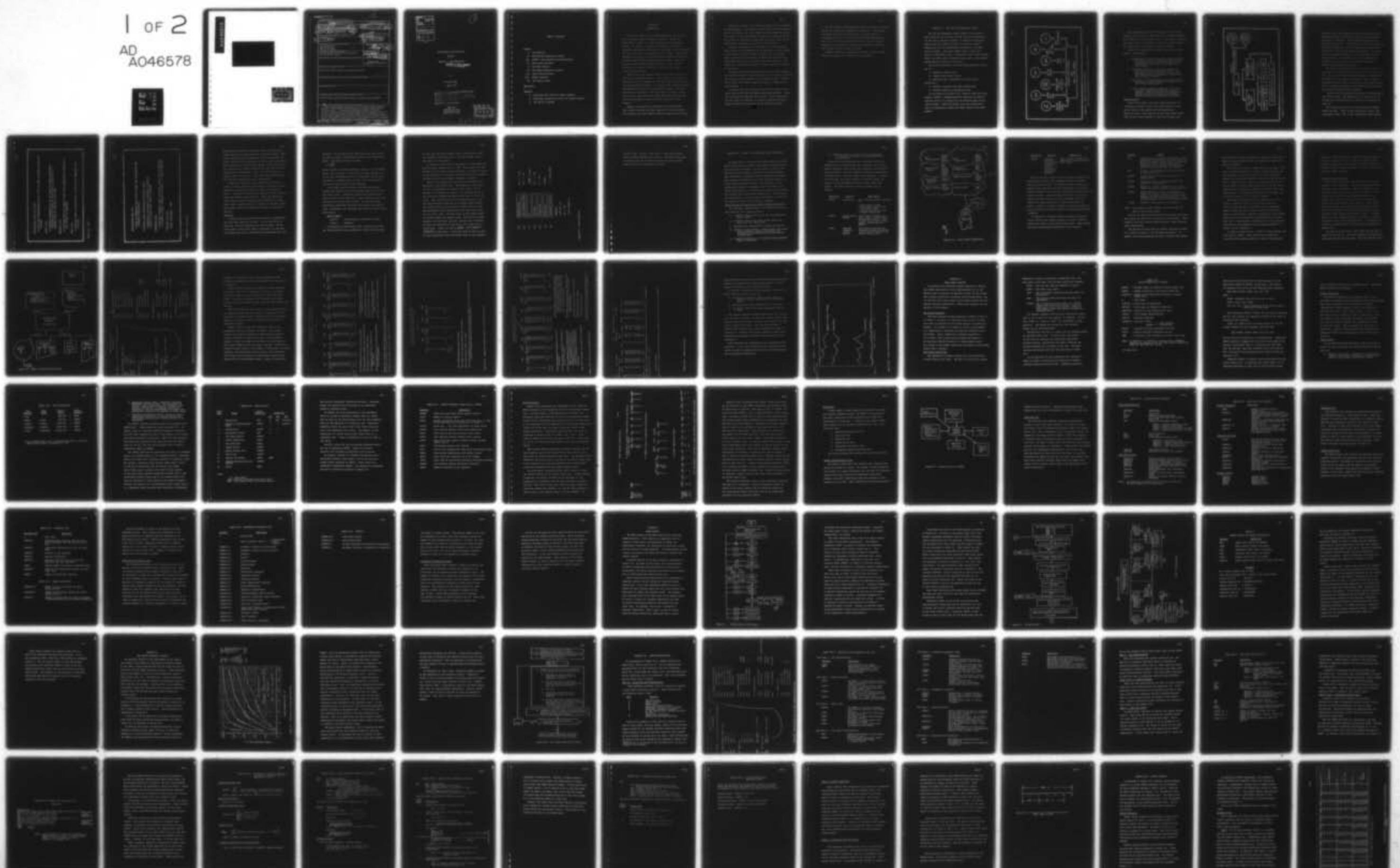
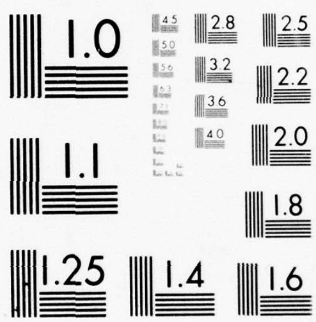
AD-A046 578 DECISION SYSTEMS DAYTON OH F/G 15/5
THE INVENTORY SYSTEM SIMULATOR, (INSSIM). VOLUME I. MODEL DESCR--ETC(U)
AUG 77 W S DEMMY F49620-77-C-0063
UNCLASSIFIED RM-77-02 AFOSR-TR-77-1244-VOL-1 NL

1 OF 2
AD
A046578





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFOSR-TR-77-1244-Vol-1	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) THE INVENTORY SYSTEM SIMULATOR, (INSSIM) VOLUME I. MODEL DESCRIPTION.		5. TYPE OF REPORT & PERIOD COVERED INTERIM
		6. PERFORMING ORG. REPORT NUMBER RM-77-02
7. AUTHOR(s) W. Steven Demmy		8. CONTRACT OR GRANT NUMBER(s) F49620-77-C-0063
9. PERFORMING ORGANIZATION NAME AND ADDRESS Decision Systems, 3575 Charlene Drive, Dayton, OH 45432		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102F 2304 A5
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Office of Scientific Research/NM, Bolling AFB, DC 20332		12. REPORT DATE August 1977
		13. NUMBER OF PAGES 119p.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

A046053

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

single-location, multi-item, EOQ, inventory, simulation model, FORTRAN

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the features of the Inventory System Simulator (INSSIM), a FORTRAN-based simulation model of a general single-location inventory system. The model was specifically designed to evaluate alternate forecasting and inventory mamagement policies proposed for use in the EOQ Buy Computation System (D062). Volume I of this report describes the philosophy, organization, and input/output features of INSSIM. Volume II documents the FORTRAN statement listings of each INSSIM program, and provides narratives describing the functions of each routine.

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED 393 082

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(3)

THE INVENTORY SYSTEM SIMULATOR,

(INSSIM)

VOLUME I: MODEL DESCRIPTION

by

W. Steven Demmy

August 1977

RM-77-02
Decision Systems
3575 Charlene Drive
Dayton, Ohio 45432

(See 1473)

Table of Contents

CHAPTER I

INTRODUCTION

This is the first of two volumes describing the features of the Inventory System Simulator (INSSIM). INSSIM is a FORTRAN-based model of a general single-echelon inventory system. The model was specifically designed to evaluate alternate forecasting and inventory management policies proposed for use in the EOQ Buy Computation System (D062). This volume describes the philosophy, organization, input/output features, and Job Control streams associated with INSSIM. Volume II documents the FORTRAN statement listings of each INSSIM routine, and provides narratives describing the function of each of these routines.

Chapter II of this report describes the major features of the EOQ Buy Computation System. This system is the primary data system for the control of stock fund items managed by the Air Force Logistics Command. In general, D062 manages discrete components that are part of the higher assembly and are not economically reparable at the depot level of supply. Major inputs to the D062 system and major management concepts used for the management of these items are described in this chapter.

Chapter III provides an overview of the INSSIM model. This chapter outlines the major features of the simulation, and discusses the major output products produced by the model.

Chapter IV reviews major concepts employed in the Inventory System Simulator. The chapter defines timing mechanism employed by the simulation, the major events that are simulated within the model, and the subroutines and data files that are required to perform the associated bookkeeping functions. Most of the major INSSIM variables are defined in this chapter.

Chapter V discusses the operations of the MAIN program. This program is the workhorse of the simulation. This routine is responsible for the control of input data, initialization of status and performance variables, the scheduling of simulation events, and the control of output products. Flow charts and a detailed discussion of this routine is presented in Chapter V.

An important feature of the INSSIM model is the capability to simulate inventory demand patterns so as to exactly duplicate the demand patterns recorded in D062 history files. Chapter VI describes the demand generation process by which this is accomplished.

To users, the most important feature of any model is its input requirements and the associated output products. These are the topics of Chapters VII and VIII. Input requirements for the model are discussed in detail in Chapter VII while output products are discussed in the following chapter.

The Job Control Language programs which are used to initiate INSSIM job streams are discussed in Chapter IX. This chapter discusses three sets of job streams. One job stream is particularly useful in the debugging of INSSIM enhancements, while the other two job streams are useful for the initial testing of a new data set, and for the conduct of production runs involving large numbers of items.

Finally, details of input/output records and job control language listings are provided in the appendices.

CHAPTER II. The EOQ Buy Computation System

The EOQ Buy Computation System (D062) is the primary data system for the control of stock fund items managed by the Air Force Logistics Command. In general, this system manages discrete components that are part of a higher assembly and are not economically reparable at the depot level of supply. The terms "repair parts," "bits and pieces," "consumable items," "stock fund items," and "expense items" are often used to describe these parts. AFLC manages approximately 500,000 of these items.

The primary functions of the EOQ Buy Computation System are to:

a. Accumulate demand data.

b. Compute depot stock levels.

c. Determine buy, termination, and long supply quantities.

d. Provide a baseline for funds projections.

e. Provide reports and management data.

As illustrated in Figure II-1, inputs to D062 come from several sources. Headquarters AFLC specifies the implied shortage factor (λ) required by the Presutti-Trepp safety stock formula. Stock list data, asset and usage counts, and file maintenance actions are other inputs to the D062 system.

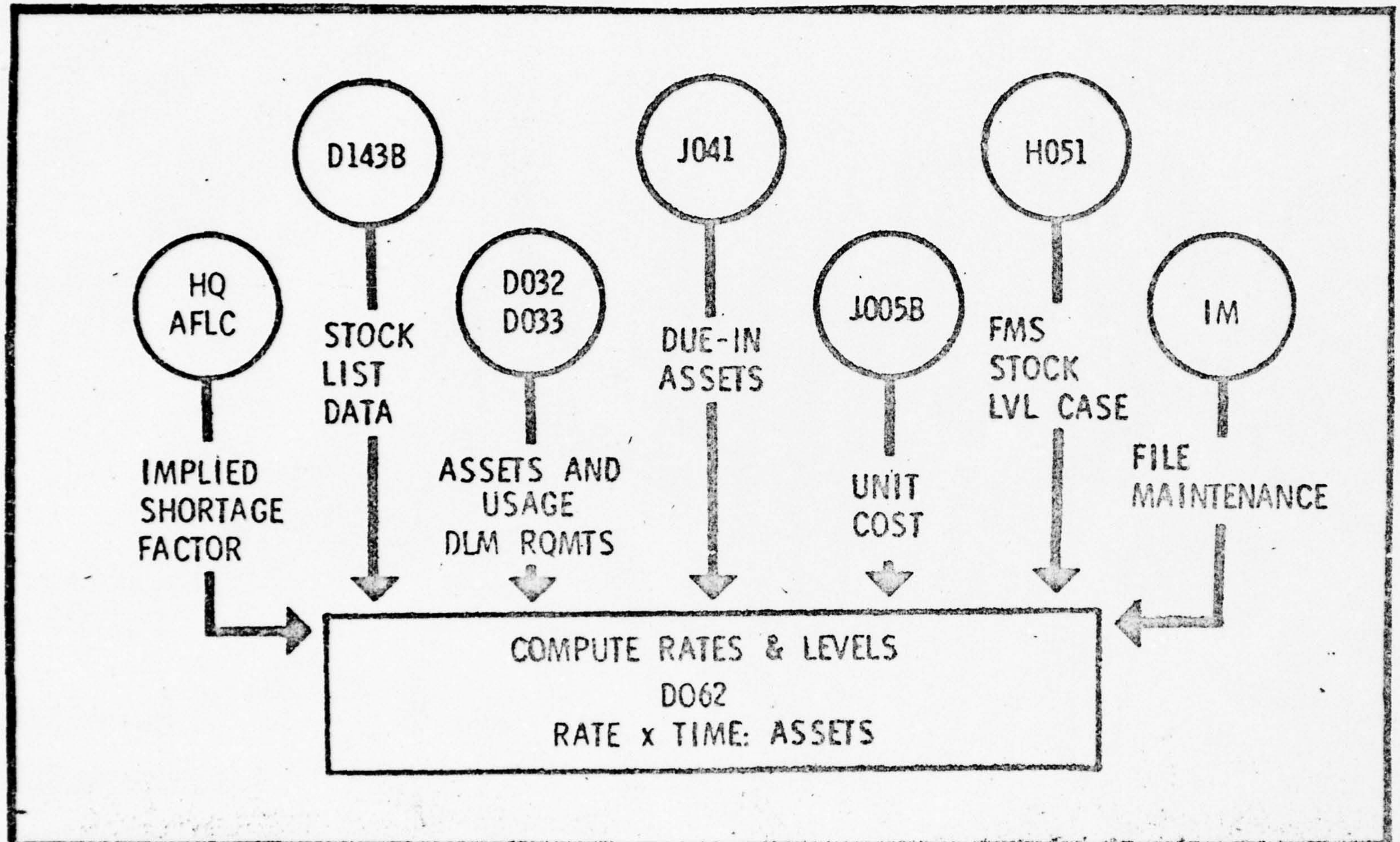


Figure II-1. Inputs to the Requirements Process

Note: The HOSI and DLM RQHTS inputs are not complemented at present.

This information is used to compute several critical numbers, called "levels," that are used to initiate management notices on a by exception basis. For example, currently available assets are compared to the levels to determine if buy, termination, or disposal actions are needed.

Figure II-2 illustrates the major outputs of the D062 System. These include:

a. Requirements Notices -- The item manager receives advance buy, buy, or termination notices when assets breach the respective levels. Item interrogations can also be requested by the item manager.

b. Management Reports -- These data products are produced for the Air Logistics Centers and Headquarters AFLC summarizing the impact of the computation by categorizing items according to actions required.

c. CSIS Data -- Data required to perform the Central Secondary Item Stratifaction is passed to the D075 system every quarter.

d. Data to Other Interfacing Systems -- D062 also provides information to the D067 and D032 systems. D067 is furnished data required to process excesses, and D032 is fed control levels required for distribution of assets.

Control Levels

In the D062 system, the terms "asset position" and "inventory position" refer to the total assets on hand and on order in the system, less any backorders. Hence, an item's asset position is the total stock available to meet future demands if there are no more buys.

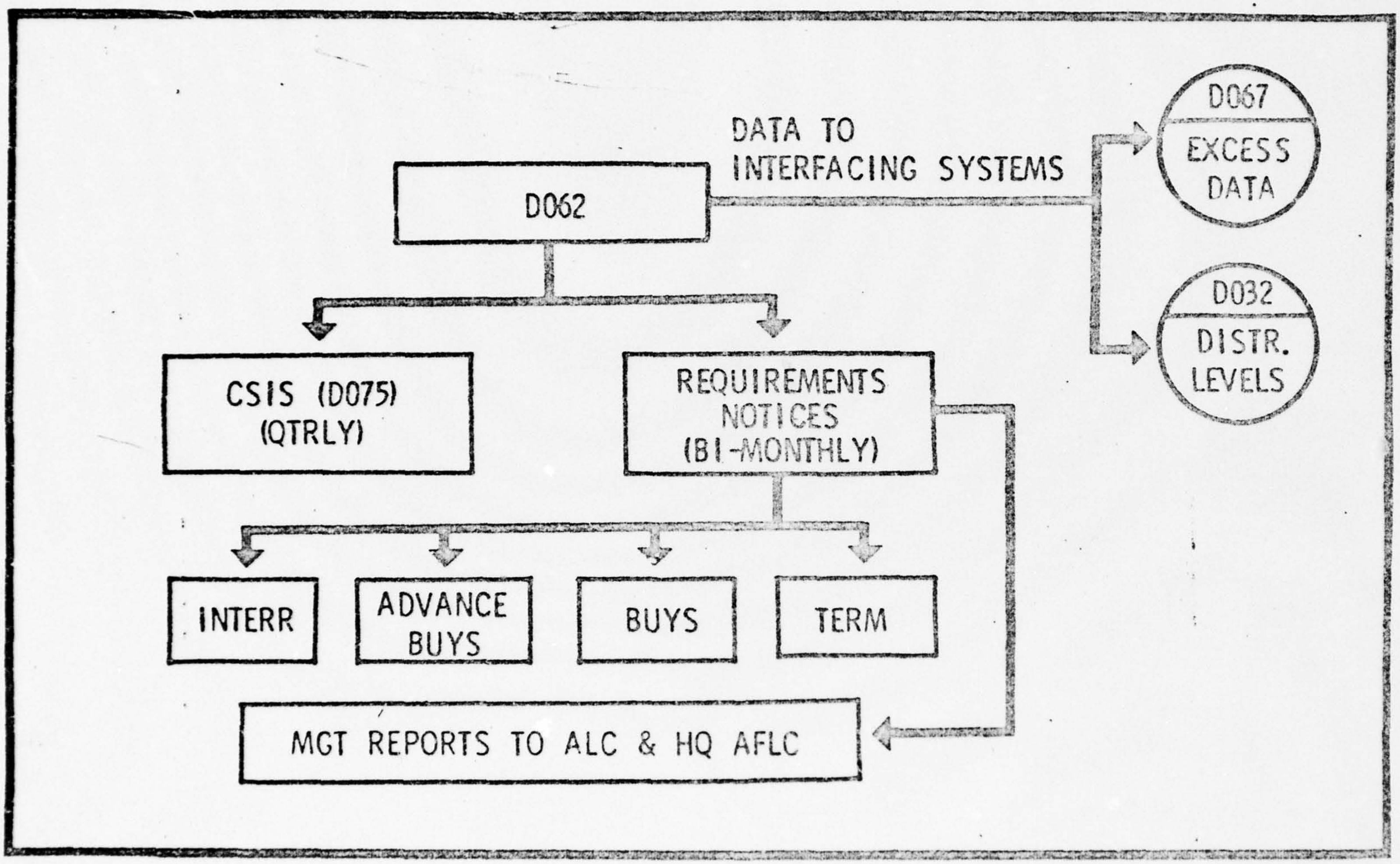


Figure II-2. Outputs from the EOQ Buy Computation System

As noted above, D062 computes various "levels" which are compared against an item's asset position to determine if any action is required. These levels are summarized in Figures II-3 and II-4. Procurement lead time is the average time that elapses from the first printout of a buy notice to the date of the first significant delivery. The safety level is variable depending on item characteristics and HQ AFLC inputs. Its purpose is to insure continuous operation in the event of unpredicted fluctuation in demand and/or extended lead times.

The reorder level (ROL) equals the number of demands expected in the procurement lead time plus the safety level. It is the point to which available assets are compared to determine if a buy action is required. When an item's inventory position equals or falls below the reorder level, the buy quantity consists of any deficiency to the reorder level plus an economic order quantity.

The data level represents four months of demands beyond the reorder level. The function of the data level is to provide early identification of items in a potential buy position.

The termination level is one years worth of demands beyond the ROL. Note: If EOQ $>$ one year's supply, the termination level = ROL + EOQ. Termination level notices

PROCUREMENT LEAD TIME (PLT)

- TIME FROM PREPARATION OF PR / MIPR TO DATE OF FIRST SIGNIFICANT DELIVERY

SAFETY LEVEL

- MATERIEL AUTHORIZED TO BE ON HAND TO INSURE CONTINUOUS OPERATION IN THE EVENT OF UNPREDICTED FLUCTUATION IN DEMANDS AND / OR EXTENDED LEAD TIME

REORDER LEVEL

- COMPUTED POINT TO WHICH ASSETS ARE COMPARED TO DETERMINE IF BUY ACTION IS REQUIRED

DATA LEVEL

- PROVIDES EARLY IDENTIFICATION OF ITEMS IN A POTENTIAL BUY POSITION

Figure II-3. Levels

TERMINATION LEVEL (TL)

- USED TO DETERMINE THOSE ITEMS WHICH ARE CANDIDATES FOR CONTRACT TERMINATION

APPROVED FORCE ACQUISITION OBJECTIVE (AFAO)

- DETERMINES BEGINNING OF ECONOMIC RETENTION
- USED AS STOCK FUND CREDIT INDICATOR

RETENTION LEVEL

- INDICATES MAXIMUM LEVEL OF STOCKS WHICH MAY BE RETAINED IN SUPPLY SYSTEM. (CONSISTS OF AFAO 1 - 5 YEAR'S REQMTS.)

