AD NUMBER

ADB007214

LIMITATION CHANGES

TO:

Approved for public release; distribution is unlimited.

FROM:

Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; JUL 1975. Other requests shall be referred to Aeronautical Systems Division, Attn: YHCD, Wright-Patterson AFB, OH 45433.

AUTHORITY

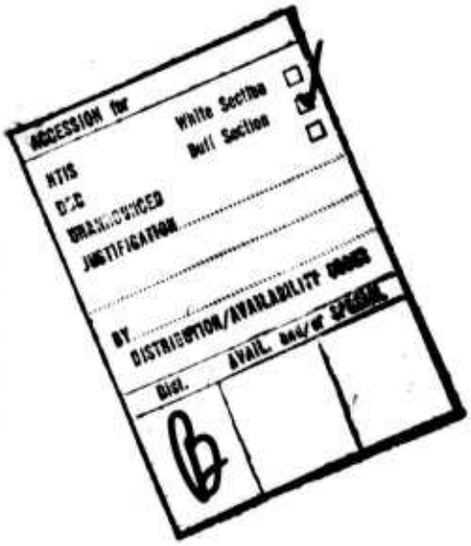
ASD ltr, 11 Jul 1977

(2)

Calspan

Technical Report

Calspan Corporation
Buffalo, New York 14221


ACCESSION for
White Section
Buff Section
NTIS
DDC
UNANNOUNCED
JUSTIFICATION
BY
DISTRIBUTION/AVAILABILITY CODES
Dist. AVAIL. and/or SPECIAL
A

Calspan

B-1 SYSTEMS APPROACH TO TRAINING
TECHNICAL MEMORANDUM SAT-5

TRAINING RESOURCES ANALYTIC MODEL (TRAM)
USER'S MANUAL

JULY 1975

PREPARED BY: William F.H. Ring
William F.H. Ring

George Galdasz
George Galdasz

John R. Menig
John R. Menig

Walter L. Stortz
Walter L. Stortz

APPROVED BY: Robert C. Sugarman
Robert C. Sugarman
SAT Task Manager

APPROVED BY: D. Barry Dahm
D. Barry Dahm, Head
Environmental & Energy
Systems Department

APPROVED BY: Richard J. Taylor for
Robert C. Kidder
Program Manager
B-1 Technical Support Program

CALSPAN CORPORATION
CONTRACT NO. F33657-75-C-0021

Calspan Corporation
Buffalo, New York 14221

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)

B-1 Systems Approach to Training. Final Report

Training Resources Analytic Model (TRAM). User's Manual.

5. TYPE OF REPORT & PERIOD COVERED

Jul 74 - Oct 75

6. PERFORMING ORG. REPORT NUMBER

SAT-5

7. AUTHOR(s)

William F.H. Ring, William L. Stortz
George Gaidasz
John R. Menig

8. CONTRACT OR GRANT NUMBER(s)

F33657-75-C-0021

9. PERFORMING ORGANIZATION NAME AND ADDRESS

Calspan Corporation
P. O. Box 235
Buffalo, New York 14221

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS

Aeronautical Systems Division
B-1 Systems Project Office
Wright-Patterson Air Force Base, OH 45433

12. REPORT DATE

JUL 1975

13. NUMBER OF PAGES

161 (12) 165p.

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

CALSPAN-TM-SAT-5

15. SECURITY CLASS. (of this report)

Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution limited to U.S. Government Agencies only: test and evaluation; July 1975. Other requests for this document must be referred to B-1 Systems Program Office, ASD/YHCD, Wright-Patterson Air Force Base, Ohio 45433.

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)

Instructional Systems Development
Systems Approach to Training
B-1
Economic Modeling
Resource Management
Computer Storage
Training
Training Resources

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

The TRAM is a multiphase set of computer programs which model a proposed training system and determines resource utilization, scheduling problems and costs. Each program, except for sorting steps, is described by a user's guide and programmer's guide. Also indicated is the relationship of TROLIE, the quick-look version of TRAM, which was developed.

(Continued on next page.)

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

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

407 727

20. Abstract (Continued.)

(cont'd p 1473A)

Phase 1 of the TRAM is used to assemble most of the input data and to check it for consistency and completeness. The output of Phase 1 is normally a tape which is passed to Phase 2. The Phase 2 program further checks linkages and network integrity and prepares lists of names, student demands, trainee source lists and resource lists. Phase 3 resolves the trainee demands into classes and determines the amount of resources used by simulating the training system. The output of Phase 3 consists of source and lag records which indicate the occurence of trainee matriculation, lags due to lack of resources, and an unused resources file. Phase 4 computes the amount of resources used by comparing the unused and original resources, and then prepares an economic analysis of the run. Phase 5 processes the trainee source and lag records and writes a report on these uses.

The TROLIE program provides a quick-look version of TRAM. TROLIE provides the data set inputs required by TRAM Phase 4. Phase 4 TRAM then performs the same economic analysis.

It is possible to write a FORTRAN program with variable dimensions in order to control the size of the program at execution. One method which is frequently used is to create a master source with symbols instead of values, which are replaced by numbers using a text manipulation program. Such a system is used in TRAM. The text manipulation program is known as VARY and is documented in the final Section of this report.

1473B

PREFACE

This document is one of several technical memoranda which have been delivered to the B-1 Systems Project Office (B-1 SPO) in performance of the Systems Approach to Training (SAT) Task under Contract Number F33657-75-C-0021. Each of the separate SAT documents is listed below. Additional copies may be requested from: B-1 Systems Project Office, Data Configuration Division, Wright-Patterson Air Force Base, Ohio.

Technical Memoranda	Number	Author(s)	Date
B-1 Systems Approach to Training, Final Report.	SAT- 1 Vol. 1	R. Sugarman S. Johnson W. Ring	July 1975
B-1 Systems Approach to Training, Final Report. Appendix A: Cost Details.	SAT- 1 Vol. 2	H. Reif W. Ring	July 1975
B-1 Systems Approach to Training, Final Report. Appendix B: Bibliography and Data Collection Trips.	SAT- 1 Vol. 3	A. Blair	July 1975
Behavioral Objectives for the Pilot, Copilot, and Offensive Systems Operator.	SAT- 2 Vol. 1 & 2	J. Mitchell W. Hinton S. Johnson	July 1975
Simulation Technology Assessment Report (STAR).	SAT- 3	S. Johnson J. Knight R. Sugarman	July 1975
Sorting Model for B-1 Aircrew Training Data. User's and Programmer's Guide.	SAT- 4	J. Menig T. Ranney	July 1975
Training Resources Analytic Model (TRAM). User's Manual.	SAT- 5	W. Ring G. Gaidasz J. Menig W. Stortz	July 1975
Training Resources Analytic Model (TRAM). Programmer's Manual.	SAT- 6	W. Ring G. Gaidasz J. Menig W. Stortz	July 1975
Task Analysis Listings.	SAT- 7	J. Mitchell T. Ranney	July 1975
Control/Display Catalog and Action Verb Thesaurus.	SAT- 8	T. Ranney A. Blair	July 1975

CALSPAN CORPORATION

JULY 1975
SAT-5

TRAINING RESOURCES ANALYTIC MODEL (TRAM)
USER'S MANUAL

William F. H. Ring
William L. Stortz
George Gaidasz
John R. Menig

SUMMARY

The TRAM* is a multiphase set of computer programs which models a proposed training system and determines resource utilization, scheduling problems and costs. Figure 1.1 presents the overall flow diagram which depicts each program within TRAM and the data sets associated with it. Each program, except for the sorting steps, is described by a user's guide and programmer's guide. Also indicated is the relationship of TROLIE** , the quick-look version of TRAM, which was developed.

Phase 1 of the TRAM is used to assemble most of the input data and to check it for consistency and completeness. The output of Phase 1 is normally a tape which is passed to Phase 2. The Phase 2 program further checks linkages and network integrity and prepares lists of names, student demands, trainee source lists and resource lists. Phase 3 resolves the trainee demands into classes and determines the amount of resources used by simulating the training system. The output of Phase 3 consists of source and lag records which indicate the occurrence of trainee matriculation, lags due to lack of resources, and an unused resources file. Phase 4 computes the amount of resources used by comparing the unused and original resources, and then prepares an economic analysis of the run. Phase 5 processes the trainee source and lag records and writes a report on these uses.

The TROLIE program provides a quick-look version of TRAM. TROLIE provides the data set inputs required by TRAM Phase 4. Phase 4 TRAM then performs the same economic analysis.

It is possible to write a FORTRAN program with variable dimensions in order to control the size of the program at execution. One method which is frequently used is to create a master source with symbols instead of values, which are replaced by numbers using a text manipulation program. Such a system is used in TRAM. The text manipulation program is known as VARY and is documented in the final Section of this report.

The following Sections describe the phases of TRAM, TROLIE, and VARY.

---

* Training Resources Analytic Model
** Training Resources Organized for Logical Integration of Expenses

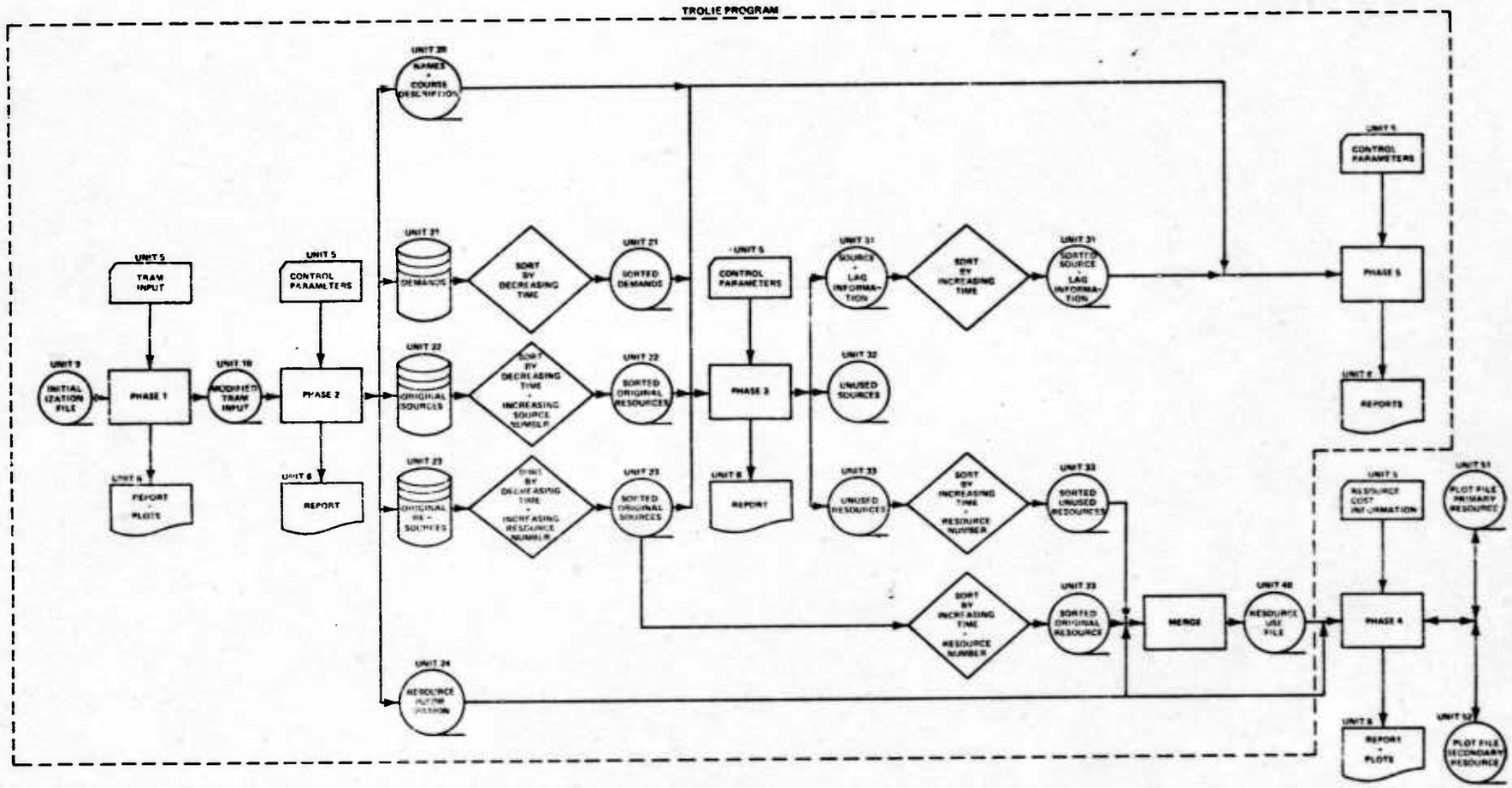


Figure 1.1

TRAM OVERALL ORGANIZATION

(Dashed lines indicate portions of TRAM replaced by TROLIE-the quick-look program)

TABLE OF CONTENTS

Section	Description	Page
1	Phase 1	1
	1.1 Purpose	1
	1.2 Description	1
	1.3 Inputs	2
	1.3.1 General Coding Information	2
	1.3.2 TRAM Input Descriptions	3
	1.4 Outputs	10
	1.5 Input Card Coding Forms	12
	1.6 Sample Run Phase 1 TRAM	27
	1.7 Input Routine Error Messages	53
2	Phase 2	54
	2.1 Purpose	54
	2.2 Description	54
	2.3 Input	58
	2.4 Output	60
	2.5 Error Messages Generated in Phase 2	69
3	Phase 3	73
	3.1 Introduction	73
	3.1.1 Contents of this Section	73
	3.1.2 Purpose	73
	3.1.3 Treatment of Time	74
	3.2 Program Description	75
	3.2.1 Overall Description	75
	3.2.2 Class Formation	79
	3.2.3 Synchronization	81
	3.2.4 Correlation	82
	3.2.5 Resource Use	83
	3.2.6 Source Allocation	84
	3.2.7 Transfers	85
	3.2.8 Data Management	86
	3.3 Description of Inputs	87
	3.3.1 Execution Control Parameters	87

TABLE OF CONTENTS (Cont.)

Section	Description	Page
3	Phase 3 (Continued.)	
	3.3.2 Training Demand Records	89
	3.3.3 Resource Inventories	89
	3.3.4 Trainee Inventories	89
	3.3.5 Description of Training Program ...	89
	3.4 Description of Outputs	90
	3.4.1 Resource Inventories	90
	3.4.2 Source Inventories	90
	3.4.3 Lag Records	90
	3.4.4 Phase 3 Messages	92
4	Merge Program	101
	4.1 Purpose	101
	4.2 Description	101
	4.3 Merge Program Error Messages	101
5	Phase 4	102
	5.1 Purpose	102
	5.2 Description	102
	5.3 Inputs	109
	5.3.1 Files from Previous TRAM Programs .	109
	5.3.2 Input Card Description	109
	5.4 Outputs	110
	5.5 Error Messages	113
6	Phase 5	115
	6.1 Purpose	115
	6.2 Description	115
	6.3 Inputs	115
	6.4 Outputs	115
	6.5 Phase 5 Error Messages	117

TABLE OF CONTENTS (Cont.)

Section	Description	Page
7	TROLIE Program	118
	7.1 Purpose	118
	7.2 Description	118
	7.3 Inputs	118
	7.4 Data Management	134
	7.5 Reports	145
	7.6 Data Set Output	145
8	Variable Dimensions	150
	8.1 VARY Program and Associated Code	150
	8.2 Inputs to the VARY Program	150

1 PHASE 1

1.1 Purpose

The purpose of phase 1 is to read the user inputs and convert them to the internal format required for phase 2. It also tests the inputs for validity and provides the necessary outputs to document the run.

1.2 Description

Phase 1 performs the following functions:

- reads the input cards and prints them
- checks all values for validity
- prints formatted tables of the inputs
- replaces user assigned names with an internal ID number
- plots a course block diagram
- sorts the data records and writes them onto unit 10 for phase 2

The values from each input card are printed as they are read. Also, each numeric value is tested against a range of acceptable values to see if it is valid. If it is not, a diagnostic message is printed, which will appear immediately after the card on the input card listing.

After all inputs have been read, they may be optionally re-printed in formatted tables. The purpose of this printout is to show all the inputs in an easily readable form.

The next processing step generates a table of all user assigned names, and replaces all references to these names with internal ID numbers. It is at this time that undefined or multiply defined names are detected and flagged.

If no errors have been detected up to this point, the course block diagram is plotted (optional). Additional error messages are printed if the processing blocks are not in the proper order for plotting.

The final processing performed in phase 1 is to sort the input data and write it to file 10 for input to phase 2.

1.3 Inputs

1.3.1 General Coding Information

The input cards for the TRAM model have fixed format fields, and a separate card is provided for each different type of information. Coding forms for each of these input cards are shown in Section 1.5.

All cards have a card name field, which is used to identify the card type. Although the different card types contain different information, most of them conform to a standard field layout. This consists of the card name field in columns 1-10, followed by two ten-column character data fields and ten five-column numeric data fields. All numeric data are read in integer format. Variables whose value can take on non-integral values are read with an implied decimal point. These values are converted internally, using the position of the implied decimal point shown on the coding forms. For example, an alert ratio of .57 would be coded as 57 on the airbase event card. Character data are left justified, and numeric data are right justified.

Cards that do not conform to the standard field layout (TASK, RUB, RUDB), must be preceded by a set header card. This card identifies the type of cards that are to follow. Note that these non-standard cards have a blank card name field, since the card type is identified by the header card. Each set is terminated by a SETEND card.

Some input cards require additional continuation cards. Such cards have parameters to give the number of each type of card which is to follow it. The formats for these additional cards are shown on the same coding form as the header card so that they may easily be coded in the proper sequence. Note that these continuation cards also have blank card name fields, since the card type is identified by the header card.

In general, these cards, or groups of cards, can be coded in any order. The only exception to this is the course and the processing block cards. The processing blocks for each course must follow the course card. Also, the processing blocks within a course must start with the graduation block and proceed towards matriculation. This is because the position of each processing block is given as an offset from the block connected to it on the right. If there is more than one block to the right, as in the case of fan-outs, the first one encountered is the one used as the reference position.

The input deck must be ended with an end card, which consists of the word "END" in the first field.

1.3.2 TRAM Input Descriptions

Time Units in TRAM

Time in TRAM appears in several ways. The basic unit of time is the calendar unit (CU). The size of the calendar unit is determined by the number of CUs in a year. If one CU is equal to a day, and a 7 day week is desired, then there are 365 CUs in a year. But if a 6 day week is desired, then there are only 6 x 52 = 312 CUs in a year. All events in the program are expressed in CUs.

The year is a second unit of time which is used in TRAM - Phases 2 and 4.

Resource and source use is accounted for in "buckets". A bucket represents the short term planning horizon for the resource or source use.

Time as a unit of a resource use is independent of CUs. Consider our 312 CU year. A simulator operated 10 hours per day, 7 days per week, could yield 3650 hours of training per year with no conflict in the definition of the CU.

Types of Trainees in TRAM

Trainees are of four types: pilots, copilots, OSO and DSO. These are encoded as 8, 4, 2, 1, respectively. Normally aircrews are trained in complete sets. However, provision is made for having an independent attrition of each type of trainee. If a different attrition formulation is used, or if there are unbalanced initial inventories at the airbases, an unbalanced number of trainees will be required. Trainees that are not a part of a complete aircrew are labeled as extras, as opposed to normal trainees. Special tasks can be defined for these extra trainees (see task type field on task description card.)

The following paragraphs describe the input data for TRAM in an appropriate order for input to the program.

Control Card

The control card must be the first card of the input deck.

Airbase Inventory Cards (One and only one per airbase)

These cards serve to name the airbases. If desired, an initial inventory of aircraft and aircrews can be placed at the airbases. The quantity of aircraft and aircrews is assumed to be constant for all time previous to the beginning of a run for the purposes of attrition calculations.

Airbase Event Cards (One or more per airbase)

At least one event must be provided for each airbase. In this way, the crew ratio, alert ratio and crew yield (hours/crew/week) can be introduced. If the rules for manning a base change, for instance if the crew ratio changes, if the PMT changes, or if a new CCTS is introduced, airbase events must be provided for the changes.

Aircraft Delivery Cards

Aircraft delivery cards are required only if the initial inventory at an airbase changes. Delivery of an aircraft is not an "airbase event".

CCTS Cards

CCTS cards are used to assign CCTS courses to demands for trainees. At least one CCTS card must be provided for each crewmember for each airbase with an inventory of aircraft. The CCTS courses which support an airbase can change, or multiple CCTS courses can be specified through the ratio parameter operating in conjunction with airbase events. The total proportion of each crewmember from any base must be 100.

PMT Group Cards

A PMT group is a set of PMT courses which run simultaneously. Typically, a PMT group is the courses required to provide recurrent training and evaluation of the crewmembers. If PMT is conducted for complete crews only, then only the pilot needs to be modeled. The remaining crewmembers can be ignored, if the resource use which they would normally introduce is attributed to the pilots, and if there is no chance of having more of one type crewmember than another, as a result of unbalanced initial inventories. The period is the frequency (CUs per training period) that crews must return for PMT training within the given group. PMT training can change by means of the airbase event reference.

PMT Cards

PMT cards follow their associated PMT Group card. These cards are used to define the PMT courses in a PMT Group. Trainees can be apportioned among PMT courses via the ratio parameter. The total proportion must be 100 for each crewmember in each PMT group. The time lost parameter is the total time away from the airbase less the time actually spent in the course, normally the 2-way transportation delay.

Training Course Cards

A course is the instruction required to train a given crewmember. As noted above, more than one course can be provided so that any combination of courses can be called upon to support any time-varying combination of airbases. A course is made up of any number of tracks. Each track leads to a unique source of trainees. Training course cards are header cards for the processing block cards which make up a course. The course has potential graduations starting at the given

date and extending regularly toward the end of the course with the given period. Within the period, sufficient graduations are scheduled to distribute the graduates as evenly as possible. The minimum number of graduations is scheduled as determined from the maximum class size. If at least one trainee is required in the period, a graduation is scheduled.

Processing Block Cards

A processing block is akin to an instructional block. The trainees are processed by the block in the sense that when trainees are present, action is taken by the program. The processing block cards are header cards for transfer cards and task cards. If trainees must have other trainees present, such as when flying an aircraft, synchronization and correlation loops are defined. Such a loop is formed by a series of block references. Block A points to Block B which points to Block C, etc. until the final block in the loop, say Block G, completes the loop by pointing back to Block A. Synchronization means that all crewmembers from a given class are present in the appropriate blocks of the loop. Correlation means that a trainee is in a block in the loop for each of the courses in the loop. Synchronization is used when one wishes to train a crew that will stay together and ultimately be assigned to the same airbase. Correlation is a much weaker constraint. Correlation is used when one wishes to have each position manned. Synchronization, since it requires a complete aircrew, can only be defined for portions of the training program in which all crewmembers are being trained. Thus a synchronization loop must include a block from all of the tracks in the courses which it links.

Blocks must be numbered uniquely in a course but need not follow any sequence. The block duration is the expected time required to complete the instruction within the block. The location information is used for plotting. The location values are relative to the next block in the course. Blocks have a standard size of 1" (height-y) by 1.4" (width-x). Blocks are assumed to be placed in the input in order from graduation (which is normally on the right of the plot) toward matriculation (on the left). If this order is not maintained, one to five transfer cards are required.

The special value of -1 transfers is used to indicate that a processing block has a source task. The special value of 0 transfers indicates that the assumed sequencing is used. Up to five tasks can be specified for each block. Block priority is not currently implemented.

Transfer Cards

Transfer cards immediately follow processing block cards to which they refer. Transfer cards are used (1) whenever the processing block card which follows a given processing block card is not the immediate predecessor block in the course, and (2) whenever more than one block precedes a given block (i.e., when tracks split off from common segments of a course.) The priority of the transfer indicates the preference of the source or sources at the end of the branch (higher numeric value indicates higher priority.) Priorities of secondary, tertiary etc., branches have secondary, tertiary, etc., effects on the calculation of priority. The ratio parameter is used to apportion trainees within a given priority level. Thus it is the transfer cards which implement source preferences. The program attempts to obtain all trainee requirements from the highest priority sources until they are exhausted. It then goes to the next highest priority source, etc. Within a given priority level the given proportion will only be preserved if (1) the sources can support the damands and if (2) the class is capable of splitting (i.e., a class of 1 cannot split, a class of 2 cannot split 3 ways, etc.).

In Case 1 the program uses sources until all sources at a given level are exhausted before calling on lower priority sources. In Case 2 the first, highest priority branches are taken.

Task Cards

A task card is required for each task in a processing block. The kinds of tasks defined are:

SYNCH	Synchronize
CORRELATE	Correlate
USERESRCE	Use of resources
SOURCE	Get trainees from a source
SCATSA	Split classes according to source availability

A given task must be defined only once. It can be defined in place behind the processing block or in a separate set of task description cards following a task header card. In this way, a single task definition can serve several processing blocks. Once a task is defined, all other processing blocks using the task invoke it by including a card with the task name only.

SYNCH Function

A single task should be defined whose function is to implement the synchronization function. A suitable task name is SYNCH.

CORRELATE Function

A single task should be defined whose function is to implement correlation. If correlation is not possible within a given period input in phase 3, an extras task is activated if it exists. Correlation can occur using normal or extras trainees.

SCATSA Function - Split Classes According to Source Availability

This task is required once and only once for any course that has more than one track. The SCATSA task is placed in the first processing block that contains all tracks in the course, that is, in the processing block furthest from the graduation block.

USERESRCE Function

A USERESRCE function task is required to expend resources. This is accomplished through the RUB - Resource Use Block. USERESRCE tasks can be either normal - (all trainees use the resources) or extras - (for extra and uncorrelatable trainees.) Task names for USERESRCE tasks are conveneintly chosen to be the same as or related to their RUB and ultimately the RUDBs.

SOURCE Function

The source task provides a means for identifying the source of a track. A unique source task must be defined for each source. Associated with the source task are RUB, RUDB, and SOURCE cards. (RUDB cards for sources contain only a name and resource field.) Thus, sources of trainees are treated almost like resources. A suitable name for the RUB and RUDB is the source name. If a processing block has a source task, the number of transfers is set to -1 to indicate that this is the end of the track. Transfer cards are not required for such a processing block.

RUB Cards

Resource Utilization Block cards are read in sets preceded by a RUB header card. Up to 6 Resource Utilization Description Blocks (RUDBs) are referenced by a given RUB. RUBs are conveniently named to indicate the RUDBs to which they refer. The purpose of the extra layer of referencing is to allow for an essentially unlimited number of resource usages through the secondary RUB parameter in the RUDB card.

RUDB Cards

Resource Utilization Description Block cards are used to indicate resource use. Resources named must be defined by an inventory card. When a class of trainees invokes an RUDB the amount of use can be dependent upon the class (i.e., a classroom and instructor) or trainee (i.e., a 1-position simulator). The use of the resource can be concentrated at the end of the processing block (INSTANT), occur uniformly throughout the duration of the processing block (UNIFORM) or in any feasible manner (ARBITRARY). ARBITRARY usage will concentrate the use as much as

possible at the latest time in the processing block, an algorithm logic that tends to maximize earlier availability of the resource. Secondary RUBs are automatically invoked when the given RUDB is invoked. An exceedingly long string of RUB-RUDB-RUB etc.'s is possible. Alternate RUDBs are invoked to the extent that a given RUDB is not feasible. That is, the primary resource is used until it is exhausted before the alternate resource is used.

Resources are provided in units. Typically these units have the dimension of time in a facility but other units are possible, such as fuel if it is to be accounted separately. Units should be defined to assure that only integral values are required.

Resource Inventory Cards and Source of Trainee Cards

These cards are used to define the sources and resources and their availability using the uniform generating function (the only one provided.) The parameters are the number of units available per bucket and the size of the bucket in calendar units (see above paragraph on RUDBs.) The first source of trainees must have the name COPILOTS even if there is no copilot recovery.

