

(NASA-CR-169194) THE ANALYSIS OF THE
STATISTICAL AND HISTORICAL INFORMATION
GATHERED DURING THE DEVELOPMENT OF THE
SHUTTLE ORBITER PRIMARY FLIGHT SOFTWARE
Final Report (Texas Engineering Experiment

N82-29960

HCAID

Unclass

G3/61 28536

Final Report

for

**"The Analysis of the Statistical
and Historical Information
Gathered During the Development
of the Shuttle Orbiter
Primary Flight Software"**

RF-82-17

Submitted by the
Texas A & M Research Foundation

to the
NASA Johnson Space Center
Mr. Robert K. Peck
Procurement Operations, Code BL
Houston, Texas 77058

Prepared by:

Dr. Dick B. Simmons
Dr. Miner P. Marchbanks, Jr.
Dr. Michael J. Quick

of the

Data Processing Center
Texas Engineering Experiment Station
College Station, Texas 77843

June 1982



Final Report

for

"The Analysis of the Statistical
and Historical Information
Gathered During the Development
of the Shuttle Orbiter
Primary Flight Software"

RF-82-17

Submitted by the
Texas A & M Research Foundation

to the
NASA Johnson Space Center
Mr. Robert K. Peck
Procurement Operations, Code BL
Houston, Texas 77058

Prepared by:

Dr. Dick B. Simmons
Dr. Miner P. Marchbanks, Jr.
Dr. Michael J. Quick

of the

Data Processing Center
Texas Engineering Experiment Station
College Station, Texas 77843

June 1982

FORWARD

This report presents the results of the project for "The Analysis of the Statistical and Historical Information Gathered During the Development of the Shuttle Orbiter Primary Flight Software". This work was performed by the Texas Engineering Experiment Station of Texas A&M University, College Station, Texas. This work was performed for the National Aeronautics and Space Administration, Johnson Space Flight Center, Houston, Texas, under Grant NCC9-2.

Principal investigators for the project were Drs. Dick B. Simmons, Miner P. Marchbanks, Jr. and Michael J. Quick. Major contributors to the project were Mr. Lee Boyajian, Mr. Blair Brenner, Ms. Karla Carroll, Mr. Richard Cox, Dr. Dan D. Drew, Ms. Janna Keel, Ms. Guylaine M. Pollock, Dr. Larry J. Ringer, Dr. Sallie Sheppard, and Mr. Dennis Wilson.

Major points of interest are covered in the body of the report with supporting information being included in the appendices.

ACKNOWLEDGEMENTS

The authors wish to express their thanks to Mr. John R. Garman and Mr. Dave Sykes of the Johnson Space Flight Center for their efforts in making the project a success. Mr. Garman planted many of the seeds for thought presented in this report. Mr. Sykes represented NASA in providing to the research team information necessary to conduct this research. He patiently explained the workings of NASA and arranged meetings with individuals from both NASA and IBM who could provide requested information.

ABSTRACT

Presented here are the results of an effort to thoroughly and objectively analyze the statistical and historical information gathered during the development of the Shuttle Orbiter Primary Flight Software. The particular areas of interest include cost of the software, reliability of the software, requirements for the software and how the requirements changed during development of the system. Data related to the current version of the software system produced some interesting results. Suggestions are made for the saving of additional data which will allow additional investigation.

CONTENTS

FORWARD	ii
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
	<u>page</u>
1. INTRODUCTION	1
Definition of Project	1
Areas of Research	2
Definition Area	2
Code Area	2
Changes Area	2
2. ACCOMPLISHMENTS	3
Literature Survey	3
Definition Area	6
List of Documentation	6
Cost Data	7
Cost Data Related to CR's and DR's	8
Code Area	10
Availability of Source Data	10
Documentation of HAL/S	11
Methodology of Analysis of Source Data	11
Size of Data	12
FSW Source Compilation	13
Overlay or Build Sequence	15
HALSTAT	15
Source Code Analyzer	16
Status of Scanner	16
Definition of Scanner Record	17
Changes Area	18
Availability of Change Data	18
Reliability Data	18
DR and CR Data	19
DR and CR Legends	19
Methodology of Analysis of Change Data	20
Results of Analysis of Change Data	20
Summary of Data Availability	23

3. FUTURE WORK	25
Continued Analysis	25
Continued Development of Cost Data	25
Continued Work with Language Scanner	25
Continued Analysis of CR and DR Data	27
Concurrent Analysis and Monitoring	27
Future Development	28
Comparison with other Projects	28
Complexity Analysis	29
Cost Analysis	29
Language Analysis	30

<u>Appendix</u>	<u>page</u>
A. DOCUMENTATION	31
B. HAL/S ANALYSIS SCANNER OUTPUT	45
C. COMPLEXITY MEASURES	55
D. HAL/S COMPLEXITY FACTORS	58
E. HAL/S SOURCE ANALYZER MODULE STRUCTURE	61
F. MEMBER LIST OF FSW APPLICATION SOURCE	62
G. MEMBER LIST OF INCLUDE SOURCE	65
H. MEMBER LIST OF SYSTEM SOURCE	68
I. DR VARIABLE LIST	70
J. DR LEGEND	72
K. CR VARIABLE LIST	86
L. CR LEGEND	89
M. COST DATA CHARTS	100
N. CHANGE DATA CHARTS	106
BIBLIOGRAPHY	201

LIST OF TABLES

<u>Table</u>	<u>page</u>
1. Major Categories of Errors	5
2. Correlation of Cost Data with Selected Variables.	9
3. Summary of Data Availability	24
4. Complexity Measure Correlation to Changes	57

LIST OF FIGURES

<u>Figure</u>	<u>page</u>
1. Structure of HAL/S Analyzer	61
2. Total hours IBM billed NASA for development.	101
3. Total hours IBM billed for verification.	102
4. Cumulative total man-hours IBM billed NASA for development. .	103
5. Cumulative total man-hours IBM billed NASA for verification.	104
6. Development and verification cost with scaled change and discrepancy reports frequency.	105
7. Frequency of DR's logged in by year-month.	107
8. Frequency of DR's for mission identifier (MISSN_ID).	108
9. DR frequency for reporting facility (FAC).	109
10. DR frequency for action department (AD).	110
11. DR frequency for priority (P).	111
12. DR frequency for status (ST).	112
13. DR frequency by year-month logged in (_ODYRMO) and priority (P).	113
14. Frequency of priority 1 DR's logged in by year-month (_ODYRMO).	114
15. Frequency of priority 2 DR's logged in by year-month (_ODYRMO).	115
16. Frequency of priority 3 DR's logged in by year-month (_ODYRMO).	116
17. Frequency of priority 4 DR's logged in by year-month (_ODYRMO).	117

18.	Frequency of priority 5 DR's logged in by year-month (_ODYRMO)	118
19.	Frequency of priority 9 DR's logged in by year-month (_ODYRMO)	119
20.	Priority (P) of DR's for mission identifier (MISSN_ID)	120
21.	Priority (P) of DR's from a reporting facility (FAC)	121
22.	Frequency of priority 1 DR's from a reporting facility (FAC) .	122
23.	Frequency of priority 2 DR's from a reporting facility (FAC) .	123
24.	Frequency of priority 3 DR's from a reporting facility (FAC) .	124
25.	Frequency of priority 4 DR's from a reporting facility (FAC) .	125
26.	Frequency of priority 5 DR's from a reporting facility (FAC) .	126
27.	Frequency of priority 9 DR's from a reporting facility (FAC) .	127
28.	Number of DR's logged without a mission identifier (MISSN_ID) .	128
29.	Number of DR's logged for site/test unique patch DR's (MISSN_ID=A)	129
30.	Number of DR's logged for PMD3 (MISSN_ID=B)	130
31.	Number of DR's logged for ECL (MISSN_ID=C)	131
32.	Number of DR's logged for SSW1 (MISSN_ID=D)	132
33.	Number of DR's logged for JSC form 1541 (MISSN_ID=E)	133
34.	Number of DR's logged for release 19, floor system STS-5 (MISSN_ID=F)	134
35.	Number of DR's logged for PMD2 (MISSN_ID=G)	135
36.	Number of DR's logged for AOA (MISSN_ID=H)	136
37.	Number of DR's logged for KSC (MISSN_ID=K)	137
38.	Number of DR's logged for STS-1, FUP5/FLT (MISSN_ID=L)	138
39.	Number of DR's logged for STS-1, FUP4 (MISSN_ID=M)	139
40.	Number of DR's logged for entry (MISSN_ID=N)	140
41.	Number of DR's logged for PMD1 (MISSN_ID=P)	141

42.	Number of DR's logged for STS-2, R18V21 flight system (MISSN_ID=R).	142
43.	Number of DR's logged for STS-2, R18V11 field system (MISSN_ID=R1).	143
44.	Number of DR's logged for STS-2, R18V12 field system (MISSN_ID=R2).	144
45.	Number of DR's logged for STS-3, R18V30 flight system (MISSN_ID=R3).	145
46.	Number of DR's logged for STS-4, R18V45 flight system (MISSN_ID=R4).	146
47.	Number of DR's logged for OV-99, R18V90 field Ssstem (MISSN_ID=R9).	147
48.	Number of DR's logged for SSW2 (MISSN_ID=S).	148
49.	Number of DR's logged for SPF, release 1 (MISSN_ID=S1).	149
50.	Number of DR's logged for SDL, rclease 36 (MISSN_ID=S6).	150
51.	Number of DR's logged for SDL, release 37 (MISSN_ID=S7).	151
52.	Number of DR's logged for OFT1 (MISSN_ID=T).	152
53.	Number of DR's logged for entry update 1 (MISSN_ID=U).	153
54.	Number of DR's logged for STS-2, R17V5 (MISSN_ID=W).	154
55.	Number of DR's logged for STS-1, FUP1 (MISSFN_ID=X)	155
56.	Number of DR's logged for STS-1, FUP2 (MISSN_ID=Y).	156
57.	Number of DR's logged for STS-1, FUP3 (MISSN_ID=Z).	157
58.	Total number of DR's logged for STS-1, FUP5/FLT (MISSN_ID=L) and STS-1, FUP4 (MISSN_ID=M).	158
59.	Total number of DR's logged for STS-2 (MISSN_ID=R), STS-2 (MISSN_ID=R1), STS-2 (MISSN_ID=R2), STS-3 (MISSN_ID=R3), and STS-4 (MISSN_ID=R4).	159
60.	Total number of DR's logged for SDL releases 36 and 37 (MISSN_ID=S6 and S7).	160
61.	Total number of DR's logged for STS-1: FUP1 (MISSN_ID=X), FUP2 (MISSN_ID=Y), and FUP3 (MISSN_ID=Z).	161

62.	CR frequency for department 1 manpower impact, year 1 (D1IMPYR1).	162
63.	CR frequency for department 1 manpower impact, year 2 (D1IMPYR2).	163
64.	CR frequency for department 1 manpower impact, year 3 (D1IMPYR3).	164
65.	CR frequency for department 1 manpower impact, year 4 (D1IMPYR4).	165
66.	CR frequency for department 2 manpower impact, year 1 (D2IMPYR1).	166
67.	CR frequency for department 2 manpower impact, year 2 (D2IMPYR2).	167
68.	CR frequency for department 2 manpower impact, year 3 (D2IMPYR3).	168
69.	CR frequency for department 2 manpower impact, year 4 (D2IMPYR4).	169
70.	CR frequency for department 3 manpower impact, year 1 (D3IMPYR1).	170
71.	CR frequency for department 3 manpower impact, year 2 (D3IMPYR2).	171
72.	CR frequency for department 3 manpower impact, year 3 (D3IMPYR3).	172
73.	CR frequency for department 3 manpower impact, year 4 (D3IMPYR4).	173
74.	CR frequency for department 4 manpower impact, year 1 (D4IMPYR1).	174
75.	CR frequency for department 4 manpower impact, year 2 (D4IMPYR2).	175
76.	CR frequency for department 4 manpower impact, year 3 (D4IMPYR3).	176
77.	CR frequency for department 4 manpower impact, year 4 (D4IMPYR4).	177
78.	CR frequency for department 5 manpower impact, year 1 (D5IMPYR1).	178

79.	CR frequency for department 5 manpower impact, year 2 (D5IMPYR2).	179
80.	CR frequency for department 5 manpower impact, year 3 (D5IMPYR3).	180
81.	CR frequency for department 5 manpower impact, year 4 (D5IMPYR4).	181
82.	CR frequency for department 6 manpower impact, year 1 (D6IMPYR1).	182
83.	CR frequency for department 6 manpower impact, year 2 (D6IMPYR2).	183
84.	CR frequency for department 6 manpower impact, year 3 (D6IMPYR3).	184
85.	CR frequency for department 6 manpower impact, year 4 (D6IMPYR4).	185
86.	CR frequency for department 7 manpower impact, year 1 (D7IMPYR1).	186
87.	CR frequency for department 7 manpower impact, year 3 (D7IMPYR3).	187
88.	CR frequency for department 7 manpower impact, year 4 (D7IMPYR4).	188
89.	CR frequency for department 1 total manpower impact (D1TOTIMP).	189
90.	CR frequency for department 2 total manpower impact (D2TOTIMP).	190
91.	CR frequency for department 3 total manpower impact (D3TOTIMP).	191
92.	CR frequency for department 4 total manpower impact (D4TOTIMP).	192
93.	CR frequency for department 5 total manpower impact (D5TOTIMP).	193
94.	CR frequency for department 6 total manpower impact (D6TOTIMP).	194
95.	CR frequency for department 7 total manpower impact (D7TOTIMP).	195

96.	DR frequency of number of months between date logged in (_OD) and development close date (BD).	196
97.	DR frequency of number of months between date logged in (_OD) and verification date (VC).	197
98.	CR frequency of number of months between date received (DTRECD) and RRB disposition date (RRBDT).	198
99.	CR frequency of number of months between date received (DTRECD) and OASCB disposition date (OASCBDT).	199
100.	CR frequency of number of months between RRB disposition date (RRBDT) and OASCB disposition date (OASCBDT).	200

INTRODUCTION

The Data Systems and Analysis Directorate (DSAD) of the Johnson Space Flight Center (JSC) has the responsibility for the implementation and maintenance of the Shuttle Orbiter Primary Flight Software. This software development project has involved a large number of software types, a large number of software management issues and a large number of software development methodology issues. During the course of the implementation of this project a great deal of statistical and historical information has been gathered and retained within JSC. A research group at Texas A&M University (TAMU) composed of members from the Advanced Technology Group of The Data Processing Center and members of the faculty of the Computing Science Department and the Institute of Statistics has undertaken the task of examining this information.

1.1 DEFINITION OF PROJECT

The object of this examination is to analyze the data gathered by JSC to determine what information can be used to provide NASA and the Software industry with additional metrics on the management and methodology issues surrounding large software implementation efforts. The software implementation data would hopefully produce data that could be of interest in areas such as cost, reliability, and requirements. With these areas of interest in mind, the task plan was organized into seven phases. The first phases involved key personnel visiting JSC to obtain information relating to the definition of the Orbiter Software Project, to determine what data is available for analysis, and to determine what analysis is to be done and products to be obtained. The latter phases will include putting data in machine readable form and producing results. The seven phases are as follows:

1. Produce a definition of the Orbiter Software Implementation
2. Define and catalog the available data.
3. Produce an estimation and prioritization of products, analyses and metrics.
4. Produce final work and product plan and schedule.

5. Conduct data gathering and integration with periodic status reports.
6. Put data in machine readable form.
7. Analysis, results and final report encompassing the above phases.

In areas of research where data was readily available, all of the above phases were completed. TAMU suggest that a continuing effort be carried on in other areas.

1.2 AREAS OF RESEARCH

The project has been organized into three areas of interest with personnel assigned to information gathering task in each of three areas. These areas are the definition area, the code area and the changes area.

1.2.1 Definition Area

The definition area of analysis is primarily concerned with information about documentation, management, cost factors, and hardware constraints in the software development process.

1.2.2 Code Area

The code area of analysis is concerned with information about the source code. Statistics that can be gathered from HALSTAT, as well as statistics that can be gathered with the TAMU code analyzer, will be used to quantize source code characteristics.

1.2.3 Changes Area

The changes area of analysis is concerned with information related to changes made to the software during the development and testing processes. Statistics are gathered with respect to Change Request (CR), Program Change Proposal (PCP), and Discrepancy Reports (DR). Categories of change, with respect to the effect of the change and the priority of the change, will be analyzed. NASA's and IBM's characterization of the changes will be analyzed as well as a TAMU characterization of the changes.

ACCOMPLISHMENTS

The work completed in this project has been categorized into the three main interest areas described in Section 1: definition, code and changes. Accomplishments in each of these areas are presented following a brief literature survey which summarizes the conclusions of other work pertinent to this project.

2.1 LITERATURE SURVEY

During the last decade software engineering has become a major area of interest in computer science research.^{6 10 12 13 16} In attempts to attain the software engineering goals of increasing system reliability and predicting software costs, researchers have conducted various program complexity and error analysis studies. This work has suggested using numerous metrics such as program length, operator/operand counts, nesting characteristics, data-sharing, and level of modularization as a means of measuring the complexity of programs and in error estimation³. The hypotheses formed in these studies have in general been tested using small data samples or student programs due to the difficulty in obtaining valid production data.

Zolnowski^{27 28} has developed a comprehensive scheme for assigning a figure of merit to a program based on an objective set of program characteristics known to be related to program complexity. She has incorporated the measures suggested by various other researchers on the assumption that the different opinions of these authors do in fact reflect many diverse aspects of program complexity. She then evaluated the effectiveness of the various measures using objective data collected from the development of production programs in FORTRAN and COBOL. The complexity characteristics are divided into four categories: instruction mix, data reference, interaction/interconnection and structure/control flow. Based on Zolnowski's results a list of factors in each of these areas was compiled for use in estimating the complexity of the NASA space shuttle software written in HAL/S. This list is presented in Appendix D.

Carver⁵ has extended the work of Zolnowski to determine the relationship between program changes and complexity characteristics for use in estimating program completeness based on changes. She defines the number of program changes to be the number of times the program code must be modi-

fied from the first version as written by the programmer to the final version at the end of the testing stage. In her model the program characteristics are measured in the first version of the program. In the study the number of changes were counted chronologically in the development of two large production software systems. Carver found that structure/control flow characteristics were the best predictors of program changes. She also identified system-related conditions such as quality of the specifications, volatility of the system, and the use of structured programming concepts to be critical for determining program changes.

The work of Henry and Kafura¹¹ is typical of several recent research projects which define and validate a set of software metrics appropriate for evaluating the structure of large-scale systems. Their metrics are based on the measurement of information flow between system components; specifically, procedure complexity, module complexity and module coupling. In this study changes made to the source code of the UNIX operating system were used in the validation effort. A strong correlation was found between the complexity measures and the occurrence of changes. Because the major elements of information flow analysis can be determined at design time, this model can be used early in the developmental process to produce a qualitative evaluation useful in identifying various types of structural flaws in the design and implementation. An overview of their work and findings which are applicable to the HAL/S software analysis are presented in Appendix C.

Thayer, Lipow and Nelson¹⁴ report on a software reliability study performed by TRW Systems and Energy for the Rome Air Development Center. The data, principally error data collected from four software development projects, was analyzed to study various types of errors in software, the effectiveness of the development and test strategies in preventing errors and the reliability of the software itself. Their final report provides guidelines for data collection and analysis on other similar projects. Of particular interest in the HAL/S software study are the categorization schemes for software errors which were developed as a part of the TRW effort. Based on detailed study of error data from four large software projects they developed 164 error categories under 16 headings. These categories are highlighted in Table 1. They concluded from their research that errors must be categorized in considerable detail to be of practical use in developing or evaluating tools. While the main categories tend to be the same for different projects, they found that the appropriate detailed categories depend on the operating environment characteristics, the language being used and the development strategy.

TABLE 1

Major Categories of Errors

Computational errors
result from errors in coded equations

Logic errors
errors in logical processing

Input/output errors
errors from i/o code rather than from interface

Data handling errors
errors in reading, writing, moving, storing
and modifying data

Operating system/system support errors
errors in OS, compiler, assembler, system support
software and system utilities

Configuration errors
catastrophic problems encountered when the
software failed to be compatible with the
system software

Interface errors
routine/routine interface errors
routine/system software interface errors
Tape processing interface errors
user interface errors
database interface errors

User requested changes
user category for enhancements and requested changes

Preset database errors
preset data primarily initialization data

Global variable/compool definition errors
errors in global variables or constants

Recurrent errors
reopened error previously categorized

Documentation errors

Requirements compliance errors
software not compliant with specification

Unidentified errors
errors with insufficient information to classify

Operator errors
problem due to operator, developer or tester

Questions
error report to record a question needing to be answered

2.2 DEFINITION AREA

The first steps accomplished by TAMU in the analysis of the space shuttle software were of an investigative nature. Many areas needed to be defined and an inventory taken of what information and data was available in the areas to be studied.

The goal in the documentation area was to determine what documentation was available during the development of the Flight Software for the Shuttle. The Research team received several boxes of manuals and documents. Inventorying the manuals received was one of the first steps in the project. Though many of the manuals received were duplicates, the list in Appendix A provided the research group with an outline of what types of information was available. The next step in the documentation area was to determine the structure of the documents and manuals for both NASA and IBM as described in Section 2.2.1.