PROJECTED NET BUY

- ROL DEFICIENCY + EOQ

Figure II -4. More Levels

are output for items with on order assets when the total asset position is greater than the termination level. This notice warns the item manager that it may be desirable to cancel delivery of at least part of the on order quantities. The approved force acquisition objective (AFAO) consists of two years worth of forecast demands plus the lead time and safety level requirements. (Note: If EOQ > 2 years of supply, AFAO = EOQ + ROL.) Items with assets greater than the AFAO are considered to be in long supply.

Finally, the retention level indicates the maximum amount of stock which may be retained in the supply system. Generally, quantities beyond this value are considered excess. Retention levels can vary from one to five years of projected demands beyond the AFAO. Retention levels for items supporting newer weapon systems generally use five years, while systems ready to be phased out may have retention levels equal to the AFAO plus one year of projected demands.

Formulas

A basic problem in any inventory system is determining how much stock should be on hand. If too much stock is procured, excessive carrying costs are incurred. On the other hand, if too little stock is procured, an item must be procured more often, and excessive procurement costs are incurred. In the D062 system, the Wilson lot size formula is used to obtain a good balance between these conflicting costs. This formula takes the form:

$$Q = \sqrt{\frac{2AD}{H}}$$

where Q equals the order quantity in units, D is the annual demand rate in units, A is the cost per order placed, and H is the cost of holding one unit in inventory for one year. At present, a holding cost H of 20% of the item unit cost per year is used in the lot size formula.

In Air Force supply systems, different procurement methods are employed for small purchases than are used for high dollar buys. Simplified procurement techniques are used for small dollar purchases. These methods may be used for purchases of less than $10,000. On the other hand, advertized, negotiated contracts are used for high dollar buys. At present, the following order costs are used in the D062 system:

COST TO ORDER

$269.87	for purchases of less than $10,000
$460.27	otherwise

In applying the EOQ formula above, impractical values for order quantities are produced for items with very high or with very low annual demand rates. Consequently, EOQ's are bounded to be no more than a 36 month supply, and no less than a 6 month supply.

At present, safety levels are computed in D062 using the Presutti-Trepp formula with $Z = \sqrt{R}$. This quantity is then bounded to be no more than (a) the number of demands expected in the procurement lead time or (b) three times the standard deviation of lead time demand, whichever is smaller.

Figure II-5 illustrates the levels computations for a particular, fast moving item. This item has a unit price of $10, a procurement lead time of nine months, and an average of 100 demands per month. The expected demand in the nine month procurement lead time is thus 9 X 100 = 900 units. The safety level for this particular item is 113. This was determined based on the PT-safety level formula. The reorder level is the sum of the safety level and the expected demand in the procurement lead time, which gives us 113 + 900 = 1013. The data level is four months worth of demands beyond the reorder level. This item has annual demands D of 1200 units per year, and the cost of holding one unit in inventory for one year is 20% X unit price = $2 per year. Hence, the EOQ is $\sqrt{2AD/H} = \sqrt{2 \times (\$269) \times (1200)/\$2}$, or 568 units. A cost per order of $269 is used in this calculation since the dollar value of the purchase

9413	RETENTION LEVEL	AFAO + 5 YRS
3413	AFAO (TERMINATION LEVEL)	ROL + 2 YR
568	EOQ	$\sqrt{\frac{2\ AD}{H}}$ (5.6 MOS)
1413	DATA LEVEL	ROL + 4 MOS
1013	REORDER LEVEL	SL + PLT
113	SAFETY LEVEL	VARIABLE
900	PROCUREMENT LEAD TIME	9 X MONTHLY DEMAND

<u>ITEM DATA</u>

UNIT PRICE $10

PLT 9 MOS

AV MOS DEMANDS 100

Figure II-5. Levels Computation

is \$10 X 568 = \$5,680. Since this is less than \$10,000, small purchase methods may be used. The AFAO is two years of stock beyond the ROL and the retention level is five years worth of demands beyond the AFAO.

CHAPTER III. INSSIM - The Inventory System Simulator

As noted above, a detailed simulation model of the D062 system was needed to evaluate the relative effectiveness of the alternate PT-formulas. In our study, we utilized the Inventory System Simulator (INSSIM) as a starting point. This simulator was developed by the Directorate of Management Sciences (AFLC/XRS) to evaluate inventory policies in single location supply systems. For our study, it was necessary to enhance the original model to provide a detailed description of the current D062 system and to provide for improved input and output capabilities. In the following discussions, we will use the term "INSSIM" to refer to the enhanced version of the original simulator.

Major Features of the Inventory System Simulator (INSSIM)

The Inventory System Simulator (as enhanced) possesses the following major characteristics:

a. A detailed description of the EOQ Buy Computation System (D062).

b. A demand process based upon actual demand histories for Air Force items.

c. Comprehensive measurement of simulation results.

d. Entensive input options -- which permit the evaluation of several proposed forecasting and inventory control policies by simple changes to input data.

e. A modular structure -- to simplify future enhancements to the model.

f. Debugging aids -- to assist in the programming of proposed rules that are not already coded in the model.

Basically, INSSIM consists of a collection of Fortran subroutines and a MAIN program that controls input requirements, schedules events within the simulation, and initiates output products. The major INSSIM routines are illustrated in Figure III-1, grouped by their major function. As shown in the figure, the major features of the D062 system are simulated using subroutines FOR576, FORUPD, LEVEL, and STATUS. The major functions of these routines are as follows:

Subroutine	Function	Description
FOR576	Forecasting	This routine provides estimates of ° gross demand rates ° serviceable return rates ° net demand rates ° average requisition size ° demand variability
FORUPD	Record demand history	This routine maintains an eight quarter moving history of simulated demand. This history is used in the forecasting calculations in FOR576.
LEVEL	Computes inventory control levels	This routine computes the inventory control levels discussed above (safety levels, reorder levels, etc.)

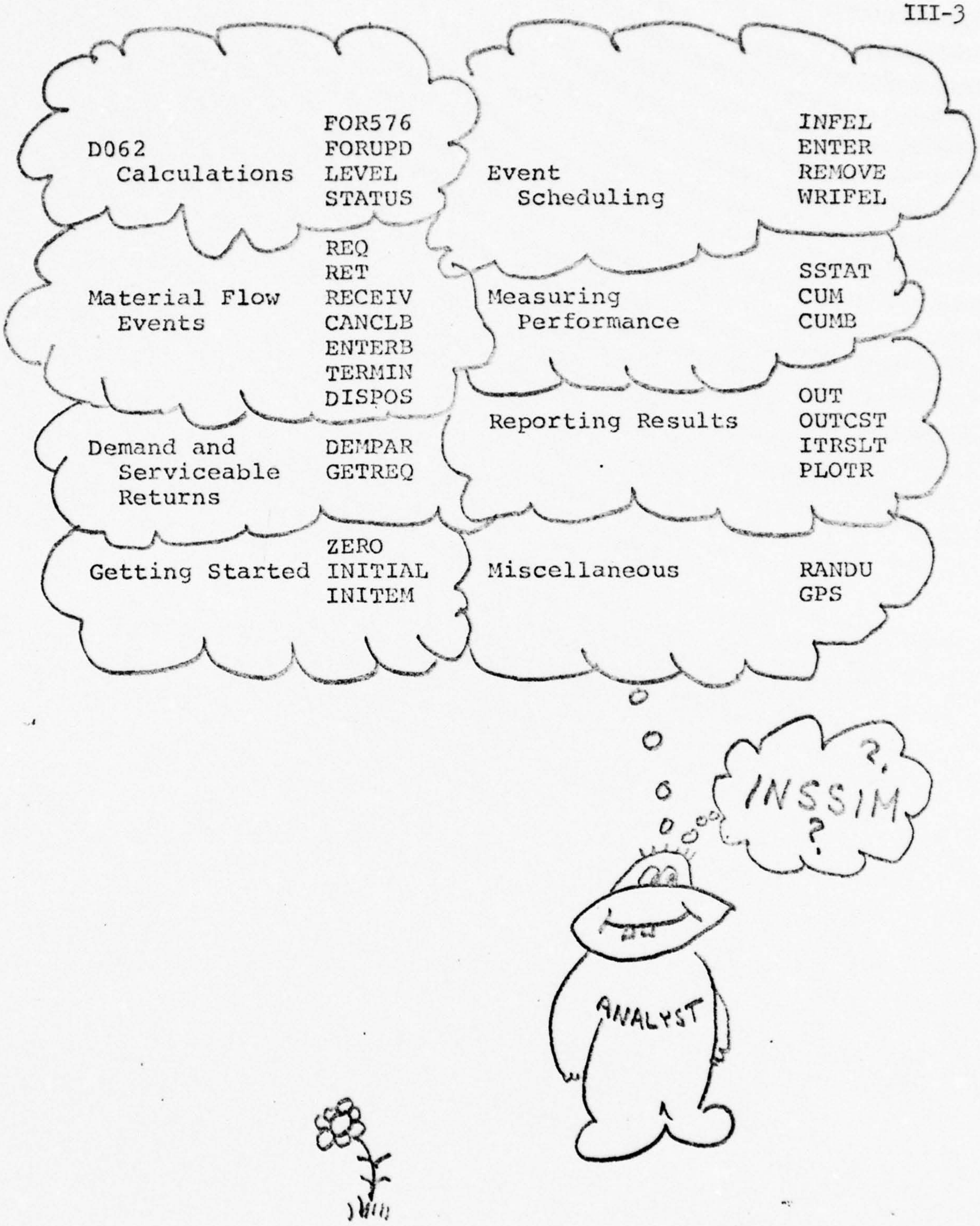


Figure III-1. Major INSSIM Subroutines

Subroutine	Function	Description
STATUS	Compares available assets to the respective control levels, and initiates appropriate actions	This routine simulates the management action portion of D062 system.

Each of the above routines contain logic describing the computational formulas and management policies currently used in the D062 system. In addition, these routines also possess logic describing several forecasting and inventory control procedures that have been suggested as alternatives to current methods. These alternate procedures may be simulated by changing one of the eight parameter cards that specify the characteristics of a given simulation run. Hence, a number of alternate inventory management proposals may be evaluated by simply changing the input specifications to INSSIM.

In addition to the D062-related subroutines described above, several other INSSIM routines are used to describe significant events in the flows of EOQ items. These events and their corresponding subroutines are as follows:

Routine	Event
REQ	A requisition is received at the depot. This represents a demand for a specific number of units of a particular EOQ item. If possible, the requisition is filled immediately; otherwise, the requisition is backordered until a replenishment order is received.
RET	A number of serviceable units are returned to the supply system.
RECEIV	A replenishment order is received by the supply system.
CANCLB	A customer with an outstanding backorder cancels the requisition.
ENTERB	Record the current requisition as a back-order, and insert it into the backorder file.
TERMIN	Action is taken to stop a replenishment order that has not yet been received. This event is initiated whenever an item's asset position exceeds the termination level, and a replenishment order is still being processed.
DISPOS	Assets in long supply are disposed of.

(Note: The routines CANCLB, TERMIN, and DISPOS were not used in the current study.)

Each of the above routines perform bookkeeping operations that update the status of on hand and on order stocks. These routines also post activity statistics that record inventory system performance.

The methods for describing the demand generation process is a critical element in any inventory simulation. In INSSIM, the demand generation process is derived from demand

and serviceable return histories for actual Air Force items. Specific computational details are handled by subroutines DEMPAR and GETREQ.

As we shall see below, a major input to our study is the actual demand histories for a sample of EOQ items from the EOQ data bank. This input defines the actual quarterly demands and serviceable returns for each item for the period FY 71 through FY 76, a total of 20 quarters worth of data. In our study, the first eight quarters of data were used to initialize the history files needed in the D062 usage rate calculations. The remaining 12 quarters of data were used to simulate demands in the inventory system.

The demand generation process is constructed so that within a particular quarter, the number of units of demand and the number of serviceable returns simulated exactly equals the actual values from the EOQ data bank. Within a given quarter, specific requisitions are generated that have the same statistical characteristics as current USAF items. Specifically, requisition sizes are generated according to the probability distributions presented in Figure II-2 of reference 2.

As shown in Figure III-1, a number of other routines are also used in INSSIM. These routines are required to initialize the simulation model, to collect and summarize performance statistics, and to assist in event scheduling and other bookkeeping tasks. Rather than discuss these routines individually, the following sections describe the ultimate results of these routines upon INSSIM input requirements and output products.

INSSIM Inputs and Outputs

Figure III-2 illustrates the major data flows of the Inventory System Simulator. Run specifications are input from File 05 in card format. This input specifies the inventory policy options that are to be simulated in the current run, as well as significant parameters (e.g., holding cost, ordering costs, and bounds on EOQ's and safety stocks) required by these policies. Other input cards specify the output options to be employed, and the size of the simulation run (e.g., number of items to be simulated, time duration for the study, etc.). A printback of the run specifications for a 100 item, 12 quarter simulation run is shown in Figure III-3. (A detailed discussion of variables shown in this figure is presented in Chapter VII.)

As shown in Figure III-2, item demand and cost data is input through File 07. This file provides item information extracted from the EOQ Data Bank. This file contains data

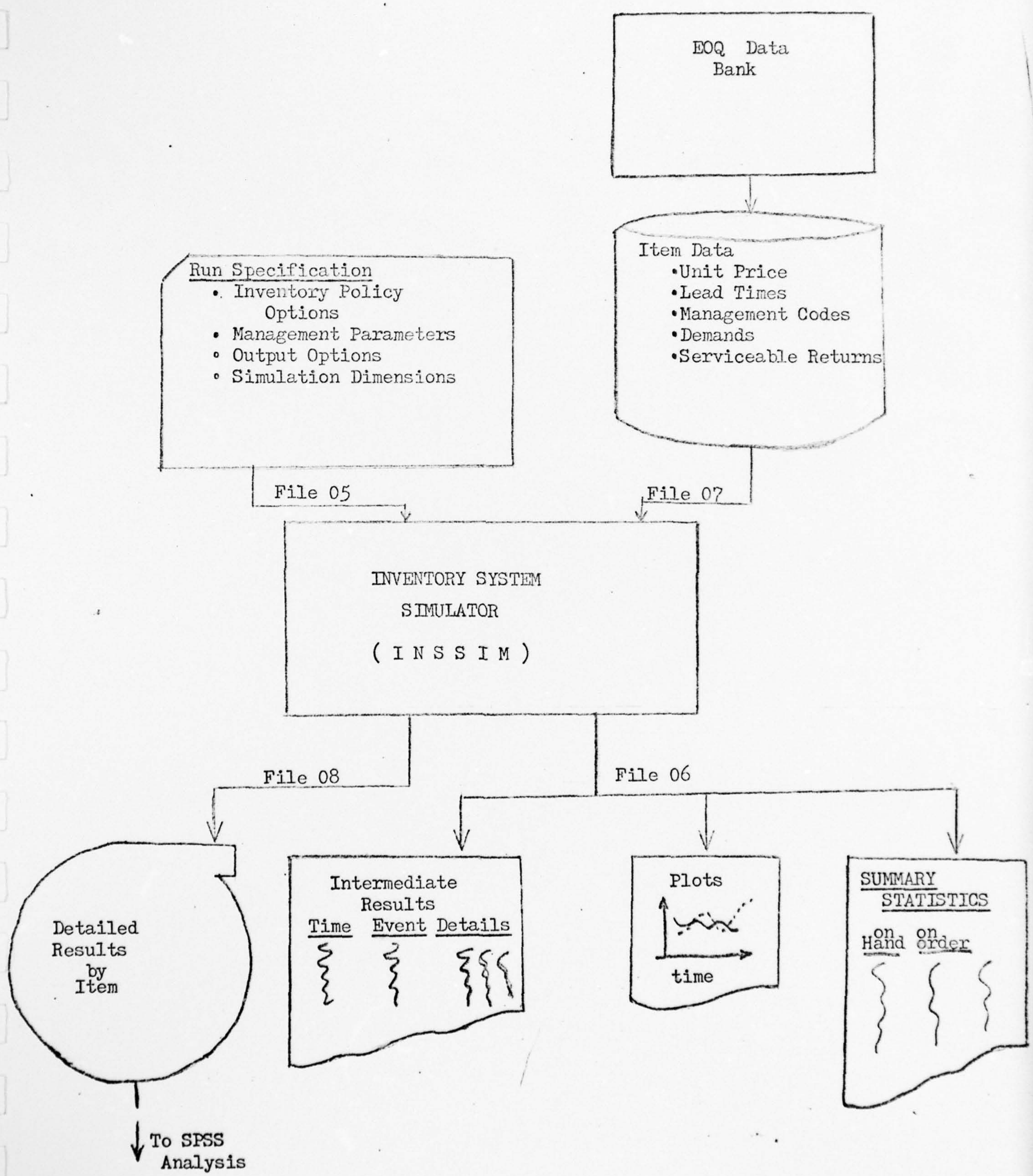


Figure III-2. INSSIM Input/Output Relationships.

```
                 INVENTORY SYSTEM SIMULATOR
                       RUN PARAMETERS

(C1)  RUN-ID.      3  TITLE. 100 ITEM SIMULATION WITH ZERO SS

(C2)  OUTPUT CONTROLS...(NOTE. 1=YES)
ITWRT          IT.WRITE=  0
IOUT           SUMMARY =  1
IGRAPH         GRAPHS  =  1
IPUNCH         PUNCH   =  1

(C3)  DEBUG FLAGS
               IDBUG =  0
               IEBUG =  1
               IFBUG =  0
               IGBUG =  0
               IHBUG =  0

ITRACE         START TRACE AT      0  FOR ITEM 1
ISTRAC         STOP TRACE AT      0

(C4)  ITEM INPUT FILES
INLU           FILE =  7
INTYPE         TYPE =  2  (1 = BCD; 2 = BINARY)
NDEM           QTRS = 24

(C5)  MANAGEMENT METHODS TO BE USED
ICDFOR         FORECAST FORMULA  =  1
ICDSIG         STD DEV  FORMULA  =  1
ICDEOQ         EOQ      FORMULA  =  2
ICDSL          SAFETY STK   CODE =  3
ICDSLL         SAFETY LIMIT CODE =  1
ICDBG          BUDGET GUIDE CODE =  1
ICDSR          RETURNS CODE      =  1

(C6)  MANAGEMENT PARAMETERS
EOQMIN         MIN EOQ(MNTHS)             6.0000
EOQMAX         MAX EOQ(MNTHS)            36.0000
SLMIN          MIN SAFETY LV(MNTHS)       0.
SLMAX          MAX SAFETY LV(MNTHS)       0.

(C7)  SYSTEM PARAMETERS
COSHLD         HOLDING COST/$-INV         0.2000
CSHOPT         SHORTAGE COST             10.0000   31.6000   100.0000
COSORD(1)      SMALL ORDER COST         269.8700
COSORD(2)      LARGE ORDER COST         460.2700
CSTBRK         COST BREAK-POINT       10000.0000

(C8)  SIMULATION SIZE
NRUN           NUMBER OF RUNS             1
INQTR          NUMBER OF QUARTERS        12
NREPL          NO. OF REPLICATIONS      100
NITEM          NO. OF ITEMS/REPL          1
```

Figure III-3. Specifications for a 100 Item 12 Quarter Simulation Run.

defining the unit price, lead times, management codes, demands and serviceable returns associated with each item to be simulated.

Output products produced by INSSIM are routed to files 06 and 08. File 08 is a magnetic tape file. It contains details by quarter on the performance of each item simulated. This file is designed for subsequent statistical analysis of the simulation results using the SPSS Statistical Package.

File 06 is routed to the printer. If all output options are requested, this file will contain a print-back of all input data, an event-by-event description of the simulation process, and detailed plots and statistical summaries of simulation results.

Figures III-4 through III-8 illustrate some of the performance summaries produced by INSSIM. For example, Figure III-4 provides statistics describing the number of units on hand and on order at the end of each simulated quarter, as well as counts of the number of units received from vendor shipments and serviceable returns. Figure III-5 presents similar counts of the number of expediting, rationing, disposal, and termination actions taken in the simulation, while Figure III-6 presents data describing average inventories and fill and backorder rates. Finally, Figure III-7

P E [illegible] O [illegible] A N [illegible] A T [illegible] I [illegible] S

... UNITS ...