1.4 Outputs

Input Card Listing

This listing shows the values exactly as they are read from each card. Field numbers are marked across the top of the page for reference by error messages. These fields correspond to five card columns Character data are spread across two such fields, and would be referred to by the number of the second field. Also, card sequence numbers are printed for later reference by error messages. This listing is always printed.

Formatted Input Tables

These tables duplicate the information shown by the input card listing, but present it in a conveniently readable form. All variables are identified by column titles, and the meanings of integer codes are printed rather than the codes

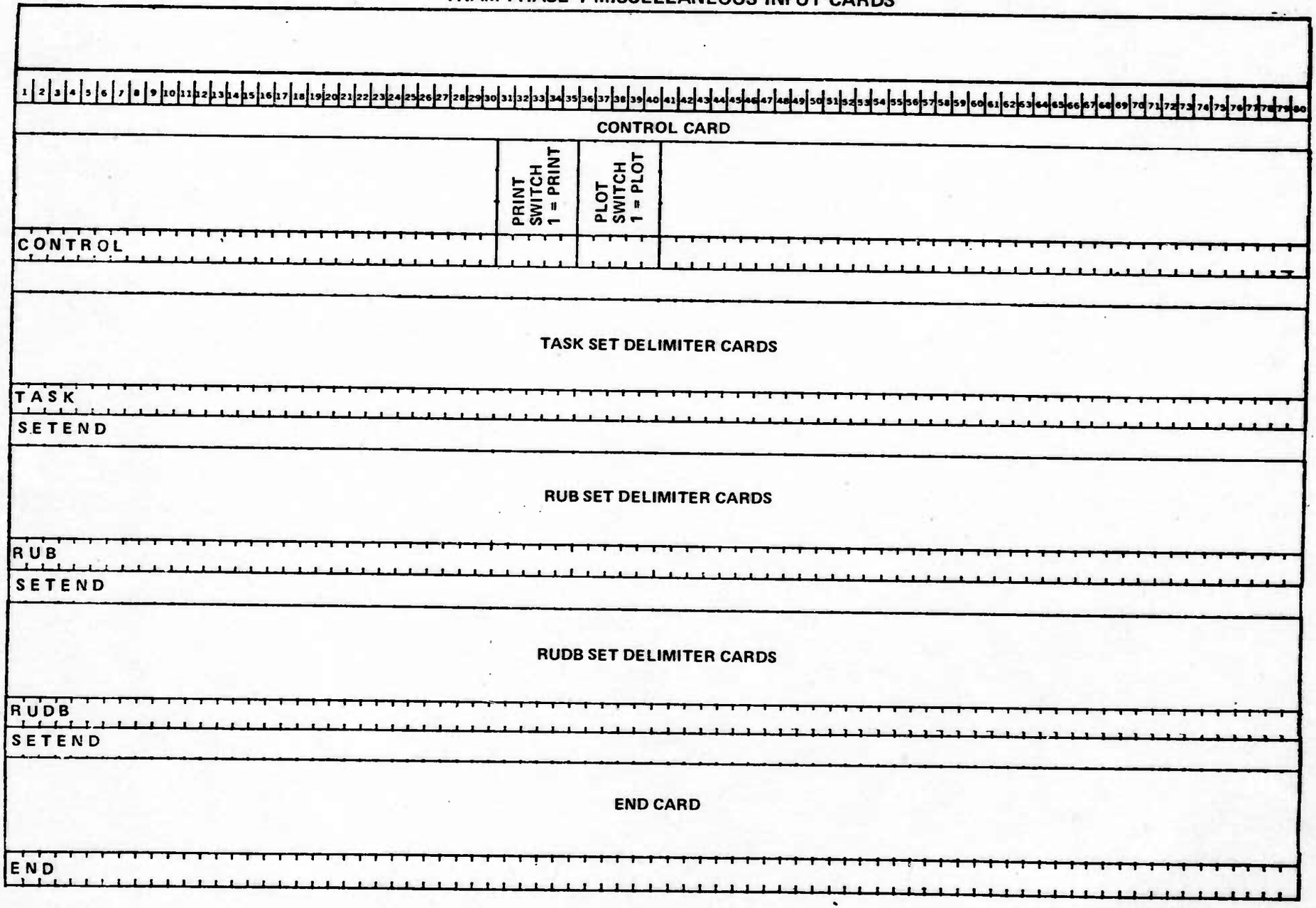
themselves. Also, these values will show the results of any input conversion that was done. The user may suppress this listing by use of the input routine control card.

Course Block Diagram

This plot shows the structure of the courses by displaying each processing block in the course as a rectangle, with the flow of students shown by connecting lines from one processing block to the next. Inside each rectangle, the block number, block name, synchronize-correlate reference (if any), and task names are shown. The course name is plotted above each graduation block. The input routine control card specifies if this plot is to be produced.

1.5 INPUT CARD CODING FORMS

TRAM PHASE 1 MISCELLANEOUS INPUT CARDS



AIRBASE INVENTORY CARDS

		AIRBASE NAME		INITIAL INVENTORY OF AIRCRAFT	INITIAL INVENTORY OF PILOTS	INITIAL INVENTORY OF COPILOTS	INITIAL INVENTORY OF OSO	INITIAL INVENTORY OF DSO	
	1 2 3 4 5 6 7 8 9 10	11 12 13 14 15 16 17 18 19 20	21 22 23 24 25 26 27 28 29 30	31 32 33 34 35	36 37 38 39 40	41 42 43 44 45	46 47 48 49 50	51 52 53 54 55	56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1	AIRBASE								
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20	AIRBASE								

AIRBASE EVENT CARDS

	EVENT NAME	AIRBASE NAME	TIME	CREW RATIO (X.XX)	ALERT RATIO (X.XX)	HOURS/CREW/WEEK	
1	ABEVENT						
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20	ABEVENT						

AIRBASE EVENT CARDS

	EVENT NAME	AIRBASE NAME	TIME	CREW RATIO (X.XX)	ALERT RATIO (X.XX)	HOURS/CREW/WEEK	
1	ABEVENT						
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20	ABEVENT						

AIRCRAFT DELIVERY CARDS

		AIR BASE NAME		DATE [CU's]	NUMBER OF AIRCRAFT	
1	DELIVERY					
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20	DELIVERY					

CCTS CARDS

	AIRBASE EVENT NAME	COURSE NAME	PERSONNEL TYPE 8 - P 4 - CP 2 - OSO 1 - DSO	RATIO (X.XX)	
1	CCTS				
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20	CCTS				

PMT GROUP HEADER CARDS

"PMTGROUP"		AIRBASE EVENT NAME	PERIOD	NUMBER OF PMT COURSES	

PMT CARDS

		COURSE NAME	PERSONNEL TYPE 8: CP 2: OSO 1: DSO	RATIO (X.XX)	TIME LOST [CU's]	

TRAINING COURSE CARDS (HEADER FOR PROC BLOCKS)

	NAME OF COURSE		COURSE TYPE 1-CCTS 2-PMT	PERSONNEL TYPE 8 - P 4 - CP 2 - OSO 1 - DSO	ABSOLUTE LOCATION OF GRADUATION BLOCK		MAXIMUM CLASS SIZE	CLASS PERIOD	PRIORITY	EARLIEST GRADUATION DATE	
					X (X.XX) in	Y (X.XX) in		[CU's]		[CU's]	
	1 2 3 4 5 6 7 8 9 10	11 12 13 14 15 16 17 18 19 20	21 22 23 24 25 26 27 28 29 30	31 32 33 34 35	36 37 38 39 40	41 42 43 44 45	46 47 48 49 50	51 52 53 54 55	56 57 58 59 60	61 62 63 64 65	66 67 68 69 70	71 72 73 74 75 76 77 78 79 80
1	COURSE										
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											
20	COURSE										

PROCESSING BLOCK CARDS

"PROCBLOCK" FOR PROCESSING BLOCK CARDS BLANK FOR TRANSFER AND TASK NAME CARDS	NAME OF PROCESSING BLOCK OR TASK	SYNCHRONIZE-CORRELATE REFERENCE			BLOCK NUMBER	BLOCK DURATION	LOCATION RELATIVE TO NEXT BLOCK		NUMBER OF TRANSFERS	NUMBER OF TASKS	PRIORITY	
		COURSE NAME	BLOCK NUMBER	TYPE 0 - NONE 1 - SYNC 2 - CORR			X (X.XX) in	Y (X.XX) in				
1 2 3 4 5 6 7 8 9 10	11 12 13 14 15 16 17 18 19 20	21 22 23 24 25 26 27 28 29 30	31 32 33 34 35	36 37 38 39 40	41 42 43 44 45	46 47 48 49 50	51 52 53 54 55	56 57 58 59 60	61 62 63 64 65	66 67 68 69 70	71 72 73 74 75	76 77 78 79 80
1												
2												
3												
4												
5												
6												
7												
8												
9												
10												
11												
12												
13												
14												
15												

TRANSFER CARDS

BLOCK NUMBER TRANSFERRED FROM	PRIORITY	RATIO (X.XX)

TASK DESCRIPTION CARDS

	TASK NAME	TASK FUNCTION NAME (SYNCH CORRELATE USERESRCE SOURCE SCATSA)	RUB NAME	TASK TYPE 1 - NORMAL 2 - EXTRAS	# OF PARAMETERS	PARAMETERS	
	1 2 3 4 5 6 7 8 9 10	11 12 13 14 15 16 17 18 19 20	21 22 23 24 25 26 27 28 29 30	31 32 33 34 35 36 37 38 39 40	41 42 43 44 45	46 47 48 49 50	51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							

RESOURCE USE BLOCK (RUB) CARDS

	RUB NAME	NAMES OF RESOURCE UTILIZATION DESCRIPTION BLOCK (RUDB) CARDS (UP TO 6)

	1 2 3 4 5 6 7 8 9 10	11 12 13 14 15 16 17 18 19 20	21 22 23 24 25 26 27 28 29 30	31 32 33 34 35 36 37 38 39 40	41 42 43 44 45 46 47 48 49 50	51 52 53 54 55 56 57 58 59 60	61 62 63 64 65 66 67 68 69 70	71 72 73 74 75 76 77 78 79 80
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								
18								
19								
20								

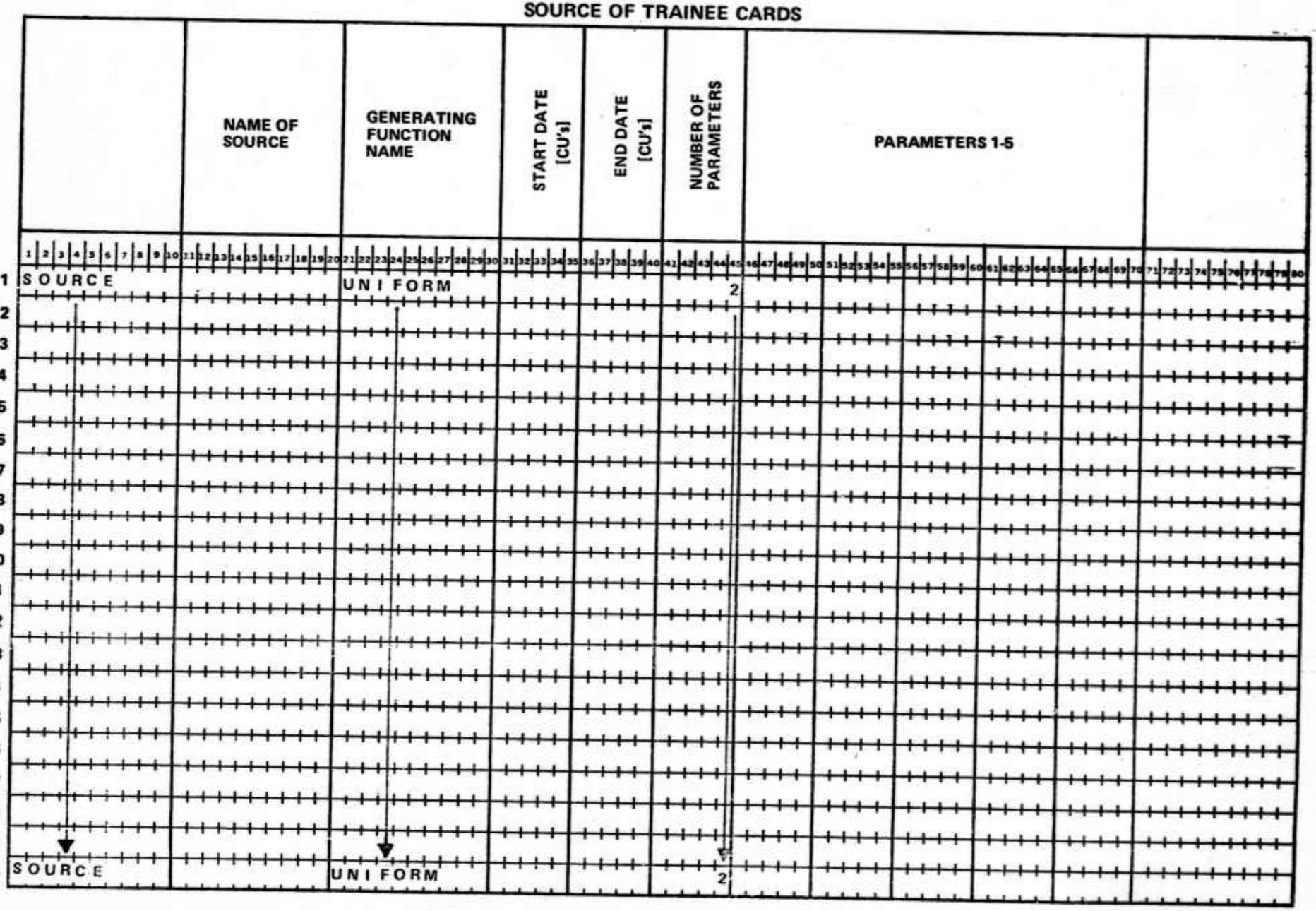
RESOURCE UTILIZATION DESCRIPTION BLOCK (RUDB) CARDS

	RUDB NAME	RESOURCE NAME	RESOURCE UTILIZATION GROUPING FUNCTION { CLASS / INDIV }	RESOURCE UTILIZATION TIMING FUNCTION { ARBITRARY / INSTANT / UNIFORM }	SECONDARY RUB	ALTERNATE RUDB	UNITS OF CONSUMPTION PER UNIT USER (II)	
	1 2 3 4 5 6 7 8 9 10	11 12 13 14 15 16 17 18 19 20	21 22 23 24 25 26 27 28 29 30	31 32 33 34 35 36 37 38 39 40	41 42 43 44 45 46 47 48 49 50	51 52 53 54 55 56 57 58 59 60	61 62 63 64 65 66 67 68 69 70	71 72 73 74 75 76 77 78 79 80
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								
18								
19								
20								

RESOURCE INVENTORY CARDS

	RESOURCE NAME	GENERATING FUNCTION NAME	START DATE [CU's]	END DATE [CU's]	NUMBER OF PARAMETERS	PARAMETERS
1	INVENTORY	UNIFORM			2	
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20	INVENTORY	UNIFORM			2	

SOURCE OF TRAINEE CARDS



1.6 SAMPLE RUN PHASE I TRAM

TRAM INPUT CARDS

FIELD NUMBER

CARD NO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	CONTROL						1	1								
2	AIRBASE		AB2				0	0	0	0	0					
3	AIRBASE		AB3				0	0	0	0	0					
4	ABEVENT		2		AB2		1	250	40	60						
5	ABEVENT		3		AB3		1	250	40	60						
6	CCTS		2		PILOTS		8	100								
7	CCTS		2		COPILOTS		4	100								
8	CCTS		2		OSOS		2	100								
9	CCTS		2		DSOS		1	100								
10	CCTS		3		PILOTS		8	100								
11	CCTS		3		COPILOTS		4	100								
12	CCTS		3		OSOS		2	100								
13	CCTS		3		DSOS		1	100								
14	PMTGROUP				2		5280	1								
15					CCTSP		5	100	10							
16	PMTGROUP				3		5280	1								
17					CCTSP		8	100	10							
18	DELIVERY		AB2				5721	1								
19	DELIVERY		AB3				5771	2								
20	COURSE		PILOTS				1	8	2000	2000	2	200	0	1		
21	PROCBLOCK		P6		COPILOTS		5	1	6	20	-200	0	0	2	0	
22			P6 U		USERESRCE		P6		0	0	0	0	0	0	0	
23			SYNCH						0	0	0	0	0	0	0	
24	PROCBLOCK		P5				0	0	5	15	-200	0	2	2	0	
25							4	1	100							
26							9	2	100							
27			P5 T		SCATSA				0	0	0	0	0	0	0	
28			P5 U		USERESRCE		P5		0	0	0	0	0	0	0	
29	PROCBLOCK		P4				0	0	4	10	-200	150	0	1	0	
30			P4 U		USERESRCE		P4		0	0	0	0	0	0	0	
31	PROCBLOCK		P3				0	0	3	5	-200	0	0	1	0	
32			P3 U		USERESRCE		P3		0	0	0	0	0	0	0	
33	PROCBLOCK		P2				0	0	2	10	-200	0	2	1	0	
34							1	1	75							
35							8	1	25							
36			P2 U		USERESRCE		P2		0	0	0	0	0	0	0	
37	PROCBLOCK		P1		OSOS		5	2	1	20	-200	150	1	2	0	
38							10	0	100							
39			P1 U		USERESRCE		P1		0	0	0	0	0	0	0	
40			CORR						0	0	0	0	0	0	0	
41	PROCBLOCK		P10				0	0	10	0	-200	0	-1	1	0	
42			P10 S		SOURCE		B52P		0	0	0	0	0	0	0	
43	PROCBLOCK		P8		PILOTS		1	2	8	15	-200	-150	1	2	0	
44							11	0	100							
45			P8 U		USERESRCE		P8		0	0	0	0	0	0	0	
46			CORR						0	0	0	0	0	0	0	
47	PROCBLOCK		P11				0	0	11	0	-200	0	-1	1	0	
48			P11 S		SOURCE		FB111P		0	0	0	0	0	0	0	
49	PROCBLOCK		P9				0	0	9	10	-200	-150	1	1	0	

TRAM INPUT CARDS

CARD NO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
50							12	0	100							
51			P9 U		USERESRCE	P9			0	0	0	0	0	0	0	
52	PROCBLOCK		P12				0	0	12	0	-200	0	-1	1	0	
53			P12 S		SOURCE	COPILOTS			0	0	0	0	0	0	0	
54	COURSE		COPILOTS				1	4	2000	1500	2	440	0	1		
55	PROCBLOCK		C6				0	0	6	20	0	0	2	1	0	
56							5	1	50							
57							11	1	50							
58			C6 U		USERESRCE	C6			0	0	0	0	0	0	0	
59	PROCBLOCK		C5		COPILOTS		11	1	5	25	-200	150	0	2	0	
60			C5 U		USERESRCE	C5			0	0	0	0	0	0	0	
61			SYNCH						0	0	0	0	0	0	0	
62	PROCBLOCK		C4		COPILOTS		10	1	4	20	-200	0	0	2	0	
63			C4 U		USERESRCE	C4			0	0	0	0	0	0	0	
64			SYNCH						0	0	0	0	0	0	0	
65	PROCBLOCK		C3				0	0	3	15	-200	-150	0	1	0	
66			C3 U		USERESRCE	C3			0	0	0	0	0	0	0	
67	PROCBLOCK		C2				0	0	2	10	-200	0	2	2	0	
68							1	1	50							
69							9	1	50							
70			C2 U		USERESRCE	C2			0	0	0	0	0	0	0	
71			C2 T		SCATSA				0	0	0	0	0	0	0	
72	PROCBLOCK		C1		PILOTS		8	2	1	5	-400	150	1	2	0	
73							12	0	100							
74			C1 U		USERESRCE	C1			0	0	0	0	0	0	0	
75			CORR						0	0	0	0	0	0	0	
76	PROCBLOCK		C12				0	0	12	0	-200	0	-1	1	0	
77			C12 S		SOURCE	B52CP			0	0	0	0	0	0	0	
78	PROCBLOCK		C11		OSOS		5	1	11	10	-200	-150	0	2	0	
79			C11 U		USERESRCE	C11			0	0	0	0	0	0	0	
80			SYNCH						0	0	0	0	0	0	0	
81	PROCBLOCK		C10		COPILOTS		4	1	10	5	-200	0	1	2	0	
82							3	0	100							
83			C10 U		USERESRCE	C10			0	0	0	0	0	0	0	
84			SYNCH						0	0	0	0	0	0	0	
85	PROCBLOCK		C9		COPILOTS		1	2	9	15	-400	-150	0	2	0	
86			C9 U		USERESRCE	C9			0	0	0	0	0	0	0	
87			CORR						0	0	0	0	0	0	0	
88	PROCBLOCK		C8				0	0	8	10	-200	0	1	1	0	
89							13	0	100							
90			C8 U		USERESRCE	C8			0	0	0	0	0	0	0	
91	PROCBLOCK		C13				0	0	13	0	-200	0	-1	1	0	
92			C13 S		SOURCE	KC135CP			0	0	0	0	0	0	0	
93	COURSE		OSOS				1	2	2000	1000	2	440	0	1		
94	PROCBLOCK		O5		OSOS		6	1	5	10	-200	0	0	2	0	
95			O5 U		USERESRCE	O5			0	0	0	0	0	0	0	
96			SYNCH						0	0	0	0	0	0	0	
97	PROCBLOCK		O4				0	0	4	22	-200	0	0	1	0	
98			O4 U		USEPESRCE	O4			0	0	0	0	0	0	0	

TRAM INPUT CARDS

CARD NO	1	3	5	6	7	8	9	10	11	12	13	14	15	16
99	PROCBLOCK	D3	COPILOTS		9	2	3	20	-800	0	2	3	0	
100					2	2	100							
101					1	1	100							
102		D3 U	USERESRCE	D3			0	0	0	0	0	0	0	
103		CORR					0	0	0	0	0	0	0	
104		D3 T	SCATSA				0	0	0	0	0	0	0	
105	PROCBLOCK	D2	DSOS		1	1	2	15	-200	150	1	2	0	
106					7	0	100							
107		D2 U	USERESRCE	D2			0	0	0	0	0	0	0	
108		SYNCH					0	0	0	0	0	0	0	
109	PROCBLOCK	D7			0	0	7	0	-200	0	-1	1	0	
110		D7 S	SOURCE	B52EN			0	0	0	0	0	0	0	
111	PROCBLOCK	D1	DSOS		4	1	1	15	-200	-150	1	1	0	
112					8	0	100							
113		SYNCH					0	0	0	0	0	0	0	
114	PROCBLOCK	D8			0	0	8	0	-200	0	-1	1	0	
115		D8 S	SOURCE	ENT			0	0	0	0	0	0	0	
116	COURSE	DSOS			1	1	2000	500	2	440	0	1		
117	PROCBLOCK	D6	PILOTS		6	1	6	8	-200	0	0	2	0	
118		D6 U	USERESRCE	D6			0	0	0	0	0	0	0	
119		SYNCH					0	0	0	0	0	0	0	
120	PROCBLOCK	D5	DSOS		3	2	5	23	-1000	0	2	3	0	
121					4	2	100							
122					2	1	100							
123		CORR					0	0	0	0	0	0	0	
124		D5 U	USERESRCE	D5			0	0	0	0	0	0	0	
125		D5 T	SCATSA				0	0	0	0	0	0	0	
126	PROCBLOCK	D4	DSOS		2	1	4	20	-200	150	0	2	0	
127		D4 U	USERESRCE	D4			0	0	0	0	0	0	0	
128		SYNCH					0	0	0	0	0	0	0	
129	PROCBLOCK	D3			0	0	3	0	-200	0	-1	1	0	
130		D3 S	SOURCE	B52EWC			0	0	0	0	0	0	0	
131	PROCBLOCK	D2	DSOS		2	1	2	7	-200	-150	0	2	0	
132		SYNCH					0	0	0	0	0	0	0	
133		D2 U	USERESRCE	D2			0	0	0	0	0	0	0	
134	PROCBLOCK	D1			0	0	1	0	-200	0	-1	1	0	
135		D1 S	SOURCE	EWCT			0	0	0	0	0	0	0	
136	COURSE	CCTSP			2	8	4000	2000	1	440	20	1		
137	PROCBLOCK	CCTSP			0	0	2	30	0	0	-1	1	0	
138		CCTSP U	USERESRCE	CCTSP			0	0	0	0	0	0	0	
139	COURSE	CCTSC			2	4	4000	1500	1	440	20	1		
140	PROCBLOCK	CCTSC			0	0	2	30	0	0	-1	1	0	
141		CCTSC U	USERESRCE	CCTSC			0	0	0	0	0	0	0	
142	COURSE	CCTSD			2	2	4000	1000	1	440	20	1		
143	PROCBLOCK	CCTSD			0	0	2	30	0	0	-1	1	0	
144		CCTSD U	USERESRCE	CCTSD			0	0	0	0	0	0	0	
145	COURSE	CCTSD			2	1	4000	500	1	440	20	1		
146	PROCBLOCK	CCTSD			0	0	2	30	0	0	-1	1	0	
147		CCTSD U	USERESRCE	CCTSD			0	0	0	0	0	0	0	

TRAM INPUT CARDS

CARD NO	FIELD NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
146	COURSE		PMTP2				2	8	2500	2000	1	440	10	1		
149	PROCBLOCK		PMTP2				0	[illegible]	2	30	[illegible]	[illegible]	-1	1	0	
150			PMTP2	U	USERESRCE		PMTP2		0	0	0	0	0	0	0	
151	COURSE		PMTC2				2	4	2500	1500	1	440	10	1		
152	PROCBLOCK		PMTC2				0	[illegible]	2	30	0	[illegible]	-1	1	[illegible]	
153			PMTC2	U	USERESRCE		PMTC2		0	0	0	0	0	0	0	
154	COURSE		PMTO2				2	2	2500	1000	1	440	10	1		
155	PROCBLOCK		PMTO2				0	0	2	30	0	[illegible]	-1	1	0	
156			PMTO2	U	USERESRCE		PMTO2		0	0	0	0	0	0	0	
157	COURSE		PMTO2				2	1	2500	500	1	440	10	1		
158	PROCBLOCK		PMTO2				0	0	2	30	0	0	-1	1	0	
159			PMTO2	U	USERESRCE		PMTO2		0	0	0	0	0	0	0	
160	COURSE		PMTP3				2	8	3000	2000	1	440	10	1		
161	PROCBLOCK		PMTP3				0	0	2	30	0	0	-1	1	0	
162			PMTP3	U	USERESRCE		PMTP3		0	0	0	0	0	0	0	
163	COURSE		PMTC3				2	4	3000	1500	1	440	10	1		
164	PROCBLOCK		PMTC3				0	0	2	30	0	0	-1	1	0	
165			PMTC3	U	USERESRCE		PMTC3		0	0	0	0	0	0	0	
166	COURSE		PMTO3				2	2	3000	1000	1	440	10	1		
167	PROCBLOCK		PMTO3				0	0	2	30	0	0	-1	1	0	
168			PMTO3	U	USERESRCE		PMTO3		0	0	0	0	0	0	0	
169	COURSE		PMTD3				2	1	3000	500	1	440	10	1		
170	PROCBLOCK		PMTD3				0	0	2	30	0	0	-1	1	0	
171			PMTD3	U	USERESRCE		PMTD3		0	0	0	0	0	0	0	
172	COURSE		PMTP5				2	8	3500	2000	1	440	10	1		
173	PROCBLOCK		PMTP5				0	0	2	30	0	0	-1	1	0	
174			PMTP5	U	USERESRCE		PMTP5		0	0	0	0	0	0	0	
175	COURSE		PMTC5				2	4	3500	1500	1	440	10	1		
176	PROCBLOCK		PMTC5				0	0	2	30	0	0	-1	1	0	
177			PMTC5	U	USERESRCE		PMTC5		0	0	0	0	0	0	0	
178	COURSE		PMTO5				2	2	3500	1000	1	440	10	1		
179	PROCBLOCK		PMTO5				0	0	2	30	0	0	-1	1	0	
180			PMTO5	U	USERESRCE		PMTO5		0	0	0	0	0	0	0	
181	COURSE		PMTD5				2	1	3500	500	1	440	10	1		
182	PROCBLOCK		PMTD5				0	0	2	30	0	0	-1	1	0	
183			PMTD5	U	USERESRCE		PMTD5		0	0	0	0	0	0	0	
184	TASK															
185			SYNCH		SYNCH				0	0	0	0	0	0	0	
186			CORR		CORRELATE				0	0	0	0	0	0	0	
187	SETEND															
188	RUB															
189			P1		CLASS 3		INST 3									
190			P2		CLASS 3		INST 3									
191			P3		SIMA 20/30											
192			P4		SIMA 30/40											
193			P5		B-1 20/30											
194			P6		SIMA 20/30		IP1 10									
195			P8		SIMA 20/30											
196			P9		SIMA 20/50											