The cost area was also an area of study. Defining this area proved to be quite difficult. Though the document Space Shuttle Orbiter Software Management Plan, produced by IBM stated that the Cost and Scheduling information was kept in machine readable form, the project team was repeatedly pointed toward the NASA 533 monthly reports.

An investigation of the DR/CR procedures and how the DR/CRs affected the development of the software was also part of the definition area of this project. The primary source of this study was the Space Shuttle Orbiter Avionics Management Plan. From this document the different code and design inspection steps were defined to the project team. At this step the team could better understand the flow of the DRs and CRs. It also became evident that trying to relate the DR and CR information to cost would be an interesting task.

2.2.1 List of Documentation

Determining the overall picture of the documents on the NASA side of the picture was a relatively simple task. Many of the NASA documents contained a Preface stating the design philosophy and the structure of the specifications. Also, contained in the Preface was a list of Level A and Level B documents. However a list of Level C or detail design documents was not so readily found. After a little investigation, a list of the Level C documents was found in the NASA document entitled "Statement of Work" dated July 25, 1980.

Putting the IBM side of the documentation together was not as easily done as the NASA side. After searching all available documents, the structure was uncovered in a manual with the wrong cover page. The manual dated 2/25/77 had erroneously been labeled Vol III - Applications: Guidance, Navigation, and Control instead of Vol I - Software Systems Overview. However, this manual did relate the NASA requirements to the

detailed design specifications written by IBM and used by the programming teams.

2.2.2 Cost Data

The area of time management and development manpower control appeared to be an area of interest and one which some useful information could be gleaned. This area, however, proved to be relatively difficult because of the lack of detailed information and the changes of reporting policies which took place during shuttle software development.

The source of information for the cost of the project was the monthly reports which IBM prepared for NASA. Since these were provided to the TAMU research team as copies of the reports, the information had to be coded and reentered into machine readable form. The major problem in analyzing information in these reports was the changes in NASA's reporting requirements. The reports changed in April of 1980 from man-equivalents as a measure to the use of hours. This necessitated the data being converted to a common format for man-time. At the same time, the categories of shuttle function reported in development also changed. In September of 1981, the report formats changed again both in categories reported and the measure of man-time reported.

The changes in categories of shuttle function was the major problem. These changes made it difficult to track the cost of the major functions of the Shuttle Software and made any sort of analysis in that area difficult.

The cost data were plotted over time. Examining the development and verification costs per month, Figure 2 (all figures will be found in Appendix M), notice that development costs show three phases: pre-May, 1980; May, 1980 through July, 1981; and post July, 1981. These may coincide with major missions. Verification costs, however, do not show this trend Figure (3). Verification costs increase to a level attained in late 1979 and then oscillate at this level thereafter.

Focusing on cumulative costs for development, Figure 4, and verification, Figure 5, the rate of expenditure has not changed significantly over the study period. The slopes of the cumulative curves have not decreased significantly, indicating that the rate of development and verification are the same throughout the study period and did not level off during the period under study.

2.2.3 Cost Data Related to CR's and DR's

Cost data were studied in relation to the number of DR's and CR's. Methods involved were graphical comparison and correlation analysis. The number of DR's and CR's were examined on the same axes as development and verification costs. Figure 6, Appendix M, shows the cost and change data (scaled by 20) together. Note that there appears to be no direct pattern across the variables.

The apparent lack of direct correlation between costs and number of changes per month was confirmed by subjecting the data to correlation analysis. Spearman nonparametric correlation coefficients, Table 2, show that development costs were related to the number of CR's, verification costs, and cumulative development and verification costs. Development costs were not related to the number of DR's for that month, but to the number of DR's 6 to 12 months prior. This would seem to indicate a mechanism of change data feedback into development, with a 6 to 12 month lag period.

Verification costs did not correlate with the number of DR's or CR's for that month. The number of DR's entered 7 to 12 months prior, and the number of CR's entered 8 to 12 months prior were significantly correlated with verification costs. Apparently there is about a one half year period required for the changes to work through the system.

Verification costs did correlate with development costs and cumulative development and verification costs.

TABLE 2

Correlation of Cost Data with Selected Variables.

Main Variable*	Correlated Variable*	r	Significance Level	
devel	cver	-.76	<.001	
	cdevel	-.69	<.001	
	crlag12	.44	.001	
	crlag1	.43	.002	
	ver	-.40	.005	
	drlag12	-.40	.005	
	drlag11	-.39	.005	
	drlag10	-.34	.02	
	crlag10	-.32	.02	
	crlag11	-.31	.03	
	numcr	.30	.03	
	drlag9	-.28	.05	
	crlag2	.28	.05	
	crlag3	.28	.05	
	drlag6	-.28	.05	
	ver	cdevel	.61	<.001
cver		.61	<.001	
crlag11		.58	<.001	
crlag12		.49	<.001	
drlag12		.44	.002	
devel		-.40	.005	
drlag7		.40	.005	
drlag11		.36	.01	
drlag9		.35	.02	
drlag10		.35	.02	
drlag8		.34	.02	
crlag9		.33	.02	
crlag10		.31	.03	
crlag8		.30	.04	
* devel		= development costs		
cdevel		= cumulative development costs		
ver	= verification cost			
cver	= cumulative verification costs			
numcr	= number of CR's for a month			
numdr	= number of DR's for a month			
crlagX	= number of CR's for the Xth month previous			
drlagX	= number of DR's for the Xth month previous			

2.3 CODE AREA

The objective of the Code Area effort was to decompose the Space Shuttle Orbiter Software into a data format that facilitated statistical gathering procedures. This effort proceeded in two directions:

1. data generation from available HAL/S utilities.
2. data generation from proposed source code analyzer or scanner.

Installation of the HAL/S utilities required a familiarization with the HAL/S 360 System and then testing procedures with data that was provided. The development of the software analyzer required a working knowledge of the HAL/S language, a formulation of the complexity factors for the HAL/S language, and a design for a generalized statistical data format.

2.3.1 Availability of Source Data

In order to analyze the Space Shuttle Orbiter Software for the proposed statistical model, a collection of software releases from one time period to a later time period was required. For this particular project only one software build or release was studied. Initially, a version of Release 16 of the Flight Software was to be the subject of this study. After receiving the first group of tapes for Release 16 from NASA, it was found that most of the data for Release 16 could not be read from tape. The tape reading problems were caused by incorrect label information and bad or missing tape file marks. The earliest available complete release of software on tape from NASA that could be used for analysis was Release 19.2. This release was the final or deliverable software from IBM to NASA and was used to fly the Space Shuttle Orbiter. The data on the tapes from NASA for this release contained:

1. FSW 19.2
 - a) Application source libraries
 - b) System source libraries
 - c) Include source libraries
2. Simulation Data Files (SDF's) for the AP-101
3. Load modules for OS/360
4. HAL/S 360 Compiler
 - a) HALSTAT

b) Linkage Editor Utilities.

Data which were unavailable from IBM were the OS/360 SDF's and the build information for Release 19.2 . In talking with IBM representatives, it was found that the OS/360 version of the SDF's were not kept up-to-date as the software release progressed into its final stages. Also, there was no documentation on how to reconstruct the build from the source members on tape. The unavailability of the OS/360 SDF'S and the build information for Release 19.2 presented problems which will be discussed in later sections.

2.3.2 Documentation of HAL/S

In order to analyze a program for factors relating to complexity and reliability, a good understanding of the native language is necessary. For the purpose of learning about HAL/S, three particular references were found to be useful. They were: HAL/S Language Specification (IR-542), Programming in HAL/S (Intermetrics), and HAL/S Programmer's Guide (IR-63-5).

With the intention of using some of the statistics generated by the utility HALSTAT, the installation of the HAL/S language compiler as well as HALSTAT became necessary. The HAL/S 360 Compiler had to be installed to provide the OS/360 SDF's for input to HALSTAT. More detail will be given concerning the reason for installation of the compiler in subsequent sections. Most of the information required for installation was given in HAL/S-360 User's Manual (IR-360-2). Implementation of the HALSTAT utility was aided by use of the HALSTAT User's Manual (IR-349) which contained helpful program and JCL examples.

During the development of the source analyzer program, the HAL/S Language Specification manual was referenced frequently concerning details of the HAL/S syntax. The concepts of automated code analysis and subsequent reliability measurements were derived from a dissertation by Jean C. Zolnowski² entitled: "A System for Measuring Program Complexity" and are given in Appendix D.

2.3.3 Methodology of Analysis of Source Data

As mentioned previously, the methods of analyzing the Space Shuttle Orbiter Software involved the use of the HAL/S utilities and development of a source code analyzer. The HAL/S utilities used to gather statistics from the Flight Software were HALSTAT and the HAL/S 360 Compiler. The HAL/S 360 Compiler provided the SDF input to HALSTAT which generated formatted reports on load module statistics. The source analyzer was designed to use the uncompiled version of the source and produce output which could be analyzed in various formats through the Statistical Analysis System (SAS) or other statistical software systems.

The installation of HALSTAT and the HAL/S 360 Compiler was finished far ahead of the final design work for the source scanner. Therefore a complete review of what HALSTAT could provide in the way of program complexity statistics was available. This allowed a decision to be made whether to continue in the HALSTAT direction or the source analyzer direction for gathering complexity statistics. The decision was based on the ease of using each method as well as the detail of statistics generated by each method. After careful consideration, it was decided to put the major emphasis on developing the source analyzer for the following reasons:

1. The number of source members that had to be compiled in order to provide HALSTAT with SDF inputs was considerably large.
2. The problems encountered when compiling the FSW source members made the input preparation step to HALSTAT difficult.
3. HALSTAT was lacking in production of complexity statistics relating to imbeddedness, nesting levels of certain constructs, and connectivity information.
4. The frequency count information given by HALSTAT only applied to certain types of instructions. For frequency counts on other categories of complexity factors, such as the number of references for a particular variable or the number of I/O instructions, either a complex program had to be devised to gather and tabulate these statistics from the HALSTAT report or these statistics had to be hand-tabulated from the report.

More details for the decision to concentrate the effort toward the source code analyzer will be discussed in the following sections.

It was noted that the effort toward using HALSTAT was not wasted while attempting to locate or generate a tool to provide program complexity statistics. The complexity statistics which were missing from HALSTAT and the difficulty in using HALSTAT provided a set of design requirements for the source code analyzer. Two examples of the requirements were that the source code analyzer would have to provide an output record with a broad definition to cover all areas of complexity analysis and that only the uncompiled source code would be analyzed. SAS or special post-processing programs will be used to group, tabulate, and categorize the output source code analysis data from the individual FSW source members.

2.3.3.1 Size of Data

The Flight Software (Rel 19.2) is subdivided into three main libraries:

1. System source

- a) Of the 350 modules that comprise the system source, approximately half are written in AP-101 assembler, and the remaining are written in HAL/S. There are about 100,000 lines of source code in the system library.
- b) Many of the system modules 'include' other system source modules.
- c) The list of modules in the System source library is contained in Appendix H.

2. Application source

- a) The application source was written in HAL/S and consists of 985 modules. It contains a total of 385,000 lines of code.
- b) The typical application module 'includes' other system and application source modules.
- c) The list of modules in the Application source library is contained in Appendix F.

3. Include source

- a) There are 673 Include modules which were written in HAL/S. They contain 200,000 lines of source code.
- b) All Include modules are self-contained (no 'includes' of other modules).
- c) The list of modules in the Include source library is contained in Appendix G.

2.3.3.2 FSW Source Compilation

Initially the HAL/S 360 Compiler was installed to expand the FSW source code. This source expansion meant that all external source code sections (SDF's or compiler templates) of the source member being analyzed would have their reference and interface sections listed in the compiler output; that is, an external COMPOOL would have its header statement and data definition statements listed, an external PROCEDURE or FUNCTION would have its header statement and data definition statements for its formal parameters listed, and an external PROGRAM would have only its header statement listed. From this expanded source listing, the source code analyzer would have an input that provided external reference and interface information. This external information is necessary to give accurate statistics regarding for example the number of variables of a particular data type (ARRAY, VECTOR, MATRIX, STRUCTURE etc.) which are encountered in a particular operation (addition, subtraction, division,

etc.,). In the case of an external variable that is of VECTOR type, its use for a particular operation in the source member being analyzed could be that of a SCALAR or VECTOR variable. This multi-use attribute also applies to external variables of the ARRAY, MATRIX, and STRUCTURE types. Therefore the type of an external variable has to be known.

Another reason for the HAL/S 360 Compiler installation was to generate 360 SDF's as input to HALSTAT. In the data received from NASA, O/S 360 load modules were supplied along with AP-101 SDF's. The inputs required by HALSTAT have to be generated from the same machine types. Also the compiled versions of the SDF's must match the version of the load module they are being used with. Since IBM did not keep or deliver the up-to-date O/S 360 SDF's to NASA, it was again necessary to install the HAL/S 360 Compile to reproduce the SDF's.

The first compilation experiments performed by the Code Group uncovered the restriction that a specific compilation order for all of the FSW source members must be followed. The INCLUDE statement is the compiler directive used to bring in the SDF or template that is included in the source member. If one source member included several external code units (SDF's or compiler templates), then these external code units had to be precompiled with their SDF's or templates stored in appropriate libraries used by the HAL/S 360 Compiler. These INCLUDED external code units could also include other external code units within their units; and so on, with no limit for the nesting of INCLUDE statements. The only way to determine the compilation order was to bring each source member onto a screen editor and record its INCLUDE list. After the master list was produced, the compilation order could be determined by breaking the list into compilation passes consisting of those members having no external units, having only one external unit, having two external units, etc., the process repeating until all the FSW source members were compiled. This method was confirmed by IBM as being the only practical way of determining the compilation order. IBM could not provide TAMU with the compilation order.

The process of recording the master INCLUDE list of the FSW source members took approximately one month of work for three people. The first and second compilation passes could be easily derived from the master list, but the third and succeeding passes required considerably more effort. The first compilation pass consisted mainly of COMPOOL source members while the second compilation pass consisted mainly of interface routines to the COMPOOL source members. The more complex compilation units (PROGRAMS) could not be easily compiled, allowing only 75% of the FSW source code to be expanded and compiled for SDF's. Because of the time constraints involved with the project, it was determined that the best method of analysis would be one that did not involve any compiling. This decision placed the analysis effort using HALSTAT at a standstill until the O/S 360 SDF's for the O/S 360 load modules could be more easily obtained.

2.3.3.3 Overlay or Build Sequence

Concomitant to the compilation order problems, the documentation needed to build a phase of the Space Shuttle Orbiter Software was not provided. By examining the FSW source code only, a source code member could not be distinguished as an overlay or a root segment, nor could it be associated to any particular compilation phase. A load module consisting of a particular phase build was provided in the data from NASA. The load module was in AP-101 format which made the associated segment names and structure of the load module difficult to obtain. Information regarding the building of the many other phases was not provided. This missing documentation limits the source code analysis to a static mode. In other words, a system, comprising of all source members INCLUDING each other, is the largest unit of complexity analysis that can be done. If the build information were available, complexity analysis could be done in a dynamic mode (complexity analysis per phase).

2.3.3.4 HALSTAT

The documentation received about the utility HALSTAT suggested that it could provide much of the analytical information sought in the Code Area effort. Specifically, HALSTAT provides compilation statistics, a memory map of the output object module, and a Global Symbol Directory (GSD). The Code Group wanted to determine exactly how useful HALSTAT was in collecting factors of complexity and reliability of any particular HAL/S source member.

After examining the output from several HALSTAT runs, it was found that some complexity statistics were provided by HALSTAT but others were either not provided or could not be derived easily from the HALSTAT output. A compilation statistic that was useful to this project's source code analysis was the frequency count by instruction type (such as how many IF's, DO's, SCHEDULE's, etc.,). Questions concerning the frequency count of SCHEDULE statements with cyclic (REPEAT) clauses or the total number of real-time statements would have to be hand-tabulated or collected using a post-processor program on the HALSTAT output. The memory map produced by HALSTAT was useful only for purposes of determining the memory size required for a source member. The Global Symbol Directory did provide useful data on the external and internal symbol reference information, but statistics regarding how many times a particular variable was referenced and the scope of the variable were not listed. Some statistics that were not provided by HALSTAT were:

1. the nesting level of subprograms and arguments to subprograms
2. the number of paramters to a subprogram
3. the number of VECTOR, MATRIX, or ARRAY shaping operations

4. the length and nesting levels of IF and DO statements.

For a complete list of complexity factors to be used in the software analysis for this project refer to Appendix D.

As a result of HALSTAT's incompleteness in providing all of the complexity statistics and the previously mentioned compilation problems, it was decided that the effort towards using HALSTAT as a complexity tool should be delayed until more work on the source code analyzer had been done. If the source code analyzer could provide all the complexity statistics with less operating set-up than HALSTAT, then in all probability the incorporation of HALSTAT as a tool would be cancelled. Of course this decision cannot be made until the results from the source code analyzer are done and its operating procedures are compared with those of using HALSTAT.

2.3.4 Source Code Analyzer

The primary objective of the source code analyzer or scanner is to produce data from which measures of complexity and reliability can easily be made. The scanner is designed to output all language constructs of a source member to a dataset that will later be processed by SAS (Statistical Analysis System software) routines. These SAS routines will be able to generate all the complexity statistics listed in Appendix D. The output format of the scanner was designed to be as accommodating as possible for implementation as a SAS input dataset. This was done by including appropriate tag and tabulation fields in the scanner output record. By merging scanner output datasets together through SAS, complexity statistics at all levels of program structure (system, subsystem, module) can be analyzed. Complexity statistics can even be gathered at the programmer level by associating, through SAS, the scanner output datasets with the INVENTORY file provided by NASA. The INVENTORY file contains the programmer's name for each source member. An application example would be to analyze particular software production groups, with respect to reliability and complexity, and rank each of the groups according to the results. In general, the scanner program can provide a statistical data format that can be used as input for a wide range of software studies.

2.3.4.1 Status of Scanner

The HAL/S source scanner has been designed according to the structure shown in Appendix E (Figure 1). SNOBOL will be used in programming the scanner because of its pattern matching and string handling capabilities. Of the modules shown in Appendix E, only parts of the Variable Handler have not been written. The Expression Handler contains the logic that operates on any form of syntactical expression. This module is

the most complex and largest piece of code of the modules within the Variable Handler. The other modules are currently being tested in an isolated mode as well as a connected mode.

2.3.4.2 Definition of Scanner Record

The HAL/S source scanner record is comprised of record fields which provide information regarding a language construct's locality, type, imbeddedness, connectivity, and identification. As shown in Appendix B, the BLOCK NAME and LINE NUMBER fields provide locality information. BLOCK NEST LEVEL, STATEMENT NEST LEVEL, EXPRESSION LEVEL, SUBSCRIPT LEVEL, and ARGUMENT LEVEL fields provide imbeddedness information. The STATEMENT and DESCRIPTION fields provide identification information while the SUBDESCRIPTION and ATTRIBUTES fields provide connectivity and type information respectively.

As mentioned previously, the scanner output record was designed to provide a format in which all complexity statistics could be gathered. Every HAL/S language construct is classified according to either one of four record types:

1. comment
2. statement
3. operator
4. variable.

The RECORD TYPE field is a tag field that broadly classifies the language construct. The LINE NUMBER field enables reference data and variable scope information to be taken since it contains the location within the source member where the construct occurred. The FILE MEMBER NAME field will be used to obtain complexity statistics, through SAS dataset merging, at levels other than the source member level. The BLOCK NAME field provides locality data at the block level for a language construct. It also allows statistics to be taken at the subprogram level. The BLOCK NEST LEVEL, STATEMENT NEST LEVEL, SUBSCRIPT LEVEL, ARGUMENT LEVEL, and EXPRESSION LEVEL fields give imbeddedness information. For instance, the depths of imbedded subprograms, IF and DO statements, the number of subprograms that an argument is passed through on one subprogram invocation, and the depths of parenthesized expressions such as those used for subscript evaluation, will be recorded. These fields give data not obtainable through HALSTAT. The DESCRIPTION and SUBDESCRIPTION fields qualify the language construct by recording such items as symbol names and statement subtypes. A variable construct will have its name placed in the DESCRIPTION field and its STRUCTURE name in the SUBDESCRIPTION field if it is of STRUCTURE type. A statement construct will have any subphrase (the REPEAT phrase on the SCHEDULE

statement, the FOR and WHILE/UNTIL phrases on the DO statement, etc.,) information recorded. The ATTRIBUTES field identifies how a variable is defined and how it is used in a statement.

2.4 CHANGES AREA

The goal of the changes area was to analyze information concerning software changes. The first step toward achieving this goal was determining what information was available and documenting that information. The available data was then ready to analyze.

2.4.1 Availability of Change Data

Originally the TAMU research team was given five tapes which were supposed to contain data on change reports (CR), discrepancy reports (DR) and program change reports (PCR). While trying to access this data, it was determined that only two of the tapes contained valid information. The two good tapes contained the OFT DR data base and the OFT CR data base.

2.4.1.1 Reliability Data

Traditional hypothesis-testing techniques may be used as a management tool by software developers or software purchasers who wish to insure that their packages have some specified reliability level. The conditions that must be met are:

1. the existence of independent collections of test data
2. a way of determining the correctness of processing of these collections
3. a way of randomly selecting test data.

Two basic approaches are available. In a fixed sample size test, the user decides on the reliability desired. The number of test cases which must be examined based upon the acceptance/rejection criteria can then be determined. In a sequential test, the desired reliability level is again pre-determined, but samples are tested one at a time until an accept/reject decision can be made.