	(1)	(2)	(3)	(4)	(5)	(6)**	(7)	(8)	(9)	(10)**
PERIOD	INVENTORY ON HAND	INVENTORY ON ORDER	RECEIPTS	RETURNS	TOTAL SHIPMENTS	PRIORITY 1 TOTAL SHIPMENTS	ORDERS PLACED	TOTAL REQUISITIONS	REQS CANCELED	PRIORITY 1 REQUISITIONS
BOP	103865	0								
1	93126	28931	0	3121	13860	13860	28931	14959	0	14959
2	88418	33183	8455	3483	16646	16646	12627	16213	0	16213
3	90113	29159	17413	2533	18251	18251	13469	19266	0	19266
4	85072	28422	13150	2472	20663	20663	12413	21627	0	21627
5	81029	30703	12957	931	17031	17031	15238	15914	0	15914
6	81757	33183	13535	909	14616	14616	16015	14591	0	14591
7	84967	21260	16181	613	13584	13584	4258	13473	0	13473
8	88260	22787	16672	799	14178	14178	18199	13328	0	13328
9	77434	32058	5862	1109	17827	17827	15073	17791	0	17791
10	74589	39341	7981	937	11763	11763	15264	13507	0	13507
11	81409	27490	19248	809	13327	13327	7397	12947	0	12947
12	87396	15637	20688	759	15460	15460	8835	14148	0	14148
TOTALS	1014470	342074	152082	18655	187206	187206	167719	187764	0	187764
AVE/YR	338157.	114025.	50694.	6218.	62402.	62402.	55906.	62588.	0.	62588.

(1) Inventory on hand at the end of the quarter.

(2) Total quantity on order at the end of the quarter.

(3) Receipts of the replenishment orders.

(4) Serviceable returns from customers of the supply system.

(5) Total shipments to fill new customer requisitions or to fill backorders.

(6) Total shipments to fill priority 1 requisitions or backorders.

(7) Total replenishment orders initiated.

(8) Total requisitions received.

(9) Total outstanding backorders that are canceled by the customer.

(10) Total priority 1 requisitions received.

Figure III-4. Receipts, returns, requisitions, and end-of-period measures.

** Note: In the current study, all demands are priority 1.

PERFORMANCE STATISTICS

... UNITS ...

PERIOD	(11) EXPEDITES	(12) RATIONING ACTIONS	(13) DISPOSALS	(14) TERMINATIONS
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
10	0	0	0	0
11	0	0	0	0
12	0	0	0	0
TOTALS	0	0	0	0
AVE/YR	0.	0.	0.	0.

(11) Total expediting actions initiated.

(12) Total of all rationing actions within the period.

(13) Total disposals within the period.

(14) Total terminations; i.e., totals for all replenishment orders that are cancelled or reduced when the inventory position exceeds the termination level.

Figure III-5. Expediting, rationing, disposal, and termination measures.

PERFORMANCE STATISTICS

UNITS

PERIOD	(15) TOTAL BACKORDERS	(16)** PRIORITY 1 BACKORDERS	(17) TOTAL BACKORDER weeks	(18)** PRIORITY 1 BACKORDER weeks	(19) INVENTORY weeks	(20) TOTAL FILLS	(21)** PRIORITY 1 FILLS	(22) TOT FILLS /TOT REQ	(23)** PRI 1 FILLS /PRI 1 REQS
1	1099	1099	3127	3127	1186380	13828	13828	0.92	0.92
2	666	666	13580	13580	1076624	15418	15418	0.95	0.95
3	1681	1681	12587	12587	1064541	17749	17749	0.92	0.92
4	2645	2645	17097	17097	1074017	18765	18765	0.87	0.87
5	1528	1528	22900	22900	1006883	13000	13000	0.82	0.82
6	1503	1503	18692	18692	991353	13577	13577	0.93	0.93
7	1392	1392	20116	20116	1045332	12524	12524	0.93	0.93
8	542	542	12731	12731	1036357	12147	12147	0.91	0.91
9	506	506	3780	3780	987006	16738	16738	0.94	0.94
10	2250	2250	16921	16921	882825	11057	11057	0.82	0.82
11	1670	1870	25436	25436	955696	9569	9569	0.74	0.74
12	558	558	13463	13463	1015204	12686	12686	0.90	0.90
•TOTALS•	16240	16240	180350	180350	12321618	167058	167058		
AVE/YR	5413.	5413.	60117.	60117.	4107206.	55686.	55686.	0.890	0.890

(15) Total backorders outstanding at the end of the period.

(16) Total priority 1 backorders outstanding at the end of the period.

(17) Total backorder-weeks observed during the period. For example,,if one item has a single requisition for 12 units in a backorder status for 3 weeks, there are 3 requisition-weeks of backorders, and 3 X 12 = 36 unit-weeks of backorders for that item.

(18) Total back-order weeks observed for priority 1.

(19) Total number of inventory-weeks observed. If there are 15 units on hand for the first 8 weeks of a quarter, and 6 units on hand for the remaining 4 weeks, a total of (15 X 8 + 6 X 4)== 144 inventory unit-weeks were observed in the period.

(20) Totals for all requisitions that were filled "off-the-shelf",i.e., that were filled without backordering.

(21) Total fills for priority 1 requisitions.

(22) Fill rate for the period.

(23) Priority 1 fill rate.

** Note: In this study, all demands are priority 1.

Figure III-6. Fill rate and backorder measures.

***** ORDER COST STATISTICS ' *****

TIME	SMALL ORDERS			LARGE ORDERS			TOTAL ORDERS		
COST TO ORDER	269.97			460.27					
	NUMBER	UNITS	DOLLARS	NUMBER	UNITS	DOLLARS	NUMBER	UNITS	DOLLARS
POP	0	0	0	0	0	0	0	0	0
1	22	27366	97649	2	1565	30896	24	28931	128545
2	15	11533	64864	1	1094	14222	16	12627	79086
3	10	13469	32130	0	0	0	10	13469	32130
4	9	9434	37661	3	2979	38727	12	12413	76388
5	14	15111	57250	1	127	13843	15	15238	71093
6	15	16015	50355	0	0	0	15	16015	50355
7	4	4258	13429	0	0	0	4	4258	13429
8	13	16996	43146	2	1203	24504	15	18199	67650
9	15	13492	55123	1	1581	20553	16	15073	75676
10	13	15139	50260	1	134	14606	14	15264	64866
11	10	6000	42077	2	1397	25567	12	7397	67644
12	9	7605	28807	2	1230	24774	11	8835	53581
TOTALS	149	156409	572751	15	11310	207692	164	167719	780443

SMALL ORDER COST 0.40210630E 05 PLUS LARGE ORDER COST 0.69048500E 04 EQUALS TOTAL ORDER COST 0.47114680E 05

EOQ BUY DOLLAR BREAK POINT 10000.00

AVERAGE COST/ITEM/YR

SMALL ORDER COST 0.13292280E 37 PLUS LARGE ORDER COST 0.13292280E 37 EQUALS TOTAL ORDER COST 0.26584560E 37

Figure III-7. Order Cost Summaries.

summarizes ordering actions using large and small purchasing methods, and Figure III-8 plots on hand stocks, on order stocks and backorders as a function of time.

In INSSIM, all statistics are accumulated according to three different measures; they are:

a. The number of distinct federal stock numbers or distinct actions associated with the current event.

b. The quantity of units associated with the event.

c. The dollar value of all units associated with the event.

For example, suppose a replenishment order for 12 units of a $10 item is placed. In this case, INSSIM records that there was one order action, 12 units were ordered, and $120 was the value of the order. The results presented in Figures III-4 through III-8 are all reported in terms of unit counts. However, INSSIM also produces similar tables summarizing the action and dollar counts recorded in the simulation.

This concludes our discussion of the Inventory System Simulator. In the next section, we will discuss how this model was used to evaluate the relative cost-effectiveness of the alternate PT-formulas.

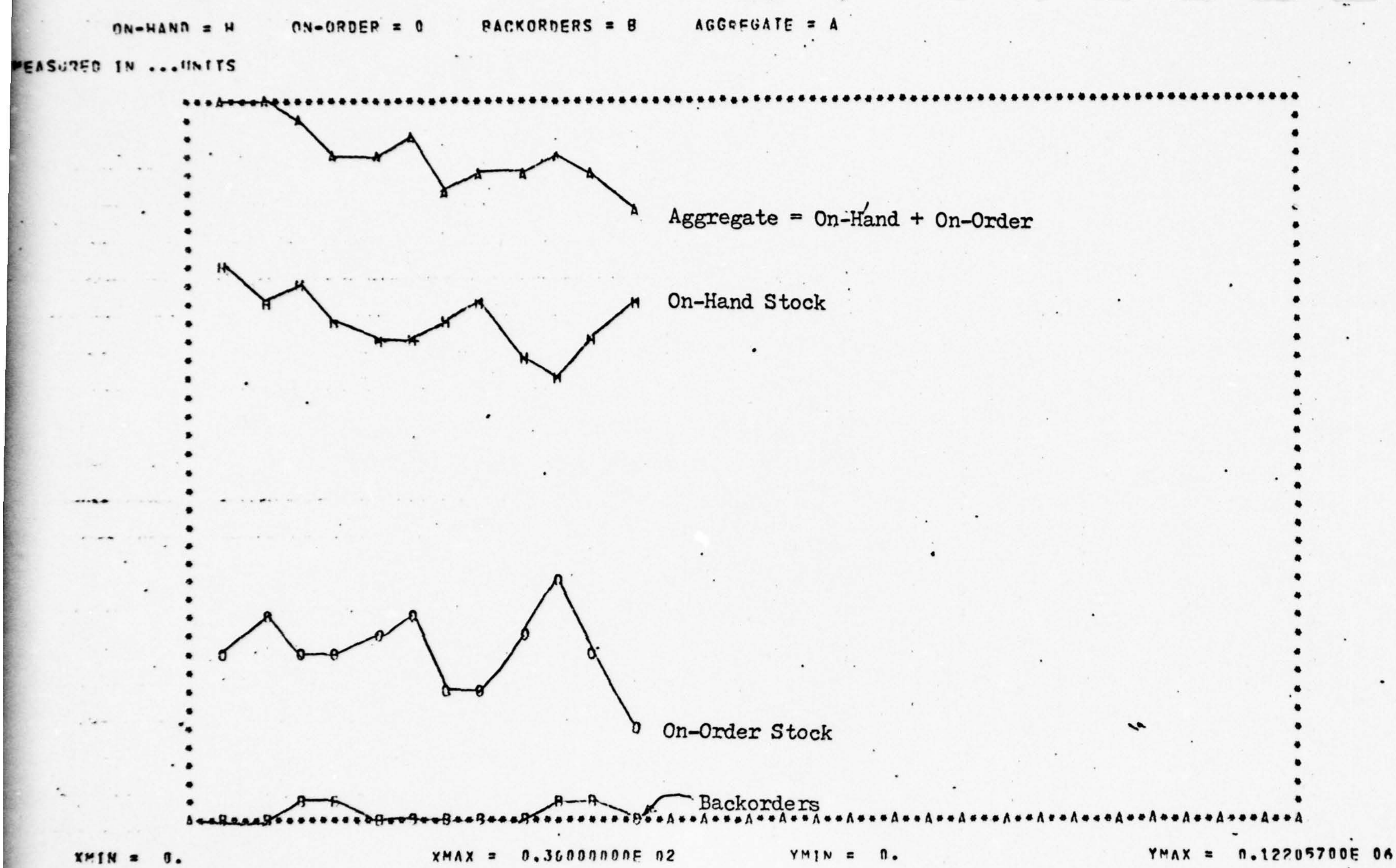


Figure III-8. On-hand and on-order assets and backorders verses time.

CHAPTER IV

MAJOR MODEL CONCEPTS

To understand the detailed program structure of INSSIM, the reader must possess a clear understanding of (a) the methods used to simulate the passage of time, (b) the events that simulate significant inventory system transactions, and (c) the data files that record current system status and that measure simulated performance. These major elements are the subject of this chapter.

The Timing Mechanism

The basic modeling concept employed in INSSIM is that of an "event." An event is a specific point in time in which the state or condition of the system changes or potentially changes. For example, the amount of stock on hand changes if a requisition is received and goods are shipped to fill the demand. Hence, receipt of a customer requisition is an event. Other events that may change the amount of on hand stock include delivery of a replenishment order by a vendor of the supply system and the return of serviceable assets from a customer.

The Future Events List

The sequencing of events through time is controlled by a Future Events List (FEL). The FEL is a list of all events scheduled to occur at some future (simulated) time. For each event on the list, the following values are recorded.

JTIME - The time the event is scheduled to occur.

JTYPE - The event type (code).

JFSN - The item number associated with the event (if applicable).

JQTY - The quantity associated with the event (if applicable).

JPRIOR - For a requisition event (code = 1), JPRIOR denotes the requisition priority, where 1 = high priority, 2 = low priority. For other event types, JPRIOR has special meanings, as discussed below.

For example, suppose a high priority requisition (event type 1) for 35 units of item number 6 is received. In this case, JTYPE = 1, JFSN = 6, JQTY = 35, and JPRIOR = (high priority). The method for setting the time variable, JTIME, will be discussed below.

Structurally, the FEL is a linked list in ascending order by scheduled event time. In such a list, new entries to the list are inserted in a previously unused data storage location, and pointers are used to indicate the correct sequencing of the entry in the list. The major variables associated with this list are shown in Table IV-1.

At the beginning of each simulation run, subroutine INFEL is called to initialize the pointers and control variables associated with the FEL. Scheduled events are

TABLE IV-1.
Future Events List Variables

NFEMAX	=	maximum number of entries on Future Events list
NENTRY	=	number of entries on the Future Events list
ILØCFE(J)	=	number of Jth unused data locations in Future Events file.
K	=	list index
JTIME(K)	=	clock time of transaction
JTYPE(K)	=	event type, as defined in Table IV-2
JPØINT(K)	=	pointer to next transaction in list
JFSN(K)	=	stock number identification*
JQTY(K)	=	quantity involved*
JPRIØR(K)	=	priority,* where JPRIØR = 1 high priority; 2 low priority
NFIRST	=	location of first transaction on chain
NTIME	=	time of earliest transaction on list
NLØC	=	location of last transaction placed on the chain

NOTE: If JTYPE(K) = 2, denoting a receipt event, JPRIØR(K) contains the pointer to the next most recent outstanding order for the same FSN, if any.

*if applicable

then entered and removed from the Future Events List by subroutines ENTER and REMOVE, respectively. New events are inserted into the FEL by the following CALL statement:

CALL ENTER (ITIME, ITYPE, IP3, IP4, IP5)

where

ITIME = scheduled time for the event to occur.

ITYPE = event type (code)

IP3, IP4, IP5 = event parameters--to be defined below--that vary by event type.

When subroutine ENTER is called, the new data is recorded, and the list pointers are updated to insert the new event in the proper time sequence.

Events are removed from the Future Events List by subroutine REMOVE; the CALL statement takes the form

CALL REMOVE (ITIME, ITYPE, IP3, IP4, IP5)

and the calling parameters are as defined above. Subroutine REMOVE removes the first entry from the FEL and updates the pointers accordingly; that is, it deletes the list entry with the smallest (earliest) scheduled clock time from the FEL, and returns the corresponding data values through the parameter list.

Subroutine WRIFEL is another routine associated with the Future Events List. In debugging new INSSIM models, it is sometimes desirable to write out the entire Future Events

List at selected times within a simulation run. Subroutine WRIFEL fulfills this function.

Timing Conventions

To simplify programming and analysis tasks, it was desirable to establish timing conventions that differ slightly from values associated with the Gregorian calendar. Specifically, INSSIM assumes that each day consists of 100 Time Units (TU). Days, weeks, months, and years are then assumed to be related as shown in Table IV-2. Since the values are used repeatedly throughout a simulation run, specific COMMON variables are established for each of these time units. These variables are also shown in Table IV-2. Values of these variables are initialized by subroutine INITAL at the beginning of each INSSIM simulation run.

INSSIM Events

In the Inventory System Simulator, three categories of events are used to simulate the EOQ Buy Computation System. They are:

a. Material flow events - describing the receipt of a customer requisition, delivery of a replenishment order, or the return of serviceable assets.

TABLE IV-2. Time Conventions*

Time Interval	Basic Value	Time in TU's		INSSIM Variable
1 Day		100 TU's	=	ITDAY
1 Week	7 Days	700 TU's	=	ITWEEK
1 Month	4 Weeks	2,800 TU's	=	ITMNTH
1 Quarter	3 Months	8,400 TU's	=	ITQTR
1 Year	4 Quarters	33,600 TU's	=	ITYEAR

* It is assumed that 1 Day = 100 Time Units (TU's), and each INSSIM Time Variable is expressed in TU's.

b. Management action events - describing important management decision points. The calculation of control levels, status reviews to determine what reorder, termination, or disposal actions are appropriate, and the establishment of budgets and procurement guidelines are examples of these events.

c. Simulation bookkeeping events - events to collect time-valued statistics, to generate demand and serviceable returns, and to signal the end of the simulation.

At present, thirteen possible events are included in the INSSIM model. These events are listed in Table IV-3. In INSSIM, there is a separate FORTRAN subroutine to describe how each event changes the state of the system, and to record associated performance statistics. Table IV-3 also presents the subroutines associated with each event and the definitions of the IP3, IP4, and IP5 parameters used in calling subroutines ENTER and REMOVE.

All INSSIM events may be classified as either (a) scheduled events, (b) random events, or (c) dependent events. Scheduled events are events with a specific schedule of occurrence times. For example, inventory status reviews (event type 5) occur at the end of every week, and the History File Update event (event type 9) occurs at the end of every quarter. On the other hand, customer requisitions (event type 1) and serviceable returns (event type 4) are random events--the time of occurrence of these events is not known in advance. Finally, the delivery of a replenishment order (event type 2) is a dependent event since the time of delivery is dependent

TABLE IV-3. INSSIM Events

Event Type	Event	Event Subroutine	Parameters IP3	IP4	IP5
1	Requisition	REQ	N	QTY	Priority
2	Receipt of Replenishment Order	RECEIV	N	"	pointer
3	Cancellation	CANCEL	N	"	
4	Serviceable Return	RET	N	"	
5	Inv Status Review	STATUS			
6	Levels Computation	LEVEL			
7	Buy Guidelines	GUIDE			
8	Budget Review	BUDGET			
9	Update History File	FORUPD			
10	End of Run	OUTPUT			
11	Special Statistics	SSTAT	KEND		
12	Generate Requisitions and Returns	DEMPAR			
13	Trace	MAIN			

where

N = item number
QTY = quantity associated with this event
KEND = week number within the current quarter

upon earlier management reordering decisions. Dependent events are caused by the occurrence of an associated random or scheduled event.

In INSSIM, the first occurrence of each scheduled event is created by subroutine INITAL; that is, INITAL puts an entry on the Future Events List for each scheduled event at the beginning of a simulation run. Subsequent scheduled events are put on the Future Events List each time a new scheduled event occurs. For example, at the conclusion of a status review event (event type 5), a subsequent type 5 event is scheduled to occur one week in the future.

Specific values for the time between scheduled events are set in subroutine INITAL. Table IV-4 defines specific time variables initialized in this routine.

At present, receipt of a customer requisition and of serviceable returns (event types 1 and 4) are the only random events simulated in INSSIM. These events are scheduled by subroutine DEMPAR. The process for accomplishing this is discussed in detail in Chapter VI.

TABLE IV-4. TIMING VARIABLES Established in INITAL

Variable	Definition
IQTRND	clock time that marks end of current quarter
ITINV	number of current quarter
ITIME	current simulation clock time (100 Time Units = 1 day). At the beginning of a simulation run, ITIME = 0.
ITLEVL	clock time of the next computation of stock levels
IDLEVL	time interval between stock level computation
ITDIV	clock time of the next division level review
IDDIV	time interval between division level reviews
ITFOR	time of the next update of demand history records (Event type 9)
IDFOR	time between history file updates
ITHQ	time of next Headquarters USAF stock fund budget review
IDTHQ	time between Headquarters USAF budget reviews
ISTOP	clock time that simulation is to be stopped
ISTAT	clock time for activating statistics collection routines
IDSTAT	time interval between statistics collection
INQTR	number of quarters to be simulated

An Illustration

Figure IV-1 illustrates the interaction of the ENTER and REMOVE routines in the management of data in the Future Events List. As noted above, at the beginning of an INSSIM run, scheduled events are entered onto the FEL by subroutine INITAL. Figure IV-1(a) illustrates data stored on the FEL at this time. At the beginning of the simulation run, the simulated clock time (ITIME) is set to zero, and seven events are placed on the Future Events List. Note that a type 5 (JTYPE = 5) event is scheduled to occur at time 40 (JTIME = 40). Next, a type 6 event is inserted on the FEL to occur at time 30. Event types 11, 10, 9, 12, and 13 are also placed on the FEL by subroutine INITAL.