TRAM INPUT CARDS

CARD NO	FIELD NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
197			C1		SIMA 20/30											
198			C2		SIMB 5											
199			C3		SIMB 10											
200			C4		SIMB 5		ICP1 10									
201			C5		SIMB 10		ICPI 10									
202			C6		SIMB 5											
203			C8		CLASS 3		INST 3		SIMA 20/30							
204			C9		SIMB 10											
205			C10		SIMB 5		ICP1 10									
206			C11		SIMB 10		ICP1 10									
207			D2		CLASS 3		INST 3		IOSO1 2							
208			D3		SIMB 5											
209			D4		SIMB 10											
210			D5		SIMB 5		IOSO1 10									
211			D2		SIMB 1		IDSO1 2									
212			D4		IDSO1 2											
213			D5		SIMB 10											
214			D6		SIMB 3											
215			PMTP2		IP2 20											
216			PMTP3		IP3 20											
217			PMTP5		IP5 20											
218			PMTC2		ICP2 20											
219			PMTC3		ICP3 20											
220			PMTC5		ICP5 20											
221			PMTO2		IOSO2 20											
222			PMTO3		IOSO3 20											
223			PMTO5		IOSO5 20											
224			PMTD2		IDSO2 20											
225			PMTD3		IDSO3 20											
226			PMTD5		IDSO5 20											
227			CCTSP		SIMA 20/30		IP1 20									
228			CCTSC		ICP1 20											
229			CCTSO		IOSO1 20											
230			CCTSD		IDSO1 20											
231			COPILOTS		COPILOTS											
232			B52P		B52P											
233			FB111P		FB111P											
234			B52CP		B52CP											
235			KC135CP		KC135CP											
236			B52BN		B52BN											
237			BNT		BNT											
238			B52EWO		B52EWO											
239			EWOT		EWOT											
240			B-1 20		B-1 20											
241			IP1 20		IP1 20											
242	SETEND															
243	RUD8															
244			B-1 20/30		SIMA		INDIV		ARBITRARY		B-1 20		SIMAP 30		20	
245			B-1 20		B-1 TIME		INDIV		ARBITRARY		IP1 20				20	

TRAM INPUT CARDS

CARD NO	FIELD NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
246			SIMA 30/3		SIMA		INDIV		ARBITRARY				SIMAP 30		20	
247			SIMAP 3		SIMAP		INDIV		ARBITRARY						3	
248			SIMA 30/4		SIMA		INDIV		ARBITRARY				SIMAP 4		30	
249			SIMAP 4		SIMAP		INDIV		ARBITRARY						40	
250			SIMA 20/5		SIMA		INDIV		ARBITRARY				SIMAP 5		2	
251			SIMAP 5		SIMAP		INDIV		ARBITRARY						50	
252			SIMB 5		SIMB		INDIV		ARBITRARY						5	
253			SIMB 10		SIMB		INDIV		ARBITRARY						10	
254			SIMB 1		SIMB		INDIV		ARBITRARY						1	
255			SIMB 3		SIMB		INDIV		ARBITRARY						3	
256			IDSO2 20		IDSO2		INDIV		ARBITRARY						20	
257			IDSO3 20		IDSO3		INDIV		ARBITRARY						20	
258			IDSO5 20		IDSO5		INDIV		ARBITRARY						20	
259			IOSO2 20		IOSO2		INDIV		ARBITRARY						20	
260			IOSO3 20		IOSO3		INDIV		ARBITRARY						20	
261			IOSO5 20		IOSO5		INDIV		ARBITRARY						20	
262			SIMB2 20		SIMB2		INDIV		ARBITRARY						20	
263			SIMB3 20		SIMB3		INDIV		ARBITRARY						20	
264			SIMB5 20		SIMB5		INDIV		ARBITRARY						20	
265			IP1 20		IP1		INDIV		ARBITRARY						20	
266			ICP1 20		ICP1		INDIV		ARBITRARY						20	
267			IDSO1 20		IDSO1		INDIV		ARBITRARY						20	
268			IOSO1 20		IOSO1		INDIV		ARBITRARY						20	
269			IP1 10		IP1		INDIV		ARBITRARY						10	
270			ICP1 10		ICP1		INDIV		ARBITRARY						10	
271			IOSO1 10		IOSO1		INDIV		ARBITRARY						10	
272			IDSO1 10		IDSO1		INDIV		ARBITRARY						10	
273			IP1 2		IP1		INDIV		ARBITRARY						2	
274			ICP1 2		ICP1		INDIV		ARBITRARY						2	
275			IOSO1 2		IOSO1		INDIV		ARBITRARY						2	
276			IDSO1 2		IDSO1		INDIV		ARBITRARY						2	
277			INST 3		INST		CLASS		UNIFORM						3	
278			CLASS 3		CLASSROOM		CLASS		UNIFORM						3	
279			IP2 20		IP2		INDIV		ARBITRARY						20	
280			IP3 20		IP3		INDIV		ARBITRARY						20	
281			IP5 20		IP5		INDIV		ARBITRARY						20	
282			ICP2 20		ICP2		INDIV		ARBITRARY						20	
283			ICP3 20		ICP3		INDIV		ARBITRARY						20	
284			ICP5 20		ICP5		INDIV		ARBITRARY						20	
285			COPILOTS		COPILOTS										0	
286			B52P		B52P										0	
287			FB111P		FB111P										0	
288			B52CP		B52CP										0	
289			KC135CP		KC135CP										0	
290			B52BN		B52BN										0	
291			BNT		BNT										0	
292			B52EWO		B52EWO										0	
293			EWOT		EWOT										0	
294	SETEND															

TRAM INPUT CARDS

CARD NO	FIELD NUMBER 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
295	INVENTORY		SIMA		UNIFORM		4000	10000	2	5	44	0	0	0		
296	INVENTORY		SIMA		UNIFORM		10001	26400	2	99999	44	0	0	0		
297	INVENTORY		SIMAP		UNIFORM		1	26400	2	00000	44	0	0	0		
298	INVENTORY		SIMB		UNIFORM		1	26400	2	00000	44	0	0	0		
299	INVENTORY		SIMB2		UNIFORM		1	26400	2	99999	44	0	0	0		
300	INVENTORY		SIMB3		UNIFORM		1	26400	2	99999	44	0	0	0		
301	INVENTORY		SIMB5		UNIFORM		1	26400	2	00000	44	0	0	0		
302	INVENTORY		IP1		UNIFORM		1	26400	2	99999	44	0	0	0		
303	INVENTORY		ICP1		UNIFORM		1	26400	2	00000	44	0	0	0		
304	INVENTORY		IOSO1		UNIFORM		1	26400	2	00000	44	0	0	0		
305	INVENTORY		IOSO1		UNIFORM		1	26400	2	99999	5280	0	0	0		
306	INVENTORY		DOLLARS		UNIFORM		1	26400	2	99999	5280	0	0	0		
307	INVENTORY		INST		UNIFORM		1	26400	2	00000	44	0	0	0		
308	INVENTORY		IP2		UNIFORM		1	26400	2	99999	44	0	0	0		
309	INVENTORY		ICP2		UNIFORM		1	26400	2	00000	44	0	0	0		
310	INVENTORY		IOSO2		UNIFORM		1	26400	2	00000	44	0	0	0		
311	INVENTORY		IOSO2		UNIFORM		1	26400	2	99999	44	0	0	0		
312	INVENTORY		IP3		UNIFORM		1	26400	2	00000	44	0	0	0		
313	INVENTORY		ICP3		UNIFORM		1	26400	2	00000	44	0	0	0		
314	INVENTORY		IOSO3		UNIFORM		1	26400	2	99999	44	0	0	0		
315	INVENTORY		IOSO3		UNIFORM		1	26400	2	99999	44	0	0	0		
316	INVENTORY		IP5		UNIFORM		1	26400	2	00000	44	0	0	0		
317	INVENTORY		ICP5		UNIFORM		1	26400	2	99999	44	0	0	0		
318	INVENTORY		IOSO5		UNIFORM		1	26400	2	99999	44	0	0	0		
319	INVENTORY		IOSO5		UNIFORM		1	26400	2	00000	44	0	0	0		
320	INVENTORY		CLASSROOM		UNIFORM		1	26400	2	40	44	0	0	0		
321	INVENTORY		B-1 TIME		UNIFORM		1	26400	2	0	440	0	0	0		
322	SOURCE		COPILOTS		UNIFORM		1	13000	2	99999	440	0	0	0		
323	SOURCE		B52P		UNIFORM		1	26400	2	99999	440	0	0	0		
324	SOURCE		FB111P		UNIFORM		1	13000	2	99999	440	0	0	0		
325	SOURCE		B52CP		UNIFORM		1	13000	2	5	440	0	0	0		
326	SOURCE		KC135CP		UNIFORM		13001	26400	2	99999	440	0	0	0		
327	SOURCE		KC135CP		UNIFORM		1	13000	2	99999	440	0	0	0		
328	SOURCE		B52BN		UNIFORM		1	26400	2	00000	440	0	0	0		
329	SOURCE		BNT		UNIFORM		1	13000	2	99999	440	0	0	0		
330	SOURCE		B52EWO		UNIFORM		1	13000	2	99999	440	0	0	0		
331	SOURCE		EWOT													
332	END															

RESOURCE INVENTORIES

RESOURCE NAME	GENERATING FUNCTION NAME	START DATE	END DATE	PARAMETERS	
SIMA	UNIFORM	[illegible]	[illegible]	5	44
SIMA	UNIFORM	1[illegible]1	264[illegible]	99999	44
SIMAP	UNIFORM	1	264[illegible]	99999	44
SIMB	UNIFORM	1	264[illegible]	99999	44
SIMB2	UNIFORM	1	264[illegible]	99999	44
SIMB3	UNIFORM	1	264[illegible]	99999	44
SIMB5	UNIFORM	1	264[illegible]	99999	44
IP1	UNIFORM	1	264[illegible]	99999	44
ICP1	UNIFORM	1	264[illegible]	99999	44
IDSO1	UNIFORM	1	264[illegible]	99999	44
IOSO1	UNIFORM	1	264[illegible]	99999	44
DOLLARS	UNIFORM	1	264[illegible]	99999	5280
INST	UNIFORM	1	264[illegible]	99999	5280
IP2	UNIFORM	1	264[illegible]	99999	44
ICP2	UNIFORM	1	264[illegible]	99999	44
IOSO2	UNIFORM	1	264[illegible]	99999	44
IDSO2	UNIFORM	1	264[illegible]	99999	44
IP3	UNIFORM	1	264[illegible]	99999	44
ICP3	UNIFORM	1	264[illegible]	99999	44
IOSO3	UNIFORM	1	264[illegible]	99999	44
IDSO3	UNIFORM	1	264[illegible]	99999	44
IP5	UNIFORM	1	264[illegible]	99999	44
ICP5	UNIFORM	1	264[illegible]	99999	44
IOSO5	UNIFORM	1	264[illegible]	99999	44
IDSO5	UNIFORM	1	264[illegible]	99999	44
CLASSROOM	UNIFORM	1	26400	99999	44
0-1 TIME	UNIFORM	1	26400	40	44

SOURCES

SOURCE NAME	GENERATING FUNCTION NAME	START DATE	END DATE	PARAMETERS	
COPILOT	UNIFORM	1	264[illegible]	[illegible]	440
B52P	UNIFORM	1	13[illegible]	00000	440
FB111P	UNIFORM	1	264[illegible]	00000	440
B52CP	UNIFORM	1	13[illegible]	99999	440
KC135CP	UNIFORM	1	13[illegible]	5	440
KC135CP	UNIFORM	13[illegible]1	264[illegible]	00000	440
B52BN	UNIFORM	1	13[illegible]	99999	440
BNT	UNIFORM	1	264[illegible]	00000	440
B52EWO	UNIFORM	1	13[illegible]	00000	440
EWOT	UNIFORM	1	13[illegible]	99999	440

AIRCRAFT DELIVERIES

AIRBASE NAME	DAT	NUMBER OF AIRCRAFT
AB2	5721	1
AB3	5771	2

TRAINING COURSES

COURSE NAME	COURSE TYPE	X	Y	MAXIMUM CLASS SIZE	CLASS PERIOD	PRIORITY	EARLIEST GRADUATION DATE	PERSONNEL TYPE
PILOTS	CCTS	[illegible]	[illegible]	2	[illegible]	[illegible]	1	PILOTS
COPILOTS	CCTS	[illegible]	15.00	2	440	[illegible]	1	COPILOTS
OSOS	CCTS	[illegible]	10.00	2	440	[illegible]	1	OSO
DSOS	CCTS	[illegible]	5.00	2	440	[illegible]	1	DSO
CCTSP	PMT	40.00	20.00	1	440	20	1	PILOTS
CCTSC	PMT	40.00	15.00	1	440	20	1	COPILOTS
CCTSO	PMT	40.00	10.00	1	440	20	1	OSO
CCTSD	PMT	40.00	5.00	1	440	20	1	DSO
PMTP2	PMT	25.00	20.00	1	440	10	1	PILOTS
PMTC2	PMT	25.00	15.00	1	440	10	1	COPILOTS
PMTO2	PMT	25.00	10.00	1	440	10	1	OSO
PMTD2	PMT	25.00	5.00	1	440	10	1	DSO
PMTP3	PMT	30.00	20.00	1	440	10	1	PILOTS
PMTC3	PMT	30.00	15.00	1	440	10	1	COPILOTS
PMTO3	PMT	30.00	10.00	1	440	10	1	OSO
PMTD3	PMT	30.00	5.00	1	440	10	1	DSO
PMTP5	PMT	35.00	20.00	1	440	10	1	PILOTS
PMTC5	PMT	35.00	15.00	1	440	10	1	COPILOTS
PMTO5	PMT	35.00	10.00	1	440	10	1	OSO
PMTD5	PMT	35.00	5.00	1	440	10	1	DSO

PROCESSING BLOCKS

BLOCK NAME	BLOCK NUMBER	SYNC-CORRELATE REFERENCE COURSE NAME	BLOCK NUMBER	TYPE	BLOCK DURATION	X	Y	BLOCK PRIORITY	TRANSFER FROM BLOCK NUMBER	TRANSFER FROM PRIORITY	TRANSFER FROM RATIO	TASK NAMES
P6	6	COPILOTS	5	SYNC	20	-2.00	1.50	0	5	0	1.00	P6 U SYNCH
P5	5		4		15	-2.00	0.0	0	4	1	1.00	P5 T
									9	2	1.00	P5 U
P4	4		0		10	-2.00	1.50	0	3	0	1.00	P4 U
P3	3		0		5	-2.00	0.0	0	2	0	1.00	P3 U
P2	2		0		10	-2.00	0.0	0	1	1	0.75	P2 U
									8	1	0.25	
P1	1	OSOS	5	CORR	20	-2.00	1.50	0	10	0	1.00	P1 U CORR
P10	10		0		0	-2.00	0.0	0				P10 S
P8	8	PILOTS	1	CORR	15	-2.00	-1.50	0	11	0	1.00	P8 U CORR
P11	11		0		0	-2.00	0.0	0				P11 S
P9	9		0		10	-2.00	-1.50	0	12	0	1.00	P9 U
P12	12		0		0	-2.00	0.0	0				P12 S
C6	6		0		20	0.0	0.0	0	5	1	0.50	C6 U
									11	1	0.50	
C5	5	COPILOTS	11	SYNC	25	-2.00	1.50	0	4	0	1.00	C5 U SYNCH
C4	4	COPILOTS	10	SYNC	20	-2.00	0.0	0	3	0	1.00	C4 U SYNCH
C3	3		0		15	-2.00	-1.50	0	2	0	1.00	C3 U
C2	2		0		10	-2.00	0.0	0	1	1	0.50	C2 U
									9	1	0.50	C2 T
C1	1	PILOTS	8	CORR	5	-4.00	1.50	0	12	0	1.00	C1 U CORR
C12	12		0		0	-2.00	0.0	0				C12 S
C11	11	OSOS	5	SYNC	10	-2.00	-1.50	0	10	0	1.00	C11 U SYNCH
C10	10	COPILOTS	4	SYNC	5	-2.00	0.0	0	3	0	1.00	C10 U SYNCH
C9	9	COPILOTS	1	CORR	15	-4.00	-1.50	0	8	0	1.00	C9 U CORR
C8	8		0		10	-2.00	0.0	0	13	0	1.00	C8 U
C13	13		0		0	-2.00	0.0	0				C13 S
O5	5	OSOS	6	SYNC	10	-2.00	0.0	0	4	0	1.00	O5 U SYNCH
O4	4		0		22	-2.00	0.0	0	3	0	1.00	O4 U
O3	3	COPILOTS	9	CORR	20	-8.00	0.0	0	2	2	1.00	O3 U
									1	1	1.00	CORR O3 T
O2	2	OSOS	1	SYNC	15	-2.00	1.50	0	7	0	1.00	O2 U SYNCH
O7	7		0		0	-2.00	0.0	0				O7 S
O1	1	OSOS	4	SYNC	15	-2.00	-1.50	0	8	0	1.00	SYNCH
O8	8		0		0	-2.00	0.0	0				O8 S
O6	6	PILOTS	6	SYNC	8	-2.00	0.0	0	5	0	1.00	O6 U SYNCH
O5	5	OSOS	3	CORR	23	-10.00	0.0	0	4	2	1.00	CORR

PROCESSING BLOCKS

BLOCK NAME	BLOCK NUMBER	SYNC-CORRELATE REFERENCE COURSE NAME	BLOCK NUMBER	TYPE	BLOCK DURATION	X	Y	BLOCK PRIORITY	TRANSFER FROM BLOCK NUMBER	TRANSFER FROM PRIORITY	TRANSFER FROM RATIO	TASK NAMES
									2	1	1.00	D5 U
												D5 T
D4	4	DSDS	2	SYNC	2[illegible]	-2.00	1.50	[illegible]	3	[illegible]	1.00	D4 U
												SYNCH
D3	3		[illegible]		[illegible]	-2.00	[illegible]	[illegible]				D3 S
D2	2	DSDS	2	SYNC	7	-2.[illegible]0	-1.50	[illegible]	1	[illegible]	1.00	SYNCH
												D2 U
D1	1		[illegible]		[illegible]	-2.00	[illegible]	[illegible]				D1 S
CCTSP	2		[illegible]		30	[illegible]	[illegible]	[illegible]				CCTSP U
CCTSC	2		[illegible]		30	[illegible]	[illegible]	[illegible]				CCTSC U
CCTSD	2		[illegible]		30	[illegible]	[illegible]	[illegible]				CCTSD U
CCTSD	2		[illegible]		30	[illegible]	[illegible]	[illegible]				CCTSD U
PMTP2	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTP2 U
PMTC2	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTC2 U
PMTD2	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTD2 U
PMTD2	2		[illegible]		3[illegible]	[illegible]	[illegible]	[illegible]				PMTD2 U
PMTP3	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTP3 U
PMTC3	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTC3 U
PMTD3	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTD3 U
PMTD3	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTD3 U
PMTP5	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTP5 U
PMTC5	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTC5 U
PMTD5	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTD5 U
PMTD5	2		[illegible]		30	[illegible]	[illegible]	[illegible]				PMTD5 U

TASKS

TASK NAME	TASK FUNCTION NAME	RUB NAME	TASK TYPE	PARAMETERS
P6 U	USERESRCE	P6	NORMAL	
P5 T	SCATSA		NORMAL	
P5 U	USERESRCE	P5	NORMAL	
P4 U	USERESRCE	P4	NORMAL	
P3 U	USERESRCE	P3	NORMAL	
P2 U	USERESRCE	P2	NORMAL	
P1 U	USERESRCE	P1	NORMAL	
P10 S	SOURCE	B52P	NORMAL	
P8 U	USERESRCE	P8	NORMAL	
P11 S	SOURCE	EP111P	NORMAL	
P9 U	USERESRCE	P9	NORMAL	
P12 S	SOURCE	COPILOTS	NORMAL	
C6 U	USERESRCE	C6	NORMAL	
C5 U	USERESRCE	C5	NORMAL	
C4 U	USERESRCE	C4	NORMAL	
C3 U	USERESRCE	C3	NORMAL	
C2 U	USERESRCE	C2	NORMAL	
C2 T	SCATSA		NORMAL	
C1 U	USERESRCE	C1	NORMAL	
C12 S	SOURCE	B52CP	NORMAL	
C11 U	USERESRCE	C11	NORMAL	
C10 U	USERESRCE	C10	NORMAL	
C9 U	USERESRCE	C9	NORMAL	
C8 U	USERESRCE	C8	NORMAL	
C13 S	SOURCE	KC135CP	NORMAL	
O5 U	USERESRCE	O5	NORMAL	
O4 U	USERESRCE	O4	NORMAL	
O3 U	USERESRCE	O3	NORMAL	
O3 T	SCATSA		NORMAL	
O2 U	USERESRCE	O2	NORMAL	
O7 S	SOURCE	B52BN	NORMAL	
O8 S	SOURCE	BNT	NORMAL	
D6 U	USERESRCE	D6	NORMAL	
D5 U	USERESRCE	D5	NORMAL	
D5 T	SCATSA		NORMAL	
D4 U	USERESRCE	D4	NORMAL	
D3 S	SOURCE	B52EWO	NORMAL	
D2 U	USERESRCE	D2	NORMAL	
D1 S	SOURCE	EWOT	NORMAL	
CCTSP U	USERESRCE	CCTSP	NORMAL	
CCTSC U	USERESRCE	CCTSC	NORMAL	
CCTSO U	USERESRCE	CCTSO	NORMAL	
CCTSD U	USERESRCE	CCTSD	NORMAL	
PMTP2 U	USERESRCE	PMTP2	NORMAL	
PMTC2 U	USERESRCE	PMTC2	NORMAL	
PMTO2 U	USERESRCE	PMTO2	NORMAL	
PMTD2 U	USERESRCE	PMTD2	NORMAL	
PMTP3 U	USERESRCE	PMTP3	NORMAL	
PMTC3 U	USERESRCE	PMTC3	NORMAL	
PMTO3 U	USERESRCE	PMTO3	NORMAL	

TASKS

TASK NAME	TASK FUNCTION NAME	RUB NAME	TASK TYPE	PARAMETERS
PMTD3 U	USERESRC	PMTD3	NORMAL	
PMTP5 U	USERESRC	PMTP5	NORMAL	
PMTC5 U	USERESRC	PMTC5	NORMAL	
PMTD5 U	USERESRC	PMTD5	NORMAL	
PMTD5 U	USERESRC	PMTD5	NORMAL	
SYNCH	SYNCH		NORMAL	
CORR	CORRELATE		NORMAL	

RESOURCE UTILIZATION BLOCKS

RUB NAME	RUB NAMES		
P1	CLASS 3	INST 3	
P2	CLASS 3	INST 3	
P3	SIMA 20/30		
P4	SIMA 20/40		
P5	B-1 20/40		
P6	SIMA 20/3	IP1 1	
P8	SIMA 20/30		
P9	SIMA 20/5		
C1	SIMA 20/30		
C2	SIMB 5		
C3	SIMB 1		
C4	SIMB 5	ICP1 1	
C5	SIMB 1	ICP1 1	
C6	SIMB 5		
C8	CLASS 3	INST 3	SIMA 20/30
C9	SIMB 10		
C10	SIMB 5	ICP1 1	
C11	SIMB 1	ICP1 1	
O2	CLASS 3	INST 3	IOSO1 2
O3	SIMB 5		
O4	SIMB 1		
O5	SIMB 5	IOSO1 1	
D2	SIMB 1	IOSO1 2	
D4	IOSO1 2		
D5	SIMB 1		
D6	SIMB 3		
PMTP2	IP2 20		
PMTP3	IP3 20		
PMTP5	IP5 20		
PMTC2	ICP2 20		
PMTC3	ICP3 20		
PMTC5	ICP5 20		
PMTO2	IOSO2 20		
PMTO3	IOSO3 20		
PMTO5	IOSO5 20		
PMTD2	IDSO2 20		
PMTD3	IDSO3 20		
PMTD5	IDSO5 20		
CCTSP	SIMA 20/30	IP1 20	
CCTSC	ICP1 20		
CCTSO	IOSO1 20		
CCTSD	IDSO1 20		
COPILOTS	COPILOTS		
B52P	B52P		
FB111P	FB111P		
B52CP	B52CP		
KC135CP	KC135CP		
B52BN	B52BN		
BNT	BNT		
B52EWO	B52EWO		