Experiments with a large amount of error data derived from several systems indicate that reliability results derived from these models are consistent with actual reliability figures.

Most current acceptance procedures are based upon a naive assumption that a large program can be exhaustively tested and delivered in an error-free condition. Because these expectations cannot be fulfilled, the manager of a software development project or the purchaser of a software product is provided with no quantitative information on which to base an acceptance decision and is thus forced to make these decisions based mostly on intuition and his own experience in similar situations. These models allow one to replace these intuition-based decisions with quantitatively-based decisions and thus constitute an important contribution to the science of management of software development efforts.

In determining what data was available to apply the models to the 'shuttle avionics software' it was determined that sufficient data was not available. The data saved during reliability testing consisted of only failures of the software to perform as expected. The application of the model requires that results from all test cases be available.

2.4.1.2 DR and CR Data

Data were supplied for discrepancy reports (DR) and change reports (CR). After overcoming initial problems with reading the data tapes, errors in data entry resulted in loss of some data, and made early analyses difficult.

Extensive analysis of the change data was difficult. This was due to no CR and incomplete DR variable documentation being supplied with the data tapes. Supplemental data was requested. Information returned pertaining to DR's was prompt and complete. However, our request for information on the CR data was long outstanding, and incompletely answered. This resulted in the DR data base being analyzed in greater depth.

2.4.1.3 DR and CR Legends

Initially, the documentation for the tape containing the DR data base consisted of a variable list and a legend. This DR legend was in the first file of the DR data base tape and was helpful, but incomplete. Many of the DR variables were not explained in the legend. These unexplained variables were: Verification Assignment for Special (GA), Pre-Build Assessment Data (PD), Pre-Build Assessment Reason (PR), T&O Closure Code (X), Verification Status Data (S), Verification Status Data (V), Future Closure Code (IMP), and Verification Baseline ID (BL). Supplemental information concerning these variables was requested and the information returned allowed a complete DR legend to be compiled. The DR legend appears in Appendix J.

The initial documentation for the tape containing the CR data base was a variable list. A legend for this tape was requested, but none was found

in existence. A list of the CR variables and data base entries was then compiled, and information concerning these was requested. When this information was returned a CR legend was written. This CR legend appears in Appendix L.

2.4.2 Methodology of Analysis of Change Data

The first goal was to transfer the change data from the tapes supplied by NASA to the Amdahl located at TAMU. The data were eventually placed in a Statistical Analysis System (SAS) data base.

Analyses were done in two phases. A descriptive analysis approach was first taken to examine what data were present and their usefulness. Examination of selected single variables was done to determine which values were predominant. This was followed by an indepth analysis examining the frequency of several variables concurrently.

All analyses were done using SAS. Plots and histograms were commonly used to display the results.

Due to the lack of a legend for the CR data, during most of the project, attention was focused primarily on the DR data.

2.4.3 Results of Analysis of Change Data

The change data base consisted of 29,219 entries, for 14,156 DR's, and 6,282 entries, for 5331 CR's. Dr data began June, 1975 and extended through December, 1981, Appendix N, Figure 7 (all figures, unless otherwise noted, are in Appendix N).

1. The frequency of dates DR's were received was examined. Figure 7 shows the data are multimodal, with two prominent trends: October, 1978 through October, 1979, and January, 1981 through December, 1981. Data were further scrutinized to explain these trends, and will be discussed later.
2. Frequency for mission identifier (MISSN ID) is shown in Figure 8. By far, most DR's, 7040, pertained to STS-1 (L). The next most common mission identifiers, STS-2 R18V21 flight system (R) and S-2 R17V5 (W), each had almost 2000 DR's logged.
3. The most common reporting facility (FAC) was the IBM verification personnel (VER), Figure 9. This facility logged over 8,000 DR's. The orbital flight test (OFT) facility entered the next highest number, just under 3200.

4. The dominant area responsible for a fix (AD) was user interface (UI), Figure 10. Other important areas include FLT computer operating system (FCOS), requirements analysis (RA), and vehicle utility (VU).
5. Priority of the DR was examined, Figure 11. Of the six possible values, priority 2 and 5 were most common. Priority 2, desirable, was specified for almost 15,000 DR's. Priority 5, dispositioned for no mod or no DR closure, was found for about 7,100 DR's. Only 435 DR's were given critical priority (1).
6. Examination of DR status (ST) was done, Figure 12. A status of closed was found for over 26,000 DR entries (Status = C, CC, and VX). The remaining entries were either awaiting verification (V), or NASA/SSD (N) approval, or in system test (T).

After examining the frequencies of certain single variables, more specific information was extracted. This involved analyzing the frequencies of two variables concurrently. Priority was scrutinized first.

1. Priority (P) was examined in relation to date the DR was received (ODYRMO), mission identifier (MISSN ID), and reporting facility (FAC). Figure 13 shows that most of the priority 1 (critical), occurred before 1980. Priorities 2 and 5 showed a shift, over time, in the number reported each month. Priority 2 was dominant before November, 1979. Beginning January, 1981, priority 5 was the mode. 1980 appeared to be a period with equal numbers of levels 2 and 5 priority DR's. Greater resolution of the frequency of DR's logged in per month, by priority, is given in Figures 14-19.

Results of examination of priority for a mission identifier are shown in Figure 20. The dominance of priorities 2 and 5 is clearly evident. Closer examination, by mission identifier, shows that priorities 2 and 5 did not occur together in great numbers. Only for MISSN_ID L and W did the two occur in abundance together. This would indicate that a particular mission was either relatively absent of real errors, or had many.

The priority of DR's originating from a reporting facility was analyzed. Figure 21 reveals which facilities generate the most DR's of a given priority and the relative abundance of priorities generated by that facility. Most priority 2 DR's originated from guidance, navigation, and control (GNC), orbital flight test (OFT), software development laboratory (SDL), and IBM verification personnel (VER). Most priority 5 DR's originated from requirements analysis (RA) and IBM verification personnel. Only for the IBM verification personnel facility did the number of priority 5 DR's approximate the number of priority 1 and 2 DR's. A closer view of the number of DR's from a facility, per priority, is given in Figures 22-27.

2. Results of examining the number of DR's generated per mission identifier, are given in Figures 28-57. These figures reflect two items, the lack of complete documentation and the length of time that mission was developed. Figure 28 displays the number of DR's with missing mission identifiers, 4408 in all.

Several patterns in the frequency per mission, over time, are evident. Most missions have DR's extending over a period of one year or less, for example C (Figure 31), G (Figure 35), and H (Figure 36). However, some missions had entries extending over several years; missions L, M, and W are examples. Each one of the latter displayed a different frequency pattern. Mission L, STS-1 FUP5 (Figure 38), shows a multimodal distribution, skewed towards the earlier dates. This is reasonable because the DR's should decline to acceptable levels before the shuttle flight. Mission M, STS-1 FUP4 (Figure 39), has a more unimodal trend. In contrast, mission W, STS-2 (Figure 54), has a distinct multimodal distribution over its 35 month span. This multimodality is probably due to multiple releases pertaining to it.

3. Four groups of similar releases were examined to determine if the number of DR's generated per release declined as the release order increased, Figures 58-61. Three of the groups displayed this trend. However, mission_id S7, SDL release 37, showed an increase of 13 percent over mission_id S6, SDL release 36, Figure 60.

Examination of CR variables was minimal due to the lack of information available about the variables.

1. Data pertaining to manpower impact were plotted: manpower impact by department and year (D1IMPYR1-D7IMPYR4), Figures 62-88 and total manpower impact by department (D1TOTIMP-D7TOTIMP), Figures 89-95. Manpower impacts by year and department all showed a similar pattern with modes of 15 units or less. Exceptions are department 6, year 3 (Figure 84) and department 7, year 3 (Figure 87); both had impacts which did not exceed one. Department 7, year 2 had no observations.

Maximum values for total manpower impact ranged from 18 for department 6 (Figure 94) to 42000 for department 5 (Figure 93). Midpoints of the modes for each department was zero, indicating that most CR's had small impact values.

The significance of these data is not understood because of the lack of information supplied by NASA and IBM.

2. The amount of time a DR or CR took to get to a particular stage in the change process was studied. The number of months needed for a DR's development close date (BD) and verification close date (VC) was determined. Figure 96 shows that DR development

close date was essentially the month it was entered. The verification close date, on the other hand, did take up to four years, with a majority of DR's closed in 18 months, Figure 97.

The amount of time a CR took to get dispositioned by the RR and OASC boards was examined. CR's were dispositioned by the RR board within 24 months, and most within 2 months after being received, Figure 98. CR's which required an OASC board disposition were processed within 20 months, with a majority within two months, Figure 99. A similar plot resulted when examining the amount of time a CR took to get from a RR board to a OASC board disposition, Figure 100.

2.5 SUMMARY OF DATA AVAILABILITY

Table 3 is a summary of the availability of data found by the research team in examining the shuttle software development. Data is coded as having been found in machine readable form or in hard copy form. The data found to be available is also coded as to whether the research team found the data beneficial in extracting useful information.

TABLE 3

Summary of Data Availability

DATA	Hard Copy	Machine Readable	Available not useful	Available useful	Not Available
COST					
Man-hours					
by date	X			X	
by dept					X
by shuttle funct					X
SOURCE CODE					
Source listings		X	X		
Source code		X		X	
Load Modules		X		X	
HAL/S Compiler		X		X	
HALSTAT		X		X	
SDF library		X	X		
Inventory file		X		X	
Build JCL					X
CR'S					
CR file		X		X	
CR legend	X			X	
DR'S					
DR file		X		X	
DR legend		X		X	
VERIFICATION					
No. of test					X
Type of test					X
Cut come of test					X

FUTURE WORK

The data which has been collected to date in the shuttle software development provides a wealth of information about the life cycle of large, real-time programs. This type of data is often not available to computer researchers, particularly data for the development of real-time software. Several studies have begun at Texas A&M using the available shuttle data and can be beneficially continued. These projects are outlined in section 3.1. As a result of our study of the available data we have identified certain analysis and data collection monitoring functions (presented in section 3.2) which could be performed at Texas A&M concurrent with on-going software development at JSC. Section 3.3 highlights additional projects which we have identified as appropriate for future work. Some of these future projects will necessitate the collection of additional data during future shuttle software development.

3.1 CONTINUED ANALYSIS

3.1.1 Continued Development of Cost Data

All the cost data from the monthly 533 reports which are prepared by IBM have been encoded and analyzed as described. The research team has been informed that more detailed documents below the 533 level exist and can be made available. These documents could provide some very interesting results once coded and analyzed.

3.1.2 Continued Work with Language Scanner

The continuation of work involving the HAL/S source analyzer will be comprised of three areas. First, the coding and initial testing of the source analyzer will be completed. Next the SAS procedures to analyze the source scanner output will be developed. Finally, statistics will be grouped concerning subsystem and system levels of the Flight Software specifically regarding elements of the language used, programming style, and complexity ranking.

In the initial testing phase of the scanner project, each test run will contain the output for one source member. Since the scanner is designed to analyze the uncompiled source, several runs may have to be made for

the external members INCLUDED in the source member being processed. The raw scanner output data will be checked for correctness and frequency count statistics will be tabulated from it. It may be possible for some source members to be compiled and used as input to HALSTAT for a listing that can be used for verifying the results from the raw scanner output data.

After the scanner has been debugged and its results verified, the scanner will be run against each FSW HAL/S source member. Those source members that are composed of AP-101 Assembler code, will have HAL/S Compiler templates written to be analyzed. These source members are mainly located in the System Source Library. Instruction mix and variable reference data from the AP-101 modules will not be available from the compiler templates, but connectivity information will be preserved. Later work will be done to design a similar type scanner to analyze the AP-101 Assembler modules to provide the instruction mix and variable reference data temporarily unobtainable. All output from each source member will be stored on tape with the FILE MEMBER NAME field of the output scanner record identifying the particular source member.

When the scanner has been run against all the FSW source members, some simple SAS routines will be run against the scanner output for one source member. These initial SAS routines will report on the number of instructions occurring by type, variables occurring by common attributes, and other statistics that can be drawn from frequency count methods. Once again, HALSTAT may be used as a tool for verifying the SAS results. More complex SAS routines will be written later to output variable scope, IF and DO nesting level, and subprogram argument nesting level information at the source member level.

After all complexity statistics are generated from each source member, the SAS output datasets can be sorted and merged to provide complexity statistics at the subsystem/system level. For example, statistics can be produced for the set of modules comprising the Process Control section of the Process Management subsystem of the Flight Computer Operating System. It will be possible to generate complexity statistics at the programmer level by merging the previously mentioned INVENTORY file with the SAS output datasets. If it is known what source members were written by a particular programmer (information which the INVENTORY file provides), then these statistics are possible. If each software production group personnel roster was made available from IBM, then complexity statistics could be obtained at the software production group level.

Using the complexity results from SAS, it will be possible to determine which of the HAL/S complexity factors listed in Appendix C should be used as guides in ranking the FSW source modules in regard to complexity. The complexity factors that will be chosen as ranking guides will be the ones that show a variance within the source modules. Once the ranking factors are established, FSW source members, software subsystems/systems, as well as programmer software collections can be ranked with one another.

2.1.3 Continued Analysis of CR and DR Data

The analyses of CR and DR data described earlier in this report give some indication that a measure of the stage of development (either as percent code completed or as number of future changes to be required) may be related to characteristics of previous CRs and DRs (e.g., the number, mission identifier, reporting facility, priority mix, etc.). It is possible that these results may be used to model software quality in a manner similar to that proposed by Mendis (1982).¹⁴

To this stage, the analysis of CR and DR data has examined two-way relationships between date and a characteristic of the CRs and DRs or between two characteristics. Future analyses of CR and DR data should include further investigation of the relationship between the historical CR and DR data for software development and measures of the reliability, or level of completion, for the software. In particular, multivariate relationships for three or more characteristics should be examined. In addition to using CR and DR data, information gathered by the HAL/S source analyzer and other characteristics such as language used, programming style, etc. may improve the modeling of the stage of development for software.

3.2 CONCURRENT ANALYSIS AND MONITORING

In order to maintain a repository of information about the development of the space shuttle software system, TAMU suggests the following data be collected for each release of the space shuttle software.

1. A copy of all distribution material made to NASA test facilities.
2. A copy of all source code libraries.
 - a) Application source
 - b) System source
 - c) Include source
3. A copy of the current version of the HAL/S compiler, HALSTAT, the linkage editor, and associated documentation.
4. A copy of the build information necessary to construct a build from the above libraries and execute all options on HALSTAT.
5. A copy of all cost information.
6. A copy of the cumulative DR and CR files.

All of these items should be in machine readable form whenever possible.

TAMU would like to serve as a repository of this information.

3.3 FUTURE DEVELOPMENT

Although many possibilities for future projects and analyses on the shuttle software development effort exist, the TAMU research team has identified four related areas for which future work seems particularly beneficial. These efforts in general involve the development and/or implementation of models which can provide input to decision-makers at NASA as well as in the computer research community at large. If these models are developed in conjunction with the continued and expanded collection of the shuttle software data as described in the previous sections, the model will also provide tools useful in tracking the evolution of the development effort.

3.3.1 Comparison with other Projects

Future research on the statistical and historical information gathered during the development of the shuttle orbiter primary flight software can be conducted in several directions. The available data from the flight software is from a unique environment, a substantial ongoing real-time software project, and thus provides an excellent groundwork for reverification, comparison, modeling, and subsequent verification of metrics. Comparisons formed on a single large real-time software system will be more significant to the computer science community than comparisons formed on small samples of experimental software. For instance, most studies that have been made are performed on small data samples that are potentially biased, while the few studies performed on larger sample sizes differ greatly in the type of errors considered and in the data collection techniques utilized. In addition, although several complexity models have been proposed in the literature, they have not all been verified, and studies attempting to verify proposed mathematical models for software reliability models are scarce.

Several objectives will be obtained by researching the verification and reverification of complexity measure and reliability predictor models on the space shuttle software. Since they will have a standard software basis for comparison, the various metrics can be ranked in regards to their effectiveness. After the models are "tuned" statements can be made concerning the various types of error predictors/analyzers, such as stating that models which measure control structure complexities are more effective than models which rely on length of code. A more effective complexity metric will be developed to pinpoint the areas of greatest complexity within a software system, thus providing management with a tool to use in allocating various resources such as personnel and time when evaluating and making changes to a software area of greater error potential.

3.3.2 Complexity Analysis

Complexity is the term generally used to describe the difficulty in developing software or in the software's resistance to modifications.² Complexity metrics are designed to measure complexity related to a human process -- namely, programming -- as contrasted with computational complexity where the area of concern is a machine procedure or machine resource allocation. A number of metrics have already been discussed in the survey of literature section of this report (see section 2.1).

Complexity analysis of the HAL/S software could be expected to provide insight into the relative quality of the program as a whole and its individual components. Such measurements would be useful in estimating costs of maintenance and enhancements of existing components. Furthermore the metrics could be used in decisions to replace or modify sections of code. Models based on the complexity metrics could be used in estimating the likelihood of errors within each program module. Since the space shuttle project is ongoing, the proposed research on complexity measures will provide continuous future error prediction to help in identifying the locations of potential errors within the project software and some indication of how soon a particular software build (or system) will be ready to be utilized. Accordingly, the results obtained can also be used for future projects and may help reduce costs by enabling management to allocate resources more optimally.

Although a number of models utilizing complexity metrics have been suggested in the literature, only limited validation of these models has been accomplished, often with data from small software projects. Appropriate models could be established and validated using the HAL/S developmental data. These models would then be available for use in decision-making in future HAL/S implementation efforts.

3.3.3 Cost Analysis

For future projects, all cost and hours information should be kept in machine readable form. Standard reporting categories should be established and adhered to throughout the project. Further subdivisions of categories could provide some flexibility and provide usable detailed information. While this data could be consolidated for managerial reporting purposes its detail could provide valuable information for future analysis. If new categories become needed they could be added to satisfy the current requirements.

The cost impact of DRs and CRs on the software is another area in which more information would be useful. The cost or time actually taken to dispose of each DR or CR should be kept in machine readable form.

Some valuable analysis could be done in this area. Statistics could be gathered in the total costs of DRs and CRs, the total cost of DRs and

CRs versus the total cost of development, the total cost of DRs and CRs in one subsystem or function of the shuttle versus another function, and perhaps the average time it takes to correct a DR.

3.3.4 Language Analysis

Software development at the present state-of-the-art cannot be evaluated independently of the programming language employed in its implementation. Although researchers note the need for the development of precisely defined universally accepted software evaluation parameters, a suitable number of metrics of this type simply do not exist at this time.¹⁹ Therefore an important area of concern in software evaluation is programming languages.

To facilitate this type of analysis certain language metrics are needed. These metrics tend to be more subjective and harder to define than program metrics which are used in evaluating the programs written in a language. The primary traditional uses for language metrics are in language selection and/or comparison and for language design.²⁰

The existing language metrics could beneficially be applied to HAL/S with several useful results. First, since HAL/S was specifically designed for spacecraft software development, some measure of its effectiveness could be obtained using language metrics. Since most of the existing metrics produce relative measures, the same measurements could be computed for other high level languages for the purpose of comparison with HAL/S. Of particular interest would be metrics which measure the applicability of a language to a specific application. Other metrics of interest might include comparison of the desirability of various language features, comparisons of the level of non-procedurality of languages (since non-procedural implies a lesser need for the programmer to specify implementation details), and comparisons of potential programmer productivity in each language.

The new Department of Defense language Ada is the most natural choice for a language with which to compare HAL/S. The results of such a comparison would be useful to NASA in making future language selections. In addition, since HAL/S has already been successfully used in large real-time software projects while Ada has not, HAL/S could be considered a standard against which Ada could be evaluated.

The language analysis of HAL/S would utilize existing program metrics as well as some designed specifically for this task. If a comparison with Ada is desired, a preliminary Ada compiler developed by New York University under contract with the U. S. Army is available and operational at Texas A&M University.

Appendix A

DOCUMENTATION

This Appendix consists of four lists of documents. These lists are as follows:

- List 1 - Specifications prepared by NASA
- List 2 - NASA documents received by the research team
- List 3 - Programs specifications prepared by IBM
- List 4 - IBM documents received by the research team

N A S A Documents

The Computer Program Development Specification (CPDS) is composed of 4 volumes which consist of 18 books. The following is a breakdown of the volumes:

- Vol 1 - Level A, Requirements not oriented to any particular end item, 7 books
- Vol 2 - Shuttle Orbiter Software Requirements, Program Notes and Waivers
- Vol 4&5 - Computer end item-oriented functional and detail requirements

The following lists the identifying numbers, volumes, and books that constitute the CPDS structure:

DOCUMENT NO.	VOLUME/BOOK	TITLE
SS-P-0002-120	I 2	System Level Requirements, Software
SS-P-0002-130	3	ALT Launch Data Bus Software Interface Requirements
SS-P-0002-140	4	Downlist/Uplink Software Requirements
* SS-P-0002-150F	5	OFT Launch Data Bus Software Interface Requirements

* SS-P-0002-170G		7	OFT System Level Requirements, Software
SS-P-0002-190		9	ALT Flight Software Initialization Load
SS-P-0002-195		9.5	STS Flight Software Initialization Load
SS-P-0002-210	II	1	ALT Program Notes and Waivers
SS-P-0002-220		2	OFTIS Program Notes and Waivers
SS-P-0002-410	IV	1	ALT Functional Level Requirements GN&C
SS-P-0002-420		2	ALT Detail level Requirements GN&C
SS-P-0002-430		3	ALT Functional level Requirements Systems Management
SS-P-0002-440		4	ALT Detail Level Requirements Systems Management
SS-P-0002-450		5	ALT Functional Level Requirements, Vehicle Utility and Data Flow-01
* SS-P-0002-510N	V	1	STS Functional Level Requirements, GN&C
SS-P-0002-515			OFT Functional Level Requirements GN&C, STS-1 and 2
* SS-P-0002-530J		3	STS Functional Level Requirements, Systems Management
* SS-P-0002-550J		5	STS Functional Level Requirements, Vehicle Utility-02
SS-P-0002-580		8	STS Operations Functional Level Requirements, Systems Managent
SS-P-0002-590		9	PASS GPC Memory Write Procedures

The Level C or Detailed requirements are contained in documents labelled Functional Sub-System Software Requirements (FSSR).