The variable NFIRST identifies the storage location of the most imminent event on the Future Events List; that is, the event with the lowest scheduled event time. The variable NENTRY identifies the total number of events on the Future Events List at the current time, while the variable NTIME specifies the clock time associated with this most imminent event.

As noted above, subroutine WRIFEL may be called to print the entire contents of the Future Events List. Figure IV-1(b) illustrates the output of WRIFEL at the beginning of the simulation run associated with the data presented in Figure IV-1(a). Note that at this time, there are seven entries on the Future Events List (NENTRY = 7), and the most imminent of these events is the seventh entry in the FEL (NFIRST = 7).

Figure IV-1. Event Routine Interactions

a) Entering of initial INSSIM events on the FEL by Subroutine INITAL

Routine	ITIME	JTIME	JTYPE	IP3	TP4	TP5	K*	NFIRST	NENTRY	NTIME
ENTER	0	40	5	0	0	0	1	1	1	40
ENTER	0	30	6	0	0	0	2	2	2	30
ENTER	0	695	11	1	0	0	3	2	3	30
ENTER	0	8400	10	0	0	0	4	2	4	30
ENTER	0	8400	9	0	0	0	5	2	5	30
ENTER	0	100	12	0	0	0	6	2	6	30
ENTER	0	0	13	0	0	0	7	7	7	0

b) WRIFEL Output at Beginning of Simulation

ITIME	NFIRST	NENTRY
0	7	7

EVENTS LIST AT TIME

No.	JTIME	JTYPE	JFSN	JQTY	JPRIOR	JPOINT
1	40	5	0	0	0	6
2	30	6	0	0	0	1
3	695	11	1	0	0	4
4	8400	10	0	0	0	5
5	8400	9	0	0	0	0
6	100	12	0	0	0	3
7	0	13	0	0	0	2

c) Initial Simulation Events

ROUTINE	ITIME	KTYPE	KFSN	KQTY	KPRIOR	K*	JTIME	JTYPE	JFSN	JQTY	JPRIOR	K*	NFIRST	NENTRY	NTIME
REMOVE	0	13	0	0	0	7									
ENTER	0						700	13	0	0	1	7	2	7	30
REMOVE	30	6	0	0	0	2									
ENTER	30						1430	6	0	0	0	2	1	7	40
REMOVE	40	5	0	0	0	1									

*K = row number in the FEL in which the current line of data is stored

Figure IV-1(c) illustrates the initial events that occur at the beginning of the INSSIM simulation run associated with the data given in IV-1(a). Note that at time = 0 (ITIME = 0) a type 13 event (KTYPE = 13) is removed from the Future Events List. As shown in Table IV-3, a type 13 event causes on-hand and on-order stock status information to be recorded for later plotting by the MAIN routine. This event also causes another type 13 event to be scheduled to occur one week (i.e., 700 time units) in the future. This scheduling is indicated by the second line of Figure IV-1(c). The next event on the Future Events List is a type 6 event. This event is removed at time 30 (ITIME = 30). As shown in Table IV-3, a type 6 event causes subroutine LEVEL to be called to compute levels for all items being simulated. When this is completed, another levels computation is scheduled to occur at time 1430 by placing a type 6 event on the Future Events List. Finally, Figure IV-1(c) indicates that a type 5 event is removed from the Future Events List at clock time 40. A type 5 event causes subroutine STATUS to be called to initiate a status review of the on-hand and on-order stocks of each inventory item. After the status review is completed, another status review event would be placed on the Future Events List.

The process illustrated above is then continued repeatedly throughout the simulation. Required simulation events are placed on the Future Events List by subroutine ENTER, and are subsequently removed from this list at the appropriate scheduled time by subroutine REMOVE.

DATA FILES

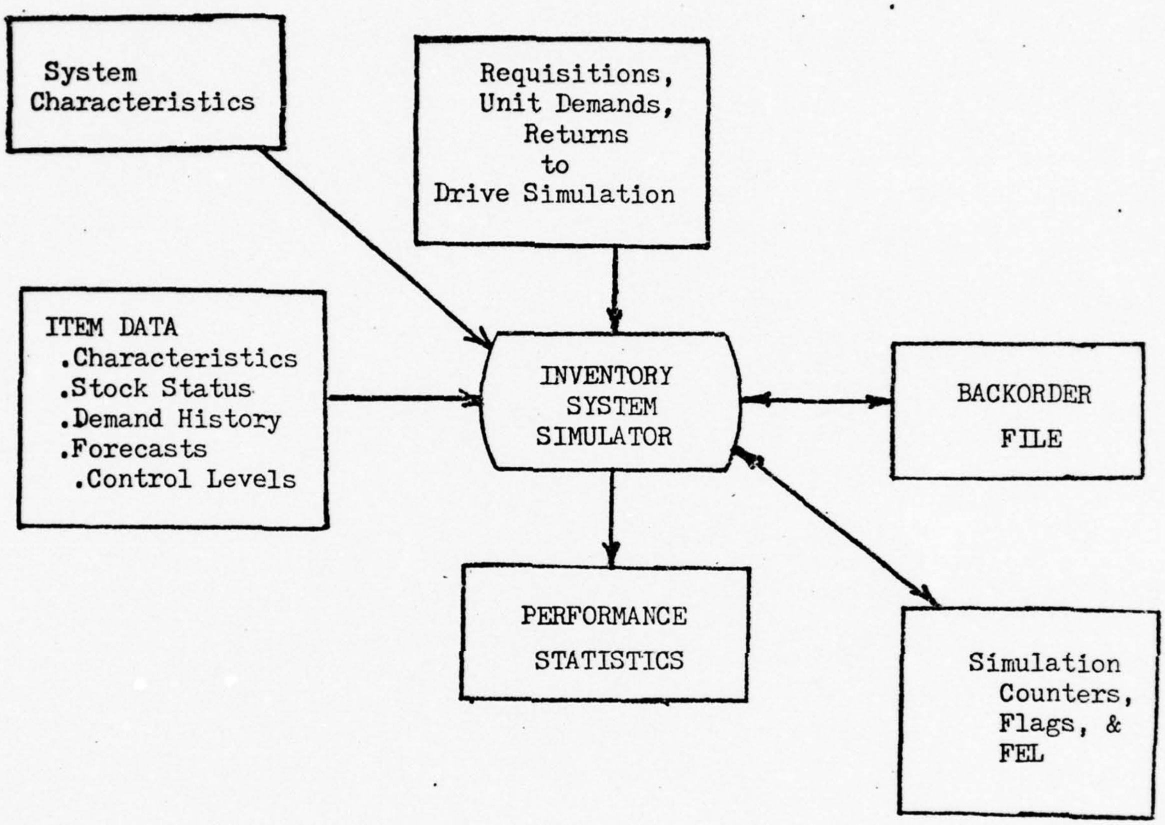
A large number of data elements are required to operate any computer simulation model, and INSSIM is no exception. The major categories of data associated with INSSIM are illustrated in figure IV-2. As shown in the figure, INSSIM data elements may be logically classified into one of the following files:

1. System Characteristics File
2. Item Data File
3. Backorder File
4. Demand Driver File
5. Performance Statistics File
6. Simulation Counters and Flags File

Let us now consider the contents of each of these files.

SYSTEM CHARACTERISTICS FILE

The Systems Characteric File contains data elements that define the system as a whole. The number of items managed by this system in a single simulation run, the cost of processing orders using small purchase and advertised procurement methods, and levels computation codes are examples of data elements in this file. This information is provided primarily



Firgure IV-2. Logical Data Files in INSSIM.

through inputs of file 05. Specific data elements input through this file will be discussed in detail in chapter VII.

ITEM DATA FILE

The item data file contains information specifically related to each item being simulated. An item's unit cost, its current level of on-hand and on-order stocks, and control levels (e.g., ROL, termination level, or retention level) are examples of this type of data element. Table IV-5 presents a detailed list of information items which are contained in this file. As shown in table IV-5, the item data file contains several subcategories of information. These define (a) specific physical and management characteristics of the item, (b) data describing the stock status of the item, (c) variables describing the demand history for the item, (d) forecast variables which quantify forecast usage rates and demand variability and (e) control levels used in the management of the item's inventories.

Table IV-5. Item Data File Variables

ITEM CHARACTERISTICS

Variable	Definition
N	Item Index
NITEM	Number of items simulated
ALC	Air Logistics Center that manages this item
FSN	Federal Stock Number, where FSN(1) = Material Management Code FSN(2) = Federal Stock Class FSN(3) = First six characters of NIIN FSN(4) = Remaining characters of NIIN
UM	Unit of measure
NOUN	Item name
MGTCD	Item Management Codes, where MGTCD(1) = Essentiality code MGTCD(2) = Supply management grouping MGTCG(3) = Weapon system code MGTCD(4) = Type computation code
UCOST(N)	Unit cost (dollars)
LTADM(N)	Administrative lead time (months)

ITEM STOCK STATUS

Variable	Definition
INVACT(N)	Actual inventory on hand (units)
INVDUE(N)	Inventory due-in from supplier (units)
NBOIU(N)	Priority I backorders, in units
NBOIR(N)	Priority I requisitions backordered
NBOTR(N)	Total requisistions backordered, all priorities
NBOTU(N)	Total backorders, all priorities, in units
NORDPT(N)*	Pointer to most recent outstanding order
NBOPT(N)*	Pointer to oldest, highest priority backorder

*Note: By convention, pointers are set to zero if there are no other elements in the data chain.

Table IV-5. Item Data File Variables

(Continued)

HISTORY VARIABLES	
NDHIS	Number of periods saved in history file
NDENT(N)	Number of forecast periods since item entered system (negative values indicated item has not yet entered system)
NDEMAC(N)	Counter to accumulate demand for Nth item during the current forecast update period.
NDEMND(N, I)	Demand for item N during Ith past period, I=1, 2,...., NDHIS
NREQ(N)	Counter to accumulate the number of requisitions for item N during the current period.
FORECAST VARIABLES	
ADR(N)	Net annual demand rate for item N the net rate of unit demands less servicable returns.
RMEAN(N)	Estimate of demand during next demand period.
RTREND(N)	Estimate of trend in demand per period.
RMAD(N)	Standard deviation of lead time demand
PERSUM(N)	Sum of past forecast errors
KNT(N)	Counter used for adaptive smoothing forecasts
RSIGLT(N)	Estimate of standard deviation of lead time demand
REQSIZ(N)	Estimate of mean requisition size for item N.
REQMAD(N)	mean absolute deviation of requisition size for item N.
CONTROL LEVELS	
IROL(N)	Reorder level
IRQTY(N)	Reorder quantity
ISUL(N)	Support level
IRL(N)	Retention level
ITL(N)	Termination level

BACKORDER FILE

The backorder file records all outstanding backorders in the inventory system at a given point in time. This file contains a record for each requisition in a backorder status. Specific data elements in this file are defined in Table IV-6. Included in these data elements are the item number associated with the backorder, the backorder quantity and priority, and the time the item was backordered. A pointer system is used to relate entries in the backorder file with the specific items associated with that backorder. These pointer variables are also defined in table IV-6.

DEMAND DRIVER FILE

The demand driver file contains data elements required to generate the demands and serviceable returns that drive the inventory system. The number of requisitions and the associated units per quarter and the number of serviceable returns to be generated each quarter are examples of data elements in this file. Table IV-7 defines specific variables associated with the demand driver file.

TABLE IV-6. BACKORDER FILE

Variable Name	Definition
J	File Index
ITMBAC(J)	simulated clock time that the Jth entry in the backorder file was placed into the file
IDFSNB(J)	item number associated with the Jth backorder
IPRIØR(J)	priority of Jth backorder
IQTYB(J)	quantity backordered
IBACPT(J)	pointer to the file entry of the next backorder for this same item
NBMAX	maximum number of entries in backorder file
ILØCBK(K)	index of Kth unused data location in backorder file
NLØCBK	number of unused data locations

TABLE IV-7. DEMAND DRIVER FILE

IDEMND(N,J)	Number of units of demand for item N during period J
IRETUR(N,J)	Number of serviceable Returns for item N during period J
IREQ(N,J)	Number of requisitions for item N in period J associated with the unit demands IDEMND(N,J)

Subroutine DEMPAR is called at the beginning of each simulated quarter. This routine uses information in the Demand Driver File to generate specific requisition and serviceable return events. DEMPAR accomplishes this in such a way that the total requistions placed within a quarter exactly equal the number of units of demand which are input through the Demand Driver File. Chapter VI discusses the demand generation process in detail.

PERFORMANCE STATISTICS FILE

The Performance Statistics File contains measures of the levels of activity observed during a simulation run. Table IV-8 defines specific data elements contained in this file. As illustrated in Table IV-8, performance measures are recorded in three different units of activity. That is, each type of activity is recorded in terms of (a) the number of actions or federal stock numbers associated with that activity, (b) the number of units of a specific item associated with that activity and (c) the dollar value of the number of units associated with that activity or action. For example, if an order is placed for a $10 item and 21 units are purchased the variable IORDER (I,1) would be increased by 1, since it counts

TABLE IV-8. PERFORMANCE STATISTICS FILE

Variable	Definition
I	period index
J	type of measure, where J = 1 actions/FSN 2 units 3 dollars
INVØH(I,J)	inventory on hand at end of period
INVØR(I,J)	inventory on-order at end of period
IRECET(I,J)	receipts
IRETRN(I,J)	returns
INVDAY(I,J)	inventory-weeks
IØRDER(I,J)	orders placed
IDISPS(I,J)	disposals
ITERM(I,J)	terminations completed
IEXPED(I,J)	expediting actions
IRATON(I,J)	rationing actions
IREQC(I,J)	total requisitions cancelled
IREQT(I,J)	total requisitions
IREQI(I,J)	priority I requisitions
IBACKT(I,J)	total backorders (end of period)
IBACKI(I,J)	priority I backorders (end of period)
IBAKDT(I,J)	total backorder weeks
IBAKDI(I,J)	priority I backorder weeks
IFILLT(I,J)	total fills (Number of requisitions filled immediately upon receipt)
IFILLI(I,J)	priority I fills
ISHIPT(I,J)	total shipments
ISHIPI(I,J)	total priority I shipments

TABLE IV-8. (CONT'D)

ISMORD(I,J)	total small orders
ILGORD(I,J)	total large orders
IBOPOH(J)	On hand inventory at beginning of simulation
IBOPOR(J)	on-order inventory at beginning of simulation

the number of orders placed. The variable IORDER (I,2) would be increased by 21 units, since this variable records the number of units ordered during period I. Finally, the variable IORDER (I,3) would be increased by 210, since an order for 21 units which cost $10 each represents a total dollar activity of $210. All of the other performance measures defined in table IV-8 are updated in a similar manner.

SIMULATION COUNTERS AND FLAGS

Every simulation model requires a series of counters and flags that are required to control the progress of the simulation, and to perform necessary bookkeeping tasks. The number of quarters to be simulated, the number of the current statistics collection interval, and the number of simulation runs to be performed are examples of these types of data elements. The variables of the future events list (FEL) defined in Table IV-1 are an important example of this type of data. Other major variables in this category are defined as inputs to INSSIM through File 05. These later variables will be discussed in detail in chapter VII.

We have now reviewed the major concepts which are the building blocks of the INSSIM simulation model. We've discussed timing mechanisms involved in the model and the events used to represent inventory system activities. The data files used to describe the status of the current system, to drive the simulation model, and to update performance statistics were also discussed. In the next chapter, we will discuss the workings of the MAIN routine. This routine is the primary INSSIM routine, in that it weaves all of the major concepts discussed here into a working model of a single location inventory control system.

CHAPTER V

MAIN PROGRAM

The MAIN program is the work horse of the Inventory System Simulator. This routine is responsible for the control of input data, initialization of status and performance variables, the scheduling of simulation events, and the control of output products. An understanding of this routine is essential for those who desire to enhance the current system.

A general outline of the MAIN program is presented in Figure V-1. As shown in the figure, the initial portion of MAIN is concerned with initializing the statistical accumulators and status variables maintained by the system, and in initializing the Future Events List.

After initialization, MAIN controls the scheduling of simulated events, and the calling of associated event subroutines. To do this, subroutine REMOVE is used to remove the most imminent event from the Future Events List. This event is called the "current" event. The program then branches based on the event code of the current event. As shown in Figure V-1, separate subroutines are used to perform bookkeeping functions associated with each event type. For example, event type 1 represents a customer requisition. When a type 1 event is removed from the Future Events List, subroutine REQ is called

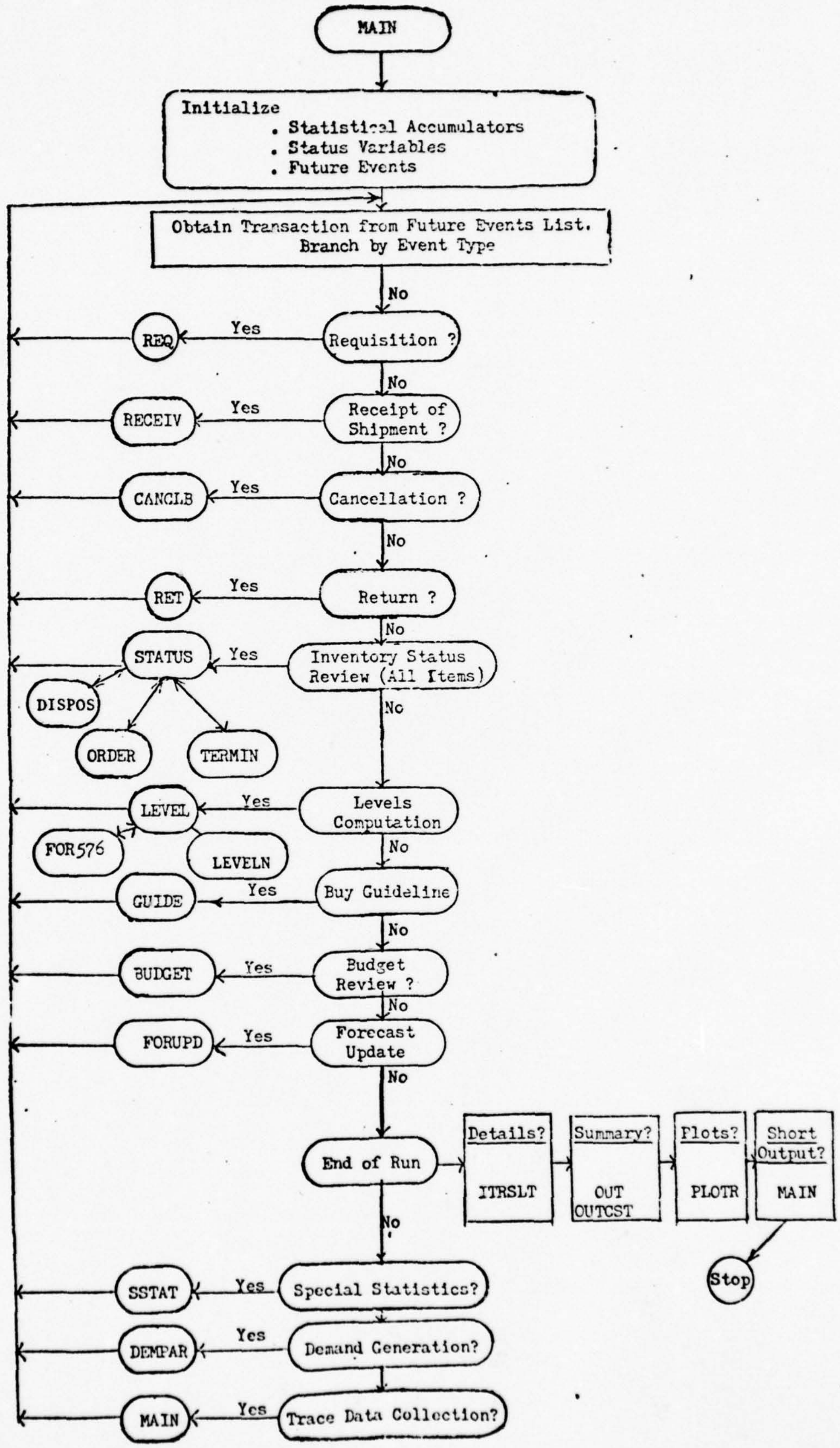


Figure V-1. General Outline of MAIN Program to perform the associated bookkeeping tasks. Similarly, for event types 2 and 3, subroutines RECEIVE and CANCLB, respectively, are called.