RESOURCE UTILIZATION BLOCKS

RUB NAME	RUB NAMES
EWOT	EWOT
B-1 20	B-1 20
IP1 20	IP1 20

RESOURCE UTILIZATION DESCRIPTION BLOCKS

RUDB NAME	RESOURCE NAME	RESOURCE UTILIZATION GROUPING FUNCTION	RESOURCE UTILIZATION TIMING FUNCTION	SECONDARY RUDB	ALTERNATE RUDB	UNITS OF CONSUMPTION PER UNIT USER
B-1 20/3[illegible]	SIMA	INDIV	ARBITRARY	B-I 2[illegible]	SIMAP 3[illegible]	2[illegible]
B-1 2[illegible]	B-I TIMS	INDIV	ARBITRARY	IP1 2[illegible]		2[illegible]
SIMA 2[illegible]/3[illegible]	SIMA	INDIV	ARBITRARY		SIMAP 3[illegible]	2[illegible]
SIMAP 3[illegible]	SIMAP	INDIV	ARBITRARY			3[illegible]
SIMA 3[illegible]/4[illegible]	SIMA	INDIV	ARBITRARY		SIMAP 4[illegible]	3[illegible]
SIMAP 4[illegible]	SIMAP	INDIV	ARBITRARY			4[illegible]
SIMA 2[illegible]/5[illegible]	SIMA	INDIV	ARBITRARY		SIMAP 5[illegible]	2[illegible]
SIMAP 5[illegible]	SIMAP	INDIV	ARBITRARY			5[illegible]
SIMB 5	SIMB	INDIV	ARBITRARY			5
SIMB 1[illegible]	SIMB	INDIV	ARBITRARY			1[illegible]
SIMB 1	SIMB	INDIV	ARBITRARY			1
SIMB 3	SIMB	INDIV	ARBITRARY			3
IDSO2 2[illegible]	IDSO2	INDIV	ARBITRARY			2[illegible]
IDSO3 2[illegible]	IDSO3	INDIV	ARBITRARY			2[illegible]
IDSO5 2[illegible]	IDSO5	INDIV	ARBITRARY			2[illegible]
IOSO2 2[illegible]	IOSO2	INDIV	ARBITRARY			2[illegible]
IOSO3 2[illegible]	IOSO3	INDIV	ARBITRARY			2[illegible]
IOSO5 2[illegible]	IOSO5	INDIV	ARBITRARY			2[illegible]
SIMB2 20	SIMB2	INDIV	ARBITRARY			20
SIMB3 20	SIMB3	INDIV	ARBITRARY			20
SIMB5 20	SIMB5	INDIV	ARBITRARY			20
IP1 20	IP1	INDIV	ARBITRARY			20
ICP1 20	ICP1	INDIV	ARBITRARY			20
IDSO1 20	IOSOI	INDIV	ARBITRARY			20
IOSO1 20	IOSO1	INDIV	ARBITRARY			20
IP1 10	IP1	INDIV	ARBITRARY			10
ICP1 10	ICP1	INDIV	ARBITRARY			10
IOSO1 10	IOSO1	INDIV	ARBITRARY			10
IDSO1 10	IDSO1	INDIV	ARBITRARY			10
IP1 2	IP1	INDIV	ARBITRARY			2
ICP1 2	ICP1	INDIV	ARBITRARY			2
IOSO1 2	IOSO1	INDIV	ARBITRARY			2
IDSO1 2	IDSO1	INDIV	ARBITRARY			2
INST 3	INST	CLASS	UNIFORM			3
CLASS 3	CLASSROOM	CLASS	UNIFORM			3
IP2 20	IP2	INDIV	ARBITRARY			2[illegible]
IP3 20	IP3	INDIV	ARBITRARY			20
IP5 20	IP5	INDIV	ARBITRARY			20
ICP2 20	ICP2	INDIV	ARBITRARY			20
ICP3 20	ICP3	INDIV	ARBITRARY			20
ICP5 20	ICP5	INDIV	ARBITRARY			20
COPILOTS	COPILOTS					[illegible]
B52P	B52P					[illegible]
FB111P	FB111P					[illegible]
B52CP	B52CP					[illegible]
KC135CP	KC135CP					[illegible]
B52BN	B52BN					[illegible]
BNT	BNT					[illegible]
B52EWO	B52EWO					[illegible]
EWOT	EWOT					[illegible]

AIRBASE INVENTORIES

AIRBASE NAME	INITIAL INVENTORY OF AIRCRAFT	INITIAL INVENTORY OF PILOTS	INITIAL INVENTORY OF COPILOTS	INITIAL INVENTORY OF OSO	INITIAL INVENTORY OF DSO
AB2					
AB3					

AIRBASE EVENTS

EVENT NAME	AIRBASE NAME	TIME	CREW RATIO	ALERT RATIO	HOURS PER CREW PER WEEK
2	AB2	1	2.5	.40	60
3	AB3	1	2.5	.40	60

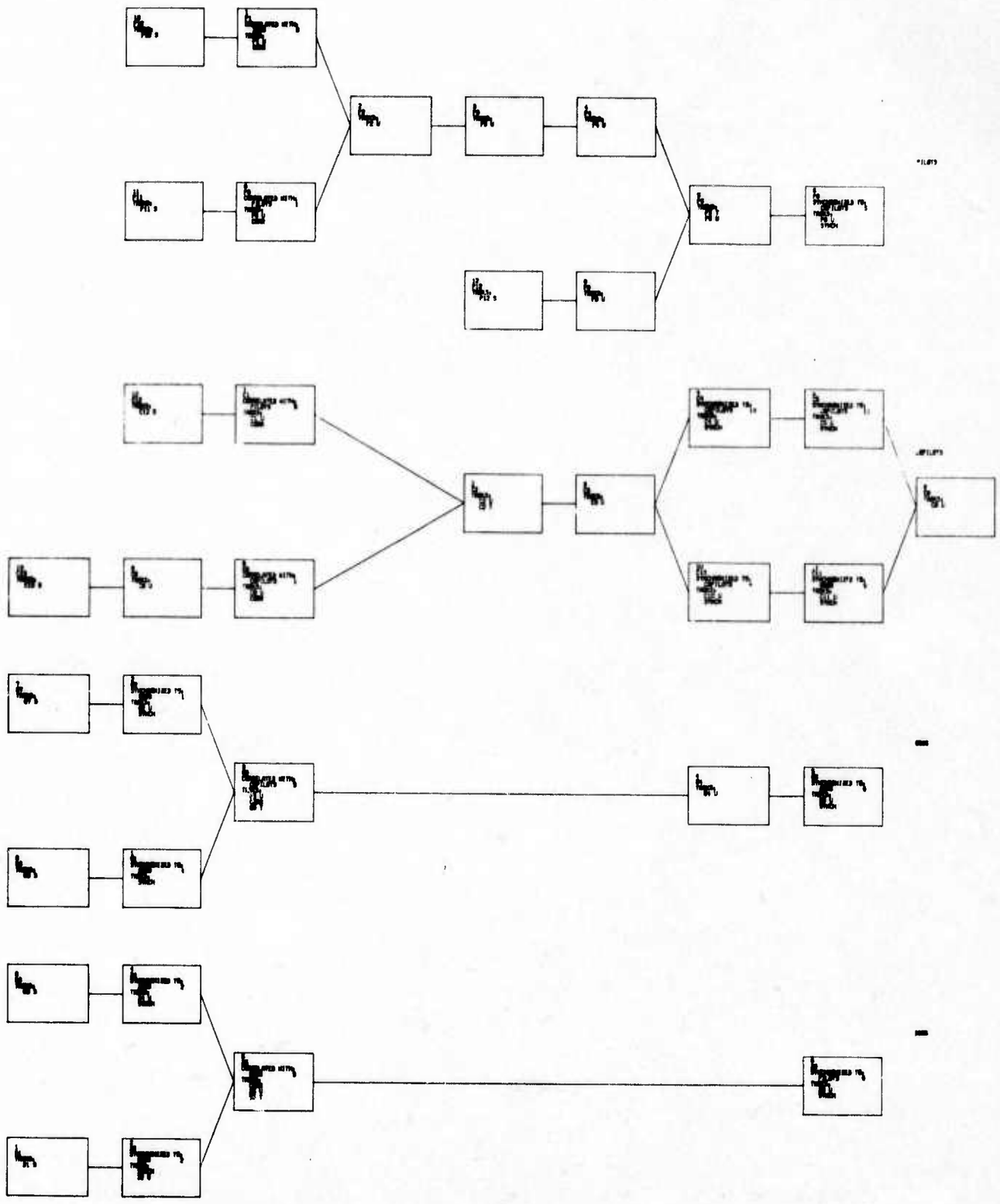
CCTS CARDS

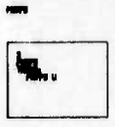
AIRBASE EVENT NAME	COURSE NAME	RATIO	PERSONNEL TYPE
2	PILOTS	1.00	PILOTS
2	COPILOTS	1.00	COPILOTS
2	OSOS	1.00	OSO
2	OSOS	1.00	OSO
3	PILOTS	1.00	PILOTS
3	COPILOTS	1.00	COPILOTS
3	OSOS	1.00	OSO
3	OSOS	1.00	OSO

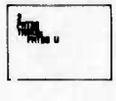
PMT CARDS

AIRBASE EVENT NAME	PERIOD	COURSE NAME	RATIO	TIME LOST	PERSONNEL TYPE
2	5280	CCTSP	1.00	1	PILOTS
3	5280	CCTSP	1.00	1	PILOTS

NORMAL END







1.7 INPUT ROUTINE ERROR MESSAGES

ABOVE CARD IS OUT OF SEQUENCE

A card which requires a header card to precede it, was encountered before the header card. (From INPUT)

ERROR AT CARD NUMBER XX, BLOCK NUMBER SYNCHRONIZED TO IS INVALID - YY

A processing block card specifies a synchronize or correlate reference to another block number which does not exist in the specified course. The card sequence number of the error is given by XX, and the invalid block number is given by YY. (From PROCB2)

ERROR AT CARD NUMBER XX, INVALID TRANSFER BLOCK NUMBER - YY

The processing block specified by card number XX specifies a transfer from a processing block which was never defined within that course. The invalid block number is given by YY. (From PROCB2)

ERROR IN SUBROUTINE IPLOT - INSUFFICIENT STORAGE AVAILABLE TO DO BLOCK DIAGRAM PLOT

The quantity of inputs was great enough so that there is not enough storage left for the plot routines work areas. The program will continue, but no plot will be produced. (From IPLOT)

ERROR IN SUBROUTINE PLOTB - BLOCK NUMBER XX WAS ENCOUNTERED BEFORE ANY BLOCK SPECIFYING A TRANSFER FROM IT

The processing blocks are out of sequence. The position of each processing block is specified as an offset from the block to the right of it (toward graduation). Therefore, each time a processing block is specified, another block must have already specified a transfer from it. (From PLOTB)

2. PHASE 2

2.1 PURPOSE

Phase 2 reformats the data passed from Phase 1, placing the data into a format usable by Phase 3. Phase 2 conducts data validity tests in addition to those performed during Phase 1. Phase 2 calculates, based on input parameters, copilots recoverable and graduation demands for each air base.

2.2 DESCRIPTION

Phase 2 performs five functions:

- reformats name tables
- reformats and tests course decription blocks
- calculates attrition and graduation demands for each air base
- calculates the number of copilots recoverable
- generates source and resource files

The name table contains the names of all functions, blocks, courses, etc. that were specified in the input to Phase 1. These tables are used in Phases 2 and 3 for printing reports. Phase 2 places these name tables in a format that can be used by Phase 2 and Phase 3.

The course description blocks are reformatted by replacing block numbers with pointers to the block storage locations. In addition, structures formed by course description blocks are examined to detect illegal conditions. Synchronization and correlation loops must form a circular list of length no larger than a preset maximum. Course tracks and paths through RUB-RUDB blocks must be of a limited length and must not loop back into themselves (i.e. they must not form circular structures.)

The air base calculations include the following: demands based on initial inventory of aircraft, initial inventory of crew members, and aircraft deliveries; attrition and demands due to attrition; and demands due to proficiency maintenance training requirements. A test is made to insure proficiency maintenance training is feasible. The demand file is sorted by decreasing time.

ERROR IN SUBROUTINE PLOTB - INSUFFICIENT WORKING STORAGE AVAILABLE - FLOW ARROWS WILL BE OMITTED

The course is structured so that many processing blocks specify transfers from block numbers which are not defined. This message is printed when the plot routine runs out of room to store the references until they are defined. Usually the processing blocks can be specified in a different order to reduce the number of such references, but if not, the program will have to be recompiled to make storage available. (From PLOTB)

ERROR ON CARD NUMBER XX, BLOCK NUMBER YY HAS BEEN PREVIOUSLY DEFINED

Two processing blocks with the same number have been defined within the same course. (From PROCB1)

ERROR ON CARD NUMBER XX, DATA BLOCK NAME PREVIOUSLY DEFINED - YY

Card number XX attempts to define a data block with the name YY, but the same name has already been used for another block. (From NAME1)

ERROR ON CARD NUMBER XX, INVALID REFERENCE - YY

The card has referenced another data block which was never defined. The undefined block name or processing block number is given by YY. (From REPLC1, REPLC2)

INSUFFICIENT STORAGE AVAILABLE FOR INPUTS

The amount of input data is greater than the amount the program can store. The program will have to be re-compiled with more storage made available to it. (From INPUT)

INSUFFICIENT STORAGE AVAILABLE TO CONSTRUCT BLOCK NAME TABLE

The quantity of input data is large enough so that there is not enough storage available to do the cross referencing of data block names. The program will have to be re-compiled to make more storage available. (From NAME1, MAIN)

INVALID CARD NAME ON ABOVE CARD

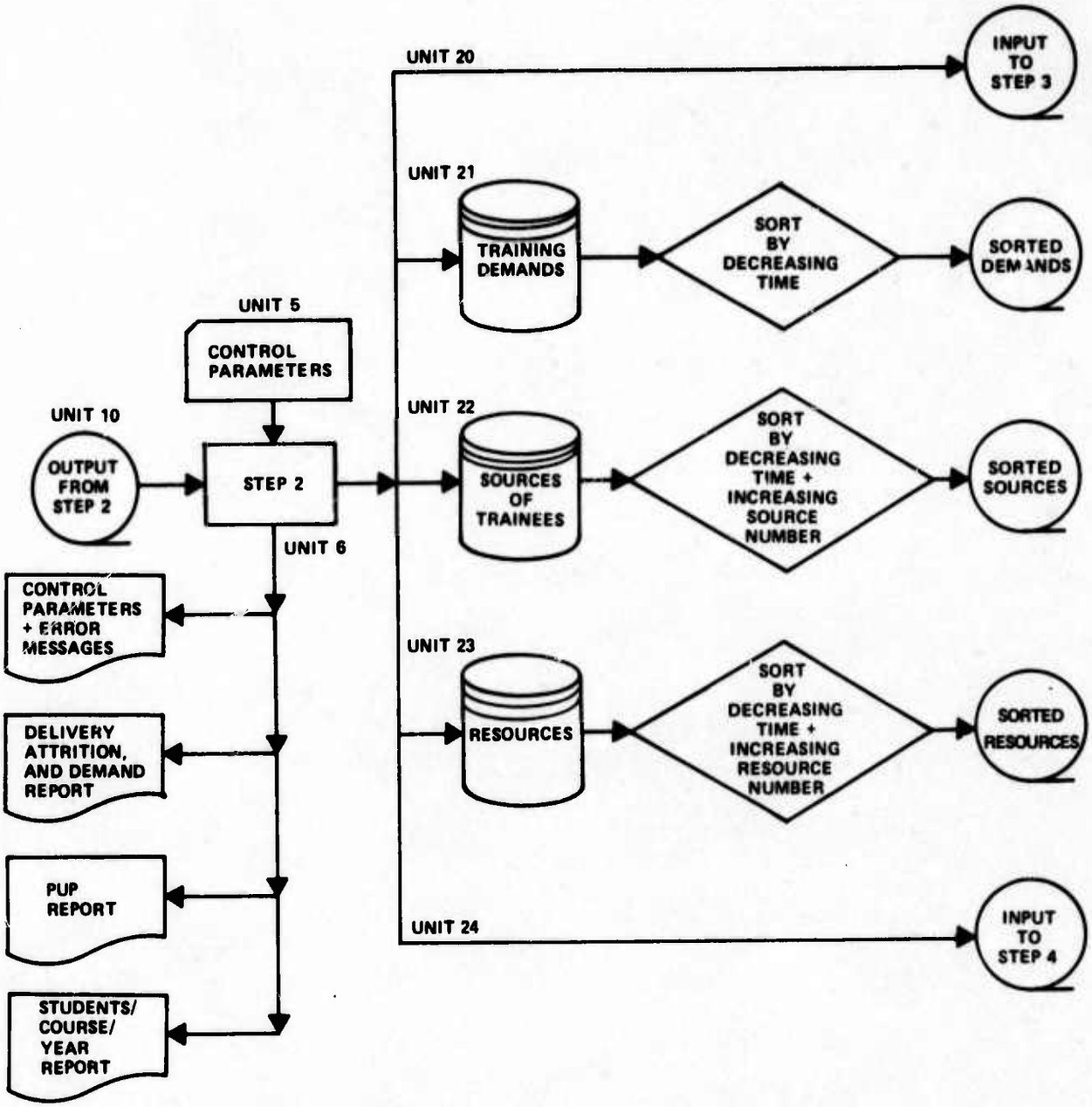
This message appears in the input card listing. The card printed immediately above the message has a card name which is not recognized by the program (card name field is columns 1-10.) (From INPUT)

INVALID VALUE IN FIELD NUMBER XX

The card printed immediately above this error message contains a numeric value which is outside the range allowed for that value. The field number XX, refers to the field number marking at the top of the input card listing. (From TEST)

RESOURCE NAME MUST NOT BE BLANK

The RUDB card which is printed above this error message does not have a resource name specified. (From INPUT)


UNIT 20
INPUT TO STEP 3
UNIT 21
TRAINING DEMANDS
SORT BY DECREASING TIME
SORTED DEMANDS
UNIT 5
CONTROL PARAMETERS
UNIT 10
OUTPUT FROM STEP 2
STEP 2
UNIT 22
SOURCES OF TRAINEES
SORT BY DECREASING TIME + INCREASING SOURCE NUMBER
SORTED SOURCES
UNIT 6
CONTROL PARAMETERS + ERROR MESSAGES
UNIT 23
RESOURCES
SORT BY DECREASING TIME + INCREASING RESOURCE NUMBER
SORTED RESOURCES
DELIVERY ATTRITION, AND DEMAND REPORT
PUP REPORT
UNIT 24
INPUT TO STEP 4
STUDENTS/ COURSE/ YEAR REPORT

It may be relevant to point out that, at this point, training demands remain in unquantized form; quantization to integer numbers of people is a Phase 3 task.

The number of copilots recoverable is calculated from the copilot attrition information. The copilots recoverable are passed to Phase 3 as a source on the source file.

Source and resource files are generated from the source and resource parameters passed from Phase 1. Phase 2 uses the generating function to partition the sources and resources into intervals. The source and resource files are sorted by decreasing time and increasing source or resource number.

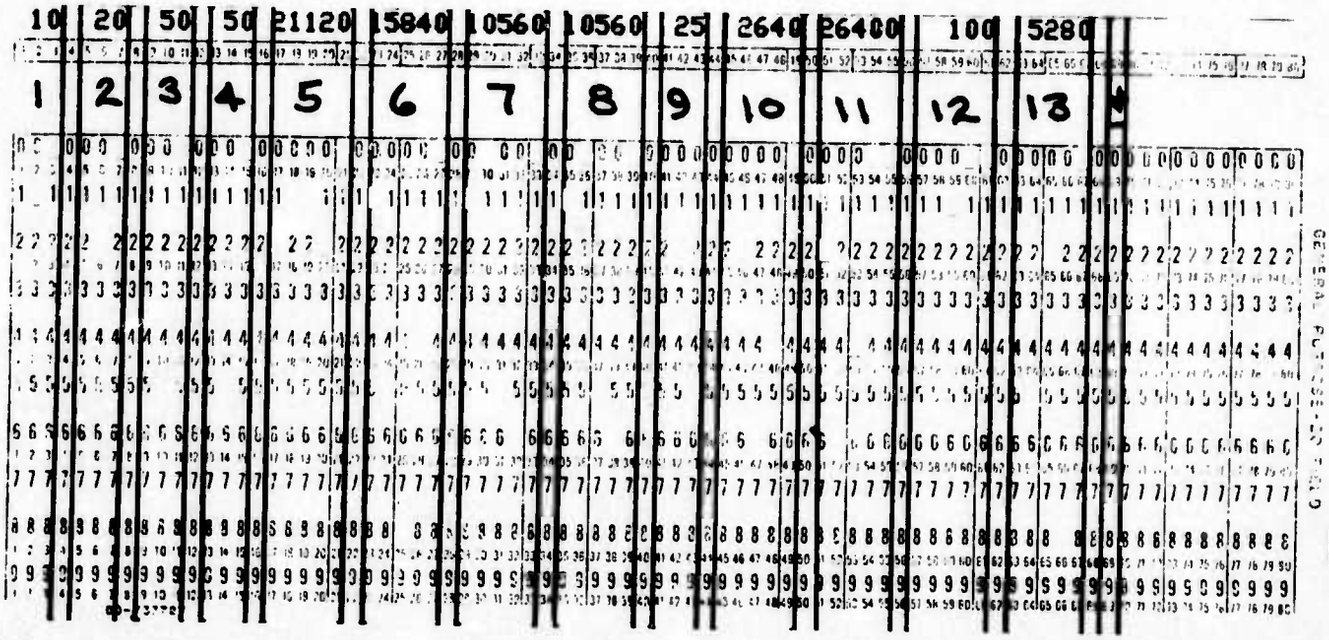
In addition to these five functions, Phase 2 passes selected parameters to Phases 3 and 4.

2.3 INPUT

Phase 3 reads a binary file (unit 10) by Phase 2 and one control card (from unit 5.)

The format of this control card is given on the following page.

FIELD	COLUMNS	FORMAT	CONTENTS
1	1-3	F3.0	Percentage of Pilot Attrition Per Year
2	5-7	F3.0	Percentage of Copilot Attrition Per Year
3	9-11	F3.0	Percentage of OSO Attrition Per Year
4	13-15	F3.0	Percentage of DSO Attrition Per Year
5	17-21	15	Delay Time For Pilots That Are Attrited (Calendar Units)
6	23-27	I5	Delay Time For Copilots That Are Attrited (Calendar Units)
7	29-33	I5	Delay Time For OSO That Are Attrited (Calendar Units)
8	35-39	15	Delay Time For DSO That Are Attrited (Calendar Units)
9	41-43	F3.0	Percentage of Attrited Copilots Recoverable
10	45-49	15	Length of Time a Copilot Can Be Recovered (Calendar Units)
11	51-55	15	End of Simulation Time (Calendar Units)
12	57-61	15	Bucket Size for Performing Air Base Calculations (Calendar Units)
13	63-67	15	Calendar Units Per Year
14	69	L1	T - When Optional Printout Is To Be Suppressed



EXAMPLE

The attrition rate of each type of crewmember is expressed as a percentage of the number of crewmembers at each base at a given time, relative to the number at a "previous" time. The difference in the present time and "previous" or reference time is, by definition, the attrition delay time. Each crewmember type has an independent attrition delay time and attrition rate. Attrition is computed one base at a time.

The number of copilots recoverable at any time is equal to the number of copilots attrited at all air bases multiplied by the percentage of copilots recoverable. These copilots are considered to be available, but the number of calendar units of availability is assumed to be variable.

The end of simulation time is the time in calendar units when the simulation is terminated.

The bucket size for performing air base calculations is used to partition time into intervals.

Calendar units per year are the number of calendar units in a year. This number defines the size of a "calendar unit", used in all calculations as the basic unit of time measurement.

The amount of output can be controlled by use of a logical switch. The default is that all reports will be printed. By setting the logical switch to T, the user can suppress several reports. This control feature will be discussed in more detail in the output section.

2.4 OUTPUT

Phase 2 writes five binary files and three reports. The binary files contain the following information:

- UNIT 20 contains names, blocks and bucket sizes to be used in Phase 3
- UNIT 21 contains graduation requirements used in Phase 3

- UNIT 22 contains sources used in Phase 3.
- UNIT 23 contains resources used in Phases 3 and 4.
- UNIT 24 contains resource information used in Phase 4.

The following reports are produced on unit 6.

- AIR BASE REPORT
- COPILOT RECOVERY REPORT
- COURSE REPORT

The airbase report has a standard output, that may be preceded by an optional higher-level-of-detail output. The standard report contains for each air base individually, and for the aggregate of air bases, the following:

- total number of people attrited
- total number of attritees replaced
- total number of new people required

The optional report adds for each air base and air base bucket:

- number of aircraft
- minimum crew (number of aircraft times crew ratio)
- for each type of crew member (pilot, copilot, OSOS, DSOS):
  - number of crewmembers attrited
  - number of attritees replaced
  - number of new people needed
  - total number at the base

See examples 2.4.1A and 2.4.1B.

The copilot recovery report contains the number of copilots recovered as a function of years (optional), and the total number of copilots recovered (standard). See example 2.4.2.

AIR BASE INVENTORIES: AB2

TIME	AIR CRAFT	MIN CREWS	PILOTS ATT	PILOTS REP	PILOTS NEW	PILOTS TOTAL	COPILOTS ATT	COPILOTS REP	COPILOTS NEW	COPILOTS TOTAL	OSOS ATT	OSOS REP	OSOS NEW	OSOS TOTAL	DSOS ATT	DSOS REP	DSOS NEW	DSOS TOTAL
22500	36	90.00	0.0	0.0	0.0	90.00	0.03	0.03	0.00	90.00	0.43	0.43	0.00	90.00	0.43	0.43	0.00	90.00
22600	36	90.00	0.0	0.0	0.0	90.00	0.03	0.03	0.00	90.00	0.43	0.43	0.00	90.00	0.43	0.43	0.00	90.00
22700	36	90.00	0.0	0.0	0.0	90.00	0.03	0.03	0.00	90.00	0.43	0.43	0.00	90.00	0.43	0.43	0.00	90.00
22800	36	90.00	0.0	0.0	0.0	90.00	0.03	0.03	0.00	90.00	0.43	0.43	0.00	90.00	0.43	0.43	0.00	90.00
22900	36	90.00	0.0	0.0	0.0	90.00	0.04	0.04	0.00	90.00	0.43	0.43	0.00	90.00	0.43	0.43	0.00	90.00
23000	36	90.00	0.0	0.0	0.0	90.00	0.04	0.04	0.00	90.00	0.47	0.47	0.00	90.00	0.47	0.47	0.00	90.00
23100	36	90.00	0.0	0.0	0.0	90.00	0.04	0.04	0.00	90.00	0.47	0.47	0.00	90.00	0.47	0.47	0.00	90.00
23200	36	90.00	0.0	0.0	0.0	90.00	0.04	0.04	0.00	90.00	0.47	0.47	0.00	90.00	0.47	0.47	0.00	90.00
23300	36	90.00	0.0	0.0	0.0	90.00	0.05	0.05	0.00	90.00	0.47	0.47	0.00	90.00	0.47	0.47	0.00	90.00
23400	36	90.00	0.0	0.0	0.0	90.00	0.05	0.05	0.0	90.00	0.52	0.52	0.00	90.00	0.52	0.52	0.00	90.00
23500	36	90.00	0.0	0.0	0.0	90.00	0.05	0.05	0.0	90.00	0.52	0.52	0.00	90.00	0.52	0.52	0.00	90.00
23600	36	90.00	0.0	0.0	0.0	90.00	0.05	0.05	0.0	90.00	0.52	0.52	0.00	90.00	0.52	0.52	0.00	90.00
23700	36	90.00	0.0	0.0	0.0	90.00	0.05	0.05	0.0	90.00	0.52	0.52	0.00	90.00	0.52	0.52	0.00	90.00
23800	36	90.00	0.0	0.0	0.0	90.00	0.06	0.06	0.0	90.00	0.52	0.52	0.00	90.00	0.52	0.52	0.00	90.00
23900	36	90.00	0.0	0.0	0.0	90.00	0.06	0.06	0.00	90.00	0.57	0.57	0.00	90.00	0.57	0.57	0.00	90.00
24000	36	90.00	0.0	0.0	0.0	90.00	0.06	0.06	0.00	90.00	0.57	0.57	0.00	90.00	0.57	0.57	0.00	90.00
24100	36	90.00	0.0	0.0	0.0	90.00	0.06	0.06	0.00	90.00	0.57	0.57	0.00	90.00	0.57	0.57	0.00	90.00
24200	36	90.00	0.0	0.0	0.0	90.00	0.07	0.07	0.00	90.00	0.57	0.57	0.00	90.00	0.57	0.57	0.00	90.00
24300	36	90.00	0.0	0.0	0.0	90.00	0.07	0.07	0.00	90.00	0.62	0.62	0.00	90.00	0.62	0.62	0.00	90.00
24400	36	90.00	0.0	0.0	0.0	90.00	0.07	0.07	0.00	90.00	0.62	0.62	0.00	90.00	0.62	0.62	0.00	90.00
24500	36	90.00	0.0	0.0	0.0	90.00	0.07	0.07	0.00	90.00	0.62	0.62	0.00	90.00	0.62	0.62	0.00	90.00
24600	36	90.00	0.0	0.0	0.0	90.00	0.07	0.07	0.00	90.00	0.62	0.62	0.00	90.00	0.62	0.62	0.00	90.00
24700	36	90.00	0.0	0.0	0.0	90.00	0.08	0.08	0.00	90.00	0.66	0.66	0.00	90.00	0.66	0.66	0.00	90.00
24800	36	90.00	0.0	0.0	0.0	90.00	0.08	0.08	0.00	90.00	0.66	0.66	0.00	90.00	0.66	0.66	0.00	90.00
24900	36	90.00	0.0	0.0	0.0	90.00	0.08	0.08	0.00	90.00	0.66	0.66	0.00	90.00	0.66	0.66	0.00	90.00
25000	36	90.00	0.0	0.0	0.0	90.00	0.08	0.08	0.00	90.00	0.66	0.66	0.00	90.00	0.66	0.66	0.00	90.00
25100	36	90.00	0.0	0.0	0.0	90.00	0.09	0.09	0.00	90.00	0.66	0.66	0.00	90.00	0.66	0.66	0.00	90.00
25200	36	90.00	0.0	0.0	0.0	90.00	0.09	0.09	0.00	90.00	0.71	0.71	0.00	90.00	0.71	0.71	0.00	90.00
25300	36	90.00	0.0	0.0	0.0	90.00	0.09	0.09	0.00	90.00	0.71	0.71	0.00	90.00	0.71	0.71	0.00	90.00
25400	36	90.00	0.0	0.0	0.0	90.00	0.09	0.09	0.00	90.00	0.71	0.71	0.00	90.00	0.71	0.71	0.00	90.00
25500	36	90.00	0.0	0.0	0.0	90.00	0.09	0.09	0.00	90.00	0.71	0.71	0.00	90.00	0.71	0.71	0.00	90.00
25600	36	90.00	0.0	0.0	0.0	90.00	0.09	0.09	0.0	90.00	0.76	0.76	0.00	90.00	0.76	0.76	0.00	90.00
25700	36	90.00	0.0	0.0	0.0	90.00	0.09	0.09	0.0	90.00	0.76	0.76	0.00	90.00	0.76	0.76	0.00	90.00
25800	36	90.00	0.0	0.0	0.0	90.00	0.09	0.09	0.0	90.00	0.76	0.76	0.00	90.00	0.76	0.76	0.00	90.00
25900	36	90.00	0.0	0.0	0.0	90.00	0.09	0.09	0.0	90.00	0.76	0.76	0.00	90.00	0.76	0.76	0.00	90.00
26000	36	90.00	0.0	0.0	0.0	90.00	0.10	0.10	0.0	90.00	0.76	0.76	0.00	90.00	0.76	0.76	0.00	90.00
26100	36	90.00	0.0	0.0	0.0	90.00	0.10	0.10	0.00	90.00	0.80	0.80	0.00	90.00	0.80	0.80	0.00	90.00
26200	36	90.00	0.0	0.0	0.0	90.00	0.10	0.10	0.00	90.00	0.80	0.80	0.00	90.00	0.80	0.80	0.00	90.00
26300	36	90.00	0.0	0.0	0.0	90.00	0.10	0.10	0.00	90.00	0.80	0.80	0.00	90.00	0.80	0.80	0.00	90.00
26400	36	90.00	0.0	0.0	0.0	90.00	0.11	0.11	0.00	90.00	0.80	0.80	0.00	90.00	0.80	0.80	0.00	90.00
TOTAL			0.0	0.0	90.0		2.8	2.8	90.0		35.6	35.6	90.0		35.6	35.6	90.0	

OPTIONAL

EXAMPLE 2.4.1A

AIR BASE INVENTORIES: GRAND TOTAL

PILOTS			COPILOTS			OSOS			DSOS		
ATT	REP	NEW	ATT	REP	NEW	ATT	REP	NEW	ATT	REP	NEW
0.0	0.0	270.0	8.5	8.5	270.0	93.1	93.1	270.0	93.1	93.1	270.0

EXAMPLE 2.4.2B

COPILOTS RECOVERED

YEAR	QUANTITY	
1	0.	OPTIONAL
2	0.	
3	0.	
4	0.	
5	2.	
TOTAL	2.	