These documents are listed below:

TITLE	NUMBER
GN&C	
Guidance FSSR	SD-74-SH-0273 SD-74-SH-0273A SD-74-SH-0273B PCN-1 SD-74-SH-0273C
Navigation FSSR	SD-74-SH-0272 SD-74-SH-0272A SD-74-SH-0272B PCN-1
SD-74-SH-0272C Flight Control FSSR	SD-74-SH-0271 SD-74-SH-0271A PCN-1 PCN-2
SD-74-SH-0271B	
RM/MSC FSSR	SD-74-SH-0270 SD-74-SH-0270A SD-74-SH-0270B PCN-1 SD-74-0207C
SOP FSSR	SD-74-SH-0269
Preliminary	PCN-1
Air Data	PCN-2 PCN-3 PCN-4 PCN-5 PCN-6
	SD-74-SH-0269A SD-74-SH-0269B
Displays and Control FSSK	SD-74-SH-296 SD-74-SH-296A SD-74-SH-296B SD-74-SH-296C
SM CPDS	SS-P-0002-440 SS-P-0002-440A SS-P-0002-440B SS-P-0002-440C SS-P-0002-440D SS-P-0002-440E SS-P-0002-440F

SM FSSR	SD-74-SH-0295 SD-74-SH-0295B SD-74-SH-0295C SD-74-SH-0295D PCN-1
Vehicle Utility-01 CPDS	SS-P-0002-460
VU-01 FSSR	SD-75-SH-0077 SD-75-SH-0077A
GN&C, Part A Vol. 2 Guidance	SD76-SH-0001C PCN-1 PCN-2 PCN-3 PCN-4
GN&C, Part A Vol. 2 Guidance-Ascent	SD76-SH-0002E PCN-1 PCN-2 PCN-3 PCN-4
GN&C, Part A, Vol. 1 Guidance-On Orbit	SD76-SH-0003B PCN-1 PCN-2
GN&C Part B,,Nav- Entry + APP A-H	SD76-SH-0004D PCN-1 PCN-2 PCN-3 PCN-4
GN&C, Part B, Nav-Ascent/RTLS + APP A-N	SD-76-SH-0005C PCN-1 PCN-2 PCN-3 PCN-4
GN&C, Part C Nav-on Orbit	SD76-SH-0006C PCN-1
GN&C, Part C FC-Entry	SD78-SH-0007B PCN-1 PCN-2

GN&C, Part C
FC-Ascent

SD76-SH-0008
Vol 1C
PCN-1
PCN-2
PCN-3
Vol 2D
PCN-1
PCN-2
PCN-3

STS Baseline is
in CR 19361A

Vol 3C
PCN-1
PCN-2

GN&C, Part C, Vol. 3
FC

On Orbit (1)
On Orbit (2)

SD-76-SH-0009
REV A
REV A
PCN-1
PCN-2

GN&C, Part D, Vol. 1
RM-Entry

SD-76-SH-0010E
PCN-1
PCN-2

GN&C, Part E, Vol 1
SOP-IMU

SD76-SH-0013A
PCN-1
PCN-2
PCN-3
PCN-4

GN&C, Part E
SOP
NAVAID

ST TRK

SD76-SH70014
Vol 1A
PCN-1
PCN-2
PCN-3
Vol 2A
PCN-1
PCN-2
PCN-3

GN&C, Part E, Vol. 3
SOP-Sensor/
Controller

SD76-SH-0015B
PCN-1
PCN-2
PCN-3

GN&C, Part B, Vol 4 SOP-FC Effectors	SD76-SH-0016B PCN-1 PCN-2 PCN-3
VU-02	SD76-SH-0019A
GN&C, VOL. 1 D&C-Display Formats	SD76-SH-0020C PCN-1 PCN-2 PCN-3
GN&C Sequence Rgmts	SD76-SH-0026B PCN-1 PCN-2
SM	SD76-SH-0027E
RMS	SD77SH-0002E
PAYLOAD MANAGEMENT	SD78-SH-0044
GPC INTERFACES	SOD76-0002 BK 1
GPC INTERFACES	SOD79-0002 BK 2

NASA DOCUMENTS RECEIVED

FUNCTIONAL DESIGN REQUIREMENTS

OFT Launch Data Bus Software Interface Requirements-

(SS-OP-0002-170G)

OFT System Level Requirements, Software - (SS-P-0002-150F)

STS Functional Level Requirements, GN&C - (SS-P-0002-510N)

STS Functional Level Requirements, System Management -

(SS-P-0002-530J)

STS Functional Level Requirements, Vehicle Utility-02 -

(SS-P-0002-550J)

MANAGEMENT

Mass Memory Unit Software Integrataion Document and Revisions -

JSC 1674

Statement of Work Space Shuttle Avionics Orbiter Software -

Schedule II

JSC - 08338 - Mass Memory Unit Computer Program

Integration Plan

Vol. I Release Control

Vol. I, Book 1 SPF Level A Generic Requirements

Vol. II Release Authority and Schedule

IBM DOCUMENTS

IBM, in response to the requirements defined by the NASA and documented in the Level A and B Computer Program Development Specifications, developed the System Design Specifications. The System Design Specifications is a series of documents consisting of Functional Design Specs (FDS) and Detail Design Specs (DDS).

The list which appears below was taken from a manual dated February 25, 1977. Therefore, this list might not be as accurate as it could be had more recent information been made available.

Vol. I - Software System Overview

Forward

Table of Contents

1. Introduction
2. Design Considerations
3. Functional Description
4. Flight Software Control Structure
5. Flight Software Dynamic Structure

Appendix A - Acronyms and Abbreviations

Appendix B - Common COMPOOL

Vol. II - System Services

Part 1 - FCOS

Forward

Table of Contents

1. Introduction

2. Functional Description

3. Software Design

3.1 Subsystem Design Overview (Include
Control Flow Trees)

3.2 Major Element 1 Overview (Includes
Control Flow Trees)

3.2.1 Module A

Function

- a) Control Interface
- b) Input Data
- c) Process Description
- d) Output Data
- e) Module References
- f) Module Type and Attributes
- g) Template References
- h) Error Handling
- i) Constraints and Assumptions

Data tables (Module Data List, I/O)

SVC Table

Control Flow (Preliminary Design)

Control Flow(s) (Detailed Design)

3.2.2 Module B

..

..

..

3.2.n Module N

3.3 Major Element 2 Overview (Includes
Flow Trees)

...

...

...

3.n Major Element n Overview (Includes
Flow Trees)

Part 2 - User Interface

(Same format as Part 1)

Part 3 - System Control

(Same format as Part 1)

Vol. III - Applications

Part 1 - GN&C

(Same format as Vol. II, Part 1)

Part 2 - SM

(Same format as Vol. II, Part 1)

Part 3 - VCO

(Same format as Vol. II, Part 1)

Part 4 - Payload

(Same format as Vol. II, Part 1)

Vol. IV - Summary Integration Specification

1. Introduction

2. Process Interaction

3. Control Block Allocation

4. Mappings

5. Critical Parameter Addresses

6. Downlist Loading

IBM DOCUMENTS RECEIVED

FUNCTIONAL DESIGN SPECS

OFT - Functional Design Specifications

DETAIL DESIGN SPECS

SDS - Volume III, Part 3, OFT DDS, Vehicle Utility and Data Flow 02

OFT - System software Designs Specs - User Interface

FCOS-UI-SC - Design Specs

SDS Volume III, Part 2 - System Management Design Specs

USER'S GUIDES

Flight Computer Operating system User's Guide (18)

Test and Operations User's Guide (18)

Shuttle Flight Operations User's Guide (18)

System Analysis User's Guide - User Interface

UI/SC User's Guide STS-1

FSW Utilities User's Guide

MANAGEMENT

Programming Standards Analysis Procedures

Programming Standards Document and Updates

Complete Software Awareness Memos Update
Shuttle Avionics Software Avionics STS-1 Operating Plan
Shuttle Avionics Software Management Plan
Reliability and Quality Assurance Plan
Flight Software Memory Sizing and CPU Loading Estimates

MANUALS

AP 101, C/M Principles of Operation
SDL - User's Guide for HALSTAT
Linkage Editor for Flight Computer (partial)

MISCELLANEOUS

Space Shuttle Orbiter Avionics Software - STS-3 Flight
Software Memory Data Base
Onboard Shuttle Software Design/Code Checklist

Appendix B

HAL/S ANALYSIS SCANNER OUTPUT

Overall Record Description

<u>Field Description</u>	<u># of Chars</u>	<u>Values</u>
RECORD TYPE	2	CM - COMMENT ST - STATEMENT OP - OPERATOR VR - VARIABLE
LINE NUMBER	6	/* LINE # OF HAL/S INPUT
RECORD NUMBER	8	/* SCANNER OUTPUT RECD #
FILE MEMBER NAME	8	/* FILE CONTAINING SRC
BLOCK NAME	32	/* PROCEDURE, FUNCTION, PROGRAM, ETC., SURROUNDING TOKEN */
BLOCK NEST LEVEL	2	/* IMBEDDEDNESS OF BLOCK
STATEMENT NEST LEVEL	2	/* IMBEDDEDNESS OF IF OR DO CONSTRUCTION
SUBSCRIPT LEVEL	2	/* LEVEL OF SUBSCRIPT EXPRESSION: A\$(B\$(C))
ARGUMENT LEVEL	2	/* LEVEL OF ARGUMENT WITHIN SUBPROGRAM INVOCATION NAME(NAME(A))
EXPRESSION LEVEL	2	/* LEVEL OF PARENS (A+(B C(E-2)))
STMT REFERENCE NUMBER	8	/* COL. 73-80 OF INPUT
STATEMENT	20	/* INSTR. KEYWORD CONTAINING TOKEN */
DESCRIPTION	40	/* SEE PARSE EXAMPLES

ATTRIBUTES
LETTER CODES */

10

/* SEQUENCE OF STORAGE

AC - ACCESS
AL - ALLIGNED
AR - ARRAY
AS - ASSIGN USAGE
AU - AUTOMATIC
BI - BIT
BO - BOOLEAN
CH - CHARACTER
CO - CONSTANT
DN - DENSE
DP - DEPENDENT
DO - DOUBLE
EQ - EQUATE
EV - EVENT
EC - EXCLUSIVE
EX - EXTERNAL
FX - FIXED
IN - INITIAL
IP - INPUT USAGE
IT - INTEGER
LB - LABEL
LT - LATCHED
LH - LEFT HAND =
LO - LOCK
MA - MATRIX
MF - MATRIX FIXED
NA - NAME
RA - RANGE
RE - REMOTE
RI - REINTRANT
RH - RIGHT HAND =
RG - RIGID
SC - SCALAR
ST - STRUCTURE
SB - SUBSCRIPT USE
TE - TEMPLATE
TP - TEMPORARY
VC - VECTOR
VF - VECTOR FIXED

Total Record length = 138 characters

PARSE EXAMPLES

HAL/S Source:

```
C THIS IS A COMMENT LINE
  /* WITH A CR99999 DESCRIPTION AS AN
    IMBEDDED COMMENT */
```

Scanner Output Recd #1

```
RECD TYPE = CM
LINE # = 000001
RECD # = 00000001
BLKNAME =
BLK NEST LVL = 00
STMT NEST LVL = 00
SRN =
STMT = COMMENT
DESCR =
ATTRIB =
```

Scanner Output Recd #2

```
RECD TYPE = CM
LINE # = 000002
RECD # = 00000002
BLKNAME =
BLK NEST LVL = 00
STMT NEST LVL = 00
SRN =
STMT = COMMENT
DESCR = CR99999
ATTRIB =
```

HAL/S Source:

```
D INCLUDE TEMPLATE APPLTEMP
D INCLUDE           INCLTXT
```

Scanner Output Recd #3

```
RECD TYPE = ST
LINE # = 000003
RECD # = 00000003
BLKNAME =
BLK NEST LVL = 00
STMT NEST LVL = 00
SRN =
STMT = INCLUDE
DESCR = APPLTEMP
ATTRIB = TE
```

Scanner Output Recd #4

```
RECD TYPE = ST
LINE # = 000004
RECD # = 00000004
BLKNAME =
BLK NEST LVL = 00
STMT NEST LVL = 00
SRN =
STMT = INCLUDE
DESCR = INCLTXT
ATTRIB =
```

HAL/S Source:

HALSPGM: PROGRAM;

Scanner Output Recd #5

RECD TYPE = ST
LINE # = 000005
RECD # = 00000005
BLKNAME = HALSPGM
BLKNEST LVL = 01
STMT NEST LVL = 00
SRN =
STMT = PROGRAM
DESCR = HALSPGM
ATTRIB =

HAL/S Source:

```
DECLARE INTEGER, SENSOR1, SENSOR2 DOUBLE;
```

Scanner Output Recd #6

```
RECD TYPE = ST  
LINE # = 000006  
RECD # = 00000006  
BLKNAME = HALSPGM  
BLK NEST LVL = 01  
STMT NEST LVL = 00  
SRN =  
STMT = DECLARE  
DESCR =  
ATTRIB =
```

Scanner Output Recd #7

```
RECD TYPE = VR  
LINE # = 000006  
RECD # = 00000007  
BLKNAME = HALSPGM  
BLK NEST LVL = 01  
STMT NEST LVL = 00  
SRN =  
STMT = DECLARE  
DESCR = SENSOR1  
ATTRIB = IT
```

Scanner Output Recd #8

```
RECD TYPE = VR  
LINE # = 000006  
RECD # = 00000008  
BLKNAME = HALSPGM  
BLK NEST LVL = 01  
STMT LVL = 00  
SRN =  
STMT = DECLARE  
DESCR = SENSOR2  
ATTRIB = ITDO
```

HAL/S Source:

CALL ADDSENSOR(SENSOR1) ASSIGN(SENSOR2);

Scanner Output Recd #9

RECD TYPE = ST
LINE # = 000007
RECD # = 00000009
BLKNAME = HALSPGM
BLK LVL = 01
STMT LVL = 00
SRN =
STMT = CALL
DESCR = ADDSENSOR
ATTRIB =

Scanner Output Recd #10

RECD TYPE = VR
LINE # = 000007
RECD # = 00000010
BLKNAME = HALSPGM
BLK LVL = 01
STMT LVL = 00
SRN =
STMT = CALL
DESCR = SENSOR1
ATTRIB = IP

Scanner Output Recd #11

RECD TYPE = VR
LINE # = 000007
RECD # = 00000011
BLKNAME = HALSPGM
BLK LVL = 01
STMT LVL = 00
SRN =
STMT = CALL
DESCR = SENSOR2
ATTRIB = AS

HAL/S Source:

ADDSSENSOR: PROCEDURE(SENS1) ASSIGN(SENS2) EXCLUSIVE;

Scanner Output Recd #12

RECDTYPE = ST
LINE # = 000008
RECD # = 00000012
BLKNAME = HALSPGM
BLK LVL = 02
STMT LVL = 00
SRN =
STMT = PROCEDURE
DESCR = ADDSENSOR
ATTRIB = EC

Scanner Output Recd #13

RECD TYPE = VR
LINE # = 000008
RECD # = 00000013
BLKNAME = ADDSENSOR
BLK LVL = 02
STMT LVL = 00
SRN =
STMT = PROCEDURE
DESCR = SENS1
ATTRIB = IP

Scanner Output Recd #14

RECD TYPE = VR
LINE # = 000008
RECD # = 00000014
BLKNAME = ADDSENSOR
BLK LVL = 02
STMT LVL = 00
SRN =
STMT = PROCEDURE
DESCR = SENS2
ATTRIB = AS

HAL/S Source:

```
DECLARE INTEGER, SENS1, SENS2 DOUBLE;  
  
SENS2 = SENS2 + SENS1;           COL 73-80  
  
CLOSE ADDSENSOR;
```

Scanner Output Recd #18

```
RECD TYPE = ST  
LINE # = 000010  
RECD # = 00000018  
BLK NAME = ADDSENSOR  
BLK LVL = 02  
STMT LVL = 00  
SRN = COL 73-80  
STMT = ASSIGN  
DESCR =  
ATTRIB =
```

Scanner Output Recd #21

```
RECD TYPE = VR  
LINE # = 000010  
RECD # = 00000021  
BLK NAME = ADDSENSOR  
BLK LVL = 02  
STMT LVL = 00  
SRN = COL 73-80  
STMT = ASSIGN  
DESCR = SENS2
```

Scanner Output Recd #19

```
RECD TYPE = VR  
LINE # = 000010  
RECD # = 00000019  
BLKNAME = ADDSENSOR  
BLK LVL = 02  
STMT LVL = 00  
SRN = COL 73-80  
STMT = ASSIGN  
DESCR = SENS2  
ATTRIB = LH
```

Scanner Output Recd #22

```
RECD TYPE = OP  
LINE # = 000010  
RECD # = 00000022  
BLK NAME = ADDSENSOR  
BLK LVL = 02  
STMT LVL = 00  
SRN = COL 73-80  
STMT = ASSIGN  
DESCR = ATTRIB = RH  
ATTRIB =
```

HAL/S Source:

```
IF SENSOR2 = 0 THEN
  IF SENSOR1 = 0 THEN DO;
    DO FOR I = 1 TO 10;
      SENSOR3.SHUTOFF = SENSOR4$(I) . SENSOR5$(I);
    END;
  END;
```

Scanner Output Recd For 1st IF

```
RECD TYPE = ST
LINE # = 000011
RECD # = 00000032
BLKNAME = HALSPGM
BLK LVL = 01
STMT LVL = 01
SRN =
STMT = IF
DESCR =
ATTRIB =
```

Scanner Recd For 2nd IF

```
RECD TYPE = ST
LINE # = 000012
RECD # = 00000042
BLKNAME = HALSPGM
BLK LVL = 01
STMT LVL = 02
SRN =
STMT = IF
DESCR =
ATTRIB =
```

Scanner Recd For 1st DO

```
RECD TYPE = ST
LINE # = 000012
RECD # = 00000044
BLKNAME = HALSPGM
BLK LVL = 01
STMT LVL = 02
SRN =
STMT = DO
DESCR = DISCRETE
ATTRIB =
```

Scanner Recd For 2nd DO

RECD TYPE = ST
LINE # = 000013
RECD # = 00000048
BLKNAME = HALSPGM
BLK LVL = 01
STMT LVL = 03
SRN =
STMT = DO
DESCR = FOR RANGE
ATTRIB =

Scanner Recd For STRUCTURED Variable

RECD TYPE = VR
LINE # = 000014
RECD # = 00000062
BLKNAME = HALSPGM
BLK LVL = 01
STMT LVL = 03
SRN =
STMT = ASSIGN
DESCR = SENSOR.SHUTOFF
ATTRIB = LH

Scanner Recd For Vector Dot Product

RECD TYPE = OP
LINE # = 000014
RECD # = 00000063
BLKNAME = HALSPGM
BLK LVL = 01
STMT LVL = 03
SRN =
STMT = ASSIGN
DESCR = .
ATTRIB =

ORIGINAL PAGE IS
OF POOR QUALITY

Appendix C

COMPLEXITY MEASURES

"Software Structure Metrics Based on Information Flow"

-Sallie Henry and Dennis Kafura

IEEE Transaction on Software Engineering, Sept. 1981

1. Introduction.

a) Objectives.

- i) To provide a practical technique for measuring large-scale systems that can serve as a design aid.

b) Characteristics.

- i) The major elements in the information flow analysis can be directly determined at design time.
- ii) The analysis reveals more of the system connections than are revealed by other ordering relations such as "calls".
- iii) The analysis defines measurements for complexity, module coupling, level interactions, and stress points from the patterns of communications.

2. Information Flow Concepts.

a) Types of Information Flows.

- i) Global Flows - There is a global flow of information from module A to module B through a global data structure D if A deposits information into D and B retrieves information from D.
- ii) Direct Local Flows - There is a local flow of information from module A to module B if A calls B.
- iii) Indirect Local Flows - There is a local flow of information from module A to module B if:

ORIGINAL PAGE IS
OF POOR QUALITY

- B calls A and A returns a value to B which B utilizes,
or
- C calls both A and B passing an output value from A to
B.

b) Representations of Information Flows (Destination \leftarrow source1,
source2, . . . ,sourceN).

- i) X.n.I denotes the value of the nth parameter of proc X
at invocation.
- ii) X.n.O denotes the value of the nth parameter of proc X
at termination.
- iii) If X is a function, X.O denotes the value returned.
- iv) X.D denotes an access by proc X to the global data ob-
ject D.

c) Advantages.

- i) The syntax of a language would not affect the form of
relations.
- ii) No distinction is made between a flow of information es-
tablished by a passed parameter and one established by a sh-
ared global data structure.