The event subroutines may in turn call other routines to assist in the update calculations. For example, subroutine STATUS is called to simulate the occurrence of an inventory status review (event type 5). This routine in turn calls subroutine STATN to determine appropriate management actions associated with a given item N. If appropriate, STATN may in turn call the routines ORDER, TERMIN, or DISPOS to represent procurement, contract termination, or disposal actions, respectively.

Event type 10 denotes the end of the simulation run. When this type of event is removed from the Future Events List, one or more output routines may be called. Specific output products produced are specified by parameters which are input through File 05. A detailed discussion of these control parameters is presented in Chapter VII. If detailed simulation results by item are to be produced, subroutine ITRSLT is called. If tabular summaries are desired, subroutines OUT and OUTCST are called. If plots of important inventory characteristics are desired, subroutine PLOTR is called. Finally, an optional abbreviated performance report may be requested that is useful in the debugging of INSSIM enhancements.

A detailed flow chart of the MAIN program is presented in Figure V-2. As shown in the figure, MAIN first sets several management parameters that are used if alternate levels calculations are not specified. Definitions for the variables GTLF, GRLF, GSULF, and GSLF shown in Figure V-2 are presented in Table V-1. These values are used in levels calculations performed by subroutine LEVELN. After initializing the above variables, MAIN reads simulation run parameters specified in File 05. These inputs are discussed in detail in Chapter VII. The "run" loop then begins. An input parameter, NRUN, specifies the number of distinct simulation runs to be performed. The variable CSHORT(K), which is provided as input in File 05, specifies the value of the Lagrange multiplier to be used in the Kth simulation run. Hence, the value of the variable CSHORT is set to the appropriate value for use in the current simulation run.

Next, MAIN initializes the random number stream, rewinds the item data file (File 07), and zeros the statistical accumulators arrays.

Subroutine INITAL is now called to initialize the time parameters associated with the simulation run, and to create the initial scheduled events by putting them on the Future Events List. Subroutine INITEM is then called to read in input data for the NITEM items that are

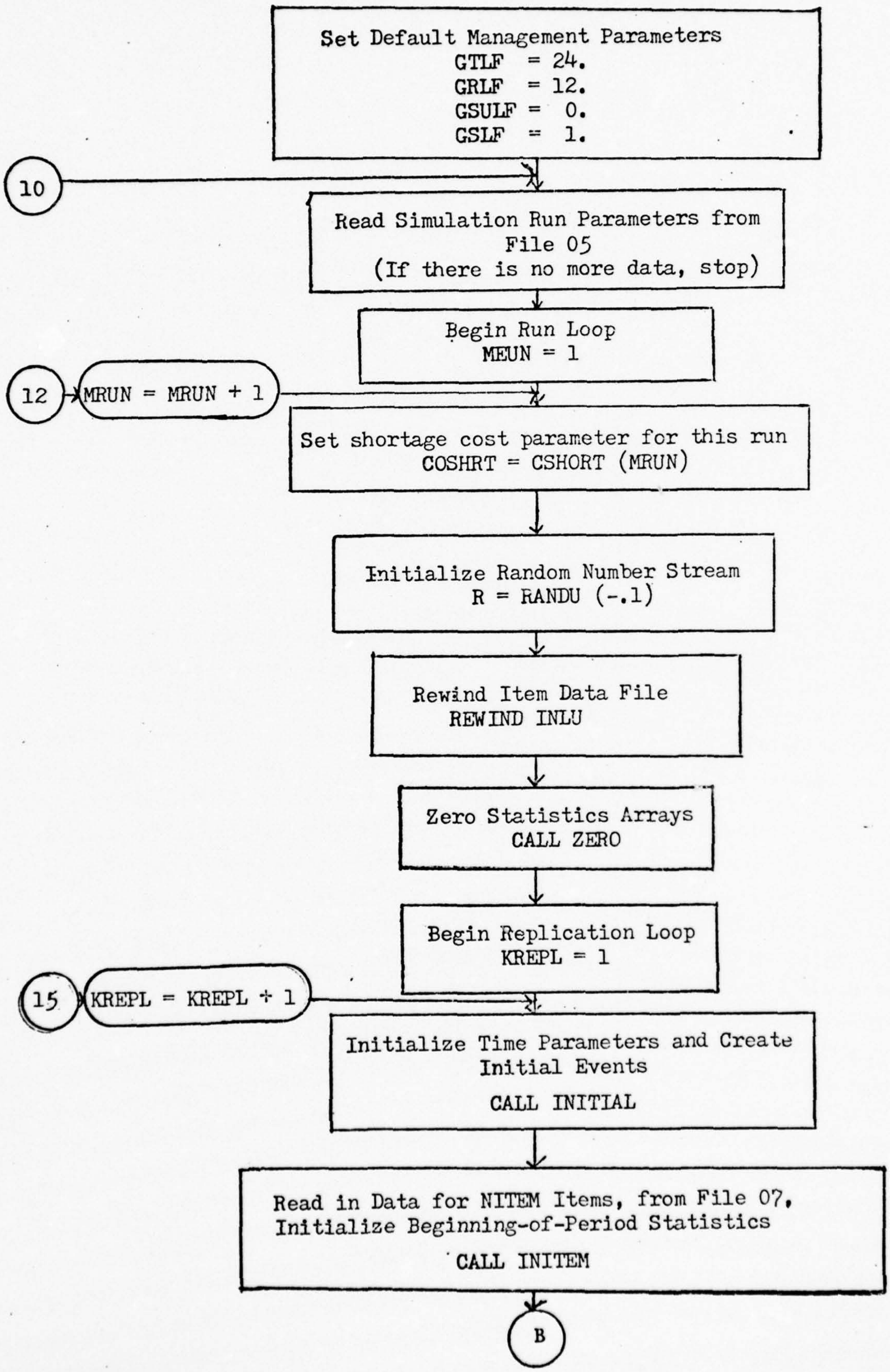


Figure V-2. The MAIN Program.

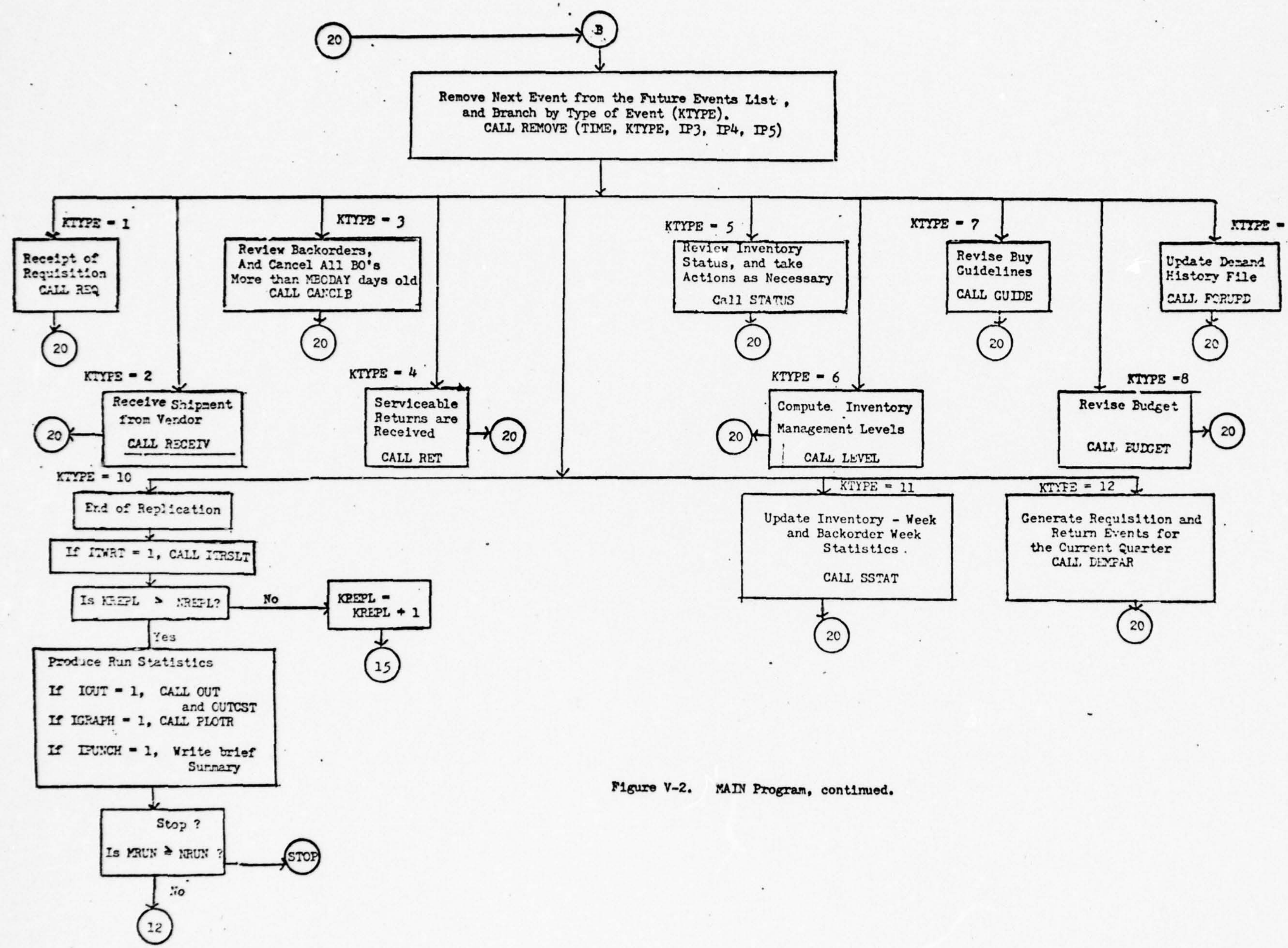


Figure V-2. MAIN Program, continued.

TABLE V-1

DEFAULT VALUES FOR LEVELS CALCULATIONS

Variable	Definition
GSLF	safety level factor (in months)
GROQF	order quantity factor (in months)
GTLF	termination level factor (in months)
GRLF	retention level factor (in months)
GSULF	support level factor (in months)
GEØQF(I)	order quantity factor for Ith demand rate class

Formulas

MR = monthly demand rate in units

lead time requirement RLT = (admin + prod lead time)*MR

safety level SL = GSLF*MR

reorder quantity ROQ = GROQF*GEOQF(I)*MR

reorder level ROL = RLT+SL+

termination level TL = ROL+GTLF*MR

retention level RL = TL+GRLF*MR

support level SUL = GSULF*RLT

to be included in the current simulation replication. This data is read in from File 07.

Simulation of specific inventory events now begins. Subroutine REMOVE is called to remove the most imminent event from the Future Events List, and MAIN then branches based on the event code of this event. Figure V-2 specifies the specific subroutines that are called during this stage of processing. Detailed narratives describing each of these programs are contained in Volume II of this report.

Observe that when a type 10 event occurs, the variable KREPL is checked to see if the replication count exceeds NREPL, the number of replications specified for the current simulation run. If not, KREPL is increased by one, the program returns to block 15 to read in a new set of item data and to perform another replication of the simulation. If, on the other hand, KREPL equals NREPL, simulation run statistics are produced. If the output control parameters IOUT equals one, subroutines OUT and OUTCST are called to present tabular summaries of the aggregate performance statistics observed during the NREPL replications of the simulation experiment. If IGRAPH equals one, subroutine PLOTR is called to produce graphical results. If IPUNCH equals one, the abbreviated statistical summary is written from MAIN.

After output products are produced, MAIN tests to find if all requested runs have been performed. If so, the simulation stops. Otherwise, MAIN returns to statement number 12. The run counter, MRUN, is then incremented, the shortage cost parameter COSHRT is reset to the new Lagrange multiplier value, and the initialization process begins again. The sequence of initialization, simulation, and output operations is repeated until all requested simulation runs are performed.

CHAPTER VI

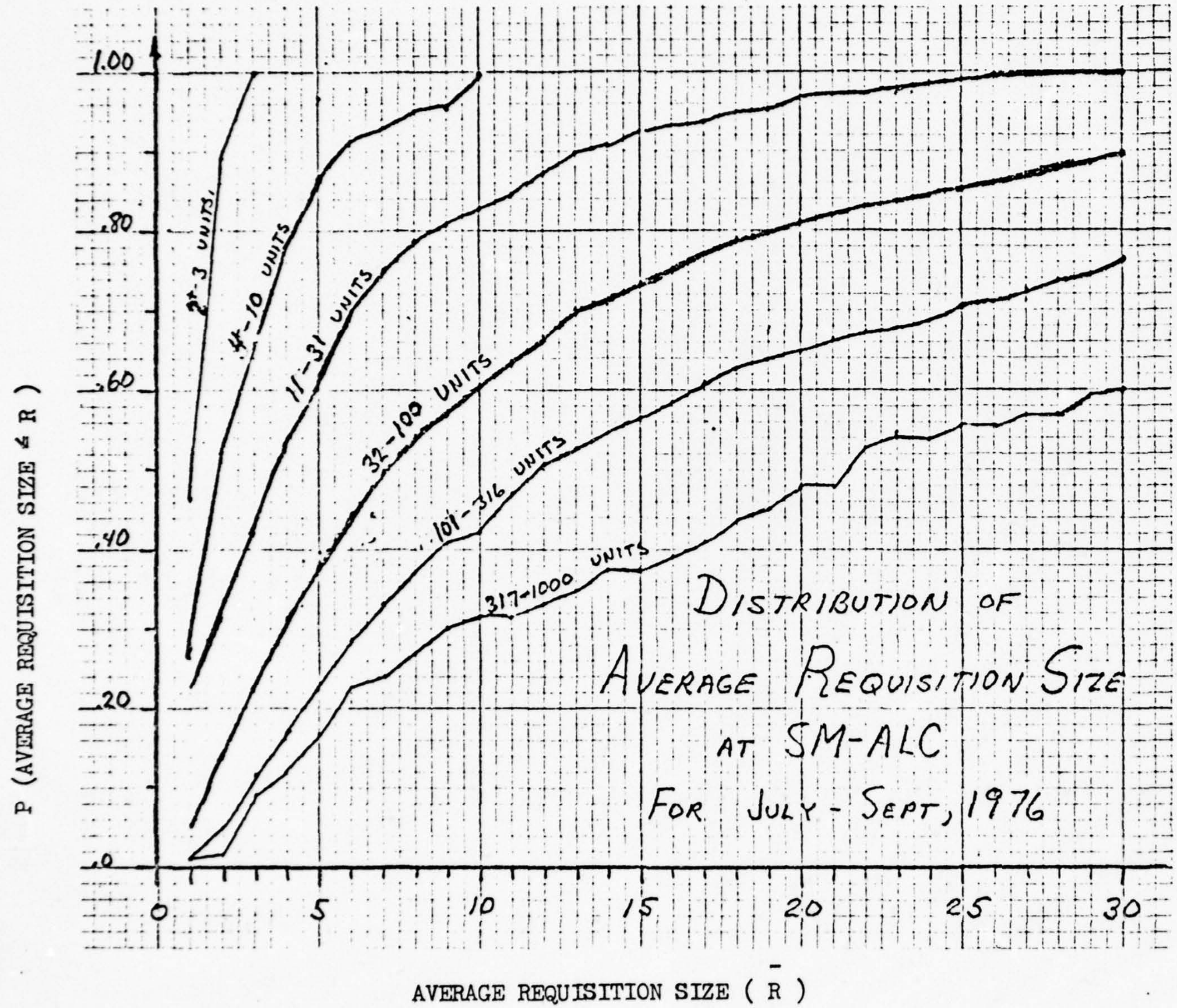
THE DEMAND GENERATION PROCESS

An important feature in the development of any simulation model is the method for representing customer demands. In our model, actual usage history for EOQ items was used as the basis for the demand generation process. These histories contain the number of units that were demanded by quarter since fiscal year 1971. Unfortunately, detailed historical information on the nature of specific requisitions placed for each item was not recorded. Consequently, it was necessary to develop a method for describing the requisition generation process using the recorded quarterly usage history as a starting point.

In developing the demand generation process, we relied heavily on the statistical results discussed in Section II of reference 2. The distribution of average requisition size was a primary result of that section. This distribution is reproduced in Figure VI-1.

A flow chart of the steps used to generate requisition sizes with the same statistical characteristics as shown in Figure VI-1 is presented in Figure VI-2.

As shown in the figure, the demand generation process begins by considering the number of units, D, that were demanded in a given historical quarter. We next determined the demand class associated with this particular level of



Demand in QTR	Number of Items
0	66,229
1	3,313
2-3	2,873
4-10	3,130
11-31	1,717'
32-100	819
101-316	245
317-1,000	67
1000+	23
Total	78,436

Figure VI-1. Distribution of Average Requisition Size.

demand. Once the appropriate demand class is identified, a Monte Carlo process is employed to simulate the specific number and size of requisitions associated with a total demand of D units. Block C of Figure VI-2 illustrates the major steps in the Monte Carlo process. The procedure begins by determining a random number X which is uniformly distributed on the interval between 0 and 1. This is conceptually equivalent to selecting a random percentile from the cumulative probability distribution of average requisition size. Next, the cumulative probability distribution illustrated in Figure VI-1 was used to determine the specific requisition size R associated with the random percentile X. This was our tentative value for the first requisition to be generated. We then check if the total cumulative units generated so far exceeds D units. If the cumulative units is less than D, the requisition generation process continues as described above. On the other hand, if the cumulative units exceeds D, the last requisition is reduced. This is to ensure that the total number of units requisitioned throughout the period exactly equals the amount D recorded in the item history record.

The above process determines a set of requisitions whose total units equal the total demand recorded in the item history record. To determine the time of arrival of each requisition, it is assumed that arrival times were uniformly distributed throughout the quarter. A Monte Carlo process is then used to determine the specific arrival time for each requisition generated. This is equivalent to assuming that the time between arrivals is exponentially distributed within a quarter.

To implement the above logic, subroutine DEMPAR is called at the beginning of each simulated quarter. Subroutine GETREQ is then called by DEMPAR to determine specific requisition sizes using the logic defined in Figure VI-2. DEMPAR then places a type 1 (requisition) event on the Future Events List (FEL) for each generated requisition. Finally, DEMPAR places a type 13 event on the FEL to occur one quarter in the future. This type 13 event triggers the next call to DEMPAR.