EXAMPLE 2.4.2

The course report contains the number of people required from each course as a function of years (optional). The total number of people required from each course and from all courses is printed (standard). See examples 2.4.3A and 2.4.3B.

In addition to these reports, the control parameters are printed. See example 2.4.4.

COURSE	TIME (YEARS) 1	2	3	4	5
PILOTS	0.	83.	118.	65.	5.
COPILOTS	0.	83.	118.	65.	14.
OSOS	0.	83.	118.	85.	78.
DSOS	0.	83.	118.	85.	78.
CCTSP	0.	0.	88.	208.	269.
CCTSC	0.	0.	88.	208.	269.
CCTSO	0.	0.	88.	208.	269.
CCTSD	0.	0.	88.	208.	269.
PMTP2	0.	40.	203.	363.	367.
PMTC2	0.	40.	203.	363.	367.
PMTO2	0.	40.	203.	363.	367.
PMTD2	0.	40.	203.	363.	367.
PMTP3	0.	79.	240.	245.	245.
PMTC3	0.	79.	240.	245.	245.
PMTO3	0.	79.	240.	245.	245.
PMTD3	0.	79.	240.	245.	245.
PMTP5	0.	0.	83.	328.	485.
PMTC5	0.	0.	83.	328.	485.
PMTO5	0.	0.	83.	328.	485.
PMTD5	0.	0.	83.	328.	485.

COURSE DEMANDS

COURSE	TOTAL
PILOTS	270.0
COPILOTS	278.5
USOS	363.1
DSOS	363.1
CCTSP	564.9
CCTSC	564.9
CCTSO	564.9
CCTSD	564.9
PMTP2	972.7
PMTC2	972.7
PMTO2	972.7
PMTD2	972.7
PMTP3	808.5
PMTC3	808.5
PMTO3	808.5
PMTD3	808.5
PMTP5	896.9
PMTC5	896.9
PMTO5	896.9
PMTD5	896.9
GRAND TOTAL	14246.5

EXAMPLE 2.4.3B

CONTROL PARAMETERS FOR STEP2

	ATTRITION PER CENT	DELAY TIME
PILOT	10.	21120
COPILOT	20.	15840
OSO	50.	10560
DSO	50.	10560

PER CENT COPILOTS RECOVERABLE = 25. AND COPILOTS RECOVERY TIME = 2640

MAXIMUM SUMULATION TIME = 26400

CALENDER UNITS PER BUCKET FOR AIR BASE CALCULATIONS = 100

CALENDER UNITS PER YEAR = 5280

EXAMPLE 2.4.4

2.5 ERROR MESSAGES GENERATED IN PHASE 2

1. TOO MANY NAMES - CHANGE VARIABLE MAXNUM AND ARRAY NAMES IN COMMON NAM

   Recompile - array is too small to store names.

2. ARRAY IWORD TOO SMALL TO STORE ALL BLOCKS

   Recompile - array too small to handle all PROC, TASK, RUB and RUDB blocks.

3. TOO MANY BLOCKS LINKED TO PROC BLOCK

   Input error - PROC blocks can have only five preceding or succeeding PROC blocks.

4. NUMBER OF COURSES AND NUMBER OF GRADUATION BLOCKS DO NOT AGREE

   Input error - There must be one and only one graduation block for each course.

5. SYN LOOP STARTING AT i EXCEEDS n BLOCKS

   Input error - Synchronization and correlation, loops are limited to a maximum length. Program must be recompiled to change this limit.

6. SYN LOOP STARTING AT i IS NOT CLOSED

   Input error - Synchronization and correlation loops must be closed. The last PROC block must point to the first PROC block in the loop.

7. SYN LOOP STARTING AT i HAS LENGTH 1

   Input error - Synchronization and correlation loops must contain more than one PROC block

8. TRACK TOO LONG IN COURSE i PROC BLOCKS ARE $p_1 \cdots p_n$

   Input error - PROC blocks in a track exceed a maximum number or the track does not terminate due to a bad transfer card. Recompile to increase maximum track length.

ERROR MESSAGES (CONT.)

9. DEPTH OF RUDB AND RUB STARTING AT RUDB i EXCEEDS n BLOCKS - THESE BLOCKS ARE $b_1 \ldots b_n$

Input error - RUDB and RUB block combinations exceed a maximum number of blocks or these blocks form an infinite loop. Recompile to increase maximum number of blocks.

10. THE NUMBER OF BUCKETS IS OUT OF RANGE MXBUCK=i

Input error - A control parameter error. Bucket size too small. The total number of buckets exceeds the amount of space allocated. Recompile to increase the number of buckets that can be handled.

11. THE NUMBER OF AIR BASES IS OUT OF RANGE NAB=n

Recompile - The number of air bases exceeds the storage allocated to air bases.

12. THE NUMBER OF AIR BASE EVENTS IS OUT OF RANGE NABE=n

Recompile - The number of air base events exceeds the storage allocated to air base events.

13. THE NUMBER OF CCT'S IS OUT OF RANGE NABC=n

Recompile - The number of CCTS courses exceeds the storage allocated to CCTS courses.

14. THE NUMBER OF PMTS IS OUT OF RANGE NABP=n

Recompile - The number of PMT groups exceeds the storage allocated to PMT courses.

15. THE NUMBER OF AIR BASE DELIVERIES IS OUT OF RANGE NABD=n

Recompile - The number of air base deliveries exceeds the storage allocated to air base deliveries.

16. TOO MANY COURSES DEFINED

Recompile - The total number of courses exceeds the space allocated to these courses.

ERROR MESSAGES (CONT.)

17. CCTS COURSE FOR PILOTS NOT DEFINED TIME=t

Input error - There is a demand for a PILOT where no CCTS course was defined.

18. CCTS COURSE FOR COPILOTS NOT DEFINED TIME=t

Input error - There is a demand for a COPILOT where no CCTS course was defined.

19. CCTS COURSE FOR OSOS NOT DEFINED TIME=t

Input error - There is a demand for a OSO where no CCTS course was defined.

20. CCTS COURSE FOR DSOS NOT DEFINED TIME=t

Input error - There is a demand for a DSO where no CCTS course was defined.

21. NO COURSE DEFINED FOR PMT INTERVAL AB=i TIME=t PMT=j

Input error - No course was defined for PMT.

22. PMT NOT FEASIBLE FOR AB=i

Input error - PMT not feasible for this air base.

23. NO AIR BASE EVENT CARD FOR AB=i

Input error - Air base event card missing for this air base.

24. THE NUMBER OF PARAMETERS SPECIFIED FOR RESOURCE i DOES NOT AGREE WITH THE GENERATING FUNCTION REQMTS

Input error - The uniform generating function requires two parameters.

25. THE BUCKET SIZE (SECOND PARAMETER) - MUST BE GREATER THAN ZERO

Input error - The bucket size for resources and sources using the uniform generating function must be greater than zero.

ERROR MESSAGES (CONT.)

26. TOO MANY RESOURCES - INCREASE ARRAY SIZES

Recompile - The number of resources exceeds space allocated to resources.

27. TOO MANY RESOURCE CARDS - INCREASE ARRAY SIZES

Recompile - The number of resource cards exceeds space allocated to resource cards.

28. THE NUMBER OF PARAMETERS SPECIFIED FOR SOURCE i DOES NOT AGREE WITH THE GENERATING FUNCTION REQMTS

Input error - The uniform generating function requires two parameters.

29. SOURCE CARD MISSING FOR COPILOTS

Input Error - A source card for COPILOTS is required.

30. TOO MANY SOURCES - INCREASE ARRAY SIZES

Recompile - The number of sources exceeds space allocated to sources.

31. TOO MANY SOURCE CARDS - INCREASE ARRAY SIZES

Recompile - The number of source cards exceeds space allocated to source cards.

32. THE NUMBER OF PMT COURSES IS OUT OF RANGE NABPC=n

Recompile - The number of PMT courses exceeds the storage allocated to PMT courses.

33. SYN LOOP STARTING AT n DOES NOT HAVE A SYNCH OR CORRELATE TASK

Synchronization task must be specified for all blocks in synchronization loop.

34. SYN LOOP STARTING AT n CONTAINS BLOCK m THAT DOES NOT HAVE A SYNCH OR CORRELATE TASK IN COMMON WITH ORIGINAL BLOCK

All blocks in synchronization loop must have the same synchronization task.

3 PHASE 3

3.1 INTRODUCTION

3.1.1 Contents of This Section

The inputs to Phase 3 are very limited (see Section 3.3.1) as is the normal output report (normally consisting of an echo of the inputs and a successful conclusion message).

The remainder of Section 3.1 gives an overview of Phase 3 of TRAM. It is followed by Section 3.2 describing the major features of the program in greater detail. Sections 3.3 and 3.4 describe the inputs and outputs.

3.1.2 Purpose

The purpose of the Phase 3 TRAM Program is to simulate the progress of related groups of trainees through sets of instructional blocks that define specific training programs. Key items of the simulation are:

1. Calculation of the resources used by the classes during training.

2. Time synchronization of instruction for members of individual crews.

3. Time correlation of instructional blocks that require more than one type of crewmember.

4. Provision for alternate training paths for trainees with different prior experience.

5. Automatic selection of trainee types based on assigned priorities and availability at matriculation time.

6. Graduation scheduling, and

7. Introduction of time lags into the instruction sequence when needed resources are not available or when necessary to achieve synchronization or correlation.

The inputs to Phase 3 consist of:

1. Time ordered training demands.

2. Resource and trainee inventories for the simulated time interval.

3. Description of the training program, and

4. Execution control parameters.

The outputs consist of:

1. Resource and trainee inventories remaining after training demands have been satisfied.

2. Records describing lags, and

3. Records showing the entry of trainees into the system.

Processing is done on an individual class and instructional block basis. That is, the progress of each individual group of trainees through each of the instructional blocks comprising the required training course is simulated in detail.

3.1.3 Treatment Of Time

The courses consist of blocks of instruction which are linked in the order in which they are taken by the trainees. By convention, the trainee proceeds from right to left. The blocks on the left are temporarily before blocks on the right.

The program processes in reverse time order. This leads to a certain awkwardness in expressions involving time such as the "latest", "previous", etc., which all have their correct "real time" meaning rather than in the sense of the order in which the processing is performed.

3.2 PROGRAM DESCRIPTION

3.2.1 Overall Description

The basic functions performed by the Phase 3 TRAM Program are:

1. Form classes from the training demands and associate them with the rightmost instructional block of the pertinent course.

2. Maintain a simulation clock. Processing is done in reverse time order. The simulation starts at the end of the simulation period and progresses to the simulation start time. Thus, the first instructional block processed for a class is the graduation block and the last instructional block processed is the matriculation block.

   The simulation is event oriented. Processing only occurs at those clock times when one or more classes are scheduled to go through a processing block or procbloc, the model analog of the instructional block.

3. Select all the classes that are scheduled for execution at the current simulation clock time. The classes are sorted in order by priority, high priority first.

4. Loop through active classes.

   a) For each active class selected, construct a list of the tasks that must be executed.

   b) Invoke the appropriate modules required to perform the individual tasks. The correlation and synchronization tasks require the merging of the tasks of several classes for execution in parallel. When this happens, the classes that were included in the correlation or synchroniation loop are marked inactive for the remainder of this time period.

   c) Test to see if the task was executed successfully. If so then the next task is executed, and when all tasks are done the next class is processed. If the task failed then the program either stops, prints a message and continues, or lags the class depending on the processing option selected on the control cards.

5. Release the storage occupied by classes that have finished executing a procbloc that has no left branches.

Figure 3.1.1 provides a brief overview of the logical flow of the program. In the following paragraphs the individual items of the flowchart are discussed in greater detail.

1. Read In Control Cards. Initialize Storage

   The formats and content of the control cards are described in Section 3.3.1.

   Initialization (performed by subroutine INIT) consists of clearing storage, establishing the pointers for the linked lists and setting of dimensional constants. Subroutine TRACKD is then invoked to parse the tree structure of the course and create the "Right-to-Left Track Descriptor Blocks" which will be used by the program to effect class transfers and allocation of trainees.

2. Need To Process Additional Training Demands?

   This test is performed to prevent the training demands queue from becoming too long. The program keeps track of the next scheduled simulation clock time. If the time of the last read training demand record is less than the next scheduled execution time, no additional training demands are read.

3. Read Training Demand Records And Place In A Queue

   This function is performed by subroutine FORMQ. The format of the training demand records is described in Section 3.3.2. Refer to Section 4.2.2 for a detailed description of the processing.

4. Create Classes From Queued Demands

   Processing is performed by subroutine FORMC. See Section 3.2.2 for detailed description.

5. Select Classes Scheduled For Execution At This Time

   Processing is performed by subroutine PREPC. Section 3.2.2 discusses the details.

6. Are There Any Active Classes?

   Since TRAM is event oriented, the absence of classes indicates that all required processing has been completed.

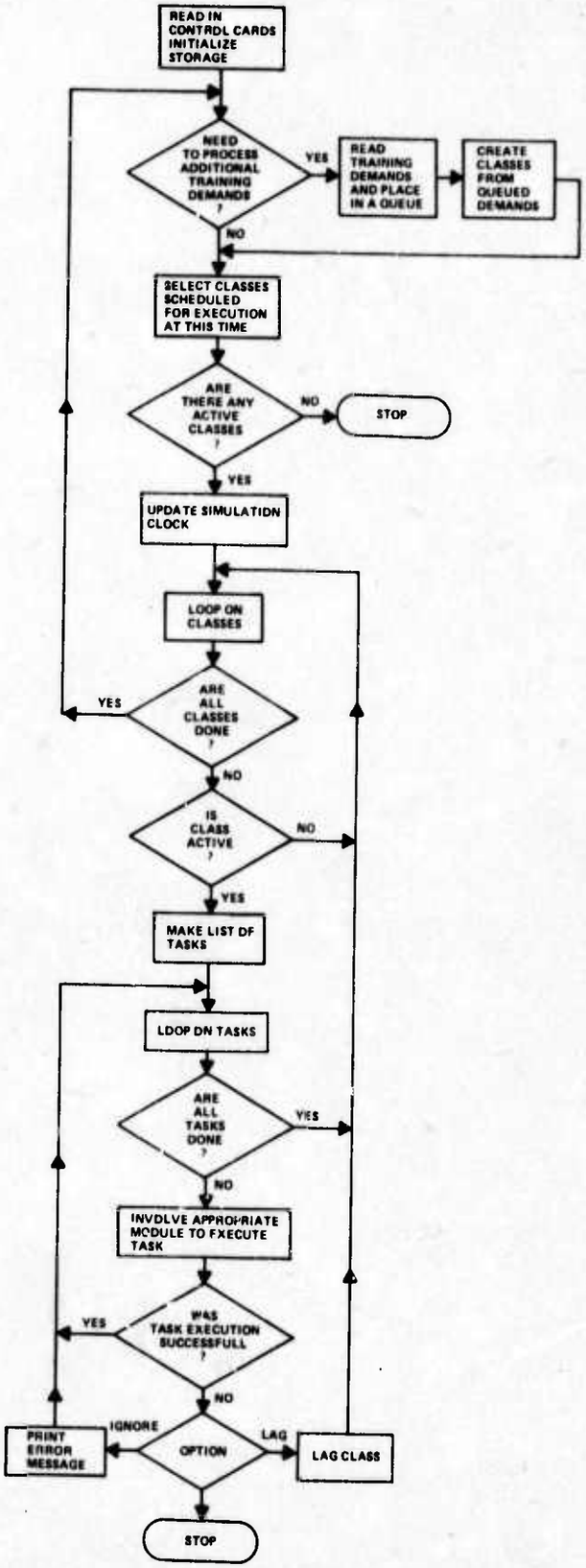


Figure3.1.1 GENERAL FLOW CHART OF PHASE 3 TRAM PROGRAM

7. Update Simulation Clock

This function is performed periodically, approximately every 50 calendar units. Whenever the simulation clock is updated the data management routines store resource and source inventories for times greater than the clock time on disk and fill the buffers with earlier inventories.

8. Loop On Classes

In subroutine PREPC pointers to all classes scheduled for execution at this time were placed in a sequential list. This list, sorted in decreasing order by priority, is the basis for this loop.

9. Is Class Active?

Initially all classes are marked as being active. When classes that are part of a synchronization or correlation loop are concatenated for processing by a class performing a synchronization or correlation task, they are marked as being inactive for the rest of this iteration.

10. Make List Of Tasks

This function is performed by subroutine LSTASK. This list is generated from the list of tasks specified in the active procbloc. The tasks are arranged in order by precedence (synchronization and correlation first, resource utilization second, followed by source allocation (SCATSA or GETSOURCE)). In addition inventory update and class transfer tasks are automatically inserted by the program into the list.

11. Loop On Tasks

This loop normally ranges on the list described in Item 10. However, when a synchronization or correlation task is performed successfully, the list is over-written by subroutine SYNCT which generates a concatenated list of the tasks that must be performed by all the classes in the synchronization or correlation loop.

12. Invoke Appropriate Module To Execute Task

These modules are:

A. SYNC (Synchronization, described in detail in Section 3.2.3)

B. CORR (Correlation, described in detail in Section3.2.4)

C. RESUSE (Resource utilization, described in detail in Section 3.2.5)

D. SCATSA (Source allocation, described in detail in Section 3.2.6)

E. UPDATE (Update of resource inventories, See Section 3.2.5)

F. GET-SOURCE (See Section 3.2.6)

G. DTRNSF (Transfer of classes, described in Section 3.2.7)

13. Was Task Execution Successful?

The only tasks that can fail are resource utilization, source allocation, synchronization and correlation. If source allocation fails the program prints an appropriate error message and stops. It is assumed that the lowest priority track for each course will be an essentially infinite source of trainees.

If a resource utilization task fails then one of three options is available:

A. Stop after printing an error message.

B. Continue after printing an error message.

C. Reschedule (LAG) the class for execution at an earlier time.

If a synchronization or correlation task fails, the classes affected are rescheduled for execution at an earlier time. Figure 3.1.2 shows the input and output data sets used by the Phase 3 TRAM program.

3.2.2 Class Formation

Training demands are calculated in Phase 2 of the TRAM Program from the schedule of aircraft deliveries, forecast attrition rates and requirements for proficiency maintenance training. These demands are passed to Phase 3 in the form of 6-word records sorted in descending order by time. Figure 3.1.2 describes the format of the training demand records.

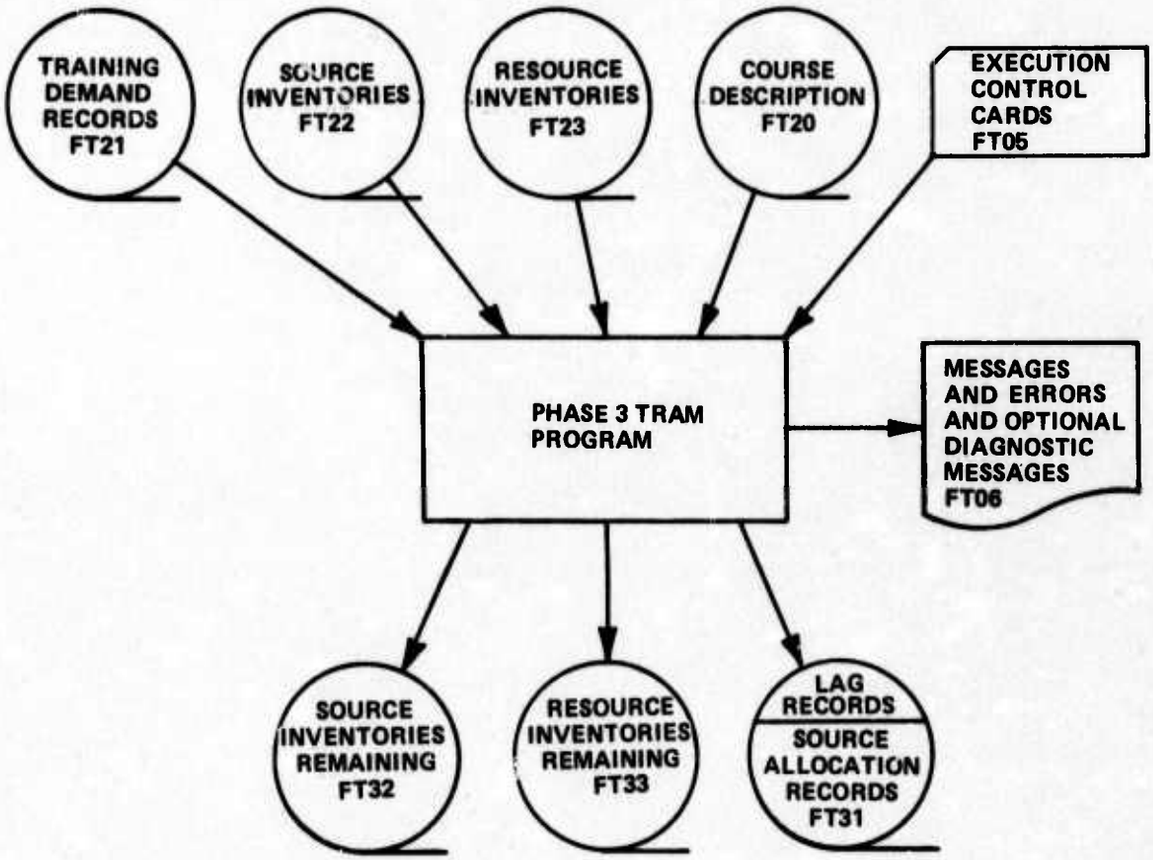


Figure 3.1.2 DATA SETS USED BY PHASE 3 TRAM

The processing of training demand records forms an integral part of the overall logic of Phase 3. After the program initialization is finished subroutine FORMQ is invoked. In this routine a training demand record is read. The time of the demand is used to calculate the set of graduation dates for all courses in the system, by means of the following relations:

1. Graduation number = (Demand Date-Date of first graduation) / intercourse period

2. Current Graduation Date = Date of first graduation + (Graduation number * intercourse period)

3. Previous Graduation Date = Current graduation date - intercourse period

The times of the earliest current graduation and the latest previous graduation are saved. The information in the training demand records is saved in a linked list addressable by individual courses. Additional records are read and stored until the demand time is less than the earliest current graduation.

Subroutine FORMC is invoked next. FORMC loops through the courses and places all demands with a time greater than or equal to the current graduation date for a course in temporary storage. These demands are added up for each course. After the current graduation requirements are known for each course, subroutine CLASCG is used to create classes.

In CLASCG the course grouping by crews specified on the input cards is used to determine the maximum number of complete crews that can be formed at this time. These classes are formed using routines MLTCLS and NEWCLS and are assigned a crew identification number. The trainees that could not be assigned to a crew are made into "extras" classes, and given a crew identification number of -1. Classes are represented as 30-word blocks stored in a linked list.

Subroutines FORMQ and FORMC are only invoked when the simulation clock time is less than the latest previous graduation time. For each cycle of the simulation clock subroutine PREPC is invoked to select the classes that are scheduled for processing at that time. The classes selected are sorted in decreasing order by class priority. Since the class priority is increased each time a class is lagged due to a resource allocation failure, the sort will put it in a better position to avoid further lags.

3.2.3 Synchronization

Synchronization, as the word is used in the TRAM Program, means that all the crew members included in the synchronization loop must perform their assigned tasks simultaneously. An example of this would be that a pilot trainee and a copilot trainee must occupy their respective positions in a two place flight simulator.

The definition of the synchronization loops is imbedded in the course description. Each procbloc can point to another procbloc with which it is synchronized. These pointers can refer to procblocs in another track of the same course or to procblocs in another course. These pointers must form a closed loop. That is, the last procbloc in the synchronization loop points to the first procbloc in the loop. These links are used to force either correlation or synchronization of the procblocs. Correlation is described in the next subsection.