3. Complexity Measures.

a) Procedures.

- i) Length * (Fan-in * Fan-out) ** 2
- ii) Features identified by the measure.
 - Lack of functionality.
 - Stress points in the system.
 - Inadequate refinement.

b) Module (All procedures that access a particular data struc-
ture).

- i) Complete sum of the complexities of the procedures with-
in the module.
- ii) Features identified by the measure.

ORIGINAL PAGE IS
OF POOR QUALITY

- Poorly designed data structures.
- Improper modularization.
- Poor module design.
- Poor functional decomposition.

c) Interfaces.

- i) (The number of procedures exporting information from module A + THE number of procedures importing information into module B) * The number of information paths.
- ii) Features identified by the measure.
 - Strength of the coupling between modules.
 - Measure of modifiability.

4. Correlation of Information Flow for Complexity Measurements.

a) Considerations concerning the correlation results:

- i) The Spearman's r test was used.
- ii) Program changes were used as an estimate for errors.
- iii) Eighty changes were considered.

TABLE 4

Complexity Measure Correlation to Changes

Measure	Correlation to Changes	Level of Significance
$(fan-in * fan-out)^{**2}$	0.98	0.028
$length * (fan-in * fan-out)^{**2}$	0.94	0.021
$(fan-in * fan-out)$	0.83	0.042
$length^{**2}$	0.60	0.078

Appendix D

HAL/S COMPLEXITY FACTORS

Following is the preliminary list of factors to measure the complexity of Shuttle Orbiter Primary Flight Software. This list was derived from the dissertation by Jean C. Zolnowski: "A System For Measuring Program Complexity".

According to her work, there are four categories of measurements that can be made to programs written in FORTRAN or COBOL. We have tried to adapt the specific measurements in each of the categories to the constructs found in the HAL/S language. Most of the measurements are in terms of "counting the numbers of ..." within a given program complex.

1. PROGRAM INTERACTIONS

a) Connection information

i) the number of:

- PROGRAM BLOCKS
- PROCEDURE BLOCKS
- FUNCTION BLOCKS
- TASK BLOCKS
- UPDATE BLOCKS
- COMPOOL BLOCKS

ii) the number of calls to each PROCEDURE and FUNCTION

iii) the nesting level of subprograms

b) Interface information

i) the number of parameters for subprograms classified by:

- data type (INTEGER, SCALAR ...)
- call type (INPUT, ASSIGN)

ORIGINAL PAGE IS
OF POOR QUALITY

- ii) the nesting level of arguments and subprogram parameters
- iii) the number of SCHEDULED real time processes
- c) Macro usage
 - i) the number of % macros
 - ii) the number of arguments for each % macro
 - iii) the number of macro calls
 - iv) the number of REPLACE macros (text replacement)

2. INSTRUCTION MIX CHARACTERISTICS

- a) the number of statements for each
 - i) PROGRAM
 - ii) PROCEDURE
 - iii) FUNCTION
 - iv) UPDATE
 - v) TASK
- b) the number of statements by type:
 - i) Conditional
 - ii) Assignment/Computation
 - iii) I/O
 - iv) Comment
- c) the number of FUNCTION references per statement
- d) the number of real time process statements
- e) the number of labels

3. DATA REFERENCE

- a) the number of variables by type: SCALAR, INTEGER, BIT, BOOLEAN, CHARACTER, VECTOR, MATRIX, STRUCTURE, EVENT
- b) for each variable

- c) the number of references
- d) the percent of program span (scope)
- e) the average number of statements between each reference
- f) the number of elements for STRUCTURED types
- g) the number of CONSTANTS
- h) the number of dimensioned variables (ARRAY)
- i) the number of INITIALIZED variables by mode of initialization (STATIC or AUTOMATIC)
- j) the number of variables with RANGE, scaling or precision conversion, NAME, RANGE, LOCK, ACCESS, EXCLUSIVE, FIXED, EXTERNAL, RIGID, DENSE, ALIGNED, attributes
- k) the number of elements in each COMPOOL
- l) the number of variables assigned through input
- m) the number of times shaping is performed on variables

4. STRUCTURE AND FLOW CHARACTERISTICS

- a) the number of conditionals by type (IF, DO)
- b) the number of loops by type (DO ... END, DO WHILE/UNTIL, DO FOR)
- c) the length of each DO loop
- d) the nesting level of DO loops
- e) the number of escapes from loops (REPEAT, EXIT, GO TO)
- f) the number of conditions, computations, function references per IF statement
- g) the nesting level of IF statements
- h) the number of jumps up and down (sequential flow altered)
- i) the number of unconditional branches

Appendix E

HAL/S SOURCE ANALYZER MODULE STRUCTURE

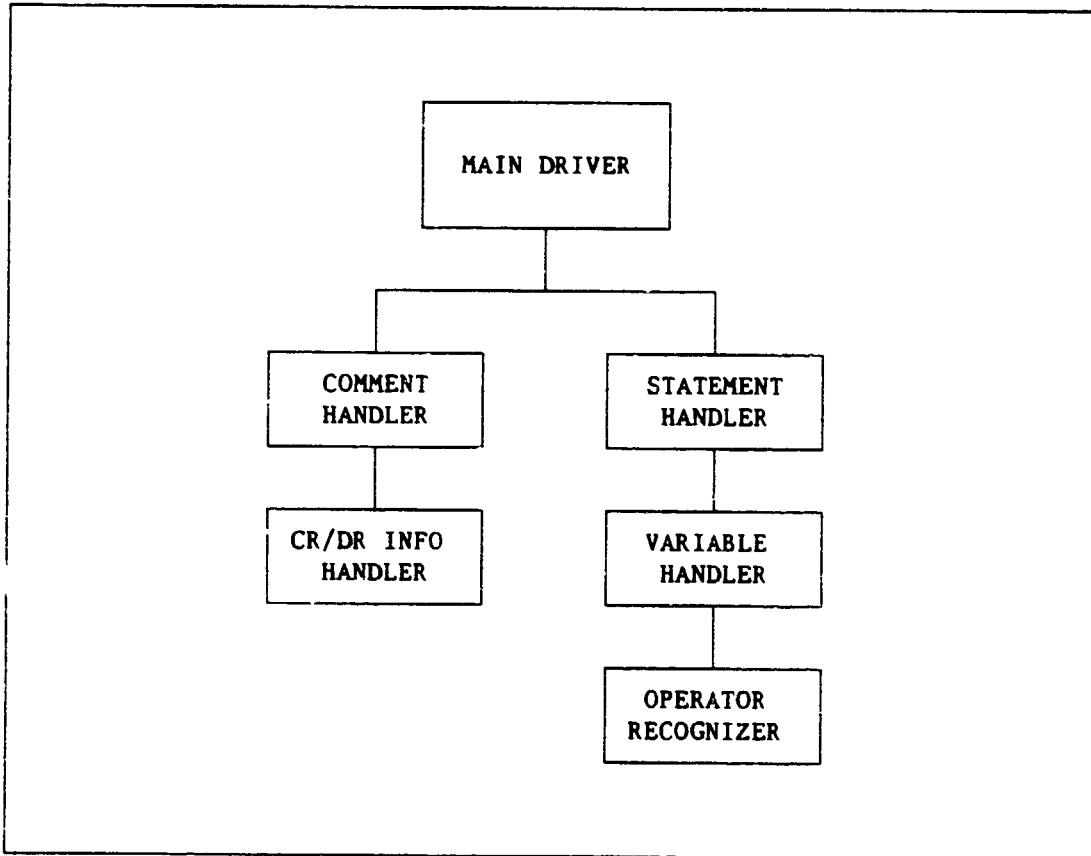


Figure 1: Structure of HAL/S Analyzer

Appendix F

MEMBER LIST OF FSW APPLICATION SOURCE

CDD032	CGE1H3	CGYMC1	CNSSSTOR	CSASMCMT	CS0760	CVAMMDIR
CDD092	CGE1H4	CGYMC3	CPADGT	CSBPBDCM	CS0780	CVBBTU
CDD142	CGE1M1	CGYSTA	CPADOW	CSCCMT	CS0790	CVCS9DL
CDF043	CGE1M2	CGZCOM	CPAMSP	CSCMMD	CS0860	CS2IX3
CDF053	CGE2H1	CGZFLD	CPASCA	CSCPAD	CS0870	CS2IX4
CDF063	CGE2M1	CGZFL2	CPASSI	CSCSCA	CS0880	CS2IX5
CDF073	CGE3H1	CGZFL3	CPCANI	CSCTEX	CS0890	CS2IX6
CDF083	CGE3H2	CGZMC2	CPCCLT	CSDHYB	CS0900	CS2IX7
CDF123	CGE3M1	CGZMC3	CPC CST	CSDINI	CS0910	CS2PAR
CDF133	CGGCOM	CGZMSLT	CPCDIT	CSDMDT	CS0920	CS2PAT
CDF153	CGGC01	CGZRMCS	CPCGXT	CSDRTCCM	CS0930	CS2PCT
CDHMMUT1	CGGC02	CGZ1DI	CPCPCI	CSIA06SC	CS0940	CS2PDT
CDIMMUT1	CGGC03	CGZ123	CPCTCI	CSIA11SC	CS0960	CS2PXT
CDVS9C	CGGC13	CGZ238	CPCVID	CSIA16SC	CS0970	CS2PX2
CDWDOWNL	CGGD01	CGZ28C	CPGGNC	CSICDA	CS2DART	CS2PX3
CGAGAX	CGGFLO	CG0180	CPKCOT	CSIPLC	CS2IFT	CS2SAT
CGAMC1	CGGFLT	CG0190	CPSSSC	CSISCMT	CS2INB	CS2000
CGA1MC	CGIGNC	CG0200	CPSSSU	CSI06CSM	CS2IPT	CS2010
CGA2MC	CGKKIP	CG0210	CP2GXT	CS111SCM	CS2IXP	CS2011
CGA3MC	CGKMC2	CG0220	CRATE	CS116SCM	CS2IX2	CS2020
CGBIH1	CGK136	CG0230	CRBMC1	CSMAMCMT	CS2IX3	CS2021
CGBIH2	CGMCOM	CGNCOM	CRCCOT	CSPCLB	CS2IX4	CS2030
CGBIM1	CGMIPC	CG0250	CRDCIL	CSPCON	CS2IX5	CS2040
CGBIM3	CGMIPF	CG0330	CREMCG	CSPTEC	CS2IX6	CS2050
CGBOBF	CGMIPM	CG0340	CRFASC	CSSCOT	CS2IX7	CVAMMDIR
CGCCOM	CGMMC2	CG0400	CRILVC	CSSDDT	CS2PAR	CVBBTU
CGCFL1	CGMMC9	CG0410	CRTDIS	CSSSPA	CS2PAT	CVCS9DL
CGCFL2	CGMMRW	CG0420	CSABCT	CSSSPCMT	CS2PCT	CVDDDCOC
CGCFL3	CGNCON	CG0430	CSADART	CSSSPINB	CS2PDT	CVDDFLCO
CGCUN1	CGNFLT	CG0440	CSAERT	CSSSPOB	CS2PXT	CVECOMBU
CGEBFS	CGNFL1	CG0500	CSAIFT	CSSTMCOT	CS2PX2	CVFTRTCO
CGEDIS	CGNFL2	CG0510	CSAINB	CS106SCM	CS2PX3	CVGGTSCO
CGEFLT	CGNMC1	CG0520	CSAIPT	CSTCAT	CS2SAT	CVHHDA
CGEGNC	CGNMC2	CG0530	CSAIXP	CSTTMC	CS2000	CVHPLD
CGEIPA	CGNMC3	CG1011	CSAPAR	CSZICCCM	CS2010	CVIMMUTI
CGEIPB	CGRRMC	CG2011	CSAPAT	CS0600	CS2011	CVJFDECP
CGEIPC	CGRRMO	CG3011	CSAPCT	CS0610	CS2020	CVKSACSC
CGEIPD	CGSAS1	CG3041	CSAPDT	CS0620	CS2021	CVLONESH
CGEIPF	CGSGNC	CG3051	CSAPFT	CS0660	CS2030	CVLTFLCO
CGE1H1	CGYCAT	CG8011	CSAPXT	CS0670	CS2040	CVBMFDCO
CGE1H2	CGYFL1	CG9011	CSASAT	CS0680	CS2050	CVHLOAD

CVMS8COM	GAGENT	GDRENT	GGOTGT	GKEKIP	GMYRGM	GPEELV
CVNMMUTI	GAHRTL	GDSSWP	GGQCOM	GKFHOR	GMZPLA	GPFORB
CVPAPTCO	GAIORB	GDTHUD	GGRENT	GKGMNV	GM1COM	GPGYAW
CVQMMUTI	GCAAER	GDWASC	GGSENT	GKIORB	GM2MAJ	GPHSBT
CVRDPERR	GCBAER	GDXAMI	GGTTAE	GKKORB	GM3PLA	GPILAN
CVSSSTCO	GCCAER	GDYASC	GGUUEP	GKMRMS	GM7SUB	GPKOMS
CVTTCSUN	GCD AER	GDZAMI	GGVPOS	GKNRCS	GM9MAS	GPLOMS
CVUSRBCO	GCEAER	GEAASC	GGWVCO	CKORMO	GNAMLS	GPN160
CVVLDBCO	GCFAER	GEBADH	GGXPGI	GKQORB	GNBTAC	GPORUD
CVWLDBPO	GCGSSM	GECRDH	GGYHAC	GKRORB	GNCBAR	GPPSRB
CVWMMULD	GCHGRT	GEEADM	GGZPSP	GKSSTA	GNDDRA	GPQSPT
CVXARFCO	GCIGRT	GEFRDM	GG1ASC	GKTUNI	GNEENT	GPRMPS
CVYADTDA	GCJASC	GEGASC	GG2ASC	GKUIMU	GNFEAR	GPSASC
CVMBFDCO	GCKGCS	GEHRTL	GG31ST	GKVREL	GNGEFT	GPTSSM
CVZMCDS	GAIORB	GEIORB	GG42ND	GKWRMS	GN1MEA	GPUMPS
CV1LSRES	GCAAER	GEJORB	GG5ORB	GKXRMC	GNJEFT	GPVSRB
CV1000	GCBAER	GEKORB	GG6ABR	GKYHFE	GNKGEO	GPW3AX
CV1010	GCCAER	GELORB	GG7PWR	GK2LND	GNLEFT	GPXSRB
CV1020	GCD AER	GEMEDH	GG9GLI	GK3ORB	GNMEFT	GPYMEC
CV1040	GCEAER	GENEDM	GHAD E0	GK4TIM	GNNANG	GPZORB
CV1050	GCFAER	GEPENT	GHBCMD	GK5MNV	GNOATT	GP0THC
CV1060	GCGSSM	GEQENT	GHCORB	GLDONO	GNPTAC	GP1ORB
CV1100	GCHGRT	GERENT	GHDUPM	GLEPIN	GNQNAV	GP2ORB
CV1120	GCIGRT	GESODH	GHEUPG	GLFFAN	GNSRVI	GP3CHE
CV2LIN	GCJASC	GETODM	GHFNRM	GLOOPS	GNTSUP	GP4CHE
CV2SIN	GCKGCS	GFATRA	GHGREF	GLUACC	GNUACC	GP5CHE
CV3GMEMW	GCLOPS	GFBRCS	GHHPHA	GL1ORB	GNVATT	GP6MPS
CV3LOT	GCMGCS	GFCREC	GHIPHA	GL2AUT	GNWATT	GP7HYD
CV3SOT	GCOASC	GFE1HT	GHRJTL	GL5NAV	GNXATT	GP8HYD
CV8011	GCPGCS	GFFORB	GHKRTL	GMAMIN	GNYATT	GP9MPS
CV9LSRES	GCQORB	GFHGRT	GHLRTL	GMBIMU	GNZATT	GQARNG
CV9011	GCRORB	GFIGRT	GHMRTL	GMCACP	GNIMEA	GQBBRG
DCDDG1	GCTFCS	GFJGRT	GHOORB	GMDRES	GN1DAT	GQCCOM
DCDDG2	GCUGCS	GFKGRT	GHPORB	GMESTA	GN2ENT	GQDHYD
DCDDG3	GCVDBA	GFL12H	GHQPEG	GMFGYO	GN3ENT	GRASWI
DCDDG8	GCWQUA	GFMJSE	GHR TAC	GMGMAJ	GN4RTL	GRBSWI
DCDDG9	GC0ORB	GFNGRT	GHS MNV	GMHACP	GN5NAV	GRC SBT
DCDDOW	GC4ORB	GFPAER	GHTDEO	GMJINE	GN6COV	GRDORB
DCDDS2	GC5ORB	GGAAUT	GHUDEO	GMKGYO	GN7ENT	GREAAF
DCDDS8	GC7ORB	GGBUPR	GHV VGO	GMLACP	GN8ASC	GRFAER
DCDDS9	GC8ORB	GGCAUT	GHWORB	GMMLAT	GN9ASC	GRGFIX
DGEGSEFR	GC9ORB	GGDAUT	GHXORB	GMNLSF	GOAGNC	GRHIMU
DGILDBIO	GDAASC	GGEENT	GHZUPP	GMOTER	GOBSMP	GRJIMU
DGOGSECO	GDEENT	GGFAUT	GH1DEO	GMPTNB	GO1ASC	GRKIMU
DGRGSERO	GDFORB	GGGAUT	GH4RTL	GMRTRA	GO2ORB	GRLIMU
DMPMMMSG	GDGDEO	GGILTV	GH6ABR	GMSIMU	GO3ENT	GRMIMU
GAALIM	GDJAMI	GGJRTL	GH7UPP	GMPFL	GO6RTL	GRNOMS
GABDIR	GDMSPI	GKGASC	GH8EAR	GMUHAN	GO8ORB	GRORCS
GADHFE	GDNHSI	GGLACC	GKBRMSW2	GMVHAN	GPABFS	GRPBFC
GAEASC	GDORTL	GGMGLI	GKCASC	GMWHAN	GPCAER	GRQRFC
GAFORB	GDPSWP	GGNSSM	GKDASC	GMXGCA	GPDAER	GRRRCS

GRSIMU	GULHOR	GWORMR	GZPBUR	RVMCON	VASSRB	VN4041KI
GRTFIX	GUMRMC	GWQORB	GZVVER	RWPPHC	VATMMDRE	VN6RCSEX
GRUFLO	GUNRMS	GWRORB	GZX6XT	RXYCIN	VAXMCIUI	VN7LIMIT
GRXADT	GUPENT	GWSORB	GZYENT	RYECNV	VAYBUFAV	VN8LIMIT
GRYTAC	GUQCON	GWVORB	GZZASC	SACPMU	VAZTCSCS	VN9MLSCK
GRZMLS	GURRMS	GWWORB	IMUCAL	SAFACQ	VBATESTT	VCM001
GR1RM1	GUSCSC	GWXORB	PCINIT	SAMITEM	VBBIOERR	VRJDTOGG
GR2ORB	GVAMLS	GWYORB	PCKITE	SBCSM	VBCMISCO	VRMRMPAR
GR4ORB	GVBTAC	GWZRCS	PCLSER	SBDITEM	VBDMDMSE	VRPRAMPC
GR5ORB	GVCBAR	GXJORB	PCMDIS	SBISM	VBEMODUL	VR1RMPCY
GR6ORB	GVDDRA	GX4DIS	PCPPSP	SBSBACKS	VBERROR	VSLMLSPR
GR7SRB	GVFRVT	GYAADT	PCRPU	SCICLN	VBQBITEA	VSPPARAM
GR8RCS	GVGVUV	GYBADT	PCWSOF	SCKPNT	VBTBUCO	VSRRCSPR
GR9RCS	GVIQUA	GYCADT	PC1SPE	SCMSUB	VB1LEVEL	VS1SPECG
GSAXFD	GVJVEC	GYDNAV	PC2SPE	SCSSPEC	VB2LEVEL	VS2SPECG
GSDFIR	GVKCOV	GYEADT	PDLIUS	SDTSM	VB9BTUSP	VS5SSTPR
GSESRB	GVLHEL	GYFADT	PDSSEQ	SLSSPEC	VCACOLLA	VS6SSTPR
GSFABT	GVMVRE	GYGADT	PL20PS	SMCKPT	VCMCSLMT	VTCTCSCO
GSGORB	GVNCOV	GYHADT	PMCCYC	SM20PS	VCUCSLCL	VTLTFLMO
GSHHYD	GVOOPS	GYIADT	PMGGNC	SPEPSP	VCYCYCUP	VULMMINT
GSIABT	GVPRUN	GYKLMS	PMOSTA	SPNINT	VDELEEDI	VUTMATIN
GSLORB	GVQSIT	GYKMLS	PMQTEC	SPPRECO	VDFDFLMO	VUUMATCY
GSMMP	GVRTAC	GYLMLS	PU1MUP	SPRPRB	VDRDDURA	VWFWFG
GSPMP	GVSQUA	GYMMLS	PU2MUP	RNCENC	VEHDPERR	VXCDCSWI
GSQASC	GVTPRE	GYNRRP	PU3MUP	SPSPSP	VFRTINPT	VXDADTST
GSRRSL	GVUUNI	GYRRAD	PU4MUP	SRESTO	VGTGTS	VXEERROR
GSSSSM	GVWQUA	GYTTAC	PU5MUP	SSAAPUFU	VG90PS9	VXFFEDDB
GSTETS	GVXQUA	GYUTAC	PU6MUP	SSBPLBAY	VHAHDA	VXFFLAGR
GSUQUA	GVYQUA	GYVAUT	PU7MUP	SSCFUELC	VJ1LSRES	VXINPUT
GRQRHC	GVZMAT	GYXADT	PU8MUP	SSFFUELC	VJ9LSRES	VXMMIDVA
GR1RMI	GV2DEL	GYADT	RASAUT	SSHHD	VKISACSI	VXOOMSDI
GR5OMS	GV3RTL	GYZSTS	RBMHWD	SSMANTMG	VKRRESPX	VXPAPT
GSVGAG	GV5STA	GY1STS	RCDDCO	SSNO2N2Q	VK3CMEMW	VXRKIP
GSWORB	GV6STA	GY3STT	RDDDDI	SSOSPDAT	VLBONESH	V01TCSSC
GSYDSC	GV7RTL	GY4STS	REXRMS	SSPEXEC	VMBBDFLA	V02TCSSA
GS5ASC	GWAORB	GY5FOV	RFPPOS	SSRREC	VMDBFDCA	V080PS8
GTAUPL	GWBORB	GY6COA	RHMHLT	SSSSTAND	VMELOAD	V090PS9
GTBUPL	GWCUNI	GY7STC	RITTEM	SSTHYDFL	VMMMMSPE	XG0180
GTCUPL	GWDRMO	GY8DAT	RJSHAN	STCCYCI	"MPITEMP	XG0210
GTDUPL	GWERM	GZBORB	RKGKIN	STMTAB	VM1BFDCY	XG0220
GUARMO	GWFREL	GZCRNG	RMCSP	STSSPEC	VM2BDYFL	XG0500
GUBRMS	GWGORB	GZDDIP	RMCENC	SULUPLIN	VM3BFDLI	XG0510
GUCIMU	GWHORB	GZGENT	RPORPO	S2ICLNUP	VM4BFSHU	XG0520
GUDORB	GWIORB	GZHDIP	RQCIPC	VAASEQUE	VNACLEAN	XG1011
GUEREL	GWJORB	GZIASC	RRPRRA	VAGTIMEO	VNECYCLI	XG3011
GUFORB	GWKORB	GZJSTA	RSCSIN	VAHCLEAN	VNFCYCLI	XG3041
GUHRMR	GWLRMCSF	GZLCON	RTVTOT	VAIMTSDR	VN1SST1C	XG3051
GUISTA	GWMORB	GZIASC	RUDKYB	VAMMDIR	VN2SST2C	
GUJORB	GWNMNV	GZNENT	RVICLN	VAPPRTYC	VN3040KI	