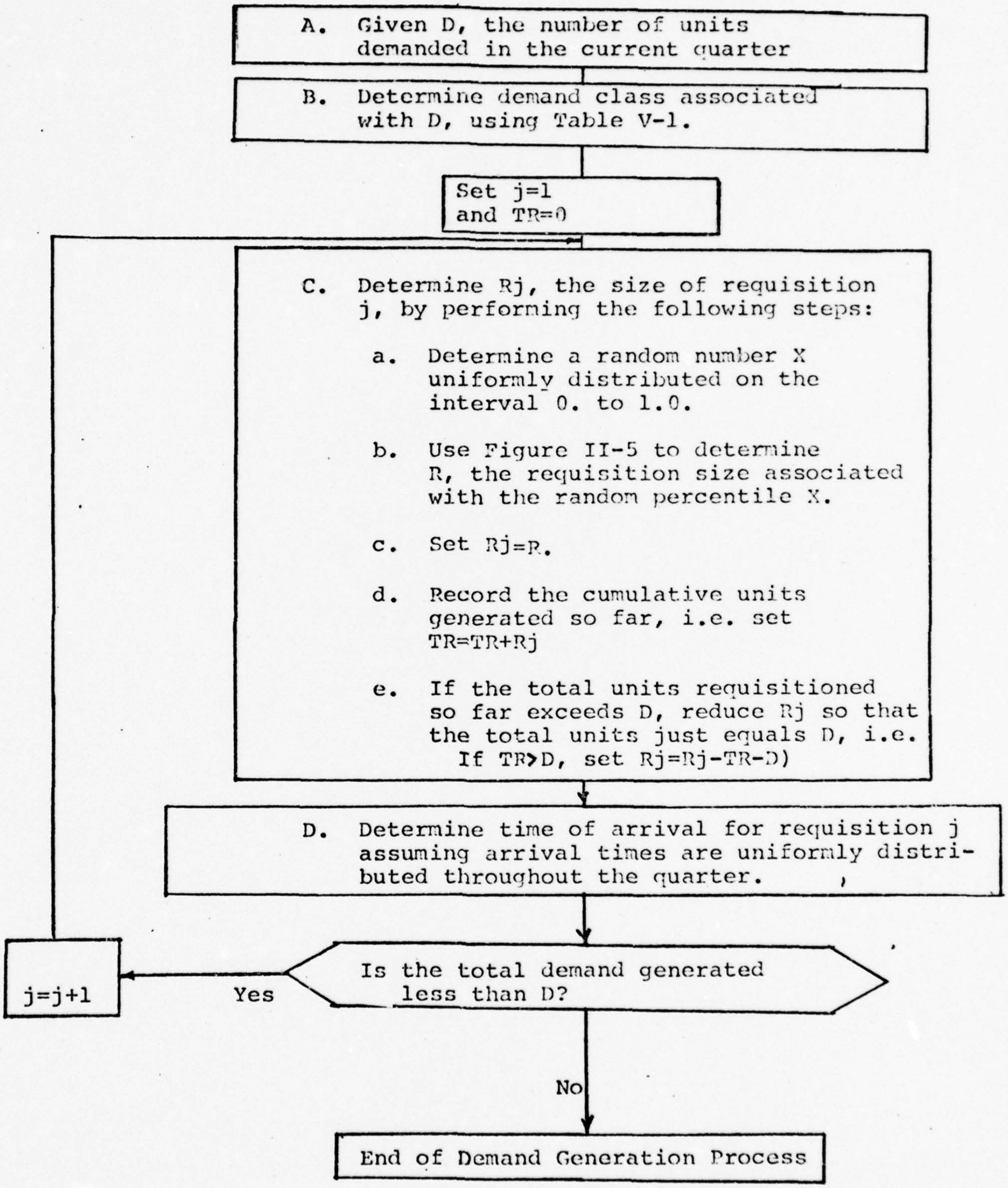


Figure VI-2. The Demand Generation Process

CHAPTER VII. INPUT SPECIFICATIONS

As illustrated in Figure III-2, INSSIM requires two input files, File 05 and File 07. File 07 specifies the characteristics of each inventory item to be simulated, while File 05 specifies the number, size, and characteristics of the simulation runs to be performed. Let us now consider each of these files in detail.

FILE 05: SIMULATION RUN SPECIFICATIONS

Eight different parameter cards are required to specify a given INSSIM simulation scenario. These cards provide the following types of data:

Card	Contents
1	Identification
2	Output Controls
3	Debug Flags
4	Input File Specifications
5	Management Methods to be Simulated
6	Management Parameters Required by the computational methods specified on card 5
7	System Cost Parameters
8	Simulation Size Specifications

Figure VII-1 presents a print back of a specific set of File 05 input data*. And this figure, labels on the left hand margin identifies the FORTRAN variables associated with each input parameter, while the specific numerical value assumed by this variable is presented on the right. Brief definitions of each of these input variables are presented in Table VII-1.

*Inputs to File 05 are specified in free-field BCD format. See page C-1, Appendix C for an example.

```
INVENTORY SYSTEM SIMULATOR                                                           V    2
                 RUN PARAMETERS

C1)  RUN-ID.     3  TITLE. 100 ITEM SIMULATION WITH ZERO SS

C2)  OUTPUT CONTROLS...(NOTE. 1=YES)
TWRT          IT.WRITE=  0
OUT           SUMMARY =  1
GRAPH         GRAPHS  =  1
PUNCH         PUNCH   =  1

C3)  DEBUG FLAGS
              IDBUG =  0
              IEBUG =  1
              IFBUG =  0
              IGBUG =  0
              IHBUG =  0

TRACE         START TRACE AT     0  FOR ITEM 1
STRAC         STOP TRACE AT      0

C4)  ITEM INPUT FILES
NLU           FILE =  7
NTYPE         TYPE =  2  (1 = BCD; 2 = BINARY)
DEM           QTRS = 24

(C5)  MANAGEMENT METHODS TO BE USED
ICDFOR        FORECAST FORMULA   =  1
ICDSIG        STD DEV  FORMULA   =  1
ICDEOQ        EOQ      FORMULA   =  2
ICDSL         SAFETY STK    CODE =  3
ICDSLL        SAFETY LIMIT CODE  =  1
ICDBG         BUDGET GUIDE CODE  =  1
ICDSR         RETURNS CODE       =  1

(C6)  MANAGEMENT PARAMETERS
EOQMIN        MIN EOQ(MNTHS)             6.0000
EOQMAX        MAX EOQ(MNTHS)            36.0000
SLMIN         MIN SAFETY LV(MNTHS)       0.
SLMAX         MAX SAFETY LV(MNTHS)       0.

(C7)  SYSTEM PARAMETERS
COSHLD        HOLDING COST/$-INV         0.2000
CSHOPT        SHORTAGE COST             10.0000     31.6000    100.0000
COSORD(1)     SMALL ORDER COST         269.8700
COSORD(2)     LARGE ORDER COST         460.2700
CSTORK        COST BREAK-POINT       10000.0000

(C8)  SIMULATION SIZE
NRUN          NUMBER OF RUNS             1
INQTR         NUMBER OF QUARTERS        12
NREPL         NO. OF REPLICATIONS      100
NITEM         NO. OF ITEMS/REPL          1
```

Figure VII-1. Specifications for a 100 Item 12 Quarter Simulation Run.

Table VII-1. Simulation Run Parameters (File 05)

Card Type 1. Run Identification

Variable	Definition
IDENT	Run identification number
TEXT	Description of the current run (up to 40 characters). The description must be included in quote marks.

Card Type 2. Output Controls

Variable	Definition
ITWRT	If ITWRT = 1, call IRSLT to write detailed simulation results for each item to File 08.
ITOUT	If ITOUT = 1, call OUT and OUTCST to print summary statistics for each simulation run.
IGRAPH	If IGRAPH = 1, call PLOTR to plot on hand, on order, requisition and backorder data verses time.
IPUNCH	If IPUNCH = 1, write a brief summary from MAIN after each simulation run.

Card Type 3. Debug Flags

Variable	Definition
IDEBUG	If IDEBUG = 1, write all entries and removals from the Future Events List.
IEBUG	If IEBUG = 1, print all item input data on File 06.
IFBUG	If IFBUG = 1, print item forecast data from FOR576.
IGBUG	If IGBUG = 1, write demand data generated by DEMPAR
IHBUG	If IHBUG = 1, write item levels calculations from LEVELN.

Card Type 4. Item Input File Definition

Variable	Definition
INLU	Logical unit number of item demand data input file.
INTYPE	Format of item input file, where 1 = Free-Field BCD format, 2 = Binary Format

Card Type 5. Inventory Management Codes

Variable	Definition
ICDFOR	Forecast Formula Code used in FOR576 to estimate annual demand rate ADR(N).
ICDSIG	Formula code for estimating the standard deviation of lead time demand RSIGLT(N) in FOR576.
ICDEOQ	EOQ formula code used in LEVELN.
ICDSL	Safety level formula code used in LEVELN.
ICDSLL	Safety level limit code, used to compute limits on the safety level in LEVELN
ICDBG	Budget Guideline code Formula code for forecasting serviceable returns in FOR576.

Card Type 6. Management Parameters

EOQMIN	Minimum EOQ, in months supply.
EOQMAX	Maximum EOQ, in months of supply.
SLMIN	Minimum safety level, in months of supply.
SLMAX	Maximum safety level, in months of supply.

Card Type 7. System Parmeters

COSHLD	Cost of holding one unit in inventory for one year, expressed as a fraction of the item's price.
CSHORT(4)	Implied shortage cost. Four values are provided as input, for use in up to 4 sequential simulation runs.
COSORD(1)	Cost of processing a "small" order, an order whose dollar value is less than CSTBRK.
COSORD(2)	Cost of processing a "large" order.
CSTBRK	Cost break-point used to differentiate between large and small orders.

Card Type 8. Simulation Run Parameters

NRUN	The number of simulation runs to be performed.
INQTR	The number of quarters to be simulated in each run.

Variable	Definition
NREPL	The number of replications to be perfomed within each simulation runs.
NITEM	The number of items to be simulated within each replication.
NDHIS	The number of quarters of information on the item data file to be used to initialize the simulation History File.

Let us now consider each of these input cards in more detail.

CARD 1: RUN IDENTIFICATION

This card identifies the current simulation run. The RUN ID is a numerical identifier which is recorded the detailed simulation results file 08. This identifier makes it possible to identify specific simulation runs using automated post-processing routines. The title simply identifies in words the type of simulation exercise being performed.

CARDS 2 and 3: OUTPUT CONTROLS

These two input cards specify the types of output products which are to be produced during the simulation run. Card 2 specifies summary products to be printed or written to tape files, while card 3 specifies specific debugging messages that are to be printed during the conduct of the simulation. Outputs specified by these parameters are discussed in more detail in the Chapter VIII.

CARD 4: ITEM INPUT FILES

As noted above, item data is entered into INSSIM through File 07. This input may be in one of two different forms: (a) binary format or (b) free-field BCD format. Card 4 specifies which of these options is to be employed in the current simulation run. BCD format is useful for the input of manually prepared data sets for testing of new INSSIM enhancements. On the other hand, binary input is useful for

Table VII-2. Item Input Data (File 07)

Variable	Definition
ALC	Air Logistics Center that manages this item
FSN	Federal Stock Numbers, where FSN(1) = Material Management Code FSN(2) = Federal Stock Class FSN(3) = First six characters of the National Item Identification Number (NIIN) FSN(4) = Remaining 3 characters of the NIIN
UCOST	Unit Cost of item in dollars
UM	Unit of Measure (e.g. each, feet, ounces, etc.)
NOUN	Item name
MGTCD	Item management codes, where MGTCD(1) = Essentiality code MGTCD(2) = Supply management grouping MGTCD(3) = Weapon system code MGTCD(4) = Type computation code
IOH	Initial on-hand assets (units)
IOR	Initial on-order units (also know as "due-ins")
IPPL	Industrial Preparedness Production Leadtime
RIPPPR	Industrial Preparedness Program Ratio
IDEMND (N, J)	Number of units of demand for item N during period J
IRETUR (N, J)	Number of serviceable returns (units) for items N in period J
IREQ (N, J)	Number of requisitions for item N in period J

production runs involving many items and many simulation replications. Binary input is useful in the later case because it generally executes much more rapidly than jobs with BCD inputs.

Appendix A illustrates the format of INSSIM input for binary runs. For binary runs, program DATAB2 is called to reformat this data to a binary equivalent. Hence, DATAB2 serves as a preprocessor for the INSSIM MAIN program.

Figure VII-2 illustrates a sample free-field BCD input file to the model. Note the data is input in the same sequence as shown in Appendix A for binary inputs.

CARDS 5 AND 6: MANAGEMENT METHODS CARDS

Together, cards 5 and 6 specify the calcuation formulas to be used in computing inventory control levels, and specify specific parameters to be used in these calculations. Subroutines FOR576 and LEVELN perform the forecasting and inventory control levels calculations specified by these cards.

The code ICDFOR specifies the calculation to be used in computing inventory position in subroutine STATN. Possible options are defined in AFLC/XRS Working Paper No. 73, by R. J. Stevens, March 1974, and in the COMMENT Statements of STATN. To simulate current D062 calculations, set ICDFOR = 1.

Figure VII-2. Sample BCD Item Input File

(File 07)

```
450$:DATA:07
460 SM 12 34 567890 123 EA 10.00 TEST- ITEM A B C D 0 0 1 1 2 2
470  3 3 3 3 6 6 6 6 350 3 6 6 3 3 6 6                         <--UNIT DEMAND
480  1 1 1 1 1 0 0 0 0   1 1 1 1 0 0 0                         --RETURNS
490  1 1 1 1 2 2 2 2 1 1 2 2 1 1 2 2                           --REQUISITIONS
500 SM 11 55 66 77 EA 1.00 TEST2- ITEM H I J K 0 0 1 1 2 3
510 2 2 2 2 4 4 4 4 133 1 0 0 1 1 0 0                          --UNIT DEMAND
520 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0                            --RETURNS
530 1 1 1 1 2 2 2 2 1 1 0 0  1 1 0 0                           --REQUISITIONS
540#
550$       ENDJOB
```

NOTE: For this data set, these are 16 periods of demand history. Hence, to read this data, Card 4 of File 05 should set INLU = 7, INTYPE = 1, and NDEM = 16.

The code ICDSIG specifies the calculation formulas to be used in computing average annual demand rates ADR(N) and the standard deviation of demand in the lead time RSIGLT(N). These calculations are performed by subroutine FOR576. Figure VII-3 presents the currently coded computational formulas. At present, there are no options to these codes; the calculation specified in Figure VII-3 will always be used.

In contrast to the forecasting formulas, there are several options for EOQ and safety level calculations. These calculations are performed by subroutine LEVELN, and the codes ICDEOQ, ICDSL, and ICDSLL specify the current computational options.

Subroutine LEVELN has several distinct computational stages. The routine first computes an economic order quantity Q according to the formula specified by the code ICDEOQ. Figure VII-4 specifies the computational options. The resulting value of Q is then limited to be no less than EOQMIN months of supply and no more than EOQMAX months of supply. Further, if Q is less than 1, Q is set equal to 1.

With Q computed, LEVELN then computes the safety level, SL, according to the formula specified by the code ICDSL. Figure VII-5 specifies the current computational options. Next, the safety level SL is limited according to the computations specified by code ICDSLL. These options are

Figure VII-3. Forecast and Standard Deviation Calculation Formulas used in Subroutine FOR576.

Quarterly Demand Rate:

$$\text{FORCST} = \sum_{n=1}^{N} \frac{(\text{Gross Demand}_n) - (\text{serviceable returns}_n)}{N}$$

where N equals the number of quarters of available data.

Annual Demand Rate:

ADR(N) = 4. * FORCST

Average Requisition Size:

$$\text{REQSIZ(N)} = \frac{\sum_{n=1}^{N} \text{GROSS DEMANDS}_n}{\sum_{n=1}^{N} \text{FREQUENCIES}_n}$$

Quarterly MAD:

$$\text{MAD}_Q = \sum_{n=1}^{N} \left| \text{Actual Quarterly Demand}_n - 3 * \text{MDR} \right| / N$$

where N number of quarters of data

Standard Deviation of Lead Time Demand

SIG = 0.5945* MADQ* (0.82375 + 0.42625* Leadtime Months)

Figure VII-4. Order Quarterly Calculation Formulas

LET:

Q = Order quarterly
RMR = Monthly demand rate (units)
ADR = Annual demand rate (units)
ADDR = Annual dollar demand rate
COSORD(I) = cost to place an order, where
I = 1 indicates small purchase and
I = 2 indicates large purchase methods
CSTBRK = Dollar breakpoint distinguishing large and small purchase methods
COSHLD = Cost to hold one dollar of stock in inventory for one year
UC = Item unit cost

The code ICDEOQ specifies the following formulas.

ICDEOQ	Calculation	
1	If (ADDR ≥ $5,000),	Q = 3* RMR
	If ($1,000 ≤ ADDR ≤ $5,000)	Q = 6* RMR
	Otherwise	Q = 12* RMR
2	Wilson lot size formula	

$$Q = \sqrt{\frac{2 * COSORD(1) * ADR(M)}{COSHLD * UC}}$$

Provided Q·C ≤ CSTBRK

Otherwise,

$$Q = \sqrt{\frac{2 * COSORD(2) * ADR}{COSHLD * UC}}$$

EOQ Size Limits

For the order quantity Q computed above,

if Q > EOQMAX *RMR, set Q = EOQMAX * RMR
if Q < EOQMIN *RMR, set Q = EOQMIN * RMR
if Q < 1, set Q = 1

Figure VII-5. Safety Level Calculation Formulas

Let
- SL = Safety level
- RMR = Monthly demand rate (units)
- RLT = Expected number of demands in a lead time
- REQSIZ = Average requisition size
- σ = Standard deviation of demand in the lead time

The code ICDSL specifies the following formulas.

ICDSL Value	Calculation
1	Safety Level equals one months supply SL = 1.* RMR
2	Safety level equals 1/4 of expected demand in the lead time SL = .25 * RLT

3 AFLCM 57-6 formula (July 1977)

a) Let $Z = \sqrt{\text{Average Requisition Size}} = \sqrt{\text{REQSIZ}}$

b) Compute K,

$$K = 0.707 \times \mathrm{LN}\left[\frac{\begin{pmatrix}\text{Implied}\\ \text{Shortage}\\ \text{Factor}\end{pmatrix}}{2\cdot\begin{pmatrix}\text{Holding}\\ \text{Cost}\end{pmatrix}\cdot\begin{pmatrix}\text{Unit}\\ \text{Cost}\end{pmatrix}}\cdot\frac{1}{Z}\cdot\frac{\sigma\cdot(1-\mathrm{EXP}(-\sqrt{2}\,\mathrm{EOQ}/\sigma))}{\sqrt{2}\,\mathrm{EOQ}}\right]$$

c) Finally, SL = K * σ

4 Presutti-Trepp formula to minimize expected units backordered.

Set Z = 1. Then use formulas for ICDSL = 3 defined above.

5 Presutti-Trepp formula to minimize expected requisitions backordered.

Set Z = average requisition size = REQSIZ.
Then use formulas for ICDSL = 3.

described in Figure VII-6. Briefly, if ICDSLL equals 2, SL is limited to be no more than SLMIN months of supply, and no more than SLMAX months of supply. On the other hand, if ICDSLL equals 1, SL is limited to be no less than SLMIN months of supply, as before, and no more than the lesser (a) three times the standard deviation of lead time demand, or (b) the expected demand in a lead time.

Finally, the codes ICDBG and ICDSR identify calculations to be employed in setting budget guidelines in predicting future values for serviceable returns. These codes are not currently utilized in the INSSIM model.

Figure VII-6. Safety Level Limit Calculations

Let

SL = Safety level computed according to one of the formulas specified in Figure VII-5.
RMR = Monthly demand rate.
RLT = Expected demand in lead time.
SIG = Standard deviation of lead time demand.
SLMIN = Minimum safety level in months of supply.
SLMAX = Maximum safety level in months of supply.

The the code ICDSLL specifies the following limit calculations.

ICDSLL Value	Calculation
1	If SL < SLMIN * RMR, set SL = SLMIN * RMR
	If SL > RLT, set SL = RLT
	If SL > 3* SIG, set SL = 3 * SIG
2	If SL < SLMIN * RMR, set SL = SLMIN * RMR
	IF SL > SLMAX * RMR, set SL = SLMAX * RMR

Figure VII-7. Levels Calculations Currently Coded

Given the results of the calculations defined in Figures VII-3 through VII-6, the levels for item N are computed as follows, and then rounded to the nearest integer value.

Reorder Level: IROL(N) = RLT + SL

Order Quantity: IROT(N) = Q

Termination Level: ITL(N) = A very large number

Retention Level: IRL(N) = a very lare number

Support Level: ISUL(N) = O.

CARD 7: SYSTEM PARAMETERS

Card 7 defines cost parameters to be used in all simulation runs defined by the current set of parameter cards. The variable COSHLD defines the cost of carrying one dollars worth of stock in inventory for one year. The variable CSHORT defines a shortage cost (a Lagrange multiplier) to be used in up to four different simulation runs. Four different shortage costs must always be specified on this card. The variable COSORD (I) defines the cost of placing an order using the Ith procurement method, where I = 1 denotes small ordering methods, while I = 2 denotes the use of large purchase order methods. Finally, the variable CSTBRK denotes the cost breakpoint, that is, the dollar value of a single buy, that distinguishes between small and large purchase order methods.

CARD 8: SIMULATION SIZE PARAMETERS

This parameter card defines the size as the simulation exercize to be performed. The variable NRUN defines the specific number of simulation runs to be accomplished according to the eight parameter cards in the current set. NRUN cannot exceed four. The variable INQTR defines the number of quarters to be simulated, while NREPL defines the number of replications of the simulation (each for INQTR quarters) to be performed by summary statistics are printed. NITEM defines the number of items to be included in a single inventory simulation replication. With present coding, NITEM should always equal 1. Finally, NDHIS defines the number of quarters of data provided as input that are to be used for initializing the INSSIM history files. At present, NDHIS should not exceed a value of 8. In addition, the total of INQTR and NDHIS cannot exceed the value of NDEM, the total number of quarters of data provided as input to the system.