In order to force synchronization, a synchronization task must be specified for each procbloc within the synchronization loop. Before the tasks that a class must execute in a procbloc are processed, the tasks are rearranged in subroutine LSTASK. In this rearrangement the synchronization (or correlation) task is placed at the start of the list. Only one such task is allowed per procbloc. When the synchronization task is executed, the program follows the synchronization loop and picks up all the classes with the proper crew number currently awaiting execution at the individual procblocs. The students in the classes awaiting execution are summed by the course they are in. Finally a check is made to see if the same number of students is in each of the courses linked by the synchronization loop. If the number of trainees in each of the courses is equal then synchronization is possible, and subroutine SYNCT is invoked to make up a new list of tasks. If the numbers of trainees in the individual courses are not the same, then synchronization is impossible at this time and the classes that are currently scheduled for execution in the synchronized procblocs must be lagged until the missing trainees catch up.

Subroutine SYNCT extracts the tasks from the procblocs associated with each of the synchronized classes and arranges them in such a way that all resource consumption tasks will be executed first. If these tasks can be executed successfully, then the resource inventories can be updated and the remaining tasks for the individual classes can be done. If any of the tasks fail because of unavailability of resources, then all the tasks are considered to have failed, no resource utilization update is done and the classes are lagged to a time when the scarce resource is again available.

3.2.4 Correlation

Correlation works very similarly to synchronization. The same method is used for linking procblocs in a closed loop and a correlation task must be specified for each of the procblocs in the loop.

The basic difference lies in the fact that the correlated classes do not need to have the same crew identification number and the equal number of trainees in each course requirement is relaxed to requiring that at least one student be present in each of the courses of the correlation loop.

Only classes that make up a crew can request correlation, but extras classes can be used to satisfy the requirement that trainees be present from each course.

If correlation is impossible at a given time then the total amount of time that the active class has been lagged is checked against the input variable MAXLAG. If the total lag time equals or exceeds the MAXLAG value, the class is marked as an extras class and processed as such. If the total lag time does not exceed the MAXLAG value, the classes are lagged.

3.2.5 Resource Use

Resource utilization by a class going through the tasks of a procbloc is specified by means of the resource utilization blocks (RUB) and the resource utilization description blocks (RUDB).

Each resource utilization task points to a RUB. Each RUB points to from 1 to 6 RUDBS. Each RUDB describes how each resource is to be consumed.

Resource inventories are not described as physical entities but rather as quantities of units of consumption available, or the utility yield per unit time. Thus the inventory for a simulator may specify that it yields 96 hours of use per week. All inventories are in integer units.

Resources are broken into two categories: Primary and Secondary. A secondary resource is one which is used only if the primary is used. One level of secondary resources is permitted. Each RUDB of a primary resource can point to an RUB of secondary resources.

Each resource, primary or secondary, can have one or more alternates. A RUDB can point to an alternate RUDB which can, in turn, point to another alternate and so on.

When the list of tasks for one or more classes (as in the case of synchronized or correlated procblocs) is compiled, the maximum time extent of the procblocs involved is saved. This time interval represents the limits for which the inventory of any resource may need to be known.

During the calculation of the resource consumption, the inventories of the pertinent resources are brought into local temporary storage (if they are not already there) and all updates are done in these local arrays.

These local inventories are updated if the resource allocation is successful. Quantities consumed are saved in other local arrays, until the secondaries have been satisfied. After all resource utilization tasks have been executed successfully, subroutine UPDATE is automatically invoked to perform a permanent update of the inventories of the affected resources to reflect the quantities actually consumed.

3.2.6 Source Allocation

The two program modules responsible for allocating trainees to the proper courses are TRACKD and SCATSA.

Instructional courses in TRAM have generally a tree structure. The graduation block (or rightmost procbloc of a course) constitutes the start of the trunk of the tree. The data path is from right to left (in reverse time sequence) with each procbloc capable of being split into five branches. This branching process can be repeated for each branch any number of times. Procblocs that have more than one left branch are called nodes. Each possible path from the graduation block to a source block (procbloc without any left branches) is called a TRACK. An individual track can have one or more nodes. This tree structure is represented within TRAM by means of TRACK DESCRIPTOR BLOCKS (TDBs). There is one TDB for each node along every track.

The purpose of the TRACKD Module is to create the TDBs required to describe the particular courses being processed. TRACKD is invoked from subroutine INIT during the program initialization phase. The processing performed by TRACKD can be separated into 3 distinct phases:

1. A right to left scan is done for each course in order to locate all procblocs that have no left branches. These procblocs are checked for the presence of a "GETSOURCE" task and pertinent information (procbloc, task, RUB and RUDB, addresses, source number, etc.) is saved in an array indirectly indexed by course number.

2. A left to right scan is performed starting at each source block isolated in Step 1. During the scan time-to-source is updated by the duration of individual procblocs.

   The scan is interrupted whenever a procbloc with more than one left branch is encountered. At these nodes the priorities and proportions are accumulated and track descriptor blocks are created. The recursive formulas for accumulating priorities and proportions are:

   $$\text{Cumulative Proportion} = \text{Current Proportion} \times \text{Previous Proportion}$$

   $$\text{Cumulative Priority} = \text{Current Priority} \times (\text{Previous Priority}/100)$$

   If more than one node is encountered along the track, the later TDB has a pointer to the previous TDB.

3. Finally, the TDB's are link sorted in decreasing order by cumulative priority within each course.

The final result is that there is a track descriptor block for each procbloc with multiple left branches within each track.

The actual allocation of students to individual sources is performed by the SCATSA module. The SCATSA task can only be defined in procblocs with multiple branches. The normal procedure is to use one SCATSA task within a course and to place it at the rightmost node.

When a class is called upon to execute a SCATSA task the following processing takes place:

1. Tracks are selected in order by priority by subroutine LSTRAK. The first time through, the track or tracks with the highest priority are selected.

2. Subroutine ALLOCD calculates the number of students that should be allocated to each track based on the proportions computed by TRACKD for this particular node. Subroutine ALLOC reads in the appropriate inventories and allocates to each track the minimum of quantity available and quantity desired.

3. If all students have been allocated by ALLOCD, Step 7 is executed next.

4. If some students have not been allocated then subroutine ALLOCA attempts to allocate them to any of the currently active tracks that have a positive inventory.

5. If all students have been allocated by ALLOCA, Step 7 is executed next.

6. If some students remain unallocated, then the track or tracks with the next highest priority are selected and processing returns to Step 2.

7. After all students have been allocated, subroutine FRMPTB is invoked to form the predetermined transfer blocks for this class. Section 3.2.7 describes how the PTBs are used to transfer classes along the selected paths.

3.2.7 Transfers

The transfer of classes from a successfully completed procbloc to the next is done automatically by the program. The processing involved in effecting a transfer depends on the number of left branches in the active procbloc and, in the case of multiple branches, whether or not the SCATSA Task has been done by the class. The four possibilities are:

1. No Left Branches This means that the class has completed the leftmost procbloc of the track. The storage space occupied by the class is released and no further processing of the class takes place. (Note: In order to generate a record of "matriculation" a "GETSOURCE" task must be specified for the leftmost procbloc of the track.)

2. One Left Branch The class block is updated to reflect:

   a) The next scheduled execution time for the class, and

   b) The address of the next procbloc to be executed.

3. More Than One Left Branch, SCATSA Not Executed The cumulative priorities and percentages for the individual branches are used to calculate how many students should be sent to each branch. This is done without any regard to the availability of trainees at the required matriculation time. New class blocks are created for each group of students. Crew identification numbers are maintained. The original class block is removed from the active list.

4. More Than One Left Branch, SCATSA Executed This case is recognized by the program by the non-zero value in the variable "IPREDT" in the class block. IPREDT contains the address of the predetermined transfer block created by SCATSA for this class-procbloc combination. The values in the predetermined transfer block are used to create new classes for each of the active tracks. The old class and PTB are removed from active storage.

   CASES 1 and 2 are handled by subroutine DTRNSF.
   CASES 3 and 4 are handled by subroutine SPLIT
   (Invoked automatically by DTRNSF.)

3.2.8 Data Management

Because of the dynamic data flow in the Phase 3 TRAM Program, the standard FORTRAN array and indexing structures are inadequate in terms of core utilization and computational efficiency.

Most of the information used by the program is grouped into blocks of data that are organized using singly linked lists. This method makes it possible to add and delete blocks to the lists without a need for periodic reorganization.

The procblocs, task blocks, resource utilization blocks (RUBs) and resource utilization description blocks (RUDBs) share a common pool of storage in common BLKS and are accessed directly by their addresses. Subroutine BLOCK is used to copy any of these blocks into local storage.

3.3 DESCRIPTION OF INPUTS

The inputs to Phase 3 of TRAM consist of:

1. Execution control parameters (on cards).
2. Time ordered training demands.
3. Resource inventories for the simulated time interval.
4. Trainee inventories for the simulated time interval.
5. A description of the training program.

Each of these inputs is discussed in the following subsections.

3.3.1 Execution Control Parameters

These inputs, contained on 7 or more cards, allow the user to control the time extent of the simulation, the amount of diagnostics generated, the program action in case of synchronization, correlation or resource allocation failure and the grouping of courses for purposes of crew formation.

The detailed formats of the input cards are as follows:

CARD #	CARD COL.	VARIABLE	FORMAT	DESCRIPTION
1	1-8	ITIMES	I8	Start time of simulation
	9-16	ITIMEE	I8	End time of simulation
2	1-50	IFLOW(50)	50I1	Control switches for program flow messages at the beginning of important subroutines If IFLOW(I)=1 flow message is printed If IFLOW(I)=0 flow message is not printed

CARD #	CARD COL.	VARIABLE	FORMAT	DESCRIPTION
3	1-50	IDUMP(50)	50I1	Control switches for diagnostic messages in important routines. If IDUMP(I)=1 diagnostic messages are printed. If IDUMP(I)=0 diagnostic messages are not printed.
4	1-8	ITRNRU	I8	FORTRAN unit for training demand records
5	I-8	MAXLAG	I8	Maximum time a class should be lagged before it starts executing 'extras' tasks instead of waiting for correlation
	9-16	IOPTF	I8	If IOPTF=0, stop if there is not enough resource available to satisfy demands. If IOPTF=1, print a message and continue execution if not enough resource is available to satisfy demands. If IOPTF=2, lag classes if not enough resource is available to satisfy demands
	17-24	IOPTF1	I8	Not used
	25-32	IOPTF2	I8	If IOPTF2=0, stop if synchronization or correlation is impossible at some time. If IOPTF2=1, print a message and continue if synchronization or correlation is impossible at some time. If IOPTF2=2, lag classes that could not be synchronized or correlated.
6	1-8	NCGRPS	I8	Number of course groups
7 to 6+NCGRPS	1-8	NCING(I)	I8	Number of courses in Ith group

CARD #	CARD COL.	VARIABLE	FORMAT	DESCRIPTION
	9-16	ICING(1,I)	18	1st course in Ith group
	17-24	ICING(2,I)	18	2nd course in Ith group
	65-72	ICING(8,I)	18	8th course in Ith group

3.3.2 Training Demand Records

Training demand records are written out by the Phase 2 TRAM program on either tape or disk. They are 6 words long and written without using a format statement.

Before use in Phase 3 of TRAM, the training demand records are sorted on time in decreasing order.

3.3.3 Resource Inventories

The resource inventory records are written out by the Phase 2 TRAM program on either disk or tape. They are 3 words long and written without using a format statement.

The resource records are sorted in decreasing order by time.

3.3.4 Trainee Inventories

The source records describe the trainee inventories. These records are written by the Phase 2 TRAM program on either disk or tape. They are 3 words long and are written without using a format statement.

The source records are sorted in decreasing order by time.

3.3.5 Description of Training Program

The Training Program (also referred to as courses) is described by means of Procblocs, Task Blocks, Resource Utilization Blocks and Resource Utilization Description Blocks. The detailed formats of these data blocks are given in the programmer's guide, Technical Memorandum SAT-6.

These blocks are read into core from FORTRAN Unit 20 when the CLOCK subroutine is invoked for the first time. The addresses of the first procbloc for each course (the Graduation Block) are stored in array IADPBI in common CBLK. Each procbloc points to the procbloc(s) lying to the left and right of it and to the tasks associated with it. Task blocks point to RUBs and RUBs in turn point to the RUDBs. This linked structure permits quick access (using subroutine BLOCK) to information required for performing the different functions of the program (i.e., Class Transfer Tasks, Resource Utilization Tasks,

etc.)

3.4 DESCRIPTION OF OUTPUTS

The outputs of the Phase 3 TRAM program consist of:

1. Echo of the inputs
2. Resource inventories remaining after training demands have been satisfied.
3. Trainee (Source) inventories remaining after training demands have been satisfied.
4. Lag records.
5. Source allocation records.
6. Warning, error and normal end messages.

3.4.1 Resource Inventories

The output resource inventory records are identical in form to the input resource inventory records.

The input inventory minus the output inventory for any given time interval is the amount of the resource consumed during that time to satisfy the training requirements.

3.4.2 Source Inventories

The output source inventory records are identical in form to the input source inventory records.

The input inventory minus the output inventory for any time interval is the number of trainees from that particular source actually assigned to the training program during that time interval.

3.4.3 Lag Records

The lag records are written out on tape or disk by the Phase 3 TRAM program whenever a class has to be lagged.

Note - Processing in TRAM 3 is done in reverse time order. (i.e., last PROCBLOC of a course is done first, first PROCBLOC is done last.) Thus, when a class is lagged, the net effect is to force something to occur at an earlier date.

Class blocks, stored in common CLASSB, are created for each new class of students entering a course at the graduation block and for every time that an existing class is split among different tracks. Class blocks are deleted from the list whenever a procbloc without a left branch is executed. Subroutine NEWCLS creates class blocks and subroutine REMCLS deletes them.

Predetermined transfer blocks, stored in common PTBC, are created by subroutine FRMPTB when a source allocation task (SCATSA) is executed. The pointer to the first PTB is placed in the class block. After a PTB is used to control a class transfer at a node, it is deleted and the space it used is released by subroutine REMPTB. The pointer in the class block is updated to point to the next PTB.

Look-up and updates of resource and source inventories are done by using subroutines GETRES, PUTRES, GETSOR and PUTSOR. Resource and source inventories are stored on tape or disk. When subroutine clock is called for the first time, the buffers allocated to the inventories are filled with data starting at the simulation clock time and extending as far back as space permits. Every time that the subroutine clock is called, inventory records for times greater than the simulation clock time are written out on tape or disk, and the core thus made available is used to read in resource and source inventories for an earlier time.

3.4.4 PHASE 3 Messages

Checking for input errors is done primarily in Phases 1 and 2.

During the execution of Phase 3 checks are made to make sure that the dimensions of the work arrays are not exceeded. When a dimension overrun does occur, the program stops after printing an appropriate error message. In most cases a recompilation of the program specifying larger dimensions will be required. It is suggested that the 'VARY' program be used for effecting the dimension change.

The Phase 3 TRAM program performs a large number of redundant checks on the data during processing. When an error is detected, the program prints an error message and stops. In the list of messages, these errors are marked 'Program Bug' because that is what they would be: program bugs. During the final testing and running of the program, none of these error conditions have occurred, and are not planned to occur under any foreseen circumstances. If any of these errors ever occur, then either the system malfunctioned or a serious program error occurred in TRAM, which will require a programmer for interpretation.

The following is a complete list of the possible stops and the recovery procedures.

1. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 1 IN SUBROUTINE ADDTDQ. STORAGE LIMITS EXCEEDED FOR COURSE TRAINING DEMAND QUEUES IN SUBROUTINE ADDTDQ. PROGRAM MUST BE RECOMPILED.

   Array ICORE in common CTDQ needs a larger dimension. Variable LIMIT1 in common CTDQ should contain the dimension.

2. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 2 IN SUBROUTINE ADDTDQ. ERROR ADDTDQ 511. ILLEGAL COURSE NUMBER NNN MAXIMUM IS MMM.

   Program Bug. (See Section 3.4.4).

3. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 3 IN SUBROUTINE ADDTDQ. JUNK IN STORAGE FOR TRAINING DEMAND RECORDS. I1FREE = NNNN

   Program Bug. (See Section 3.4.4).

4. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 4 IN SUBROUTINE ALLOC. PROGRAM ERROR SUB. ALLOC. IGSTME,ITIME1,ITIME2 = NNN,NNN,NNN

   Program Bug. (See Section 3.4.4).

5. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 5 IN SUBROUTINE ALLOC. ILLEGAL VALUE RETURNED BY GETSOR. NSORCE,IGSTME,ITIME1,ITIME2, INVSRC = NNN NNN NNN NNN NNN

   Program Bug. (See Section 3.4.4).

15. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 15 IN DTRNSF.
ERROR 401 DTRNSF. MORE THAN 1 FORWARD BRANCH IN PROCBLOC.

Error message is followed by a listing of the erroneous PROCBLOC.
Correct inputs to step 1 to insure that a tree structure is followed for the course description.
Rerun steps 1,2 and 3.

16. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 16 IN SUBROUTINE EXECT.
ILLEGAL TASK FUNCTION NNN IN PROCBLOC III.

Correct the task function name in the inputs to step 1 and rerun steps 1,2 and 3.

17. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 17 IN SUBROUTINE EXECT
TASK FAILURE. PROGRAM STOP IN EXECT.

Error message is followed by listing of the class block and the task block.
Either insure that enough resource inventory is available to satisfy demands or select the CONTINUE or LAG options for the bypassing of this error type.

18. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 18 IN SUBROUTINE CORR.
NO SYNC LINK IN PROCBLOC.

Error message is followed by a listing of the PROCBLOC in error.
User specified a CORR task for the PROCBLOC that was not included in a synchronization loop.
Correct input to step 1 and rerun steps 1,2 and 3.

19. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 19 IN SUBROUTINE FRMPTB.
ERROR FRMPTB 411. TRAINEES ASSIGNED TO WRONG TDB NUMSTA, NUMBLK,
NPROCB = NNN NNN NNN

Program Bug. (See Section 3.4.4.).

20. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 20 IN SUBROUTINE GENTDB.
PROGRAM ERROR. SUB. GENTDB. ISRCE1,NOSRCS = NNN NNN

Arrays in common SORDSC need larger dimensions.

21. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 21 IN SUBROUTINE GENTDB.
ILLEGAL NUMBER OF FORWARD BRANCHES IN PROCBLOC.

Error message is followed by a listing of the erroneous PROCBLOC.
Correct inputs to step 1 to insure that a tree structure is followed for the course description.
Rerun steps 1,2 and 3.

22. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 22 IN SUBROUTINE GENTDB. ILLEGAL NUMBER OF LEFT BRANCHES IN PROCBLOC.

Error message is followed by a listing of the erroneous PROCBLOC.

23. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 23 IN SUBROUTINE GENTDB. BACK POINTER DOES NOT MATCH PREVIOUS PROCBLOC.

Error message is followed by listings of the two erroneous PROCBLOCS.
Program Bug (See Section 3.4.4).

24. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 24 IN SUBROUTINE GENTDB. COURSE DIMENSIONS EXCEEDED. NCOURS = NNN

Arrays in common CBLK need to increased.

25. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 25 IN SUBROUTINE LSTASK. ILLEGAL TASK FUNCTION,NPROCB,ITASK,ITSKFN = NNN NNN NNN

Correct the name of the task function in the inputs to step 1 and rerun steps 1, 2 and 3.

26. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 26 IN SUBROUTINE LSTSRC. SOURCE PROCBLOC WITHOUT A GETSOURCE TASK.

Error message is followed by a list of the erroneous PROCBLOC.
Correct inputs to step 1 to insure that all source PROCBLOCS contain a 'GETSOURCE' task.
Rerun steps 1,2 and 3.

27. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 27 IN SUBROUTINE LSTSRC. SOURCE PROCBLOC HAS MORE THAN 1 GETSOURCE TASK.

Error message is followed by a list of the erroneous PROCBLOC.
Correct inputs to step 1 to insure that all source PROCBLOCS contain one and only 1 'GETSOURCE'task.
Rerun steps 1,2 and 3.

28. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 28 IN SUBROUTINE LSTSRC. SOURCE PROCBLOC HAS INVALID NUMBER OF TASKS.

Error message is followed by list of erroneous PROCBLOC
Program Bug (See Section 3.4.4).

29. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 29 IN SUBROUTINE LSTSRC NUMBER OF PROCBLOCS IN COURSE EXCEED WORKING STORAGE IN SUBROUTINE LSTSRC.

Either increase the dimension of array IADRSB in subroutine LSTSRC or reduce the maximum number of PROCBLOCS in each course to fit the dimension.

6. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 6 IN SUBROUTINE ALLOCA
PROGRAM ERROR. SUB. ALLOCA. NASGND,NUMSTD = NNN NNN

Program Bug. (See Section 3.4.4).

7. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 7 IN SUBROUTINE ALLOCD
PROGRAM ERROR. SUB. ALLOCD. NASGND,NUMSTD = NNN NNN

Program Bug. (See Section 3.4.4).

8. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 8 IN SUBROUTINE CALQ.
ILLEGAL RESOURCE UTILIZATION GROUPING FUNCTION NNN

Program Bug. (See Section 3.4.4).

9. ERROR IN STEP 3 of TRAM. ERROR NUMBER 9 IN SUBROUTINE CALQ.
ILLEGAL RESOURCE UTILIZATION TIMING FUNCTION NNN

Correct input to step 1 and rerun steps 1, 2 and 3.

10. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 10 IN SUBROUTINE FORMC
ERROR 601. SUB FORMC. ARRAY LIMITS EXCEEDED.

Arrays in common WRKA need a larger dimension.
Variable LIMTR in common WRKA should contain the dimension.

11. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 11 IN SUBROUTINE CORR
PROCBLOC NNN HAS THE FOLLOWING NNN CLASSES ASSOCIATED WITH IT
NNN NNN NNN ...

Program Bug. (See Section 3.4.4).

12. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 12 IN SUBROUTINE CORR
COURSE NUMBER IN CLASS BLOCK IS NNN BUT COURSE NUMBER IN
PROCBLOC IS NNN.

Error message is followed by a dump of the linked class
block storage area and the PROCBLOC involved.
Program Bug. (See Section 3.4.4).

13. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 13 IN SUBROUTINE DETLAG.
ERROR 401. SUB DETLAG. ARRAY LIMITS EXCEEDED.
NXT,LIMNXT = NNN NNN

Array INVRES in common RES needs a larger dimension.
Variable LIMNXT should contain the dimension.

14. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 14 IN DETLAG.
ERROR 501. SUB DETLAG. BAD TIMES RETURNED BY GETRES.
IPTIME,ITIMEC,LOT1,LOT2 = NNN NNN NNN NNN

Program Bug. (See Section 3.4.4).

30. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 30 IN SUBROUTINE LSTSRC. MORE THAN 1 RUDB SPECIFIED FOR SOURCE.

Error message is followed by listings of the PROCBLOC, task block and resource utilization blocks. Correct the inputs to step1 to insure that only 1 source is called for by the 'GETCOURSE' task.
Rerun steps 1,2 and 3.

31. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 31 IN SUBROUTINE LSTSRC. ALTERNATE RUDBS AND SECONDARY RUBS ARE NOT PERMITTED FOR SOURCES.

Error message is followed by listing of PROCBLOC, task block, resource utilization block and resource utilization description block.
Correct inputs to step 1 and rerun steps 1, 2 and 3.

32. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 32 IN SUBROUTINE NEWCLS . LIMITS FOR CLASS STORAGE ARRAYS EXCEEDED.

Error message is followed by a dump of the class block storage array.
Array ICLASS in common CLASSB needs a larger dimension.
Variable LIMITC in common CLASSB should contain the value of the dimension.

33. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 33 IN SUBROUTINE PBLOCK.

Last PROCBLOC printed did not have a '1' in the block type entry.

34. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 34 IN SUBROUTINE PREPC. LIMITS FOR CURRENT CLASS STORAGE EXCEEDED.

Error is followed by a dump of the class block storage array.
Arrays in common CCLS need a larger dimension.
Test in the IF statement following statement number 110 in PREPC should be modified to reflect the larger dimension.

35. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 35 IN SUBROUTINE REMCLS. BAD LINKS IN CLASS STORAGE.

Error message is preceeded by a dump of the class block storage array.
Program Bug. (See Section 3.4.4).

36. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 36 IN SUBROUTINE REMPTB. BAD LINKS IN PTB STORAGE.

Error message is preceeded by a dump of the predetermined transfer blocks storage array.
Program Bug. (See Section 3.4.4).

37. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 37 IN SUBROUTINE RESINV.
ERROR RESINV 1101. INVALID TIME RETURNED BY GETRES.

IRESNO	LITIM1	LITIM2		
NNNN	NNNN	NNNN		
LOTIM1	LOTIM2	NXT	NXT1	NB
NNNN	NNNN	NNNN	NNNN	NNNN
I	INVRES			
NN	NNN			

:

Program Bug. (See Section 3.4.4).

38. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 38 IN SUBROUTINE RESINV.
ERROR RESINV 1201. DIMENSION OF ARRAY INVRES EXCEEDED.
NXT = NNNN)

Array INVRES in common RES needs a larger dimension.
Variable LIMNXT should contain the value of the dimension.

39. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 39 IN SUBROUTINE RESUSE.
AT TIME NNNN RESOURCE NUMBER NNN WAS EXHAUSTED. PROGRAM STOP.

Either insure that enough resource inventory is available to satisfy demands or select the CONTINUE or LAG options (IOPTF=1 or 2) for bypassing errors of this type.

40. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 40 IN SUBROUTINE RESUSE.
AT TIME NNNN RESOURCE NUMBER NNNN WAS EXHAUSTED. EXECUTION CONTINUES.

Warning. Final resource inventories will not reflect true consumption. Either increase the scarce resources or select the LAG option (IOPTF=2).

41. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 41 IN SUBROUTINE RESUSE
AT TIME NNNN RESOURCE NUMBER NNNN WAS EXHAUSTED. PROGRAM STOP

Same as #39 (but is for primary resources).

42. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 42 IN SUBROUTINE RESUSE.
AT TIME NNNN RESOURCE NUMBER NNNN WAS EXHAUSTED. EXECUTION CONTINUES.

Same as #40 (but is for primary resources).

43. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 43 IN SUBROUTINE RUSER.
PGM.ERROR.RUSER 411. ITOTQ = N

Program Bug. (See Section 3.4.4).

44. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 44 IN SUBROUTINE SCATSA.
ERROR 411 SCATSA. NO TDBS FOR COURSE n,L1,L2= n n

45. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 45 IN SUBROUTINE SCATSA.
ERROR 421. SCATSA. ALL SOURCES EXHAUSTED. NUMERS,ICLSTM,
NDXCLS = n n n

Correct inputs to step 1 to insure that the lowest priority track(s) of each course provide an essentially infinite supply of trainees.

46. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 46 IN SUBROUTINE SPLIT.
ERROR SPLIT 521. WRONG PROPORTIONS IN PREDETERMINED TRANSFER BLOCK.

NUMCRS	NOSTDS	IPRTYC	ICLSTM	NPROCB	ISTATS	IPREDT
n	n	n	n	n	n	n
I	PROP	NEXTPT				
n	n	n				
⋮	⋮	⋮				

Program Bug. (See Section 3.4.4).

47. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 47 IN SUBROUTINE SVRUS1.
ERROR 301. SUBSVRUS1. ARRAY LIMITS EXCEEDED.
NSAVE,ISAVE,INDX1,INDIX2,LIMIS,LIMNS = n n n n n n

If NSAVE is greater than or equal to LIMNS then arrays IADI1,IADI2 and IADS1 need larger dimensions.
If (ISAVE+INDX2-INDX1+1) is greater than or equal to LIMIS then array IAUSED needs a larger dimension.
Variables LIMNS and/or LIMIS should be changed to reflect the increased dimensions.

48. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 48 IN SUBROUTINE SYNC.
NO SYNC LINK IN PROCBLOC

Error message is followed by list of erroneous PROCBLOC.
A SYNC task was specified for a PROCBLOC that is not included in a synchronization loop. Correct inputs to step 1 to insure that all PROCBLOCS having SYNC or CORR tasks are part of closed synchronization loops. Rerun steps 1,2 and 3.

49. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 49 IN SUBROUTINE SYNC
PROCBLOC n HAS THE FOLLOWING n CLASSES ASSOCIATED WITH IT

Program Bug . (See Section 3.4.4).

50. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 50 IN SUBROUTINE SYNC
COURSE NUMBER IN CLASS BLOCK IS n BUT COURSE NUMBER IN PROCBLOC IS n

Error message is followed by a dump of the class blocks linked storage and a listing of the erroneous PROCBLOC
Program Bug . (See Section 3.4.4).

51. ERROR IN STEP 3 OF TRAM ERROR NUMBER 51 IN SUBROUTINE TBLOCK.

The task block printed last does not have a '2' in the 'block type' field.
Program Bug. (See Section 3.4.4).

52. PROGRAM ERROR. CURRENT TIME NOT DECREMENTED.
LSTIME,ITIMEC= n n
LIST OF ACTIVE CLASSES

ERROR IN STEP 3 OF TRAM. ERROR NUMBER 52 IN SUBROUTINE TRAM3

Program Bug. (See Section 3.4.4).

53. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 53 IN SUBROUTINE WRUB

The resource utilization description block printed last does not have a 3 in the 'block type' field.
Program Bug. (See Section 3.4.4).

54. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 54 IN SUBROUITNE WRUDB.

The resource utilization description block printed last does not have a '4' in the 'block type' field.
Program Bug. (See Section 3.4.4).

55. STEP 3 OF TRAM COMPLETED NORMALLY.

Normal job termination message.

56. ERROR IN STEP 3 of TRAM. ERROR NUMBER 56 IN SUBROUTINE INIT.
ILLEGAL SIMULATION START AND/OR STOP TIMES. ITIMES, ITIMEE= n n

Correct input card and rerun Step 3
ITIMES must be less than ITIMEE and not less than zero.
ITIMEE must be less than 99999

57. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 57 IN SUBROUTINE INIT
ILLEGAL FORTRAN INPUT UNIT FOR TRAINING DEMAND RECORD. ITRNRU = n

Correct input card and rerun Step 3
Permissible value range is 1-99

58. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 58 IN SUBROUTINE INIT
ILLEGAL VALUE FOR MAXIMUM CORRELATION LAG. MAXLAG= n

Correct input card and rerun Step 3
Permissible value range is 0-1000

59. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 59 IN SUBROUTINE INIT.
ILLEGAL OPTIONS SELECTED. IOPTF, IOPTF2 = n n

Correct input card and rerun Step 3
Permissible value range is 0-2

60. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 60 IN SUBROUTINE INIT.
ILLEGAL NUMBER OF COURSE GROUPS. 15 IS MAXIMUM ALLOWED.
NCGRPS = n

Correct input card and rerun Step 3

61. ERROR IN STEP 3 OF TRAM. ERROR NUMBER 61 IN SUBROUTINE INIT.
ILLEGAL COURSE NUMBER IN LIST = n n n n

Correct input card and rerun Step 3

4. MERGE PROGRAM

4.1 PURPOSE

The purpose of this program is to merge the original resources file from phase 2 and the unused resources file from phase 3 into a single resource use file for input to phase 4.

4.2 DESCRIPTION

The merge program reads matched pairs of records from the two input merge files, and outputs single records containing the data from the original pair. There are no inputs to this program other than the binary files passed from the earlier TRAM job steps (files 23, 24 and 33). The main output is also a binary file, which is passed to phase 4. The only printed output consists of diagnostic error messages, or the message "MERGE COMPLETED", which indicates that no errors were detected.

4.3 MERGE PROGRAM ERROR MESSAGES

ERRORNUMBER OF RESOURCES EXCEEDS MAXIMUM

The merge program does not have enough storage to process all of the resources. The program will have to be re-compiled to increase the array dimensions.

ERRORRECORD IS MISSING FROM FILE 23

A record on file 33 was read that logically precedes the last record read from file 23. This problem should never occur, but could be caused by an abnormal condition during TRAM phase 2, phase 3 or the sort steps for the two files.

ERRORRECORD READ ON FILE 23 BEFORE ITS EXPECTED TIME

The expected time is the time of the last record for that resource, plus its bucket size. This problem should never occur, but abnormal condition during TRAM phase 2, or in the sort step for file 23.

5 PHASE 4

5.1 PURPOSE

The purpose of the Phase 4 TRAM program is to develop the costs associated with the training system. The costs are presented in current dollars and in then-year dollars, assuming a constant compounded inflation rate. The then-year dollars can also be used to discount future costs by means of negative inflation rates.

5.2 DESCRIPTION

The Phase 3 TRAM program provides a time history of training resource usage in buckets. A bucket is a time period that is on the order of a student's near term planning horizon so as to avoid the detailed (micro) scheduling of resources, but still preserve problems in resource allocation. These buckets are far too small for planning purposes, so they are summarized by the reporting period. The phase 4 TRAM has a user specified periodic reporting period and an automatic yearly reporting period. The periodic reporting period can be either greater or less than the yearly period, to obtain, for example, quarterly reports or 5-year summaries.

In order to (1) properly treat simulators and trainers that can be separated into components for individual training or combined to provide full or partial crew training, and (2) allow computation of facility related costs, the concepts of resource generating units (RGU), primary resources, secondary resources, and resource generator components must be introduced. A primary resource for the purposes of Phase 4 is a resource whose use is determined by Phase 3. (Phase 3 has secondary resources as well.) Secondary resources are used in both programs to simplify coding of the use of resources such as simulators that have associated (secondary) use of personnel, facilities, utilities, etc. An RGU is a device such as a trainer that generates some number of resource units in a bucket. For example, the resource is trainee contact hours and the resource generator device is the trainer. A resource generator unit can correspond to a single real unit such as a simulator, but could also refer to multiple devices such as a room full of carrels.

A resource generator component is a part of an RGU that is separable. It is envisioned that resource components will be used to treat the special problem of computing the use of trainers that have multiple display panels using a single display board and computer, and for computing the use of separable trainers and simulators. For example, if the DSO station is used for 30 hours per week, the full crew simulator for 250 hours, the OSD station for 120 hours and the flight station (pilot and copilot) for 200 hours, the number of simulator hours required is 450 since the combination of full simulator and flight station is the constraining use of the total facility.

Plots of the use of any of the resources can be called for through the inputs. Resource-use plots indicate the average, peak and available levels of use, and the number of RGUs required as a function of time.

Hardware once purchased is assumed to last throughout the training period with an allowance for update, repair and maintenance. Instructional personnel are RGUs also. Personnel, once assigned, can later be transferred. However, they suffer some inertia in their use. It is assumed that a yearly planning horizon is appropriate for such RGUs. If so indicated, returnable RGUs are adjusted yearly to match the demand.

Costs developed for each resource use are given in Table 5-1.

TABLE 5-1

COST DATA

• RDT&E	One time if resource is used Distributed uniformly over N years before the first delivery
• Initial Investment	Based on a first-unit cost and a cost-quantity relationship Attributed to the time of delivery
• Recurring Investment	Based on the current number of resource generator units
• Recurring Investment	Based on a yearly charge independent of number of units as long as at least one unit is owned
• Operations & Maintenance	Based on the current number of resource generator units
• Operations & Maintenance	Based on the current use of the resource

As the quantity of devices increas, the cost to buy and maintain per unit decreases. The system is said to "learn" how to make or use the device. The learning ratio, L_R, is the ratio of the average cost of N units to the average cost of 2N units. Normally $0. \leq L_R \leq 1$. The cost of N units is

$$C_N = N \times (\text{cost for 1 RGU}) \times (L_R)^{\text{Log}_2 N}$$

The learning ratio is applied to Initial Investment per RGU, Recurring Investment per RGU and O+M per RGU. To have the learning ratio not applied, a negative value of these inputs is indicated. In this case, the cost of N units is given by

$$C_N = N \times (\text{Cost for 1 RGU})$$

If RGUs are present in a given year and M+N units are required in the next year, the initial investment for the additional units is calculated by

$$C_N = C_{N+M} - C_N$$

Returnable RGUs are returned without refund. Should they be later required, a new purchase in accordance with the above formulation will be tallied.

Summaries of time phased RDT&E, Investment, Recurring Investment, O&M and totals of this data are produced in current and then-year dollars.

The inputs to the program are given in Table 5.2. Table 5.2 contains two examples, which illustrate the use of the primary, secondary and components of resources. Example I represents the case of a DSO Station separable Full Mission Simulator that can provide only one "scenario" at a time; thus it can be used separately in the sense of operating without a pilot. The "control" component is an essential component and is defined for all of the parts of the FULL Mission Simulator. The components other than the control unit are tallied to record the use of each station. Example II is for a Better Simulator that can run more than one problem at a time. This simulator's use is determined by the maximum overall components (in contrast to Example I where the use is determined by the sum of all "Full M,S. - Control Use.)

TABLE 5.2

INPUTS TO PHASE 4

	Example I		Example II
• For Each Primary Resource			
• Name	Full MS-DSO Station		Better MS-DSO Sta
• Cost Data (See Table 1)	All set to 0		All set to 0
• Number of use units an RGU can produce	5000		5000
• Plot and transferable indicators			
• Number of secondary resources associated with this primary resource	2		1
For each such secondary resource			
• Secondary Resource or component name	Full MS-Control	Full MS-DSO Sta	Better MS-DSO Sta
• Units of use per RGU	0	0	0
• Units of use per unit use of primary resource	1	1	1
• For Each Secondary Resource			
• Name	Full Mission Sim		Better Mission Sta
• Cost Data (See Table 1)	(appropriate values)		(appropriate value
• Number of use units an RGU can produce	5000		5000
• Number of components	5		3
• Plot and transferable indicators			
For each component			
• Name	Full M.S.-Control		Better MS Flight S
	Full M.S.-Pilot Sta		Better MS-OSO Sta
	Full M.S.-Copilot Sta		Better MS-DSO Sta
	Full M.S.-OSO Sta		
	Full M.S.-DSO Sta		

TRAM PHASE 4 PARAMETER CARD

		START TIME [CU's]	END TIME [CU's]	PERIODIC REPORTING PERIOD [CU's]	INFLATION RATE (PER YEAR) (X.XX%)	LEARNING RATE (X.XX%)	
1	PARMS						

PRIMARY RESOURCE

"PR"	NAME	# OF USE UNITS AN RGU CAN PRODUCE PER YEAR	COST DATA (FLOATING POINT - THOUSANDS OF DOLLARS)							PLOT SWITCH 0 = NO 1 = YES	TRANSFERABLE SWITCH 0 = NO 1 = YES	# OF SRC'S	
			RDTE	RDTE PERIOD (INTEGER) YEARS	INITIAL INVESTMENT PER RGU (OR - n)	RECURRING INVESTMENT PER RGU (OR - n)	RECURRING INVESTMENT PER YEAR	O&M PER RGU (OR - n)	O&M/UNIT OF USE				
1 2 3 4 5	6 7 8 9 10 11 12 13 14 15	16 17 18 19 20	21 22 23 24 25	26 27 28 29 30	31 32 33 34 35	36 37 38 39 40	41 42 43 44 45	46 47 48 49 50	51 52 53 54 55	56 57 58 59 60	61 62 63 64 65	66 67 68 69 70	71 72 73 74 75 76 77 78 79 80
1													
2													
3													
4													
5													
6													
7	107												
8													
9													
10													
11													
12													
13													
14													
15													
16													
17													
		SRC NAME	UNITS OF USE/RGU	UNITS OF USE/UNIT OF PR USE									

SECONDARY RESOURCE

"SR"	NAME	# OF USE UNITS AN RGU CAN PRODUCE/YEAR	COST DATA (FLOATING POINT - THOUSANDS OF DOLLARS)							PLOT SWITCH 0 = NO 1 = YES	TRANSFERABLE SWITCH 0 = NO 1 = YES	# OF SRC'S	
			RDTE	RDTE PERIOD (INTEGER) YEARS	INITIAL INVESTMENT PER RGU (OR - n)	RECURRING INVESTMENT PER RGU (OR - n)	RECURRING INVESTMENT PER YEAR	O&M PER RGU (OR - n)	O&M/UNIT OF USE				
1 2 3 4 5	6 7 8 9 10 11 12 13 14 15	16 17 18 19 20	21 22 23 24 25	26 27 28 29 30	31 32 33 34 35	36 37 38 39 40	41 42 43 44 45	46 47 48 49 50	51 52 53 54 55	56 57 58 59 60	61 62 63 64 65	66 67 68 69 72	72 73 74 75 76 77 78 79 80
1													
2													
3													
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													
14													
15													
16													

SECONDARY RESOURCE COMPONENT (SRC) NAMES

5.3 INPUTS

5.3.1 Files From Previous TRAM Programs

The phase 4 program requires two files that were written by previous TRAM job steps. The first of these is the resource information file, which was written to unit 24 by phase 2. The other is the resource use file, which was written to unit 40 by the merge program.

5.3.2 Input Card Description

The input cards for phase 4 have fixed format fields, and a separate card is provided for each type of information. All of these cards have a name field reserved in columns 1-5. This field is used to identify the card type. Coding forms for each of these cards are shown on the following pages. A description of the values contained on these cards is contained in the program description (Section 5.2).

The parameter card must always be the first card of the input deck. This card is followed by the primary resource definitions. Each primary resource definition consists of the PR header card, followed by the number of SRC use cards that was specified on the PR card. The secondary resource definitions follow the primary resource definitions. The SR header card supplies similar information pertaining to the secondary resource, as the PR header card did for the primary resource. The SR card is always followed by one SRC name card, which contains the number of secondary resource component names that were specified on the header card. The input deck must be ended with a final card consisting of the word END punched in the first field.

5.4 OUTPUTS

The following outputs are produced by phase 4:

1. INPUT CARD LISTING

This listing shows the values exactly as they are read from each card. Any errors detected in the inputs will be flagged by error messages.

2. PARAMETER VALUES

This page shows the values of the parameters used by the program. The parameters consist of those values specified on the parameter card, along with the number of calendar units per year, which is read from file 24.

3. PRIMARY RESOURCE DEFINITIONS

This printout shows the primary resource definitions in a tabular format, with each value identified by a column heading. This table includes the bucket sizes that were read from file 24 and matched with the primary resource definitions in the card inputs.

4. SECONDARY RESOURCE DEFINITIONS

This listing is a table of the secondary resources that were specified by the card inputs, and is similar to the table of primary resource definitions.

5. PERIODIC REPORT

This report summarizes the usage of the primary resources during the reporting period.

6. YEARLY REPORT

This report summarizes the resource usage during the year, and also shows the costs resulting from that usage. A separate summary is printed for primary and secondary resources.

7. FINAL COST SUMMARY

This report is printed at the end of the run, and summarizes the costs incurred in each category for each year. The costs are shown in both current and inflated dollars. The report will go back into negative years, if RDTE cost was incurred before the start of the run.

8. USE PLOTS

These plots are produced for each primary and secondary resource that had a 1 specified in the plot switch field of the input cards. Each plot contains four curves: the actual use available, the maximum use available, the amount used, and the number of RGUs on hand for the resource.

9. TEMPORARY FILES

Units 51 and 52 are used as temporary storage for plot data. These files are described in the programmer's guide.

TRAM STEP 4 PARAMETERS

START TIME	0
END TIME	15
PERIODIC REPORTING PERIOD	2
INFLATION RATE (PERCENT)	6.5
LEARNING RATE (PERCENT)	50.0
NUMBER OF CALENDAR UNITS/YEAR	5

5.5 ERROR MESSAGES

The following error messages may be produced by phase 4.

**ERROR IN SUBROUTINE FPLOT
INSUFFICIENT STORAGE AVAILABLE FOR PLOT OF PRIMARY RESOURCE - XX

The name of the primary resource that caused the error is given by XX. The program will have to be re-compiled to increase the array dimensions. (From Subroutine FPLOT.)

**ERROR IN SUBROUTINE FPLOT
INSUFFICIENT STORAGE AVAILABLE FOR PLOT OF SECONDARY RESOURCE - XX

The name of the secondary resource that caused the error is given by XX. The program will have to be re-compiled to increase the array dimensions. (From Subroutine FPLOT.)

ERRORABOVE PRIMARY RESOURCE DOES NOT MATCH WITH ANY FROM PREVIOUS TRAM STEP

The name of the primary resource was not found among those passed from TRAM phase 2 via file 24. (From Subroutine INIT.)

ERROR CARD IS NOT A SECONDARY RESOURCE

A card has been encountered while reading the secondary resource, that is not a seocndary resource definition. (From Subroutine INIT.)

ERROR INSUFFICIENT STORAGE AVAILABLE TO STORE PRIMARY RESOURCES

The program will have to be re-compiled to increase the array dimensions. (From subroutine INIT.)

ERROR INVALID CARD ID

The card printed immediately above this error message, has a card name punched in columns 1-5 that the program does not recognize. (From Subroutine INIT.)

ERROR INVALID PARAMETER CARD

The first input card did not have the card name PARMS punched into columns 1-5. (From Subroutine INIT)

ERROR NO PRIMARY RESOURCES HAVE BEEN DEFINED

No processing can be done without the resource definitions, so the program halts. (From Subroutine INIT.)

ERROR PRIMARY RESOURCE NOT DEFINED - XX

The primary resource named XX, was passed from the previous TRAM step, but was not defined by the card inputs for this step. (From Subroutine INIT.)

ERROR RDTE COSTS HAVE GONE BEYOND YEAR NUMBER XX

RDTE costs have been incurred in a year prior to the year given by XX (a negative number), which is the maximum allowed. If the inputs are correct, and do cause these costs in negative years, the program will have to be re-compiled to increase the array dimensions. (From Subroutine RDTE)

ERROR SECONDARY RESOURCE COMPONENT NAME IS UNDEFINED - XX

The use of the secondary resource component XX was specified by a primary resource, but was never defined. (From Subroutine INIT.)

ERROR TOO MANY SECONDARY RESOURCE COMPONENTS FOR AVAILABLE STORAGE

The program will have to be re-compiled to increase the array dimensions (From Subroutine INIT.)

ERROR TOO MANY SECONDARY RESOURCES FOR AVAILABLE STORAGE

The program will have to be re-compiled to increase the array dimensions (From Subroutine INIT.)

6 Phase 5

6.1 PURPOSE

The purpose of Phase 5 of TRAM is to report on the usage of trainees and on the time lags that occurred in the training system.

6.2 DESCRIPTION

This program produces two output reports; the lag report and the source report. Each of these reports is produced at a periodic interval and at a yearly rate. The periodic reports are independent of the yearly reports, and they can be suppressed completely by specifying a large reporting period. In addition, the two reports are printed at the end of the run to cover the period from the last yearly report up to the end time of the run.

6.3 INPUTS

Binary Files

The following two files are required for input to Phase 5:

20 - names file from Phase 2
31 - source lag file from Phase 3

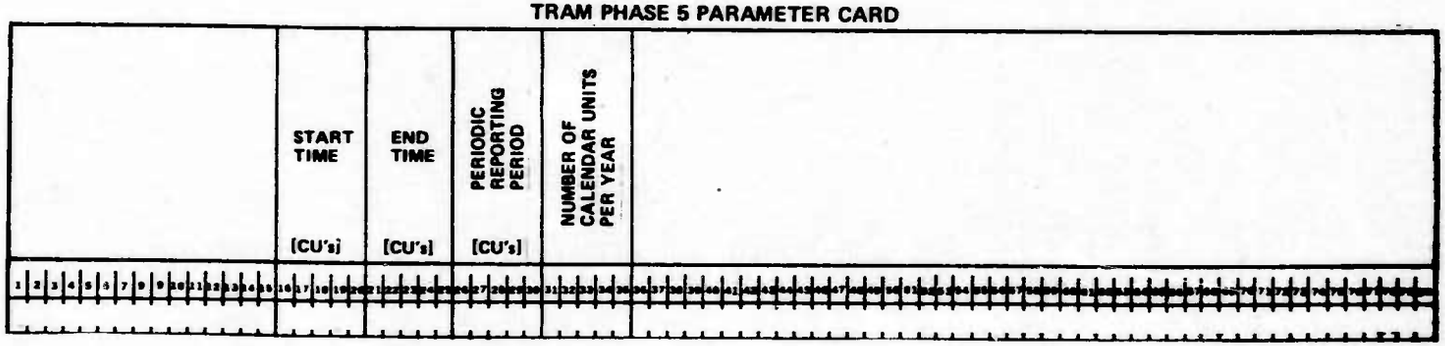
Card Inputs

A single input card is read by Phase 5. A coding form for this card is shown on the next page. The start time and end time specify the period over which this program is to be run. The periodic reporting period gives the length of time between the periodic reports, and the number of calendar units per year specifies how often the yearly reports are to be produced.

6.4 OUTPUTS

The Phase 5 outputs consist of two printed reports. Samples of these reports are shown on the following pages.

TRAM PHASE 5 PARAMETER CARD



Lag Report

The lag report lists each course, processing block, and task that was logged during the period being reported. For each of these, the amount of time logged during the period, and the total lag time to date are shown. The final lag report, which is produced at the end of the run, gives this information for all those that were lagged at any time, rather than just those that occurred during the reporting period.

Source Report

The source report gives the number of trainees used during the reporting period and their average training time. The cumulative totals and averages are also shown. These figures are listed by both course name and by source name.

6.5 PHASE 5 ERROR MESSAGES

**ERROR - INSUFFICIENT STORAGE AVAILABLE FOR LAG REPORT

The program will have to be re-compiled to increase the array dimensions.

**ERROR - INSUFFICIENT STORAGE AVAILABLE FOR SOURCE REPORT

The program will have to be re-compiled to increase the array dimensions.

7. TROLIE PROGRAM

7.1 PURPOSE

The purpose of the TROLIE* program is to provide a rapid and easy method of developing the costs of an aircrew training system. As such, it fits into the set of programs collectively known as TRAM,** hence the acronym TROLIE. TROLIE uses some inputs normally provided to TRAM and can creat the input data used by Phase 4 TRAM to make an economic analysis of training costs. Description of TRAM phases appear elsewhere in this report. This section describes the TROLIE program.

7.2 DESCRIPTION

The training is patterned after that required for manning the B-1. Training consists of an initial training phase called CCTS (Combat Crew Training Squadron) and a periodic training phase called PMT (Proficiency Maintenance Training). There are five courses in CCTS -- pilots, copilots, OSOs, DSOs and Extras. The first four of these areB-1 aircrewmen. Extras will be discussed below. Associated with each type of trainee are tracks. A separate track is assigned to each source of trainees.

Five tracks are allocated to each position. Associated with each air base is a set of PMT tracks. A given PMT track can serve one or more air bases.

The program functions for an input number of years, calculating the aircrews trained and the resources used. A number of air bases are defined by names and the PMT tracks which the airmen at that base attend. Deliveries are tallied by noting the base, year and number of aircraft delivered. The aircraft on a base determine the number of full crews at that base by the input crew ratio.

---

* Training Resources Organized for Logical Integration of Expenses

** Training Resources Analytic Model

Associated with each CCTS track, there is a source. More than one track can use the same source. The capacity of sources is defined by entries in a matrix giving the number of trainees available from each source for each year of interest.

Resource use by trainees is given by a matrix which is the amount of each resource used by a unit trainee in each track. A unit trainee in PMT is all the training a full crew receives in a year. In CCTS the individual crewmembers are unit trainees. The extras are treated as a trainee pair.

The aircraft deployed at a given base are used to compute the number of full crews at that base. (Round-up is used to deal with fractional crews.) Crews are required from CCTS whenever an aircraft is delivered or when an existing crew suffers attrition. Attrition is modeled as a proportion of the existing crews in the system (round-off is used to deal with fractional crews).

Since newly deployed crews are unlikely to suffer attrition immediately upon completion of their training, a delay (input) is inserted. The attrition in a given year is calculated on the basis of the crews present in the given delay period earlier.

In addition to normal attrition, the copilot position suffers "attrition" due to upgrading to pilot (PUPs). The extra copilots which must be trained are called extras. They are treated as pairs so that half of the extras can act as pilot to the other half in instructional blocks which involve flight-crew coordination.

The number of extra classes is half (with round-up) of the number of copilot upgrades. The copilot upgrade is the preferred source of pilot trainees. The source data for copilot upgrades is calculated by the program.

Output consists of data which verifies the inputs, the time-phased training demands, the time-phase trainee source use, and the time-phased resource use.

A data set can be written which is functionally equivalent to the TRAM Phases 1 through 3. Phase 4 of TRAM is normally run in conjunction with TROLIE to provide cost data.

7.3 INPUTS

Input data are indicated in Table 7-1.

Card 1 Contains the floating point (real) data.

Card 2 Contains the fixed point (integer) data. The first year date is for output labeling. The number of calendar units per year is used to determine the year in which deliveries are made and to be passed to Phase 4 TRAM. The row of calendar units in a year must be greater than 3.

Card 3 Contains the air base names, one per card. The format allows for the first two characters to be in columns 9 and 10. Normally TRAM input cards are used. In this case, the field actually starts in column 11.

Card 4 Contains three inputs. IDT is a two-card list of PMT track numbers. The arrays ILTID and IUTID are indexed on air base and point to sections of the PMT track list. Let

ILTID = 1, 2, 4, etc.

IUTID = 2, 3, 5, etc.

IDT = 26, 27, 28, 29, 30, etc.

Then air base 1 uses PMT tracks 26 and 27, air base 2 uses PMT tracks 27 and 28, air base 3 uses PMT tracks 26, 29, and 30, etc.

Card Set 5 is a set of deliveries. A variable number of deliveries can be inserted. The list can be terminated by an END card but any set of 4 characters not equal to "DELI" can be used. The card format is again that of TRAM where the first field of 10 in a delivery card reads "DELIVERY ".

Card Set 6 contains the source index list. The first source is copilots. The first five elements are for the 5 pilots' tracks, the next five for the five copilots' tracks, the next five for the five OSO tracks, the next five for the five DSO tracks and the final five for

the five extras tracks. Sources are chosen in order of their track number within a given position. Sources are drawn upon in the order of pilot, copilot, DSO, OSO and extras. PMT tracks have their associated air base as the source and are not input explicitly.

Card Set 7 is a matrix of trainee sources available, with one or two cards required per year depending on whether there are more or less than 20 sources.

Card Set 8 is a list of resource names, one per card. This is used for heading data and to pass on to Phase 4 TRAM.