Appendix G

MEMBER LIST OF INCLUDE SOURCE

AIGDEULO	DAG#RAM	DCD13225	VB17DECL	VS4X4	XREFSSW	DFG0860
AMT0001	DAMTEMP	DCD142HS	VB2LMCIU	VS5X5	XREFVSM	DFG0870
ANNUNTBL	DAPPFMPT	DCD14201	VB21MDMS	VS6X6	ZMMAREAS	DFG0880
ANTMACS	DAPPFSD	DCD14205	VB22BSRV	VS7X7	ZPRIOTIM	DFG0890
A01XTAB	DBG#RAM	DCD14212	VB23SCUM	VS8X8	DCD15305	DFG0900
A01YTAB	DCDMMBEG	DCD14225	VB24PROM	VTLMMDIR	DCD15312	DFG0910
A01YTMSC	DCDMMDCL	DCD144HS	VB25SCUP	VUCMDWDS	DCD15325	DFG0920
A02XTAB	DCDMMDMP	DCD14401	VB26ADCV	VXACOMPA	DCD160HS	DFG0930
A02YTAB	DCDMMEND	DCD14405	VB27BTE4	V01X	DCD16001	DFG0940
A02YTMSC	DCDMNBEG	DCD14412	VB31DFI	V01YTFI	DCD16005	DFG0960
A03XTAB	DCDMNDMP	DCD14425	VB32PDI	V02X	DCD16012	DFG0970
A03YTAB	DCDMNEND	DCD146HS	VB33NRAM	V03XDFL	DCD16025	DFG2SMP2
A03YTMSC	DCD120HS	DCD14601	VB36NMOD	V03YDFL	DCG#RAM	DFG2000
A04XTAB	DCD12001	DCD14605	VCOAMT08	V04XDFL	DCG#RAM	DFG2010
A04YTAB	DCD12005	DCD14612	VCOAMT12	V1AXFCS	DDUMACS	DFG2011
A04YTMSC	DCD12012	DCD14625	VCOAMT13	V1AYFCS	DD0001D	DFG2020
A05XTAB	DCD12025	DCD148HS	VCOMDT	V1BXFCS	DD0010D	DFG2021
BDPMMACS	DCD121HS	DCD14801	VCO0400	V1CXFCS	DD0020D	DFG2030
BMODMACS	DCD12101	DCD14805	VCO0410	V1DXFCS	DD0060D	DFG0240
BRMACS	DCD12105	DCD14812	VCO1000	V1EXFCS	DD0990D	DFG0250
CAMEMACS	DCD12112	DCD14825	VCO1010	V1FXFCS	DEG#RAM	DFG3041
CAMIMACS	DCD12125	DCD152HS	VCO1020	V1GXFCS	DEUMACS	DFG4FSP
CDVCOMPO	DCD122HS	DCD15201	VCO1050	V1HXFCS	DFGMDAS	DFG40PS0
CGBIM2	DCD12201	DCD15205	VCO1060	V1IXFCS	DFGMDAT	DFG4SMP4
CMRMACS	DCD12205	DCD15212	VCO1100	V1JXFCS	DFGMDATZ	DFG4SPC1
CMSMACS	DCD12212	DCD15225	VCO1120	V1KXFCS	DFGMDAX	DFG4SP60
CPAPID	DCD12225	DCD153HS	VC08011	V1LXFCS	DFGMDAZ	DFG4TIME
CPDMSP	DCD123HS	DCD15301	VC09011	V1MXFCS	DFGNULN	DFLMMDIR
CSACDA	DCD12301	TCSMACS	VN1X1	V1NXFCS	DFGPROT	DFMPTMSK
CSAPLC	DCD12305	TIOMACS	VN1Y1	V1OXFCS	DFGSPEC	DFT0020
CSCA06SC	DCD12312	TMPMACS	VN2X2	V1PXFCS	DFGTIME	DGIGOAHE
CSCA11SC	DCD12325	TMYTABS	VN3X3	V1QXFCS	DFG0600	DGIHEREC
CSCA16SC	DCD124HS	TRCMACS	VN4X4	V1RXFCS	DFG0610	DGIINTER
CSCOTSRC	DCD12401	VB01ERRC	VN5X5	V1SXFCS	DFG0620	DGIINTWD
CSCSCMT	DCD12405	VB02ITEM	VN6X6	V1TXFCS	DFG0660	DGISTATR
CSC06SCH	DCD12412	VB1LMCIU	VN7X7	V1UXFCS	DFG0670	DGISTATU
CSC11SCH	DCD12425	VB11MDMS	VN8X8	V1VXFCS	DFG0680	DGNCFMPT
CSC16SCH	DCD132HS	VB12PCMU	VS1X1	V1WXFCS	DFG0760	DGNCFSD
CSSCOTR	DCD13201	VB13DDUS	VS1Y1	V1XXFCS	DFG0770	DG0180D
CSSSXTR	DCD13205	VB15REPL	VS2X2	XREFDCI	DFG0780	DG0210D
CWWRMACS	DCD13212	VB16DECL	VS3X3	XREFGNC	DFG0790	DG0220D

DG0500D	GNOGLD	G6D602	MG0250M	MS0096M	PFFOMACS	S13XMDT
DG0510D	GNOGLE	G8AMTP07	MG0330M	MS0600M	PFMDMACS	S13YMDT
DG0520D	GNOGLF	G8D420	MG0340M	MS0610M	PF1MACS	S14XMDT
DG1011D	GNOGLU	G8D430	MG0400M	MS0620M	PF2MACS	S14YMDT
DG3011D	GNSBET	G8D440	MG0410M	MS0660M	PHODMACS	S15XMDT
DG3041D	GOEDAT	G8D801	MG0420M	MS0670M	PMUMACS	S15YMDT
DG3051D	GOESSP	G9OSMACS	MG0430M	MS0680M	PREMACS	S16XMDT
DIVOGNC	GOFANT	HNHM	MG0440M	MS0760M	PROTMACS	S16YMDT
DIVOSSW	GOGSCH	IMUMACS	MG0500M	MS0780M	PSP1MACS	S17XMDT
DIVOVSM	GOIMFL	IOGNC	MG0510M	MS0790M	PSP2MACS	S17YMDT
DMA#MACS	GOINPT	IOMACS	MG0520M	MS0860M	RDISMACS	S18XMDT
DPLFMPT	GOLITE	IOSSW	MG0530M	MS0870M	RLTHMACS	S18YMDT
DPLFSD	GOOATT	IOVSM	MG1011M	MS0880M	RSFSMACS	S19XMDT
DSSFMP	GOOPMR	IPING1	MG1040M	MS0890M	RSTDMACS	S19YMDT
DSSFSD	GOOSSM	IPING2	MG1060M	MS0900M	SDADCL	S20XMDT
DVUFMPT	GOPDAT	IPING3	MG1120M	MS0910M	SETDMACS	S20YMDT
DVUFSD	GOQCOD	IPING8	MG2011M	MS0920M	SLTHMACS	S21XMDT
D26XS2G1	GOQUAT	IPING9	MG2021M	MS0930M	SMFDA	S21YMDT
FAOTMACS	GOSCHO	IPINS2	MG3011M	MS0940M	SMSSMACS	S22XMDT
FCBTMACS	GOTMID	IPINS9	MG3041M	MS0960M	SMSTAT	S22YMDT
FFSMACS	GPMRCS	IPINV9	MG3051M	MS0970M	SRBMACS	S23XMDT
FLEXDATA	GRIATT	IPINX0	MG8011M	MS1000M	STBYMACS	S23YMDT
FLEXTBL	GRIMUL	IPX0	MMUMACS	MS1010M	STRPCT	S24XMDT
FPMMDIR	GSOVEN	LDBMACS	MP0710M	MS1020M	STRPDT	S24YMDT
FTBAMT11	GXRCLR	MDAX4TMS	MP0720M	MS1050M	STUB	S25XMDT
FTBD0710	GXOTER	MDAY4TMS	MP0721M	MS0160M	SUMMACS	S25YMDT
FTBD0721	GXQCLR	MDPMACS	MP0730M	MS1100M	SYNCMACS	S26XMDT
FTBD0731	G1AMTP04	MD0001M	MP0731M	MS1120M	S01XMDT	S26YMDT
FTBD0741	G1D101	MD0002M	MP0740M	MS2000M	S01YMDT	S27XMDT
FTBD0751	G1D510	MD0003M	MP0741M	MS2010M	S02XMDT	S28XMDT
FTBD0762	G1D520	MD0010M	MP0750M	MS2011M	S02YMDT	S29XMDT
FTBD7011	G2AMTP05	MD0011M	MP0751M	MS2020M	S03XMDT	S30XMDT
DFG2040	G2D200	MD0020M	MP0752M	MS2021M	S03YMDT	S31XMDT
DFG2050	G2D201	MD0030M	MP0760M	MS2030M	S04XMDT	S32XMDT
FXFLMACS	G2D20101	MD0031M	MP0762M	MS2040M	S04YMDT	S33XMDT
GD1HSI6	G2D202	MD0040M	MP5001M	MS2050M	S05XMDT	S34XMDT
GD2HSI3	G2D210	MD0041M	MP5002M	MS8011M	S05YMDT	S35XMDT
GD3HSISC	G2D220	MD0060M	MP7011M	MS9011M	S06XMDT	S36XMDT
GEDISP	G2D230	MD0990M	MP9011M	MS9999M	S06YMDT	S37XMDT
GE1HIB	G2D250	MD9999M	MS0043M	MTUCMACS	S07XMDT	S38XMDT
GE2HIA	G2D330	MECMACS	MS0044M	MTUMACS	S07YMDT	S39XMDT
GE3HII	G2D340	HETMACS	MS0045M	NHCSECTS	S08XMDT	S40XMDT
GE4MIB	G3AMTP06	MG0042M	MS0061M	NHMANXRF	S08YMDT	S41XMDT
GE5MOB	G3D018	MG0092M	MS0062M	OPSCMACS	S09XMDT	S42XMDT
GE6:IA	G3D053	MG0093M	MS0070M	OPSIMACS	S09YMDT	S43XMDT
GE7MOA	G3D190	MG0180M	MS0081M	OPSTMACS	S10XMDT	S44XMDT
GE8MOB	G3D301	MG0190M	MS0082M	OPSUPMAC	S10YMDT	S45XMDT
GFORCS	G3D304	MG0200M	MS0083M	CVLYMACS	S11XMDT	S46XMDT
GKPMNV	G3D305	MG0210M	MS0084M	PDIMACS	S11YMDT	S47XMDT
GM5TOR	G3D500	MG0220M	MS0094M	PDSTEMPL	S12XMDT	S48XMDT
GNC91040	G6D601	MG0230M	MS0095M	PFBTMACS	S12YMDT	S49XMDT

S50XMDT S55XMDT S60XMDT S65XMDT S70XMDT S75XMDT
S51XMDT S56XMDT S61XMDT S66XMDT S71XMDT S76XMDT
S52XMDT S57XMDT S62XMDT S67XMDT S72XMDT S77XMDT
S53XMDT S58XMDT S63XMDT S68XMDT S73XMDT S78XMDT
S54XMDT S59XMDT S64XMDT S69XMDT S74XMDT

Appendix H

MEMBER LIST OF SYSTEM SOURCE

#CFIUCGR	CDG064	CDR04D	DMNEW	FCMSFAIL	FIOGNIPG	FIONSPPG
#DFIONDX	CDG074	CDR05D	DMRRESUM	FCMSFCAM	FIOG9OPG	FIOPBYG9
#ZFIUCGR	CDG084	CDR06D	DMTERR	FCMSFINT	FIOHFEPG	FIOPBYS2
AIBGPCLO	CDG114	CDR07D	DMZLOG	FCMSMASK	FIOHFE02	FIOPBYTB
AIESIP	CDG124	CDR08D	DMKEY	FCMSSLPT	FIOHFE16	FIOPBY02
AIGDEUP	CDG134	CDR11D	DM2APP	FCMSSYNC	FIOHFE38	FIOPBY16
AMTP02	CDG154	CDR12D	DM3DIS	FCMSVC	FIOHFE89	FIOPBY38
AMTP04	CDH025	CDR13D	DXXCSE	FCMSVOTE	FIOHISAM	FIOPBY89
AMTP05	CDH045	CDR15D	FCMBCEMD	FCMSWMON	FIOICCPG	FIOPDG9
AMTP06	CDH055	CDSDIC	FCMBMTPG	FCMTBLG9	FIOIMUPG	FIOPDHF
AMTP07	CDH065	CDTANNUN	FCMBM2S2	FCMTBLPG	FIOIDbPG	FIOPDIPG
AMTP08	CH0754	CDU#LNK	FCMBM202	FCMTBLS2	FIOLGERR	FIOPDISP
AMTP11	CDH085	CDXDAT	FCMBM16	FCMTBL02	FIOICIPG	FIOPDS&U
AMTP12	CDH115	CDYDIC	FCMBM38	FCMTBL16	FIOICNTL	FIOPDSM2
AMTP13	CDH125	CDZANNUN	FCMBM89	FCMTBL38	FIOIDMPG	FIOPDSPG
AMTP15	CDH135	CD0001	FCMBOOT	FCMTBL39	FIOIDPPG	FIOPDSRB
ARAGPCSW	CDH155	CD0002	FCMBUSCM	FCMTRACE	FIOIDPPG	FIOPDVU
ARC#GPC	CDJRWD	CD0003	FCMBUSPC	FCMTRCLG	FIOIFEG9	FIOPMUPG
ARDCSBUS	CDKDIC	CD0010	FCMCBLKS	FCMTSYNC	FIOIFEPG	FIOPRMPG
ARDFBBS	CDLANNUN	CD0020	FCMCKSUM	FCMUPLOD	FIOIFEO2	FIOPSPPG
ARDRSBUS	CDM#COM	CD0060	FCMCOM	FCMZCONS	FIOIFE16	FIOPURGE
ARFDPSCO	CDND03	CD07XX	FCMCSYNC	FDDUMMY	FIOIFE38	FIORTBUF
ARG#REC	CDND04	CD0990	FCMDSCRM	FIOACTMD	FIOIGCMP	FIOSMDPG
ARK#DOW	CDND05	CD1#DFB1	FCNFD1	FIOACT02	FIOIGCV	FIOSMFFG
ASCTIMEM	CDND06	CZ1#COM	FCMINBCE	FIOADCCL	FIOIGDSP	FIOSRBPB
ASGCYCLI	CDND07	CZ2COMMO	FCMINIOP	FIOADCNS	FIOIGERR	FIOSTMSC
ASHRWCYC	CDND08	CZ3#COM	FCMINMSC	FIOCBLKS	FIOIGMTR	FIOSVC
ASLTM	CDND09	DCI#CYC	FCMINSSL	FIOC DATG	FIOIGSNC	FIOSVCP
ASMAUX	CDND10	DCI#DATA	FCMISYNC	FIOC DATS	FIOIGSTR	FPMCANCL
ASNGMEMW	CDND11	DCI#PDE	FCMLINIT	FIOCMLPT	FIOIGTQE	FPMCHPCT
CAASCCOM	CDND12	DCI#STK	FCMNGBOV	FIOCOUNT	FIOIMGTG	FPMCHTQE
CABCOMPO	CDND13	DFLDCU	FCMNGPOV	FIOCTR	FIOIMMSC	FPA_LOSE
CB021	CDND14	DGMWRT	FCMNGPT	FIOCYCSM	FIOIMUPG	FPMC VTFL
CDFIT	CDND15	DGNLIGHT	FCMNINIT	FIOCYCTB	FIOIMODSM	FPMC VTFX
CD022	CDH075	DG9LIGHT	FCMPAT26	FIOCYC02	FIOIMDTB	FPMDCU
CDD102	CDN102	DIM#ICC	FCMPMOD	FIODDUPG	FIOIMOD02	FPMDISP
CDF023	CDN202	DIRIOR	FCMPROTD	FIODEUPG	FIOIMS2DT	FPMDSABL
CL7110	CDN302	DIS#PLAY	FCMPSA	FIOERRLA	FIOIMS2PG	FPMENABL
CDG024	CDPDIC	DLALIGHT	FCMRSTFC	FIOERRLB	FIOIMWPG	FPMERLOG
CDG044	CDQANNUN	DMAMAC	FCMRTBLE	FIOERRLC	FIOIMVUDT	FPM EVAL
CDG054	CDR02D	DMCSUPR	FCMSAVE	FIOGNCDL	FIOIMVUPG	FPM EVDFQ

FPMVENQ	FPMIHPC2	FPMRESET	FPM SWITCH	FPMZSYNC	PCH04SRC	PCH14SRC
FPMFCLOS	FPMIHPCM	FPMRIPCT	FPMTERM	IDLE#OPS	PCH05SRC	PCH15SRC
FPMFRPCT	FPMINHSP	FPM SCHED	FPM TMCVT	LOADTABL	PCH06SRC	PCH26SRC
FPMFXMTU	FPMITUPD	FPM SDERR	FPM TMDQ	MCDS#IN	PCH07SRC	TFLCDU
FPMGMTIM	FPM TUF	FPM SET	FPM TMEQ	MCDS#PRC	PCH08SRC	XD0001
FPMIDLE	FPM TUM	FPM SIGNAL	FPM TMMAL	MDT XT BLE	PCH09SRC	XD0010
FPMIDTIM	FPMOPSCN	FPM SIO	FPMUPMTU	PATCH	PCH10SRC	XD0020
FPMIHIM	FPMREL	FPM SVC	FPMUPTOX	PCH02SRC	PCH12SRC	XD0060
FPMIHPC1	FPMRES	FPM SVCEP	FPM WAIT	PCH03SRC	PCH13SRC	XD0990

Appendix I

DR VARIABLE LIST

<u>Variable</u>	<u>Description</u>
FILE_ID	File ID, always "F"
DR_NUM	DR Number
MISSN_ID	Suffix (Mission Identifier)
FAC	Reporting Facility
_OD	Date Logged In (MM/DD/Y)
_ODYRMO	Year-Month Logged In (YMM)
DV	Discriminant Version
ITR	IBM Transmittal Number
NTR	NASA Transmittal Number
GA	Verification Assignment for "Special"
REF	Reference CR/DR, Other
P	Priority of DR
VER	Target Closure Code (Replaced by actual Build ID for 2A,2B,2C,2D,2E,2F)
CP	Actual Closure Code (Meaning different for SDL DRs)
PB	Pre-Build Assessment Data
G0	Verification Assignment for Department HE3
PR	Pre-Build Assessment Reason
X	T&O Closure Code
BD	Development Close Date (MM/DD/Y)
BDYRMO	Year-Month of Development Close Date
S	Verification Status Data
G1	Verification Assignment for Department HF6
G2	Verification Assignment for Department HC9
G3	Verification Assignment for Department HG5
G4	Verification Assignment for Department HB7
G5	Verification Assignment for Department HF5
G6	Verification Assignment for Department HD4
G7	Verification Assignment for Department HF7
G8	Verification Assignment for Department HC8
F	Change/New Identifier
PD	SDL Build Date
G9	Verification Assignment for Department HD7
D	Title/Description
TN	IBM Transmittal Date (MM/DD/Y)
TNYRMO	Year-Month of IBM Transmittal Date (YMM)
FN	NASA Transmittal Received Date (MM/DD/Y)
FNYRMO	Year-Month of NASA Transmittal Received Date (YMM).

