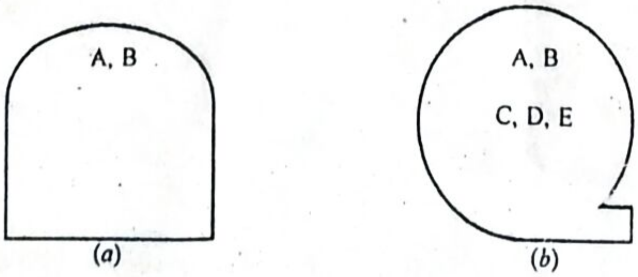


Fig. 10.9. GENERATE Blocks

Examples:

GENERATE 20, 5, 0 will create a transaction every 20 ± 5 time units, with first transaction at time $t = 0$.

GENERATE 20, 5 will create a transaction every 20 ± 5 time units, with first transaction at $t = 20$.

GENERATE 30, 6, 100 will create a transition every 30 ± 6 time units, with first transaction entering the system at $t = 100$.

GENERATE 10, 4, 20, 120 will create a transaction every 10 ± 4 time units, first transaction entering the system at t = 20. A total of 120 transactions will be created.

GENERATE 12 will create a transaction every 12 time units, that is after fixed intervals of 12 time units. The first will be created at the end of first time interval, *i.e.*, at $t = 12$.

In case all the operands are not to be used, just commas are put and no space or blank is left in between the commas.

GENERATE 45, 250 will create a transaction every 45 time units and a total of 250 transactions will be created.

GENERATE,,, 10 will create ten transactions immediately.

TRANSFER: The TRANSFER block is used when the transactions are moved in a non-sequential manner. This is like the GO TO statement used in FORTRAN and other programming languages. Though there are several forms of TRANSFER block, two most commonly employed are described below:

(*a*) **The unconditional TRANSFER block**

The form of this block is TRANSFER, (Label). The label is the label of the block to which the transaction is to be transferred.

TRANSFER, NEXT

TRANSFER, DOWN

TRANSFER, UP

Are some of the examples.

(*b*) **The conditional TRANSFER block**

The form of this block is TRANSFER, xyz, block 1, block 2

xyz is a maximum three digit decimal number like .005, .25, .5, which gives the fraction of time, the transaction is to be transferred to block 2. The remaining time transfer will be to block 1.

TRANSFER .25, NEXT 1, NEXT 2 will transfer the transaction 25% of the time to block with label NEXT 2 and remaining 75% of the time to block with label NEXT 1. The symbol for the TRANSFER block is shown in Fig. 10.10.

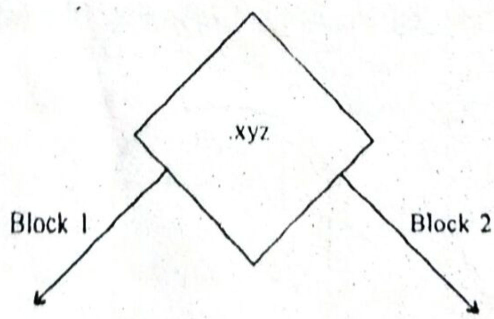


Fig. 10.10: TRANSFER Block

DEPART: The DEPART block is used in connection with a QUEUE block. When a transaction is in a QUEUE block, it must at some time leave the block. The DEPART block enable the transfer to leave the QUEUE block. The symbol is shown in Fig. 10.11.

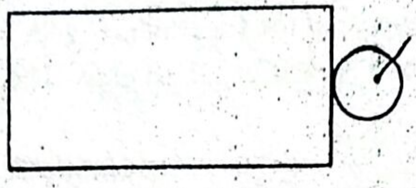


Fig. 10.11: DEPART Block

10.16 SIMULATE, START and END Statements

Every GPSS program will have these statements.

SIMULATE statement is generally the first statement of the program, but it is not necessary to be always the first. The general form of SIMULATE statement is

SIMULATE *n*

The operand *n* is optional. It is used to limit the running time of the program. SIMULATE 5 means that the program can run upto a maximum of 5 minutes. SIMULATE 5s means that maximum run time is 5 seconds.

START: Every GPSS program must have a START statement, the form of which is START *n*, where *n* must be a non-zero integer, which gives the maximum number of runs. It is like a counter, the value of which goes on depreciating with each completed run.

START 5 will limit the number of runs to 5.

THE END statement is always the last statement in a program.

10.17 Merits of GPSS

GPSS is the most widely used simulation language, for the following reasons:

1. GPSS is written in machine language and is very fast from the execution point of view. The simulation which may requires hours of CPU time in FORTRAN, can be executed in minute in GPSS.
2. GPSS being block-based is very fast in developing the simulation models. A model which may requires months of writing program in FORTRAN or other general-purpose languages, can be written in days in GPSS, once it is rightly understood.
3. The programs written in GPSS can be very easily modified, while it requires a lot of effort and time when modifying the program written in a general-purpose languages.
4. GPSS is much more user friendly as compared to the general-purpose languages.
5. GPSS is a multi-vendor language and is being continually updated.
6. It is widely available.
7. GPSS allows animation in its models.
8. The programs written in GPSS are small, comprise of much less computer code lines as compared to general-purpose languages.

Example 10.3 : Visitors enter an Art exhibition every 4 ± 3 minutes. At the entry every one has to buy an entry ticket and collect a copy of exhibition layout plan, which taken 2 ± 1 minutes. There are three halls in which exhibits are displayed. These have been named as H_1, H_2 and H_3. After getting the ticket, 70% of the visitors go to hall H_1 and the remaining 30% to hall H_2. If a visitors misses a hall he or she is not allowed to go back to that hall. When visitors leave hall H_1, 80% go to hall H_2 and remaining to hall H_3. Every one who leaves hall H_2 goes to hall H_3. The time required to see the exhibits in halls H_1, H_2 and H_3 are 12 ± 5, 20 ± 8 and 15 ± 5 minutes respectively. Draw a GPSS block diagram for the exhibition.