Card Set 9 is the resource use matrix, one card (or set of cards) per track. The resource use is for a unit class which is 1 for normal air crew members, 2 copilots for extras and 1 year's training for a full crew for PMT tracks.

```
 INPUTS   TO TROLIE MODEL
NO OF YEARS              15
FIRST YEAR             1975
NO OF AIRBASES           13
NO OF SOURCES            12
NO OF TRACKS             38
NO OF RESOURCES          79
CALENDAR UNITS/YEAR    1500
ATTRITION DELAY           2
ATTRITION RATIO        0.30
CREW RATIO             2.00
PUP UPGRADE RATIO      0.20

  TAPES ARE TO BE PRODUCED
```

Figure 7.3.1 Input Data

AIRBASE LIST

BASE	NAME	ILP	IUP	TRACKS
1	TINKER	1	1	26
2	ELLSWORTH	2	2	27
3	BARKSDALE	3	3	28
4	MINOT	4	4	29
5	MCCONNEL	5	5	30
6	W-RAMA	6	6	31
7	DYESS	7	7	32
8	KINCHELOE	8	8	33
9	CARSWELL	9	9	34
10	GNDFORKS	10	10	35
11	MCGEE-TYS	11	11	36
12	GRISSOM	12	12	37
13	NIAG-FALLS	13	13	38

Figure 7.3.2 Airbase List

DELIVERY LIST											
YEAR	BASE	2	3	4	5	6	7	8	9	10	11	12
1976	0	0	0	0	0	0	0	0	0	0	0	
1977	0	0	0	0	0	0	0	0	0	0	0	
1978	0	0	0	0	0	0	0	0	0	0	0	
1979	0	0	0	0	0	0	0	0	0	0		
1980	0	0	0	0	0	0	0	0	0	0		
1981	0	5	0	0	0	0	0	0	0	0		
1982	0	10	15	15	5	0	0	0	0			
1983	0	0	0	0	10	15	15	8	0			
1984	0	0	0	0	0	0	0	7	15			
1985	14	0	0	0	0	0	0	0	0			
1986	1	0	0	0	0	0	0	0				
1987	0	0	0	0	0	0	0	0				
1988	0	0	0	0	0	0	0	0				
1989	0	0	0	0	0	0	0					
1990	0	0	0	0	0	0	0					

Figure 7.3.3 Yearly Delivery List

TRACK	TRACK SOURCE
1	1
2	2
3	3
4	4
5	4
6	5
7	6
8	7
9	7
10	7
11	8
12	9
13	10
14	10
15	10
16	11
17	12
18	12
19	12
20	12
21	5
22	6
23	7
24	7
25	7

Figure 7.3.4 Track-Source Tie-Together

==> SOURCE AVAILABILITY<===

YEAR	SOURCE ==> 1	2	3	4	5
1976	0	12	96	9999	
1977	0	12	96	9999	
1978	0	12	96	9999	
1979	0	12	96	9999	
1980	0	12	96	9999	
1981	0	12	96	9999	
1982	0	12	96	999	
1983	0	12	96	99	
1984	0	12	96		
1985	0	12	96		
1986	0	12	96		
1987	0	12	96		
1988	0	12	96		
1989	0	12	9		
1990	0	12			

Figure 7.3.5 Source Availability Matrix

==> RESOURCE USE DATA <===

TRACK	RESOURCE ==> BRIEF ROOM	CARREL-GP	DEVICE ONE	B-1 AIRCFT	INS
1	0	0	0	0	
2	89	43	31	4	
3	96	40	43	4	
4	108	68	48	5	
5	0	0	0	0	
6	2	40	43	0	
7	2	68	48	0	
8	2	68	48	0	
9	0	0	0	0	
10	0	0	0		
11	14	78	0		
12	19	84	0		
13	27	110	0		
14	0	0	0		
15	0	0	0		
16	35	50	0		
17	45	60	0		
18	0	0	0		
19	0	0	0		
20	0	0	0		
21	192	80	86		
22	130	136	96		
23	130	136	96		
24	0	0	0		
25	0	0	0		
26	0	0	0		
27	0	0	0		
28	0	0	0		
29	0	0	0		
30	0	0			
31	0	0			
32	0	0			
33	0	0			
34	0	0			
35	0	0			
36	0	0			
37	0	0			
38	0	0			

Figure 7.3.6 Resources Used by Trainee Type (Track)

CCTS SUMMARY

YEAR	DELIVERIES	AC TOTAL	NEW CREW	REP CREW	T CREW	PUPS
1976	0	0	0	0	0	r
1977	0	0	0	0	0	
1978	0	0	0	0	0	
1979	0	0	0	0	0	
1980	0	0	0	0	0	
1981	5	5	10	0	10	
1982	45	50	90	0	100	
1983	48	98	96	3	196	
1984	48	146	96	30	29	
1985	48	194	96	59	?	
1986	1	195	2	88		
1987	0	195	0	116		
1988	0	195	0	117		
1989	0	195	0	117		
1990	0	195	0	117		

Figure 7.3.7 CCTS Training Summary

YEAR	BASE	DELIVERIES	TOTAL	CREWS
1976	1	0	0	0
	2	0	0	0
	3	0	0	0
	4	0	0	0
	5	0	0	0
	6	0	0	0
	7	0	0	0
	8	0	0	0
	9	0	0	0
	10	0	0	0
	11	0	0	0
	12	0	0	0
	13	0	0	0
1977	1	0	0	0
	2	0	0	0
	3	0	0	0
	4	0	0	0
	5	0	0	0
	6	0	0	0
	7	0	0	0
	8	0	0	0
	9	0	0	0
	10	0	0	0
	11	0	0	0
	12	0	0	0
	13	0	0	0
1978	1	0	0	0
	2	0	0	0
	3	0	0	0
	4	0	0	0
	5	0	0	0
	6	0	0	
	7	0	0	
	8	0		
	9	0		
	10	0		
	11			
	12			
	13			
1979				

Figure 7.3.8 Detailed Delivery Report

PMT TRACK TRAINEES

	26	27	28	29	30	31	32	33	34	35	36	37	38
1976	0	0	0	0	0	0	0	0	0	0	0	0	0
1977	0	0	0	0	0	0	0	0	0	0	0	0	0
1978	0	0	0	0	0	0	0	0	0	0	0	0	0
1979	0	0	0	0	0	0	0	0	0	0	0	0	0
1980	0	0	0	0	0	0	0	0	0	0	0	0	0
1981	0	10	0	0	0	0	0	0	0	0	0	0	0
1982	0	30	30	30	10	0	0	0	0	0	0	0	0
1983	0	30	30	30	30	30	30	16	0	0	0	0	0
1984	0	30	30	30	30	30	30	30	30	30	22	0	0
1985	28	30	30	30	30	30	30	30	30	30	30	30	30
1986	30	30	30	30	30	30	30	30	30	30	30	30	30
1987	30	30	30	30	30	30	30	30	30	30	30	30	30
1988	30	30	30	30	30	30	30	30	30	30	30	30	30
1989	30	30	30	30	30	30	30	30	30	30	30	30	30
1990	30	30	30	30	30	30	30	30	30	30	30	30	30

Figure 7.3.9 PMT Training

==> SOURCE USE <===

YEAR	SOURCE ==> 1	2	3	4	5
1976	0	0	0	0	0
1977	0	0	0	0	0
1978	0	0	0	0	0
1979	0	0	0	0	
1980	0	0	0	0	
1981	0	10	0	0	
1982	0	12	78	0	
1983	2	12	85	0	
1984	20	12	94	0	
1985	39	12	96	8	
1986	58	12	20	0	
1987	78	12	26		
1988	78	12	27		
1989	78	12	27		
1990	78	12	27		

Figure 7.3.10 Source Use

==> TRACK USEAGE <===

YEAR	TRACK ==> 1	2	3	4	5
1976	0	0	0	0	0
1977	0	0	0	0	0
1978	0	0	0	0	
1979	0	0	0	0	
1980	0	0	0	0	
1981	0	10	0	0	
1982	0	12	78	0	
1983	2	12	85	0	
1984	20	12	94	0	
1985	39	12	96	8	
1986	58	12	20		
1987	78	12	26		
1988	78	12	27		
1989	78	12	27		
1990	78	12	27		

Figure 7.3.11 Training by Track

==> RESOURCE USE <===

YEAR	RESOURCE ==> BRIEF ROOM	CARREL-GP	DEVICE ONE	B-1 AIRCFT	IN'
1976	0	0	0	0	
1977	0	0	0	0	
1978	0	0	0	0	
1979	0	0	0	0	
1980	0	0	0		
1981	1400	2110	740		
1982	13536	19224	7596		
1983	14872	21320	8395		
1984	18688	28628	10942		
1985	22958	36760	13764		
1986	12110	20512	7826		
1987	15270	27732	10322		
1988	15432	27984	10413		
1989	15432	27984	10413		
1990	15432	27984	1041?		

Figure 7.3.12 Resources Used

7.4 DATA MANAGEMENT

Because of the dynamic data flow in the Phase 3 TRAM Program, the standard FORTRAN array and indexing structures are inadequate in terms of core utilization and computational efficiency.

Most of the information used by the program is grouped into blocks of data that are organized using singly-linked lists. This method makes it possible to add and delete blocks to the lists without a need for periodic reorganization.

The procblocs, task blocks, resource utilization blocks (RUBS) and resource utilization description blocks (RUDBS) share a common pool of storage in common BLKS and are accessed directly by their addresses. Subroutine BLOCK is used to copy any of these blocks into local storage.

TRAM STEP 4 INPUT CARDS

```
PARMS      0    15      2    6.5  50.0
PR     PRIMARY  1       5      0.0        0     -1.000      1.000      1.000      1.000      1.000    1    1    1
       SRC 1            1      1
PR     PRIMARY  2      10     10.000      1     -1.000     -1.000      1.000     -1.000      1.000    1    1    1
       SRC 2            0      1
PR     PRIMARY  3       2      1.000      1     -1.000     -1.000      1.000     -1.000      1.000    1    1    4
       SRC 3            1      0
       SRC 5            1      0
       SRC 7            0      1
       SRC 9            0      1
PR     PRIMARY  5       0      0.0        0      0.0        0.0        0.0        0.0        0.0      0    0    2
       SRC 11           0      1
       SRC 13           0      1
PR     PRIMARY  6       0      0.0        0      0.0        0.0        0.0        0.0        0.0      0    0    1
       SRC 14           0      1
PR     PRIMARY  4       1      1.000      2      1.000      0.0        0.0        0.0        0.0      1    0    2
       SRC 4            1      0
       SRC 6            1      0
SR     SR 1             0      0.0        0      0.0        0.0        0.0        0.0        0.0      1    1    2
       SRC 1        SRC 2
SR     SR 2             0      0.0        0      0.0        0.0        0.0        0.0        0.0      1    1    2
       SRC 3        SRC 4
SR     SR 3             0      0.0        0      0.0        0.0        0.0        0.0        0.0      1    1    2
       SRC 5        SRC 6
SR     SR 4             0      0.0        0      0.0        0.0        0.0        0.0        0.0      1    1    2
       SRC 7        SRC 8
SR     SR 5             0      0.0        0      0.0        0.0        0.0        0.0        0.0      1    0    2
       SRC 9        SRC 10
SR     SR 6             1      0.0        0      1.000      0.0        0.0        0.0        0.0      1    1    2
       SRC 11       SRC 12
SR     SR 7             1      0.0        0      1.000     -1.000      0.0       -1.000      0.0      1    0    1
       SRC 13
SR     SR 8             1      1.000      1      0.0        0.0        0.0        0.0        0.0      1    1    2
       SRC 14       SRC 15
SR     SR 9             0      0.0        0      0.0        0.0        0.0        0.0        0.0      0    0    5
       SRCX 1       SRCX 2     SRCX 3     SRCX 4     SRCX 5
END
```

PRIMARY RESOURCE DEFINITIONS

NAME	USE UNITS/ RGU/YEAR	RDTE	N	INITIAL INVESTMENT COST/RGU	RI/RGU	RI/YEAR	OM/RGU	OM/UNIT OF USE	PLOT SW	TRANS SW	BUCKET SIZE	SRC NAME	UNITS OF USE/RGU	UNITS OF USE/UNIT PR USE
PRIMARY 1	5	0	0	-1000	1000	1000	1000	1000	1	1	1	SRC 1	1	1
PRIMARY 2	10	10000	1	-1000	-1000	1000	-1000	1000	1	1	2	SRC 2	0	1
PRIMARY 3	2	1000	1	-1000	-1000	1000	-1000	1000	1	1	3	SRC 3	1	0
												SRC 5	1	0
												SRC 7	0	1
												SRC 9	0	1
PRIMARY 4	1	1000	2	1000	0	0	0	0	1	0	3	SRC 4	1	0
												SRC 6	1	0
PRIMARY 5	0	0	0	0	0	0	0	0	0	0	5	SRC 11	0	1
												SRC 13	0	1
PRIMARY 6	0	0	0	0	0	0	0	0	0	0	1	SRC 14	0	1

SECONDARY RESOURCE DEFINITIONS

NAME	USE UNITS/ RGU/YEAR	RDTE	N	INITIAL INVESTMENT COST/RGU	RI/RGU	RI/YEAR	OM/RGU	OM/UNIT OF USE	PLOT SW	TRANS SW	COMPONENT NAMES		
SR 1	0	0	0	0	0	0	0	0	1	1	SRC 1	SRC 2	
SR 2	0	0	0	0	0	0	0	0	1	1	SRC 3	SRC 4	
SR 3	0	0	0	0	0	0	0	0	1	1	SRC 5	SRC 6	
SR 4	0	0	0	0	0	0	0	0	1	1	SRC 7	SRC 8	
SR 5	0	0	0	0	0	0	0	0	1	0	SRC 9	SRC 10	
SR 6	1	0	0	1000	0	0	0	0	1	1	SRC 11	SRC 12	
SR 7	1	0	0	1000	-1000	0	-1000	0	1	0	SRC 13		
SR 8	1	1000	1	0	0	0	0	0	1	1	SRC 14	SRC 15	
SR 9	0	0	0	0	0	0	0	0	0	0	SRCX 1	SRCX 2	SRCX 3
											SRCX 4	SRCX 5	

PERIODIC REPORT

START TIME 0
END TIME 2

PR NAME	PEAK USE/BUCKET	NUMBER OF BUCKETS	TOTAL USE	AVERAGE USE/BUCKET
PRIMARY 1	1	2	2	1
PRIMARY 2	0	0	0	0
PRIMARY 3	0	0	0	0
PRIMARY 4	0	0	0	0
PRIMARY 5	0	0	0	0
PRIMARY 6	1	2	2	1

YEARLY REPORT FOR YEAR NUMBER 1
PRIMARY RESOURCES

PR NAME	NO RGUS	NO RGUS PURCHASED	AVAILABLE USE/BUCKET	AVERAGE USE/BUCKET	PEAK USE/BUCKET	TOTAL USE	INITIAL INVESTMENT	RECURING INVESTMENT	OPERATIONS AND MAINTENANCE
PRIMARY 1	1	1	5	1	1	3	1000	2000	4000
PRIMARY 2	1	1	10	2	2	2	1000	2000	3000
PRIMARY 3	2	2	4	3	3	3	2000	3000	5000
PRIMARY 4	4	4	4	4	4	4	1000	0	0
PRIMARY 5	0	0	0	0	0	0	0	0	0
PRIMARY 6	0	0	0	1	1	5	0	0	0

YEARLY REPORT FOR YEAR NUMBER 1
SECONDARY RESOURCES

SR NAME	NO RGUS	NO RGUS PURCHASED	AVAILABLE USE/YEAR	ACTUAL USE/YEAR	COMPONENT USE 1	2	3	4	5	INITIAL INVESTMENT	RECURING INVESTMENT	OPERATIONS AND MAINTENANCE
SR 1	0	0	0	4	4	2	0	0	0	0	0	0
SR 2	0	0	0	4	2	4	0	0	0	0	0	0
SR 3	0	0	0	4	2	4	0	0	0	0	0	0
SR 4	0	0	0	3	3	0	0	0	0	0	0	0
SR 5	0	0	0	3	3	0	0	0	0	0	0	0
SR 6	0	0	0	0	0	0	0	0	0	0	0	0
SR 7	0	0	0	0	0	0	0	0	0	0	0	0
SR 8	5	5	5	5	5	0	0	0	0	0	0	0
SR 9	0	0	0	0	0	0	0	0	0	0	0	0

FINAL COST SUMMARY

YEAR NUMBER	CURRENT DOLLAR COSTS					INFLATED COSTS				
	RDTE	INITIAL INVESTMENT	RECURRING INVESTMENT	OPERATIONS AND MAINTENANCE	TOTAL COST FOR THE YEAR	RDTE	INITIAL INVESTMENT	RECURRING INVESTMENT	OPERATIONS AND MAINTENANCE	TOTAL COST FOR THE YEAR
-1	500.	0.	0.	0.	500.	441.	0.	0.	0.	441.
0	12500.	0.	0.	0.	12500.	11737.	0.	0.	0.	11737.
1	0.	5000.	7000.	12000.	24000.	0.	5000.	7000.	12000.	24000.
2	0.	1000.	8000.	15000.	24000.	0.	1065.	8520.	15975.	25560.
3	0.	2000.	5000.	7000.	14000.	0.	2268.	5671.	7940.	15879.
TOTALS	13000.	8000.	20000.	34000.	75000.	12178.	8333.	21191.	35915.	77617.

INPUT PARAMETERS

START TIME	10
END TIME	28
PERIODIC REPORTING PERIOD	2
NUMBER OF CALENDAR UNITS PER YEAR	8

LAG REPORT

PERIOD START TIME 26
PERIOD END TIME 28

COURSE NAMES	CU S LAGGED THIS PERIOD	CU S LAGGED TO DATE
PILOTS	2	4
COPILOTS	0	2
CCTSP	0	2
CCTSC	0	2

PROC BLOCK NAMES	CUS LAGGED THIS PERIOD	CUS LAGGED TO DATE
P SCATSA	0	2
P U1	0	2
P U2	0	2
P S1	0	2
P U3	2	2

TASK NAMES	CUS LAGGED THIS PERIOD	CUS LAGGED TO DATE
SYN	0	1
SCATSA	0	1
EXTRA	0	1
P U1	0	1
RUB1	0	4
P S1	2	2

SOURCE REPORT

PERIOD START TIME 26
PERIOD END TIME 28

COURSE NAMES	******** THIS PERIOD ******** TRAINEES USED	AVG TRAINING TIME	********** CUMULATIVE ********** TRAINEES USED	AVG TRAINING TIME
PILOTS	1	1.0	5	1.0
COPILOTS	4	1.0	15	1.7
CCTSP	2	4.0	6	4.0

SOURCE NAMES	******** THIS PERIOD ******** TRAINEES USED	AVG TRAINING TIME	********** CUMULATIVE ********** TRAINEES USED	AVG TRAINING TIME
COPILOTS	1	1.0	5	1.0
S1	4	1.0	15	1.7
S2	0	0.0	2	4.0
S3	0	0.0	2	4.0
S4	2	4.0	2	4.0

7.5 REPORTS

Sample outputs are contained on the following pages. The reports are:

1. Parameters - The first 2 cards
2. Air Base List - Base names, indices, and PMT track lists
3. Delivery List - Years by bases
4. CCTS Track List - Sources index for each CCTS track
5. Source Availability - Trainees available years by sources
6. Resource Use Data - Resources used by unit trainees - tracks by resources
7. CCTS Summary - Yearly A/C deliveries, total A/C deployment, new crews, replacement crews, total crews in the system, pilot upgrades, total full crew training, and extras pairs training
8. Detailed Base Delivery List
9. PMT Track Trainees - Number of PMT unit trainees per track, years by track
10. Source Use Matrix - Use of trainees from each source, years by source
11. Track Use - Trainees taught by years and track
12. Resource Use - Resources used, years by resource.

7.6 DATA SET OUTPUT

Two files are produced.

FORTRAN Unit 1 contains resource use records. The records must be sorted by date and resources number. Each logical record contains

- Date in CUs
- Resource number
- Resource originally available
- Resource remaining.

The resource originally available is nominally 99999 units. A final record with the time 999999 is produced as an end of file record.

FORTRAN Unit 2 is the resource name file. It contains

Record

1 Number of resources
2 Resource names
3 Number of calendar units per year
4 Bucket size for each resource (=NCU for TROLIE)

Table 7.4.1

Table 1

TROLIE INPUT DATA

Card or Card Set	Cols. or Format	Variable Name	Type	Units	Definition	Notes
1	1-10	AR	Real	Crew/Year	Attrition Ratio	
	11-21	CR	Real	Crew/A-C	Crew Ratio	
	21-30	PUPR	Real	Copilots/year	Copilot Upgrade Ratio	
2	1-5	NY	Integer	Years	Total Time	≤ 24
	6-10	NYO	Integer	Years	First Year Date	
	11-15	NB	Integer		No. Air Bases	≤ 15
	16-20	NS	Integer		No. Sources	≤ 25
	21-25	NT	Integer		No. Tracks	≤ 40
	26-30	NR	Integer		No. Resources	≤ 80
	31-35	NCU	Integer	CUs/year	No. Calendar Units/year	≥ 3
	36-40	IDELAY	Integer	Years	Delay before Attrition	
	41-45	ITAPE	Integer	I/O	Non-Zero Inhibits Outputs	
3	9-20	NAB	Alphanumeric		Air Base Names (Normally only Cols. 11-20 are used)	(NB cards)
4A	40I2	ILTID	Integer		Lower PMT Track ID Pointer	
	40I2	IUTID	Integer		Upper PMT Track ID Pointer	
4B	40I2	IDT	Integer		Track ID List (2 cards)	IDT(I) > 25
5A	1-4	A	Alphabetic		"DELI" (Cards normally read DELIVERY)	
	9-20	MAB	Alphabetic		Air Base Name	
	31-35	IDTCU	Integer	CU*	Delivery Date	
	36-40	NACS	Integer		Number of Aircraft Delivered	

*Normally, only Cols 11-20 are used.

Table 7.4.1 (continued)

Card or Card Set	Cols. or Format	Variable Name	Type	Units	Definition	Notes
5B	1-4	A	Alphabetic		"END	
6	40I2	ISTAB	Integer		Source List	
7	20I4	NSPSY	Integer		Trainer Sources Available	1 Card per year unless $20 < NS \leq 25$
8	A10	RESNAM	Alphabetic		Resource Names	(1 per card)
9	20I4	IRCPRT	Integer		Resource use per Unit Class	(1 or more cards per track, new card for each track)

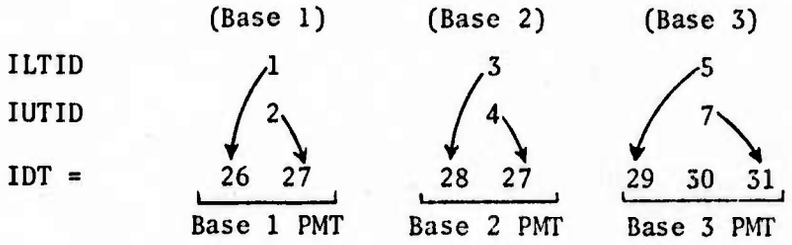
Notes on Preparation of Inputs

The only complex input is card set 4 consisting of the array IDT and its pointer arrays ILTID and IUTID.

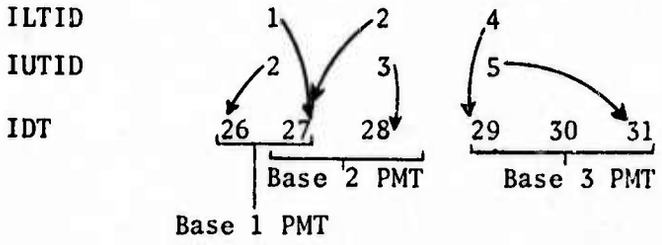
These inputs are most conveniently created by first making a list of the tracks which are to be used for PMT at each airbase. For example,

	Base 1	Base 2	Base 3
PMT Tracks	26, 27	28 ,27	29, 30, 31

The Table IDT and the arrays ILTID and IUTID could be coded



Or one could code



8 VARIABLE DIMENSIONS

8.1 VARY PROGRAM AND ASSOCIATED CODE

Much of the TRAM has been arranged to allow the user to re-dimension the arrays to fit a particular training system. This becomes important when core storage is limited, since one may be forced to give up core not needed for procblocks say, in order to accommodate an increased number of classes. Particularly in Phase 3, it is cumbersome to change the dimensions in even a few common blocks.

The idea in the variable dimension option is as follows: Replace each common block in each subroutine with ones having machine-recognizable keysymbols instead of dimensions. The source code then becomes a master deck upon which an editing step (the VARY program) is performed. This step replaces the keysymbols in the dimension statements by numerical values. The resulting code may then be compiled. The possibility of errors introduced into the dimension statements is thereby reduced to the problem of correctly specifying the value of the keysymbols.

The variable dimension feature of the TRAM program is implemented through the use of a text manipulation program VARY. VARY processes FORTRAN statements to create the variable dimensions. Statements containing dimensions to be modified are indicated by an "X" in Column 1. The program searches for the X on the input data cards. Finding such a card, it then looks for special character strings of the form #vvvv where vvvv is a 3- or 4-character name. The length of the name indicates maximum reasonable value for the variable. A 4-character name (#PTA) has a limit of 9999, a 5-character name (#CHKP) has the limit 99999. A table is supplied to the VARY program which gives the dimension desired for variables with a given character string. For example, #CHKP could be defined to be 1500 as it would be in the fixed dimension program. (See Table 8.1.1.)

8.2 INPUTS TO THE VARY PROGRAM

There are three sets of input data for VARY. The input consists of a control card, an arbitrary number of keysymbol cards, and the source code to be modified (See Table 8.1.2.)

TABLE 8.1.1

VARY PARAMETERS FOR VPHASE3

#RSRTS	-	max. number of resources needed to satisfy any RUB.
#BUCKS	-	max. number of buckets needed for 1 primary resource and associated second aries.
#CRSP1	-	max. number of courses + 1.
#PTBS	-	dimension of linked storage for predetermined transfer block.
#CG	-	max. number of course groups.
#CIN	-	max. number of courses in a course group.
#TRKS	-	max. number of tracks.
#RLTDB	-	max. number of track descriptor blocks.
#CRSE1	-	max. number of courses +1 (synonym for #CRSP1).
#CRSES	-	max. number of courses.
#LMIT1	-	dimension of linked storage for training demand records.
#LMITC	-	dimension of linked storage for class blocks.
#TDF1C	-	max. number of track descriptor for 1 course.
#CLASS	-	dimension of linked storage for class blocks (synonym for #LM1T)
#CCLS	-	max. number of simultaneously active classes.
#STSKS	-	max. number of tasks to be executed simultaneously.
#PBLKS	-	max. number of procblocs.
#CBSZE	-	number of words in classblock (30)
#TRESB	-	max. number of resource buckets needed to do resource utilization tasks for a group of synchronized classes.
#RES	-	max. number of resources.
#SRCES	-	max. number of sources.

Table 8.1.1 (continued)

VARY PARAMETERS FOR VCLOCK, VNAMES AND VPHASE2

#A	-	max. number of air bases.
#E	-	max. number of air base event cards.
#AC	-	max. number of CCTS cards.
#PG	-	max. number of PMT group cards.
#PC	-	max. number of PMT course cards.
#AD	-	max. number of air base delivery cards.
#AB	-	max. number of air base buckets.
#Y	-	max. number of years.
#CR	-	max. number of courses (CCTS + PMT).
#BLKS	-	number of words in pool of storage for PROC,TASK,RUB and RUDB blocks.
#NAM	-	max. number of names in pool of names.
#RS	-	max. number of resources.
#RC	-	max. number of resource cards.
#SR	-	max. number of sources.
#SC	-	max. number of source cards.
#RPL	-	max. number of resources in pool of resources.
#SPL	-	max. number of sources in pool of sources.
#STK	-	max. size of stack.

TABLE 8.1.2

CONTROL CARD (Format 615, 10AI)

Variable	Value	Definition
IPRINT	0/1	No/yes prints copy of modified deck
IPUNCH	0/0	No/yes punches copy of modified deck
IUNIT	1-99≠6,7	Unit used for source input
IØUT	1-99≠5,6	Unit used for deck output
NCØLS	72 (typical)	Columns to check
LØKEY	1 (typical)	Column location of flag for modifiable card
ICHG	X (typical)	Flag for modifiable card
ICØDE	# (typical)	First character of keysymbol
I0	0	Numeric character references
I9	9	
ICOMM	,	
IRPAR	)	Alphanumeric characters reference
ILPAR	(	
IREPLC	blank	
ISLSH	/	

Keysymbol Cards (Format 80AI)

KEY (1-6)	Keysymbol
IKEY (7-12)	Keysymbol replacement
COMENT (13-80)	Arbitrary comment

END Card (Columns 1-3)

The source code is found on Unit IUNIT and the output is placed on Unit IØUT.