PF	Principle Function ID
TC	Verification Test Code ID
TD	Development Build Target ID
ST	Status of the DR
IMP	Future System Closure Code
BL	Verification Baseline ID
PN	Program Name (SDL Only)
VC	Verification Closure Date
AD	Action Department

Appendix J

DR LEGEND

FILE_ID -- File ID
Always "F"

DR_NUM -- DR Number; a unique number for tracking purposes.

MISSN_ID -- Suffix (Mission Identifier); indicates applicable FSW systems.

None=SDL

A=Site/Test Unique Patch DRs

B=PMD3

C=ECL

D=SSW1

E=JSC Form 1541 (Rev Jun 75)

F=Release 19 (Floor System STS-5)

G=PMD2

H=AOA

J=Invalid Entry; there should be no entries of "J".

K=KSC

L=STS-1 FUP5/FLT (Flight System)

M=STS-1 FUP4 (07/30 System for Field Users)

N=Entry

P=PMD1

Q=Invalid Entry; there should be no entries of "Q".

R=STS-2 (R18V21 Flight System)

R1=STS-2 (R18V11 Field System)

R2=STS-2 (R18V12 Field System)

R3=STS-3 (R18V30 Flight System)

R4=STS-4 (R18V40 Flight System)

R5=Invalid Entry; there should be no entries of "R5".

R9=OV-99 (R18V90 Field System)

S=SSW2

S1=SPF Release 1

S2=SPF Release 2

S6=SDL Release 36

S7=SDL Release 37

T=OFT1

U=Entry Update 1

V=Invalid Entry; there should be no entries of "V".

W=STS-2 (R17V5)

X=STS-1 FUP1 (12/11 System for Field Users)

Y=STS-1 FUP2 (03/19 System for Field Users)

Z=STS-1 FUP3 (05/21 System for Field Users)

FAC -- Reporting Facility; OFT area initiating the DR.

*=

AAA=Avionics Architecture
AASD=Avionics Application Software Development
ASD=Avionics Software Development
ASSD=Avionics System Software Development
B30=
CI=Configuration Inspection
CM=Control Monitor
CSC=Computer Sciences Corporation
CSD=Charles Stark Draper Laboratory
CSDL=Charles Stark Draper Laboratory
CT=Communications and Tracking
C3=
DL=Downlist
DSL=
FACC=Ford Aerospace Communications Corporation
FCO=Flight Computer Operating System
FCOS=Flight Computer Operating System
FIT=Flight Computer Interface Tester
FNC=
FSL=Flight Systems Laboratory
FSW=Flight Software
GBS=Ground Based Shuttle
GNC=Guidance, Navigation & Control
GNCC=Guidance, Navigation & Control--Ex/Seq/Guid
GNCD=Guidance, Navigation & Control--Navy & Spec Proc
GNCE=Guidance, Navigation & Control--Fit Control Rm
GNCO=Guidance, Navigation & Control
HC2=IBM Department
HC8=IBM Department
HE1=IBM Department
HE8=IBM Department
HF9=IBM Department
HG8=IBM Department
HHB=IBM Department
HHS=IBM Department
HH4=IBM Department
HH5=IBM Department
HH6=IBM Department
HH7=IBM Department
HH8=IBM Department
HR7=IBM Department
I=
IBM=International Business Machines
IT=
I2=Intermetrics Inc. (HAL)
JAI=Jefferson Association Inc.
KSC=Kennedy Space Center
LEC=Lockheed
MDAC=McDonnell Douglas Aircraft Co.
MDC=McDonnell Douglas Co.

MDTS=
MOD=
MRI=MRI (Subcontractor)
NAS=National Aerospace System
NASA=National Aeronautics & Space Administration
OAS=Orbiter Avionics Software
OFT=Orbital Flight Test
OPE=IBM Operations
OPER=Operations
OPR=RTCC Operations
OWE=IBM (Owego, NY)
OWEG=IBM (Owego, NY)
PDAD=
PL=Payload
PO=Project Office
RA=Requirements Analysis
RI=Record ID
RMS=Remote Manipulator System
RTCC=Real Time Computer Complex
S L=
SAI=Shuttle Avionics Integration Laboratory
SAIL=Shuttle Avionics Integration Laboratory
SC=System Control
SDL=Software Development Laboratory
SDR=Software Design Requirements
SDRD=
SDRO=
SFC=Selection Filter Control
SFO=
SFS=Shuttle Flight Support
SFSG=Shuttle Flight Support--GNC
SFSS=Shuttle Flight Support--SSW, VU, SM, DL, RMS
SI=System Integration
SID=System Integration Department
SM/P=Systems Management/Payload Management
SMD=Systems Management Development
SMP=SM Offline
SMPL=SM Payload
SMS=Shuttle Mission Simulator
SPF1=Software Production Facility 1
SSD=Spacecraft Software Division
SSW=System Software
ST=State
SID=
T+O=Test & Operations
T+O=Test & Operations
T&O=Test & Operations
T/O=Test & Operations
TRW=TRW
TSO=
UI=User Interface
VCO=Vehicle Checkout

VER=IBM Verification Personnel
VU=Vehicle Utility
8.1=

_OD -- Date logged in (MM/DD/Y); date DR is entered into data base by Project Office.

_ODYRMO -- Year-Month logged in (YMM)

DV -- Discrepant Version; floor version on which the discrepancy was observed.

ITR -- IBM Transmittal Number

NTR -- NASA Transmittal Number

GA -- Verification Assignment for "special"; used to identify DRs requiring signoff by Tony Macina (4A,4B) or Rich Cucco (3F).

REF -- Reference CR/OR other; reference area for a related DR, PCR, PCA, or PTR.

P -- Priority of DR (for scheduling purposes)

- 1=Critical
- 2=Desirable
- 3=Proposed Disposition
- 4=Dispositioned for Future Build
- 5=Dispositioned for No Mod or No DR Closure
- 9=User note; these are not discrepancies, but this variable was used to track them as if they were.

VER -- Target Closure Code (Replaced by actual Build ID for 2A, 2B, 2C, 2D, 2E, 2F). Implemented version; Build release/version number of DR fix or other closure. (No Mod, Dup DR, Etc.)

AWP=Old SDL Closure Code
CODE RVW=Code Review
COSTLY
CR=Change Request Required
CR REQ=Change Request Required
CR REQD=Change Request Required
DOC=Documentation Error
DOC ERR=Documentation Error
DUP DR=Duplicate DR
DUPDR=Duplicate DR
DUPE=Duplicate DR
DUPE DR=Duplicate DR
DUPEDR=Duplicate DR
EOS=Enhanced Operation System
H/D=Hardware Error
H/W=Hardware Error

H/W ERR=Hardware Error
 HDW=Hardware Error
 HRDWR=Hardware Error
 HPWR=Hardware Error
 I/O ERK=Input/Output Error
 INS INFO=Insufficient Information
 INVALID
 JCL ERR=Job Control Language Error
 KP ERR=Old SDL Closure Code
 MOD=Old SDL Closure Code
 MODE=Old SDL Closure Code
 MODED=Old SDL Closure Code
 N/A=Not Applicable
 NO DATA
 NO DR
 NO H/W=Not Hardware
 NO HW=Not Hardware
 NO MOD=No Mod
 NO MODE=No Mod
 NO MODS=No Mod
 NO REQ
 NOMOD=No Mod
 NOT H/W=Not Hardware
 NOT HDW=Not Hardware
 NOT OPER=Not Operator
 OP ERR=Operator Error
 OP NOTE=Operator Note
 OPER=Operator Error
 OPER ERR=Operator Error
 OPERR=Operator Error
 PATCH
 PRCCHG=Old SDL Closure Code
 PROC=Processing
 REPAIR
 RL NOTE=Old SDL Closure Code
 S/W=Software
 SSW1=System Software 1
 SYS DL=Old SDL Closure Code
 SYS DLT=Old SDL Closure Code
 UNKNOWN
 US ERR=User Error
 US NOTE=User Note
 USEERR=User Error
 USER ER=User Error
 USER ERR=User Error
 USERR=User Error
 USR ER=User Error
 USR ERR=User Error

CP -- Actual Closure Code (meaning different for SDL DRs).

OFT CP - DR Categorization

1=Under Investigation

1A=Sufficient Data
 1B=Insufficient Data
 1C=Insufficient Data - Unable to Recreate
 1D=Insufficient Data/Too Costly to Recreate
 2=Fix Closure
 2A=Source Mod
 2B=Patch
 2C=Patch and Release Note
 2D=I-Load Patch
 2E=PSF I-Load Corrections
 2F=Build/GFE/MMU Mod
 3=No DR Closure
 3A=User Error or No Requirement or Not Applicable
 3A**=User Error; ** indicates NASA SSD concurrence
 3B=Duplicate DR
 3C=CR Required (New Requirement)
 3D=CR Required (Make Req. Match Code)
 3E=Hardware/Set Up/Support Software
 3F=User note
 3F**=User note; ** indicates NASA SSD concurrence
 3F*P=User Note for Non-Flight System
 3F-X-U=User Note (same as 3F, "-X-U" is an error)
 3G=User's Guide
 3J=GFE Error/NASA Generated Opnote Required
 3K=GFE Error/Source or Patch Req
 4=No Mod Closure
 4A=Waiver
 4B=Waiver and Op Note
 4B,X=W=Waiver and Op Note (Same as 4B, "-X=W-" is an error)
 4C=Program Standards or Philosophical Issue or Fix at Next
 Opportunity when Module Opened for other CR or DR
 Implementation.
 4D=No Fix/Release Note
 4DF=*=-No Fix/Release Note (same as 4D, "-*-" is an error)
 4E=Insufficient Data
 4F=Unexplained Anomaly
 4G=Requirement Intentions Met
 4H=Waiver and Users Guide
 4I=Function No Longer Used
 5=System No Longer Supported
 5A=System Deleted
 6=Unsupported Function
 6A=Development Action Deferred to Next System
 6B=Verification Action Deferred From Previous System
 SDL CP - Computer Program End Item (CPEI) Name
 ALL=
 APASM=
 APES=Advanced Processor Emulator System
 ASC=Ascent
 ASM=AP-101 Assembler
 AUTO=Automatic
 BLD=

CA=
CB=
CDC=
CF=
CF/UA=
COM=
COMM=
COMR=
COR=
CSC=Computer Sciences Corporation
CSECT
DASS=
DPS=Data Processing Subsystem
EOS=Enhanced Operating System
FC=Flight Computer
FDH=
FDSW=
FD4=
FEID=Flight Equipment Interface Development
FSW=Flight Software
FSW12=Flight Software Release 12
FSW13=Flight Software Release 13
FSW14=Flight Software Release 14
FSW15=Flight Software Release 15
FSW16=Flight Software Release 16
FSW17=Flight Software Release 17
GBS=
GEMINI
HAL=High-Order Aerospace Language
HAL/S=High-Order Aerospace Language/Shuttle
HALS=High-Order Aerospace Language/Shuttle
HDW=Hardware
HF2=IBM Department
HH5=IBM Department
HH8=IBM Department
IBM=International Business Machines
ICS=Interpretive Computer Simulator
IR=Inner Roll
I2=Intermetrics Inc. (HAL)
LE=AP-101 Linkage Editor
LEC=Lockheed
MMB=
MMP=
MMU=Mass Memory Unit
MO=
MOD=
MVS=Middle Value Select
NAS=National Aerospace System
NASA=National Aeronautics & Space Administration
OBS=On Board Software
OFT1S=Orbital Flight Test Integrated System
OFT15=

OS=
OSS=
OTH=
PMF=Program Management Facility
PP=Platform Positioning
REQ=Requirement
RF/IS=
SDL=Software Development Laboratory
SDL01
SDL5
SDL6
SDL7
SDL8
SDL9
SDL10
SDL11
SDL12
SDL13
SDL14
SDL15
SDL16
SFT=
SIM=Simulator
SIS=SAIL Interface System
SISO=Space Informations Systems Operations
SM2=
SODA=
SOLID=
SPMD=Software Performance Monitoring Device
TAPE
TEK=
TMS=
TSO=
UA=
VA=
101=HAL/AP-101 Compiler
11.7=
12=
12.4=
12.5=
12.7=
12.8=
12.9=
13=
35=
36.0=
360=HAL/360 Compiler

PB -- Pre-build Assessment Data; Code changes detected earlier than Build. Used to track DRs against pre-build process (design/code inspection and unit test).

***=Has not yet been assessed.

C=DR should have been found by code inspection.
D=DR should have been found by design inspection.
T=DR should have been found by unit test.
000=DR should not have been found before build.

GO -- Verification assignment for Dept. HE3

The first character is a status.

P=Primary
S=Secondary
C=Closed
J=Joint

The second character indicates a specific test analyst.

PR -- Pre-Build Assessment Reason; reason found test.

X -- T&O Closure Code; Test & Operations applicability to pass areas 1, 2, and 3. The three positions represent the three mass memory areas.

For non-code change dispositions:

Blank=Not Applicable to Area
N=No DR
W=Waiver
O=Open
P=Patch
R=Release Note
S=Source Mod
D=System Deleted
U=User note

For code fix:

A "P" in each position indicates that the DR was patched in that area of mass memory.

BD -- Development Close Date (MM/DD/Y); Release/Target date.

Target is projected date of fix (MM/DD)

Release is date of build or other type of closure (MM/DD/Y)

BDYRMO -- Year-Month of Development Close Date

W -- Verification Status Data

Used by Tony Macina & Verification for waiver correlation.

V -- Verification Status Data

Used by Tony Macina & Verification for waiver correlation.

G1 -- Verification Assignment for Dept. HF6

Dept. HF6 is a department within the Avionics Software Verification Dept., which reviews GN&C DRs.

The first character is a status:

P=Primary
S=Secondary
C=Closed
J=Joint

The second character indicates a specific test analyst.

G2 -- Verification Assignment for Dept. HC9

Dept. HC9 is a department within the Avionics Software Verification Dept., which reviews GN&C DRs.

The first character is a status:

P=Primary
S=Secondary
C=Closed
J=Joint

The second character indicates a specific test analyst.

G3 -- Verification Assignment for Dept. HG5

Dept. HG5 is a department within the Avionics Software Verification Dept., which reviews GN&C DRs.

The first character is a status:

P=Primary
S=Secondary
C=Closed
J=Joint

The second character indicates a specific test analyst.

G4 -- Verification Assignment for Dept. HB7

Dept. HB7 is a department within the Avionics Software Verification Dept., which reviews GN&C DRs.

The first character is a status:

P=Primary
S=Secondary
C=Closed
J=Joint

The second character indicates a specific test analyst.

G5 -- Verification Assignment for Dept. HF5

Dept. HF5 is a department within the Avionics Software Verification Dept., which reviews System Service DRs.

The first character is a status:

P=Primary
S=Secondary
C=Closed
J=Joint

The second character indicates a specific test analyst.

G6 -- Verification Assignment for Dept. HD4

The first character is a status:

P=Primary
S=Secondary
C=Closed
J=Joint

The second character indicates a specific test analyst.

G7 -- Verification Assignment for Dept. HF7

Dept. HF7 is a department within the Avionics Software

Verification Dept., which reviews Performance DRs.
The first character is a status:

P=Primary
S=Secondary
C=Closed
J=Joint

The second character indicates a specific test analyst.

G8 -- Verification Assignment for Dept. HC8
Dept. HC8 is a department within the Avionics Software
Verification Dept., which reviews Performance DRs.

The first character is a status:

P=Primary
S=Secondary
C=Closed
J=Joint

The second character indicates a specific test analyst.

F -- Change/New Identifier
*=indicates changes since previous run.
N=New

PD -- SDL Build Date

G9 -- Verification Assignment for Dept. HD7

The first character is a status:

P=Primary
S=Secondary
C=Closed
J=Joint

The second character indicates a specific test analyst.

D -- Title/Description; condensed description of problem.

TN -- IBM transmittal date (MM/DD/Y)

TNYRMO -- Year-Month of IBM transmittal date (YMM)

FN -- NASA transmittal received date (MM/DD/Y)

FNYRMO -- Year-Month of NASA transmittal received date (YMM)

PF -- Principle Function Id
Not Used

TC -- Verification Test Code ID

TD -- Development Build Target ID; Development target date (MM/DD)

ST -- Status of the DR.
C=Closed
CC=Closed by PWSW and Verification

CP=Closed with a patch
 D=Verification Action Deferred
 F=Fixed (SDL only)
 M=Composite Mover (SDL)
 N=Awaiting NASA/SSD approval
 O=Open
 OI=Planned development work will close the DR or applicability
 OFT is uncertain.
 ON=Open with an OP note written
 OP=Open patched
 T=In system test (SDL)
 V=Awaiting verification approval
 VO=Waiting Closure Description
 VX=Closed by Development; Awaiting review by Macina or Gucco.
 X=Dummy Status used on entries for page control.

IMP -- Future System Closure Code; Used by verification to indicate how the DR is being closed on the next system.

BL -- Verification Baseline ID; Used by verification for budgeting. When a DR has been included in a man-power estimate, this is set. The "blank" DRs indicate costs over the "baseline".

PN -- Program Name (SDL only)

VC -- Verification Closure Date

AD -- Action Department; OFT area responsible for fix.

AASD=Avionics Application Software Development

ASA=Avionics Software Arch/Sys Analysis

ASD=

ASSD=Avionics System Software Development

C71=

DASS=

DDI=Discrete Digital Inputs

DL=Downlist

D/L=Desorbit/Landing

EK=

EOS=Enhanced Operating System

FCOS=FLT Computer Operating System

FDH=

FSW=Flight Software

GNC=Guidance, Navigation and Control

GNC A=Guidance, Navigation and Control

GNC B=Guidance, Navigation and Control - Display

GNC C=Guidance, Navigation and Control - Ex/Seq/Guid

GNC D=Guidance, Navigation and Control - Nav & Spec Proc

GNC E=Guidance, Navigation and Control - Flt Cntrl/RM

GNC F=Guidance, Navigation and Control - Iloads

GNC G=Guidance, Navigation and Control

GNC O=Guidance, Navigation and Control

GNC I=Guidance, Navigation and Control

GNC2=Guidance, Navigation and Control
HDW=Hardware
HD7=IBM Department
HE9=IBM Department
HF1=IBM Department
HF2=IBM Department
HF7=IBM Department
HF9=IBM Department
HG7=IBM Department
HG8=IBM Department
HH4=IBM Department
HH5=IBM Department
HH6=IBM Department
HH7=IBM Department
HH8=IBM Department
HH9=IBM Department
HK9=IBM Department
HR2=IBM Department
H/W=Hardware (EOS)
I2=Intermetrics Inc. (HAL)
KSC=Kennedy Space Center
OPE=IBM Operations
OPER=Operations
OSS=
OTH=
OWEG=IBM (Owega, NY)
PRC=Procedures (EOS)
PROC=Procedures
PYLD=Payload
RA=Requirements Analysis
RMS=Remote Manipulator System
RQR=Requirements
SC=System Control
SDL=Software Development Laboratory
SFSG=Shuttle Flight Support - GNC
SFSS=Shuttle Flight Support - SSW, VU, SM, DL, RMS
SI=System Integration
SID=System Integration
SM=Systems Management
SMP=SM Offline
SMPL=SM Payload
SS=Systems Software
SSCT=Top Level Design Team
SSW=System Software
SW=Software
SYSA=Systems Analysis Group
S/W=Software (EOS)
Tape
T+O=Test & Operations
T&O=Test & Operations
UI=User Interface
VCO=Vehicle Checkout

VU=Vehicle Utility

Appendix K

CR VARIABLE LIST

<u>Variable</u>	<u>Description</u>
FILE_ID	File ID, always "P"
PCR_NUM	PCR Number
CR_NUM	CR Number
CRREV_ID	CR Revision ID
OASCB_EF	OASCB Effectivity
RRBDT	RRB Disposition Date (MM/DD/Y)
RRBYRMO	Year-Month of RRB Disposition (YMM)
CCBDDT	CCB Disposition Date (MM/DD/Y)
CCBYRMO	Year-Month of CCB Disposition (YMM)
OASCBDT	OASCB Disposition Date (MM/DD/Y)
OASCBYRM	Year-Month of OASCB Disposition (YMM)
RRBDIS	RRB Disposition
CCBDIS	CCB Disposition
OASCBDIS	OASCB Disposition
DEPT1_ID	Department 1 ID
DEPT2_ID	Department 2 ID
DEPT3_ID	Department 3 ID
DEPT4_ID	Department 4 ID
DEPT5_ID	Department 5 ID
DEPT6_ID	Department 6 ID
DEPT7_ID	Department 7 ID
DEPT1_VT	Department 1 Vote
DEPT2_VT	Department 2 Vote
DEPT3_VT	Department 3 Vote
DEPT4_VT	Department 4 Vote
DEPT5_VT	Department 5 Vote
DEPT6_VT	Department 6 Vote
DEPT7_VT	Department 7 Vote
BCB_EF	Baseline Control Board Effectivity
DTRCD	Date Received
YRMOREC	Year-Month Received (YMM)
FNLDIS	Final Disposition
PRCTL_ID	Print Control ID
CATCODE	Category Code
BCBR16	BCB Indicator for Release 16
NNLSTS1	Never-Never List Indicator for STS-1
BCBR17	BCB Indicator for Release 17
NNLR17	Never-Never List Indicator for Release 17
BCBR18	BCB Indicator for Release 18