Figure VII-8 illustrates the relationships among the variables NDHIS, INQTR, and NDEM. The figure illustrates the case in which there are 24 quarters of data read as input through File 07; that is, NDEM = 24. Eight of these data values are used to initialize the Item History File (NDHIS = 8). Finally, in this example the simulation proceeds for 12 quarters, beginning with the ninth quarter of data read from File 07, and ending with the 20th quarter. Data for quarters 21 through 24 are not used in this example.

This concludes our discussion of input parameters to the INSSIM model. In the next chapter, we will discuss output products produced by the simulation model.

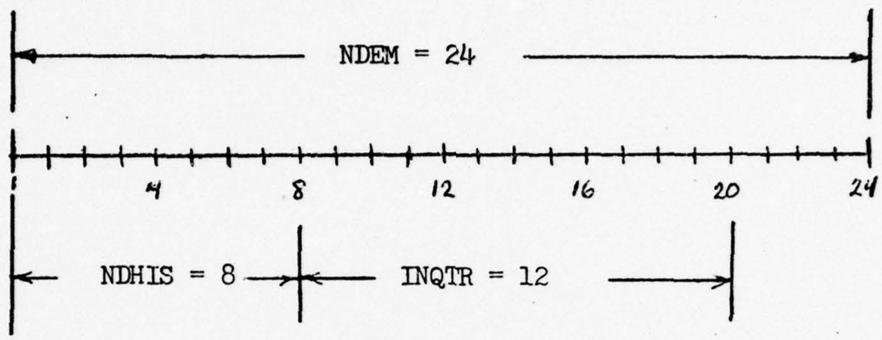


Figure VII-8. Illustration of Relationships Among Variables NDHIS, INQTR, and NDEM.

CHAPTER VIII. OUTPUT PRODUCTS

As discussed in Chapter VII, specific output products produced in a given INSSIM simulation run are specified by input parameters defined on cards 2 and 3. There are two basic categories of outputs: (a) summaries of simulation results, and (b) debugging aids. The latter category of outputs are particularly useful in the development of enhancements to the INSSIM simulation model. Let us now consider each of these types of output products in more detail.

Results Summaries

Result summary products are specified by input parameters defined on card 2. If the parameter ITWRT = 1, detailed results are written as output in binary to File 08 for each item simulated. The format of File 08 is defined in appendix D of this report. This file is particularly useful for conducting involved post-processing calculations using statistical packages such as SPSS or OMNITAB.

Another summary products are the tabular displays and graphical outputs discussed in Chapter III. These products are illustrated in Figures III-9 through III-13 and will not be discussed further here. The tabular summaries are produced when the parameter IOUT = 1, while the graphical outputs are produced when the parameter IGRAPH = 1.

In developing INSSIM enhancements, the voluminous tabular summaries and graphical outputs are usually not necessary. Consequently, another form of summary statistics has been developed; this abbreviated output is illustrated in Figure VII-1. This output product specifies key performance statistics of particular interest in most inventory simulations. This output is produced whenever the parameter IPUNCH = 1.

Let us now discuss debugging options available in the model.

Debugging Aids

Card 3 specifies the values of five debug flags and two "trace" variables that are useful in debugging INSSIM enhancements. Let us consider the meanings of each of these input parameters.

IDBUG. If the input parameter IDBUG = 1, a message will be printed each time an event is entered or removed from the future events list. Consequently, this option will provide a listing of every event that takes place during a given INSSIM simulation run. Figure IV-1 illustrates the format and variable definitions associated with these event messages. In addition, when IDBUG = 1, stock status information is printed each time that subroutine STATN is called. The format of this output is illustrated in Figure VIII-2. As shown in the figure, this output message identifies the number of units that are on-hand,

IDENT	MRUN	REP	J	QTR	IORDER	INVOH	INVOR	IBACKT	IBAKDT	IRETRN	IREJT	FILLT
34	3	120	1	1	60	96	56	8	41	22	277	270
34	3	120	1	2	20	92	62	38	257	23	287	259
34	3	120	1	3	33	92	59	30	430	22	268	241
34	3	120	1	4	25	98	49	8	147	27	284	264
34	3	120	1	5	27	97	44	7	91	20	324	307
34	3	120	1	6	23	97	41	9	89	23	275	264
34	3	120	1	7	25	97	42	7	123	21	258	243
34	3	120	1	8	20	97	38	9	34	22	273	262
34	3	120	1	9	36	95	47	13	153	20	314	294
34	3	120	1	10	30	95	50	10	109	17	295	284
34	3	120	1	11	31	94	53	22	310	20	290	251
34	3	120	1	12	35	93	53	15	233	20	262	236
34	3	120	2	1	1103	9620	7103	45	329	175	2188	2142
34	3	120	2	2	1234	8941	7957	260	1968	302	1976	1745
34	3	120	2	3	1855	9562	6749	360	3967	158	2306	1994
34	3	120	2	4	2692	11257	5584	52	1201	393	2241	2107
34	3	120	2	5	1493	10752	5739	23	525	131	1945	1842
34	3	120	2	6	1633	10839	5425	9	716	151	1997	1783
34	3	120	2	7	1887	10310	6159	99	804	180	1949	1795
34	3	120	2	8	1836	10903	5742	174	729	119	1657	1677
34	3	120	2	9	2494	10359	6514	171	2452	83	2346	2087
34	3	120	2	10	1906	10270	6355	50	819	51	2084	1983
34	3	120	2	11	2084	10390	6301	148	3555	92	2205	1715
34	3	120	2	12	2677	9932	7344	331	3783	83	2361	1978
34	3	120	3	1	666515	1938038	666517	5968	42970	12907	311980	305668
34	3	120	3	2	178508	1727493	825847	30513	237972	19739	266004	239646
34	3	120	3	3	275991	1729097	875055	47280	430666	14894	259748	223572
34	3	120	3	4	235369	1884487	708185	18918	367366	53190	272082	239281
34	3	120	3	5	255745	1799746	747781	4649	131881	12952	300449	284943
34	3	120	3	6	209271	1789000	753056	3439	97842	16059	229480	210202
34	3	120	3	7	301501	1718519	864289	8858	170107	25036	291223	259048
34	3	120	3	8	184804	1785487	767966	9154	35348	17430	231782	221287
34	3	120	3	9	356650	1668453	894169	29566	240840	11733	379508	341255
34	3	120	3	10	293786	1579035	937639	15668	286710	7684	333416	307233
34	3	120	3	11	341316	1561795	1043617	28330	300776	8893	284028	250930
34	3	120	3	12	391427	1603878	1180420	28430	396908	8703	316258	256390

J = 1 ⇒ ACTIONS/FSN's

J = 2 ⇒ UNITS

J = 3 ⇒ DOLLARS

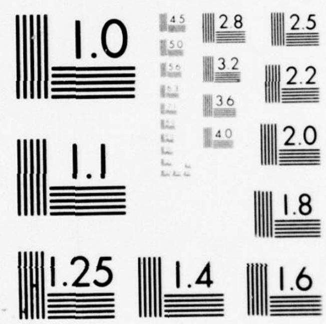
Note: For variable definitions, see Table IV-8.

Figure VIII-1. Selected Summary Statistics Printed When IPUNCH=1

AD-A046 578 DECISION SYSTEMS DAYTON OH F/G 15/5
THE INVENTORY SYSTEM SIMULATOR, (INSSIM). VOLUME I. MODEL DESCR--ETC(U)
AUG 77 W S DEMMY F49620-77-C-0063
UNCLASSIFIED RM-77-02 AFOSR-TR-77-1244-VOL-1 NL

2 OF 2
AD
A046578

END
DATE
FILMED
12-77
DDC



MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

If IDBUG = 1

```
****STATN--UNITS OH=   42  DUE-IN=    0  BO=    0  INPOS=   42  REQ-SHORT=    0
```

If IEBUG = 1

```
    1 OC    16500262550   EA      168.50 CYL ASSY       5 316ZB          1     41     3      4      8 0.
 IDEMND  DEMAND/QTR     4       4       1       3       0       4       4       3       0       0
                        1       4       0       2       0       0       5       0       0       0
                        0       0       0       0
 IRETUR  RETURN/QTR     0       0       0       0       0       0       0       0       0       0
                        0       0       0       0       0       0       0       0       0       0
                        0       0       0       0
 IREQ    REQ   /QTR     2       2       1       3       0       2       1       3       0       1
                        0       0       0       0       0       0       0       0       0       0
                        0       0       0       0
```

If IFBUG = 1

```
****FOR576--N,NDENT,NLOOP,ICDFOR,ICDSIG, LTADM,LIPROD
     1       8      24       1       1       3       4
     DEMAND, RETURNS, AND FREQ IN CURRENT HISTORY FILE
     0       3       4       4       0       3       1       4
     0       0       0       0       0       0       0       0
     0       3       1       2       0       3       1       2
FORCST      ADR     AVE-REQ    UCOST       OMAD      RSIGLT
2.8732    11.4929    1.5833   168.5000    1.1884    2.6990
```

If IEBUG = 1

```
REMOVE        100        12        0        0        0        6
DEMPAR-----       YEAR NO.  1     QTR NO.  1     WEEK NO.  1
   ****DEMPAR--N=     1   NDENT=     8  IDPER=    1  IDEMND=    9  IRETRN=     0  IREQ=     2
   ****DEMPAR--IT=      6435   N=   1   R=0.387   INITAL REQ-SIZE=   2  FINAL REQ-SIZE=   2
ENTER        100                                          6435      1        1       2        1       6       3       7     695
   ****DEMPAR--IT=      8361   N=   1   R=0.308   INITAL REQ-SIZE=   1  FINAL REQ-SIZE=   1
ENTER        100                                          8361      1        1       1        1       8       3       8     695
   ****DEMPAR--IT=      1165   N=   1   R=0.529   INITAL REQ-SIZE=   2  FINAL REQ-SIZE=   2
ENTER        100                                          1165      1        1       2        1       9       3       9     695
   ****DEMPAR--IT=      1916   N=   1   R=0.229   INITAL REQ-SIZE=   1  FINAL REQ-SIZE=   1
ENTER        100                                          1916      1        1       1        1      10       3      10     695
   ****DEMPAR--IT=      7890   N=   1   R=0.096   INITAL REQ-SIZE=   1  FINAL REQ-SIZE=   1
ENTER        100                                          7890      1        1       1        1      11       3      11     695
   ****DEMPAR--IT=      4707   N=   1   R=0.986   INITAL REQ-SIZE=  10  FINAL REQ-SIZE=   2
ENTER        100                                          4707      1        1       2        1      12       3      12     695
ENTER        100                                          8500     12        1       0        0      13       3      13     695
```

If IHBUG = 1

```
****LEVELN--N=    1  IRUTY=   14  IROL=    7  ITL=99999  IRL=99999  ISUL=    0
```

Figure VIII-2. Intermediate Results Output Options due-in, and in a backorder status. The inventory position (INPOS) is simply equal to the sum of on-hand plus due-in assets, less any outstanding backorders. The variable REQ-SHORT identifies the number of requisitions that are backordered at that specific point in time.

IEBUG. If the parameter IEBUG = 1, subroutine INITEM writes the details of all item demand data input from File 07. The format of this output message is illustrated in Figure III-2, and corresponds exactly with the data layout specified in appendix A.

IFBUG. The IFBUG parameter causes the printing of forecasting information, as well as the printing of demands, serviceable returns, and requisition frequency data recorded in the current history file. The format of this information is illustrated in Figure VII-2.

IGBUG. The IGBUG parameter causes the printing of requisition generation events created by subroutine DEMPAR. Figure VII-2 illustrates a specific printout that occurred when the flags IDBUG = IGBUG = 1. Note from the figure that at time 100, a type 12 event (i.e., a DEMPAR event) is removed from the future events list. When subroutine DEMPAR is called, the associated number of demands (IDEMMD) to be generated is nine, as is circled in the figure. The requisition generation process discussed in Chapter VI is then initiated. The first requisition generated is for a requistion size of 2 units. Note that this requisition for two units (event code 1) is entered on the future events list to occur at time 6435. Note that the second requisition is for one unit, and that this second requisition is entered onto the future events list to occur at time 8361. Subsequently, requisitions for two, one, one, and finally two units are generated in succession and placed onto the future events list. Note that the total of all of these generated requisitions equals nine units, the total number of units that were present in the demands history record for the current item. Finally, note that a type 12 event is placed on the future events list to occur at time 8500. This will cause the next CALL to subroutine DEMPAR.

IHBUG. The IHBUG parameter specifies the printing of levels that are computed by subroutine LEVELN. The format of the associated printout is illustrated at the bottom of Figure VII-2. As shown in the figure, the levels for item number 1 involve an order quantity of 14 units, and a reorder level of 7 units. For this item, termination and retention levels are set to very large numbers (99999), and the support level is set to zero.

ITRACE, ISTRAC. At times, the analyst will not want to produce the many lines of print that result when IDBUG = 1, although he may be interested in obtaining an event by event printout during a specific time interval in the simulation. The flags ITRACE and ISTRAC provide this capability. The variable ITRACE specifies the point in simulated time that a detailed trace of simulation events is to start. At that point in time, the flag IDBUG is set equal to 1. This causes the event by event printout to start. The flag remains equal to 1 until the simulated clock time ISTRAC. At this time, IDBUG is reset to zero, and the detailed event by event printouts cease.

This concludes our discussion of INSSIM output products. In the next chapter, we will discuss the job control language statements required to assemble and execute INSSIM simulation runs.

CHAPTER IX

JOB CONTROL CARDS

Execution of an INSSIM simulation run is controlled by a set of Job Control Language statement cards. These cards cause the appropriate subroutines to be acquired from CREATE permanent files and assembled into an executable package. These routines also assure that input files are read from the appropriate physical devices, and that output products are directed to appropriate locations.

We have found there are three basic job streams that are useful in the conduct of an INSSIM simulation study. These job streams are documented in Appendix C of this volume. Briefly, these job streams are as follows:

INSSIM.T	This job stream is used for testing of INSSIM enhancements, and for debugging newly developed subroutines.
INSSIM.A	This job stream provides for the printing of all INSSIM input data, and produces tabular summaries of simulation performance statistics. Graphical displays of these statistics as a function of simulated time are also produced.
INSSIM.B	This job stream is used for performing several simulation runs at a single time. The job stream is particularly useful in developing cost effectivenesss curves for alternate inventory management policies.

In performing a given INSSIM.T study, it is usually useful to begin with the INSSIM.T job stream. This job stream would be used for testing of new code that defines forecasting or inventory control policies that are to be evaluated. In this stage, the detailed debugging statements defines on input card 4 will probably be used extensively. The INSSIM.T job stream will usually utilize File 07 input data in free-field BCD format. No pre-processing step is necessary when data is input in this form.

Once new calculation formulas have been fully debugged, the INSSIM.A and INSSIM.B job streams are employed. The INSSIM.A job stream is useful for initial testing of data to be used in production runs. The graphical outputs of INSSIM.A are particularly useful in determining whether there are "weird" data elements that are included in the proposed data set. Once a data set has been identified that appears suitable for simulation study, the INSSIM.B job stream is utilized. This job stream is designed to provide a large number of simulation runs at the lowest possible computer cost.

Both the INSSIM.A and INSSIM.B job streams utilize a pre-processing step. In these job streams, subroutine DATAB2 reads input from File 07, and writes the same data in binary format to File 09. This latter file then becomes input to the INSSIM MAIN program. In performing production runs, there will usually be a large amount of data to be read in through File 07. By converting this information to binary, the data processing steps may be speeded up considerably.

APPENDIX A.

INSSIM ITEM DATA FILE

(FILE Ø7)

RECORD LAYOUTS

INPUT /OUTPUT RECORD

SYSTEM NUMBER

TITLE	INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE
INSSIM ITEM DATA RECORD (RECORD TYPE = 1)	I	07		

DATA RECORD LENGTH: 20 WORDS

TAPE RECORD LENGTH/ BLOCKING FACTOR:

REMARKS: FILE 07 IS A BINARY SEQUENTIAL FILE, AND INPUT FOR EACH FSN CONSISTS OF A SET OF RECORD TYPES 1, 2, 3, AND 4, READ IN THAT ORDER.

NO.	DATA ELEMENT NAME	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
1	ALC	ALC	1	2	AN
2	FEDERAL STOCK NUMBER (FSN)				
	MMC	FSN(1)	2	2	AN
	FSC	FSN(2)	3	4	AN
	NIIN	FSN(3) & FSN(4)	4 5	6 3	AN AN
3	RECORD TYPE = 1**	IR	6	1	N
4	UNIT OF MEASURE	UM	7	2	AN
5	UNIT PRICE (FORMAT F9.2)	UCOST	8	9	N
6	NOUN	NOUN	9 10	10	AN
7	PROCESSING CODES				
	ITEM ESSENTIALITY	MGTCD(1)	11	1	N
	SUPPLY MANAGEMENT GROUPING	MGTCD(2)	12	1	AN
	WEAPON SYSTEM CODE	MGTCD(3)	13	4	AN
	TYPE COMPUTATION	MGTCD(4)	14	2	AN
8	ON-HAND ASSETS	IOH	15	7	N
9	ON-ORDER ASSETS	IOR	16	7	AN
10	ADMINISTRATIVE LEAD TIME (MONTHS)	LTADM	17	2	N
11	PRODUCTION LEAD TIME (MONTHS)	LTPROD	18	2	N
12	IPP LEAD TIME (MONTHS)	IPPL	19	2	N
13	IPP PROGRAM RATIO (F4.2)	RIPPPR	20	4	N

** NOTE: FOR ACD INPUT, DATA ELEMENT NO 3 (RECORD TYPE) IS NOT READ IN

INPUT /OUTPUT RECORD

SYSTEM NUMBER

TITLE: INSSIM DEMAND RECORD (RECORD TYPE = 2)

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE
I	Ø7		

DATA RECORD LENGTH: 30 BINARY WORDS

TAPE RECORD LENGTH/ BLOCKING FACTOR:

REMARKS: SEE BELOW

NO.	DATA ELEMENT NAME	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
1	ALC**	ALC	1	2	AN
2	FEDERAL STOCK NUMBER**				
	MMC	FSN(1)	2	2	AN
	FSC	FSN(2)	3	4	AN
	NIIN	FSN(3) & FSN(4)	4 5	9	AN
3	RECORD TYPE = 2**	IR	6	1	N
4	TRANSFER + SALES DEMAND				
	QTR 1 (I7)	IDEMND(N,1)	7	7	N
	QTRS 2, 3, ..., THRU 24 (23I7)	IDEMND(N,J), J=2,...,24	8 THRU 30		N

** FOR BCD INPUT, DATA IS FREE-FIELD, AND DATA ELEMENTS 1, 2, AND 3 ARE NOT TO BE IN THE INPUT STREAM.

INPUT /OUTPUT RECORD		SYSTEM NUMBER		PAGE 3 OF 4 PAGES
TITLE INSSIM SERVICEABLE RETURNS RECORD (RECORD TYPE =3)	INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE
DATA RECORD LENGTH: 30 BINARY WORDS — TAPE RECORD LENGTH/ BLOCKING FACTOR:	I	Ø7		
REMARKS: SEE BELOW				

NO.	DATA ELEMENT NAME	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
1	ALC **	ALC	1	2	AN
2	FEDERAL STOCK NUMBER **				
	MMC	FSN(1)	2	2	AN
	FSC	FSN(2)	3	4	AN
	NIIN	FSN(3) & FSN(4)	4 5	9	AN
3	RECORD TYPE = 3 **	IR	6	1	N
4	TRANSFER + SALES SERVICEABLE RETURNS				
	QTR 1 (I7)	IRETUR(N,1)	7	7	N
	QTRS 2, 3, ..., THRU 24 (23 I7)	IRETUR(N,J), J=2,...,24	8 THRU 30		N

** FOR BCD INPUT, DATA IS FREE-FIELD, AND DATA ELEMENTS 1, 2, AND 3 ARE NOT TO BE IN THE INPUT STREAM.

INPUT /OUTPUT RECORD	SYSTEM NUMBER	PAGE 4 OF 4 PAGES

TITLE	INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE
INSSIM REQUISITION FREQUENCY RECORD (RECORD TYPE = 4)	I	Ø7		

DATA RECORD LENGTH: 30 BINARY WORDS

TAPE RECORD LENGTH/ BLOCKING FACTOR:

REMARKS:

SEE BELOW

NO.	DATA ELEMENT NAME	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
1	ALC**	ALC	1	2	AN
2	FEDERAL STOCK NUMBER**				
	MMC	FSN(1)	2	2	AN
	FSC	FSN(2)	3	4	AN
	NIIN	FSN(3) & FSN(4)	4 5	9	AN
3	RECORD TYPE = 4**	IR	6	1	N
4	FREQUENCY OF DEMANDS -- BY QTR				
	QTR 1 (I7)	IREQ(N,1)	7	7	N
	QTRS 2, 3, ..., THRU 24 (23I7)	IREQ(N,J), J=2,...,24	8 THRU 30	161	N

** FOR BCD INPUT, DATA IS FREE-FIELD, AND DATA ELEMENTS 1, 2, AND 3 ARE NOT TO BE IN THE INPUT STREAM.

APPENDIX B.

INSSIM SIMULATION RESULTS FILE

(FILE Ø8)

RECORD LAYOUTS

INPUT /OUTPUT RECORD	SYSTEM NUMBER INSSIM	PAGE 1 OF 2 PAGES

TITLE SIMULATION RESULTS RECORD

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE
O	Ø8		

DATA RECORD LENGTH

TAPE RECORD LENGTH/ BLOCKING FACTOR

REMARKS ALL INSSIM INTEGER VARIABLES LISTED ARE CONVERTED TO FLOATING POINT VARIABLES, AND OUTPUT IN BINARY FORMAT.

NO.	DATA ELEMENT NAME	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
1	RUN IDENTIFICATION	IDENT	1	4	N
2	SAFETY LEVEL FORMULA CODE	ICDSL	2	1	N
3	RUN NUMBER	MRUN	3	1	N
4	ALC	ALS	4	2	AN
5	FEDERAL STOCK NUMBER				
	MMC	FSN(1)	5	2	AN
	FSC	FSN(2)	6	4	AN
	NIIN	FSN(3) FSN(4)	7 8	6 3	AN AN
6	MANAGEMENT CODES				
	ITEM ESSENTIALITY CODE	MGTCD(1)	9	1	N
	SUPPLY MANAGEMENT GROUPING CODE	MGTCD(2)	10	1	AN
	WEAPON SYSTEM CODE	MGTCD(3)	11	4	AN
	TYPE COMPUTATION CODE	MGTCD(4)	12	2	AN
7	UNIT PRICE	UCOST(1)	13	9	F9.2
8	TOTAL LEAD TIME (ADMINISTRATIVE LT + PRODUCTION LT)	LTADM(1)+ LTPROD(1)	14	7	N
9	INITIAL ON-HAND + DUE-IN INVENTORY (UNITS)	IOH+IOR	15	7	N

INPUT /OUTPUT RECORD	SYSTEM NUMBER INSSIM	PAGE 2 OF 2 PAGES

TITLE SIMULATION RESULTS RECORD (CONTINUED)	INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE
DATA RECORD LENGTH / TAPE RECORD LENGTH/ BLOCKING FACTOR				
REMARKS				

NO.	DATA ELEMENT NAME	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
10	REQUISITIONS BY QTR*	IREQT(I,1)			
11	DEMAND (UNITS) BY QTR*	IREQT(I,2)			
12	RETURNS (UNITS) BY QTR*	IRETRN(I,2)			
13	REQUISITION BACKORDER-WEEKS BY QTR*	IBAKDT(I,1)			
14	UNIT BACKORDER-WEEKS BY QTR*	IBAKDT(I,2)			
15	BUY-UNITS BY QTR*	IORDER(I,2)			
16	INVENTORY UNIT-WEEKS BY QTR*	INVDAY(I,2)			
17	FILLS -- REQUISITIONS -- BY QTR*	IFILLT(I,1)			
18	FILLS -- UNITS -- BY QTR*	IFILLT(I,2)			
19	TOTAL COST OF ORDERING -- in $ -- BY QTR* = COSORD(1)*ISMORD(I,1) + COSORD(2)*ILGORD(I,1)	—			
	* FOR QTRS 1, 2, ..., THRU INQTR, THE NUMBER OF SIMULATED PERIODS				

APPENDIX C.

JOB CONTROL PROGRAMS

```
20     $     IDENT    WP1279,XRS/DEMMY         INSSIM.A
30     $     USERID   REQS$IZE
35     $     LIMITS   05,,,5K
40     $     OPTION   FORTRAN,NOMAP
50     $     SELECT   REQS/DATAB2.O
55     $     SELECT   REQS/DEMPAR.O
56     $     SELECT   REQS/RANDU.O
60     $     EXECUTE
70     $     PRMFL    07,R,S,REQS/SIM.D1
80     $     FILE     09,XIS
85     $     OPTION   FORTRAN,NOMAP
90     $     SELECT   REQS/MAIN.O
100    $     SELECT   REQS/STATUS.O
110    $     SELECT   REQS/DEMPAR.O
120    $     SELECT   REQS/ENTERB.O
130    $     SELECT   REQS/FILLBO.O
140    $     SELECT   REQS/FOR576.O
150    $     SELECT   REQS/FORUPD.O
160    $     SELECT   REQS/INITAL.O
170    $     SELECT   REQS/INITEM.O
180    $     SELECT   REQS/LEVEL.O
190    $     SELECT   REQS/LEVELN.O
200    $     SELECT   REQS/ORDER.O
210    $     SELECT   REQS/OUTCST.O
220    $     SELECT   REQS/RECEIV.O
230    $     SELECT   REQS/RET.O
240    $     SELECT   REQS/ZERO.O
250    $     SELECT   REQS/ENTER.O
260    $     SELECT   REQS/SSTAT.O
270    $     SELECT   REQS/OUT.O
280    $     SELECT   REQS/REQ.O
290    $     SELECT   REQS/REMOVE.O
300    $     SELECT   REQS/WRIFEL.O
310    $     SELECT   REQS/STATN.O
320    $     SELECT   REQS/CUM.O
330    $     SELECT   REQS/CUMB.O
340    $     SELECT   REQS/INFEL.O
350    $     SELECT   REQS/ITRSLT.O
360    $     SELECT   REQS/RANDU.O
370    $     SELECT   REQS/PLOTR.O
380    $     SELECT   REQS/GP.O
390    $     EXECUTE
400    $     LIMITS   05,30K,,10K
410    $     REMOTE   P*,AC
420    $     REMOTE   $$,AC
430    $     DATA     05
440     3  "100 ITEM SIMULATION WITH ZERO SS"
450     0 1 1 1     C2
460     0 1 0 0 0     0  0     C3
470     7 2 24
480     1 1 2 3 1 1 1      C5
490     6.  36.  0.  0.    C6
500       .20  10. 31.6 100. 316. 269.87 460.27  10000.
510       1  12  100  1  8      C8
520    $     FILE     07,XIR
530    $     ENDJOB
```

FILE 05 DATA

```
20    $    IDENT     WP1279,XRS/DEMMY     INSSIM.B
30    $    USERID    REQS$IZE
40    $    LIMITS    50,,,10K
50    $    OPTION    FORTRAN,NOMAP
60    $    SELECT    REQS/DATAB2.O
70    $    SELECT    REQS/DEMPAR.O
80    $    SELECT    REQS/RANDU.O
90    $    EXECUTE
100   $    PRMFL     07,R,S,REQS/SIM.D1
110   $    FILE      09,X1S
120   $    OPTION    FORTRAN,NOMAP
130   $    SELECT    REQS/MAIN.O
140   $    SELECT    REQS/STATUS.O
150   $    SELECT    REQS/DEMPAR.O
160   $    SELECT    REQS/ENTERB.O
170   $    SELECT    REQS/FILLBO.O
180   $    SELECT    REQS/FOR576.O
190   $    SELECT    REQS/FORUPD.O
200   $    SELECT    REQS/INITAL.O
210   $    SELECT    REQS/INITEM.O
220   $    SELECT    REQS/LEVEL.O
230   $    SELECT    REQS/LEVELN.O
240   $    SELECT    REQS/ORDER.O
250   $    SELECT    REQS/OUTCST.O
260   $    SELECT    REQS/RECEIV.O
270   $    SELECT    REQS/RET.O
280   $    SELECT    REQS/ZERO.O
290   $    SELECT    REQS/ENTER.O
300   $    SELECT    REQS/SSTAT.O
310   $    SELECT    REQS/OUT.O
320   $    SELECT    REQS/REQ.O
330   $    SELECT    REQS/REMOVE.O
340   $    SELECT    REQS/WRIFEL.O
350   $    SELECT    REQS/STATN.O
360   $    SELECT    REQS/CUM.O
370   $    SELECT    REQS/CUMB.O
380   $    SELECT    REQS/INFEL.O
390   $    SELECT    REQS/ITRSLT.O
400   $    SELECT    REQS/RANDU.O
410   $    SELECT    REQS/PLOTR.O
420   $    SELECT    REQS/GP.O
430   $    EXECUTE
440   $    LIMITS    50,30K,,10K
450   $    REMOTE    P*,AC
460   $    REMOTE    $$,AC
470   $    DATA      05
480          22   100 ITEMS  ---
490         1 1 0 0       C2
500  DATA   0 0 0 0 0      0 0      C3
510  SET    7 2 24
520  #1     1 1 2 3 1 1 1         C5
```

```
6.  36.  0.  0.      C6
 .20  10. 31.6 100. 316. 269.87 460.27  10000.
1 12 100 1 8            C8
23   100 ITEMS ---
1 1 0 0          C2
0 0 0 0 0     0  0       C3
```

DATA SET #2

```
7 2 24
1 1 2 3 1 1 1          C5
6.  36.  0.  40.     C6
 .20  10. 31.6 100. 316. 269.87 460.27  10000.
4 12 100 1 8                C8
33   100 ITEMS ---
1 1 0 0          C2
0 0 0 0 0     0  0       C3
```

DATA SET #3

```
7 2 24
1 1 2 3 1 1 1          C5
6.  36.  0.  40.     C6
 .20  600. 1000. 3162. 10000. 269.87 460.27  10000.
4 12 100 1 8                C8
 24   100 ITEMS ---
1 1 0 0          C2
```

DATA SET #4

```
0 0 0 0 0     0  0       C3
7 2 24
1 1 2 4 1 1 1          C5
6.  36.  0.  40.     C6
 .20  10. 31.6 100. 316. 269.87 460.27  10000.
4 12 100 1 8                C8
 34   100 ITEMS ---
1 1 0 0          C2
```

DATA SET #5

```
0 0 0 0 0     0  0       C3
7 2 24
1 1 2 4 1 1 1          C5
 6.  36.  0.  40.     C6
 .20  600. 1000. 3162. 10000. 269.87 460.27  10000.
4 12 100 1 8                C8
 25   100 ITEMS ---
1 1 0 0          C2
0 0 0 0 0     0  0       C3
```

DATA SET #6

```
7 2 24
1 1 2 5 1 1 1          C5
6.  36.  0.  40.     C6
 .20  10. 31.6 100. 316. 269.87 460.27  10000.
4 12 100 1 8                C8
 35   100 ITEMS ---
1 1 0 0          C2
```

DATA SET #7

```
0 0 0 0 0     0  0       C3
7 2 24
1 1 2 5 1 1 1          C5
6.  36.  0.  40.     C6
 .20  600. 1000. 3162. 10000. 269.87 460.27  10000.
4 12 100 1 8           C8
$       FILE     07,X1R
$       TAPE     08,X2D,,75985,,SMRSLTS/RING
$       ENDJOB
```

INSSIMA.T

```
$      IDENT     WP1279,XRS/DEMMY        INSSIM.T
$      USERID    REQS$IZE
$      OPTION    FORTRAN,NOMAP
$      SELECT    REQS/MAIN.O
$      SELECT    REQS/STATUS.O
$      SELECT    REQS/DEMPAR.O
$      SELECT    REQS/ENTERB.O
$      SELECT    REQS/FILLBO.O
$      SELECT    REQS/FOR576.O
$      SELECT    REQS/FORUPD.O
$      SELECT    REQS/INITAL.O
$      SELECT    REQS/INITEM.O
$      SELECT    REQS/LEVEL.O
$      SELECT    REQS/LEVELN.O
$      SELECT    REQS/ORDER.O
$      SELECT    REQS/OUTCST.O
$      SELECT    REQS/RECEIV.O
$      SELECT    REQS/RET.O
$      SELECT    REQS/ZERO.O
$      SELECT    REQS/ENTER.O
$      SELECT    REQS/SSTAT.O
$      SELECT    REQS/OUT.O
$      SELECT    REQS/REQ.O
$      SELECT    REQS/REMOVE.O
$      SELECT    REQS/WRIFEL.O
$      SELECT    REQS/STATN.O
$      SELECT    REQS/CUM.O
$      SELECT    REQS/CUMB.O
$      SELECT    REQS/INFEL.O
$      SELECT    REQS/ITRSLT.O
$      SELECT    REQS/RANDU.O
$      EXECUTE
```

```
320      $       LIMITS  10,25K,,5K
330      $       REMOTE  09,AC
340      $       REMOTE  P*,AC
350      $       REMOTE  $$,AC
360      $       DATA    05
370       6 'INSSIM.T DEMPAR TEST RUN'
380       0 0 0 1      C2
390       1 1 0 1 0    0      0      C3
400       7 1 16
410       1 1 2 3 1 1 1        C5
420       6.  36.  0.  40.     C6
430       .20  10. 50. 100. 400.  269.87  460.27  10000.    C7
440       1 2 1 1 8     C8
450      $       DATA    07
460       SM 12 34 567890 123 EA 10.00 TEST- ITEM A B C D 0 0 1 1 2 2
470        3 3 3 3 6 6 6 6 350 3 6 6 3 3 6 6
480        1 1 1 1 1 0 0 0 0   1 1 1 1 0 0 0
490        1 1 1 1 2 2 2 2 1 1 2 2 1 1 2 2
500       SM 11 55 66 77 EA 1.00 TEST2- ITEM H I J K 0 0 1 1 2 3
510       2 2 2 2 4 4 4 4 133 1 0 0 1 1 0 0
520       0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
530       1 1 1 1 2 2 2 2 1 1 0 0  1 1 0 0
540
550      $       ENDJOB
```

```
$       IDENT    WP1279,XRS/DEMMY      INSSIM.C
$       USERID   REQS$IZE
$       LIMITS   06,,,10K
$       OPTION   FORTRAN,NOMAP
$       SELECT   REQS/DEMPAR.O
$       SELECT   REQS/RANDU.O
$       SELECT   REQS/DATAB2.O
$       EXECUTE
$       PRMFL    07,R,S,REQS/SIM.DI
$       FILE     09,XIS
$       OPTION   FORTRAN,NOMAP
$       SELECT   REQS/MAIN.O
$       SELECT   REQS/STATUS.O
$       SELECT   REQS/DEMPAR.O
$       SELECT   REQS/ENTERB.O
$       SELECT   REQS/FILLBO.O
$       SELECT   REQS/FOR576.O
$       SELECT   REQS/FORUPD.O
$       SELECT   REQS/INITAL.O
$       SELECT   REQS/INITEM.O
$       SELECT   REQS/LEVEL.O
$       SELECT   REQS/LEVELN.O
$       SELECT   REQS/ORDER.O
$       SELECT   REQS/OUTCST.O
$       SELECT   REQS/RECEIV.O
$       SELECT   REQS/RET.O
$       SELECT   REQS/ZERO.O
$       SELECT   REQS/ENTER.O
$       SELECT   REQS/SSTAT.O
$       SELECT   REQS/OUT.O
$       SELECT   REQS/REQ.O
$       SELECT   REQS/REMOVE.O
$       SELECT   REQS/WRIFEL.O
$       SELECT   REQS/STATN.O
$       SELECT   REQS/CUM.O
$       SELECT   REQS/CUMB.O
$       SELECT   REQS/INFEL.O
$       SELECT   REQS/ITRSLT.O
$       SELECT   REQS/RANDU.O
$       SELECT   REQS/PLOTR.O
$       SELECT   REQS/GP.O
$       EXECUTE
$       LIMITS   06,30K,,10K
$       REMOTE   P*,AC
$       REMOTE   $$,AC
$       DATA     05
  41   '100 ITEMS'
 0 1 0 0        C2
 0 0 0 0 0      0 0      C3
 7 2 24
 1 1 2 5 1 1 1         C5
  6.  36.  0.  40.
 .20 1000. 31.6 100. 316.  269.87 460.27 10000.
 1 12 100 1 8           C8
$       FILE     07,X1R
$       ENDJOB
```

INSSIM.D

```
20      $      IDENT     WP1279,XRS/DEMMY     INSSIM.D
30      $      USERID    REQS$IZE
40      $      LIMITS    25,,,1ØK
5Ø      $      OPTION    FORTRAN,NOMAP
6Ø      $      SELECT    REQS/DATAB2.0
7Ø      $      SELECT    REQS/DEMPAR.0
8Ø      $      SELECT    REQS/RANDU.0
9Ø      $      EXECUTE
1ØØ     $      TAPE      Ø7,X3D,,71515,,SIM.D2
11Ø     $      FILE      Ø9,XIS
12Ø     $      OPTION    FORTRAN,NOMAP
13Ø     $      SELECT    REQS/MAIN.0
14Ø     $      SELECT    REQS/STATUS.0
15Ø     $      SELECT    REQS/DEMPAR.0
16Ø     $      SELECT    REQS/ENTERB.0
17Ø     $      SELECT    REQS/FILLBO.0
18Ø     $      SELECT    REQS/FOR576.0
19Ø     $      SELECT    REQS/FORUPD.0
2ØØ     $      SELECT    REQS/INITAL.0
21Ø     $      SELECT    REQS/INITEM.0
22Ø     $      SELECT    REQS/LEVEL.0
23Ø     $      SELECT    REQS/LEVELN.0
24Ø     $      SELECT    REQS/ORDER.0
25Ø     $      SELECT    REQS/OUTCST.0
26Ø     $      SELECT    REQS/RECEIV.0
27Ø     $      SELECT    REQS/RET.0
28Ø     $      SELECT    REQS/ZERO.0
29Ø     $      SELECT    REQS/ENTER.0
3ØØ     $      SELECT    REQS/SSTAT.0
31Ø     $      SELECT    REQS/OUT.0
32Ø     $      SELECT    REQS/REQ.0
33Ø     $      SELECT    REQS/REMOVE.0
34Ø     $      SELECT    REQS/WRIFEL.0
35Ø     $      SELECT    REQS/STATN.0
36Ø     $      SELECT    REQS/CUM.0
37Ø     $      SELECT    REQS/CUMB.0
38Ø     $      SELECT    REQS/INFEL.0
39Ø     $      SELECT    REQS/ITRSLT.0
4ØØ     $      SELECT    REQS/RANDU.0
41Ø     $      SELECT    REQS/PLOTR.0
42Ø     $      SELECT    REQS/GP.0
43Ø     $      EXECUTE
44Ø     $      LIMITS    5Ø,3ØK,,1ØK
45Ø     $      REMOTE    P*,AC
46Ø     $      REMOTE    $$,AC
47Ø     $      DATA      Ø5
```

```
33  '100 ITEMS  ---'
1 1 Ø Ø        C2
Ø Ø Ø Ø Ø    Ø  Ø     C3
7 2 24
1 1 2 3 1 1 1       C5
6.  36.  Ø.  40.    C6
 .2Ø  6ØØ. 1ØØØ. 3162. 1ØØØØ. 269.87 46Ø.27  1ØØØØ.
4 12 1ØØ 1 8            C8
 34  '1ØØ ITEMS ---'
1 1 Ø Ø        C2
Ø Ø Ø Ø Ø    Ø  Ø     C3
7 2 24
1 1 2 4 1 1 1     . C5
6.  36.  Ø.  4Ø.    C6
 .2Ø  6ØØ. 1ØØØ. 3162. 1ØØØØ. 269.87 46Ø.27  1ØØØØ.
4 12 1ØØ 1 8            C8
$       FILE    Ø7,X1R
$       TAPE    Ø8,X2D,,732Ø5,,RSL.II/RING
$       ENDJOB
```