NNLR18	Never-Never List Indicator for Release 18
BCBR19	BCB Indicator for Release 19
NNLR19	Never-Never List Indicator for Release 19
BCBR20	BCB Indicator for Release 20
NNLR20	Never-Never List Indicator for Release 20
BCBR18_3	BCB Indicator for Release 18.30 (STS-3)
BCBR18_4	BCB Indicator for Release 18.40 (STS-4)
OV99SYS	BCB Indicator for OV99 System
NAIN	Not Applicable to Pass Indicator
BFSIN	BFS Only Indicator
RELINVER	Release 16, 17, 18, 19, 20 Indicators, Verification
TITLE	Title
D1IMPYR1	Dept. 1 Manpower Impact - Year 1
D1IMPYR2	- Year 2
D1IMPYR3	- Year 3
D1IMPYR4	- Year 4
D2IMPYR1	Dept. 2 Manpower Impact - Year 1
D2IMPYR2	- Year 2
D2IMPYR3	- Year 3
D2IMPYR4	- Year 4
D3IMPYR1	Dept. 3 Manpower Impact - Year 1
D3IMPYR2	- Year 2
D3IMPYR3	- Year 3
D3IMPYR4	- Year 4
D4IMPYR1	Dept. 4 Manpower Impact - Year 1
D4IMPYR2	- Year 2
D4IMPYR3	- Year 3
D4IMPYR4	- Year 4
D5IMPYR1	Dept. 5 Manpower Impact - Year 1
D5IMPYR2	- Year 2
D5IMPYR3	- Year 3
D5IMPYR4	- Year 4
D6IMPYR1	Dept. 6 Manpower Impact - Year 1
D6IMPYR2	- Year 2
D6IMPYR3	- Year 3
D6IMPYR4	- Year 4
D7IMPYR1	Dept. 7 Manpower Impact - Year 1
D7IMPYR2	- Year 2
D7IMPYR3	- Year 3
D7IMPYR4	- Year 4
D1TOTIMP	Dept. 1 Total Manpower Impact
D2TOTIMP	Dept. 2 Total Manpower Impact
D3TOTIMP	Dept. 3 Total Manpower Impact
D4TOTIMP	Dept. 4 Total Manpower Impact
D5TOTIMP	Dept. 5 Total Manpower Impact
D6TOTIMP	Dept. 6 Total Manpower Impact
D7TOTIMP	Dept. 7 Total Manpower Impact
COMMENT1	Comments (Part 1)
COMMENT2	Comments (Part 2, Continuation of Part 1)
VCOS1	Early Release Indicator for Vehicle Checkout System #1

VCOS2	Early Release Indicator for Vehicle Checkout System #2
VCOS3	Early Release Indicator for Vehicle Checkout System #3
PALMS1	Early Release Indicator for Palmdale System #1
SS1	Early Release Indicator for System Software #1
ENTRYCL	Early Release Indicator for Entry Closed Loop (10/03/77)
PALMS2	Early Release Indicator for Palmdale System #2 (12/05/77)
SS2	Early Release Indicator for System Software #2
AOA	Early Release Indicator for Abort Once Around (02/06/78)
PALMS3	Early Release Indicator for Palmdale System #3 (03/06/78)
ENTRYS	Early Release Indicator for Entry System (05/01/78)
KSCS	Early Release Indicator for KSC System (06/05/78)
ENTRYUD1	Early Release Indicator for Entry Update #1 (07/03/78)
OFT1FACI	Early Release Indicator for OFT1 FACI System (09/04/78)
R12	Early Release Indicator for Release 12 (12/11/78)
CYC2ILD	Early Release Indicator for Cycle 2 I-Loads (02/05/79)
R13	Early Release Indicator for Release 13
R14	Early Release Indicator for Release 14
R15	Early Release Indicator for Release 15

Appendix L

CR LEGEND

FILE_ID -- File ID
Always "P"

PCR_NUM -- PCR Number; a unique number for tracking purposes.

CR_NUM -- CR Number; a unique number for tracking purposes.

CRREV_ID -- CR Revision ID; reflects re-issues of change requests due to changes.

A=First Reissue
B=Second Reissue
C=Third Reissue
D=Fourth Reissue
E=Fifth Reissue
F=Sixth Reissue
G=Seventh Reissue
H=Eighth Reissue

OASCB_EF -- Orbiter Avionics Software Control Board Effectivity; the Release ID of the mission the OASCB stated this CR was approved for. This fell into disuse when a separate board determined mission effectivity.

ALT=Approach and Landing Test
BLD=
BOTH=Both ALT and OFT Missions
DIT=Data ID Table
EOS=Extended Operating System
FEID=Flight Equipment Interface Development
FRF=STS-1 Flight Readiness Firing
IPL=Initial Program Load
MIP=
MMP=Mass Memory Procedures Document CR
N/A=Not Applicable
OFT=Orbital Flight Test
OFT1=Orbital Flight Test 1
OFT2=Orbital Flight Test 2
OFT3=Orbital Flight Test 3
OTHR=Non-Primary Avionics Software CR
OV99=Orbital Vehicle 99 Checkout System
POFT=Post-OFT (Operational Shuttle)
PRI=Primary Avionics Software CR
PS-1=Patch Set 1

PS-2=Patch Set 2
R-21=Release 21
ROM=Read Only Memory
SCL=
SDL=Software Development Laboratory
SFS=Shuttle Flight Support
SID=System Integration Department
SPF=Software Production Facility
SPF2=Software Production Facility 2
SS-1=STS-1 Mission
SS-2=STS-2 Mission
SS-3=STS-3 Mission
SS-6=STS-6 Mission
SS-7=STS-7 Mission
SS1=STS-1 Mission
SS2=STS-2 Mission
STS--STS Mission
STS1=STS-1 Mission
STS2=STS-2 Mission
STS3=STS-3 Mission
STS4=STS-4 Mission
STS5=STS-5 Mission
STS6=STS-6 Mission
V 19=Software Release 19
V 21=Software Release 21
V-05=Software Release 5
V-11=Software Release 11
V-12=Software Release 12
V-16=Software Release 16
V-17=Software Release 17
V-18=Software Release 18
V-19=Software Release 19
V-20=Software Release 20
V-21=Software Release 21
VERF=Verification
V18=Software Release 18
V19=Software Release 19
18=Software Release 18 (STS-3)
19=Software Release 19 (STS-5, STS-6)
20=Software Release 20 (STS-7, STS-8)

RRBDT -- RRB Disposition Date (MM/DD/Y)

RRBYRMO -- Year-Month of RRB Disposition (YMM)

CCBDDT -- Change Control Board Disposition Date (MM/DD/Y)

CCBYRMO -- Year-Month of CCB Disposition (YMM)

OASCBDT -- OASCB Disposition Date (MM/DD/Y)

OASCBYRM -- Year-Month of OASCB Disposition (YMM)

RRBDIS -- RRB Disposition

A=Approved

D=Disapproved

N=Invalid Entry; There should be no entries of "N".

R=Invalid Entry; There should be no entries of "R".

W=Withdrawn

l=Invalid Entry; There should be no entries of "l".

CCBDIS -- CCB Disposition

A=Approved

D=Disapproved

W=Withdrawn

O=Invalid Entry; There should be no entries of "O".

OASCBDIS -- OASCB Disposition

A=Approved

D=Disapproved

W=Withdrawn

DEPT1_ID -- Department 1 ID

GNC=Guidance Navigation and Control

GNC/C=Guidance Navigation and Control--Ex/Seq/Guid

DEPT2_ID -- Department 2 ID

SM=Systems Management

SM/PL=Systems Management/Payload Management

SYS ANL=Systems Analysis

DEPT3_ID -- Department 3 ID

FCOS/CI=Flight Computer Operation System/Configuration Inspection

FCOS/UI=Flight Computer Operation System/User Interface

DEPT4_ID -- Department 4 ID

VERIF=Verification

DEPT5_ID -- Department 5 ID

SDL=Software Development Laboratory

DEPT6_ID -- Department 6 ID

SMP/DL=Systems Management Offline/Downlist

SYS ANAL=Systems Analysis

SYS ANL=Systems Analysis

DEPT7_ID -- Department 7 ID

FEID=Flight Equipment Interface Development

PL=Payload

SM/PL=Systems Management/Payload

DEPT1_VT -- Department 1 Vote

A=Approved by this Department

C=Invalid Entry; There should be no entries of "C".

N=Not Applicable to this Department

O=Invalid Entry; There should be no entries of "O".
R=Reject Current CR; Awaiting a Revision.
U=Vote Pending
W=Recommend Withdrawal

DEPT2_VT -- Department 2 Vote
A=Approved by this Department
N=Not Applicable to this Department
R=Reject Current CR; Awaiting a Revision.
U=Vote Pending
W=Recommend Withdrawal

DEPT3_VT -- Department 3 Vote
A=Approved by this Department
N=Not Applicable to this Department
R=Reject Current CR; Awaiting a Revision.
U=Vote Pending
W=Recommend Withdrawal

DEPT4_VT -- Department 4 Vote
A=Approved by this Department
N=Not Applicable to this Department
R=Reject Current CR; Awaiting a Revision.
U=Vote Pending
W=Recommend Withdrawal

DEPT5_VT -- Department 5 Vote
A=Approved by this Department
N=Not Applicable to this Department
R=Reject Current CR; Awaiting a Revision.
U=Vote Pending
W=Recommend Withdrawal
4=Invalid Entry; There should be no entries of "4".

DEPT6_VT -- Department 6 Vote
A=Approved by this Department
N=Not Applicable to this Department
R=Reject Current CR; Awaiting a Revision.
U=Vote Pending
W=Recommend Withdrawal

DEPT7_VT -- Department 7 Vote
A=Approved by this Department
N=Not Applicable to this Department
R=Reject Current CR; Awaiting a Revision.
U=Vote Pending
W=Recommend Withdrawal

BCB_EF -- Baseline Control Board Effectivity
AC=Ascent 09/04/78 Drop
AOA=Abort Once Around
ASC=Ascent 09/04/78 Drop

AU=Ascent Update 12/11/78
BFS=Backup Flight System
CY-3=Cycle 3 I-Load Update
EC=Events Controller
ECL=Entry Closed Loop 10/03/77
ENT=Entry 05/01/78
EU1=Entry Update #1 07/03/78
EU2=Entry Update #2 09/04/78
EU3=Entry Update #3 12/11/78
GNC=Guidance, Navigation and Control
KSC=Kennedy Space Center
KCSU=VU/Downlist KSC Update 09/04/78
N/A=Not Applicable
NA=Not Applicable
NNL=Never-Never List (ALT); Approved CRs that were not implemented.
NNL1=Never-Never List (STS-1); Approved CRs that were not implemented.
NNL2=Never-Never List (STS-2); Approved CRs that were not implemented.
OC=Orbit Closed Loop 05/01/78
OCL=Orbit Closed Loop 05/01/78
OFTU=Orbital Flight Test Update
OFT1=Orbital Flight Test 1
ONLY=
ORB=Orbit 09/04/78
OU=Orbit Update 12/11/78
OV99=Orbital Vehicle 99 Checkout System
PDP=Payload Bay Door System 05/01/78
PMD1=Palmdale #1
PMD2=Palmdale #2
PMD3=Palmdale #3
RMSC=Early RMS Drop
RMSD=Early RMS Drop
R17=Release 17
R18=Release 18
R19=Release 19
R21=Release 21
R36=Old SDL Release 36
SMC=
SMC3=
SM2=
SS-1=STS-1 Mission
SS-2=STS-2 Mission
SSC=
SSC3=
SSW1=
SSW2=
SS1+=
SS1C=
SS2C=
STS2=STS-2 Mission

STS3=STS-3 Mission
STS4=STS-4 Mission
STS5=STS-5 Mission
STS6=STS-6 Mission
STS9=STS-9 Mission
TBS=
V-18=Software Release 18
V-19=Software Release 19
V-20=Software Release 20
V-21=Software Release 21
VCAN=
VC1=Vehicle Checkout #1
VC2=Vehicle Checkout #2
VC3=Vehicle Checkout #3
OFTU=OFT (STS-1) Update 12/11/78
10.0=Software Release 10.0
10.1=Software Release 10.1

DTRECD -- Date Received

YRMOREC -- Year-Month Received (YMM)

FNLDIS -- Final Disposition

APPROVD=Approved
DISAPP=Disapproved
DISAPPV=Disapproved
WITHDRN=Withdrawn

PRCTL_ID -- Print Control ID

90=Used to suppress printing of a record for "current record" reports.

CATCODE -- Category Code

CONCE=
FO/FS=Fail Operational/Fail Safe Study
PCRC=
REQ=Requirements
REQMT=Requirements
SCRUB=Scrub Library
02/26/1D=Invalid Entry; There should be no "02/26/1D" entries.
03/04/1A=Invalid Entry; There should be no "03/04/1A" entries.
03/04/1D=Invalid Entry; There should be no "03/04/1D" entries.
03/04/1W=Invalid Entry; There should be no "03/04/1W" entries.

BCBR16 -- Baseline Control Board Indicator for Release 16

B=This CR Appears on BCB 5110 Listings.

NNLSTS1 -- Never-Never List Indicator for STS-1. Never-Never lists provide accounting for CRs that were approved, but never implemented.

*=This CR appears in Never-Never list.

N=This CR appears in Never-Never list.

BCBR17 -- Baseline Control Board Indicator for Release 17
 B=This CR Appears on BCB 5110 Listings
 V=Invalid Entry; There should be no "V" entries.

NNLR17 -- Never-Never List Indicator for Release 17. Never-Never lists provide accounting for CRs that were approved, but never implemented.
 *-This CR appears in a Never-Never List.
 N=This CR appears in a Never-Never List.

BCBR18 -- Baseline Control Board Indicator for Release 18
 B=This CR Appears on BCB 5110 Listings.

NNLR18 -- Never-Never List Indicator for Release 18. Never-Never lists provide accounting for CRs that were approved, but never implemented.
 *-This CR appears in a Never-Never List.
 N=This CR appears in a Never-Never List.

BCBR19 -- Baseline Control Board Indicator for Release 19
 B=This CR Appears on BCB 5110 Listings.

NNLR19 -- Never-Never List Indicator for Release 19. Never-Never lists provide accounting for CRs that approved, but never implemented.
 *-This CR appears in a Never-Never List.
 N=This CR appears in a Never-Never Li.

BCBR20 -- Baseline Control Board Indicator for Release 20.
 B=This CR Appears on BCB 5110 Listings.

NNLR20 -- Never-Never List Indicator for Release 20. Never-Never lists provide accounting for CRs that were approved, but never implemented.
 *-This CR appears in a Never-Never List.
 N=This CR appears in a Never-Never List.

BCBR18_3 -- Baseline Control Board Indicator for Release 18.30
 (STS-3)
 B=This CR Appears on BCB 5110 Listings.

BCBR18_4 -- Baseline Control Board Indicator for Release 18.40
 (STS-4)
 B=This CR Appears on BCB 5110 Listings.

OV99SYS -- Baseline Control Board Indicator for Orbital Vehicle 99 Checkout System.
 B=This CR Appears on BCB 5110 Listings.

NAIN -- Not Applicable to Pass Indicator
 D=Invalid Entry; There should be no "D" entries.
 M=Invalid Entry; There should be no "M" entries.

N=Invalid Entry; There should be no "N" entries.

P=Invalid Entry; There should be no "P" entries.

1=DIT

2=OTT

3=VEH ONLY

4=IPL

5=GMEM R/W

6=MIP

7=BFS

8=DLL

9=MISC

BFSIN -- Backup Flight System Only Indicator

*=This CR is applicable to Backup Flight System only.

RELINVER -- Release 16,17,18,19,20 Indicators - Verification

B=This CR Appears in BCB 5110 Data Base.

V=This CR Appears in Verification CR Data Base.

TITLE -- Title

D1IMPYR1 -- Department 1 Manpower Impact - Year 1

D1IMPYR2 -- Department 1 Manpower Impact - Year 2

D1IMPYR3 -- Department 1 Manpower Impact - Year 3

D1IMPYR4 -- Department 1 Manpower Impact - Year 4

D2IMPYR1 -- Department 2 Manpower Impact - Year 1

D2IMPYR2 -- Department 2 Manpower Impact - Year 2

D2IMPYR3 -- Department 2 Manpower Impact - Year 3

D2IMPYR4 -- Department 2 Manpower Impact - Year 4

D3IMPYR1 -- Department 3 Manpower Impact - Year 1

D3IMPYR2 -- Department 3 Manpower Impact - Year 2

D3IMPYR3 -- Department 3 Manpower Impact - Year 3

D3IMPYR4 -- Department 3 Manpower Impact - Year 4

D4IMPYR1 -- Department 4 Manpower Impact - Year 1

D4IMPYR2 -- Department 4 Manpower Impact - Year 2

D4IMPYR3 -- Department 4 Manpower Impact - Year 3

D4IMPYR4 -- Department 4 Manpower Impact - Year 4

D5IMPYR1 -- Department 5 Manpower Impact - Year 1
D5IMPYR2 -- Department 5 Manpower Impact - Year 2
D5IMPYR3 -- Department 5 Manpower Impact - Year 3
D5IMPYR4 -- Department 5 Manpower Impact - Year 4
D6IMPYR1 -- Department 6 Manpower Impact - Year 1
D6IMPYR2 -- Department 6 Manpower Impact - Year 2
D6IMPYR3 -- Department 6 Manpower Impact - Year 3
D6IMPYR4 -- Department 6 Manpower Impact - Year 4
D7IMPYR1 -- Department 7 Manpower Impact - Year 1
D7IMPYR2 -- Department 7 Manpower Impact - Year 2
D7IMPYR3 -- Department 7 Manpower Impact - Year 3
D7IMPYR4 -- Department 7 Manpower Impact - Year 4
D1TOTIMP -- Department 1 Total Manpower Impact
D2TOTIMP -- Department 2 Total Manpower Impact
D3TOTIMP -- Department 3 Total Manpower Impact
D4TOTIMP -- Department 4 Total Manpower Impact
D5TOTIMP -- Department 5 Total Manpower Impact
D6TOTIMP -- Department 6 Total Manpower Impact
D7TOTIMP -- Department 7 Total Manpower Impact
COMMENT1 -- Comments (Part 1)
COMMENT2 -- Comments (Part 2, Continuation of Part 1)
VCOS1 -- Early Release Indicator for Vehicle Checkout System #1.
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only
VCOS2 -- Early Release Indicator for Vehicle Checkout System #2.
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

VCOS3 -- Early Release Indicator for Vehicle Checkout System #3.
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

PALMS1 -- Early Release Indicator for Palmdale System #1.
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

SS1 -- Early Release Indicator for System Software #1.
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

ENTRYCL -- Early Release Indicator for Entry Closed Loop (10/03/77)
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

PALMS2 -- Early Release Indicator for Palmdale System #2 (12/05/77)
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

SS2 -- Early Release Indicator for System Software #2
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

AOA -- Early Release Indicator for Abort Once Around (02/06/78)
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

PALMS3 -- Early Release Indicator for Palmdale System #3 (03/06/78)
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

ENTRYS -- Early Release Indicator for Entry System (05/01/78)
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

KSCS -- Early Release Indicator for KSC System (06/05/78)
 *=This CR Appears in BCB 5110 Data Bases
 C=This CR Appears in old BCB Charts
 D=This CR Appears in Development Plans Only

ENTRYUD1 -- Early Release Indicator for Entry Update #1 (07/03/78)
 *=This CR Appears in BCB 5110 Data Bases

C=This CR Appears in old BCB Charts
D=This CR Appears in Development Plans Only

OFT1FACI -- Early Release Indicator for OFT1 FACI System (09/04/78)

*=This CR Appears in BCB 5110 Data Bases
C=This CR Appears in old BCB Charts
D=This CR Appears in Development Plans Only

R12 -- Early Release Indicator for Release 12 (12/11/78)

*=This CR Appears in BCB 5110 Data Bases
C=This CR Appears in old BCB Charts
D=This CR Appears in Development Plans Only

CYC2ILD -- Early Release Indicator for Cycle 2 I-Loads (02/05/79)

*=This CR Appears in BCB 5110 Data Bases
C=This CR Appears in old BCB Charts
D=This CR Appears in Development Plans Only

R13 -- Early Release Indicator for Release 13

*=This CR Appears in BCB 5110 Data Bases
C=This CR Appears in old BCB Charts
D=This CR Appears in Development Plans Only

R14 -- Early Release Indicator for Release 14

*=This CR Appears in BCB 5110 Data Bases
C=This CR Appears in old BCB Charts
D=This CR Appears in Development Plans Only

R1 -- Early Release Indicator for Release 15

*=This CR Appears in BCB 5110 Data Bases
C=This CR Appears in old BCB Charts
D=This CR Appears in Development Plans Only

Appendix M

COST DATA CHARTS

Data, provided by IBM, pertaining to costs for development and verification were examined. Data were plotted over time using monthly and cumulative values to observe how the amount and rate of spending changed over time. Change data were also plotted in conjunction with cost data to visualize how they changed concurrently.

ORIGINAL PAGE IS
OF POOR QUALITY

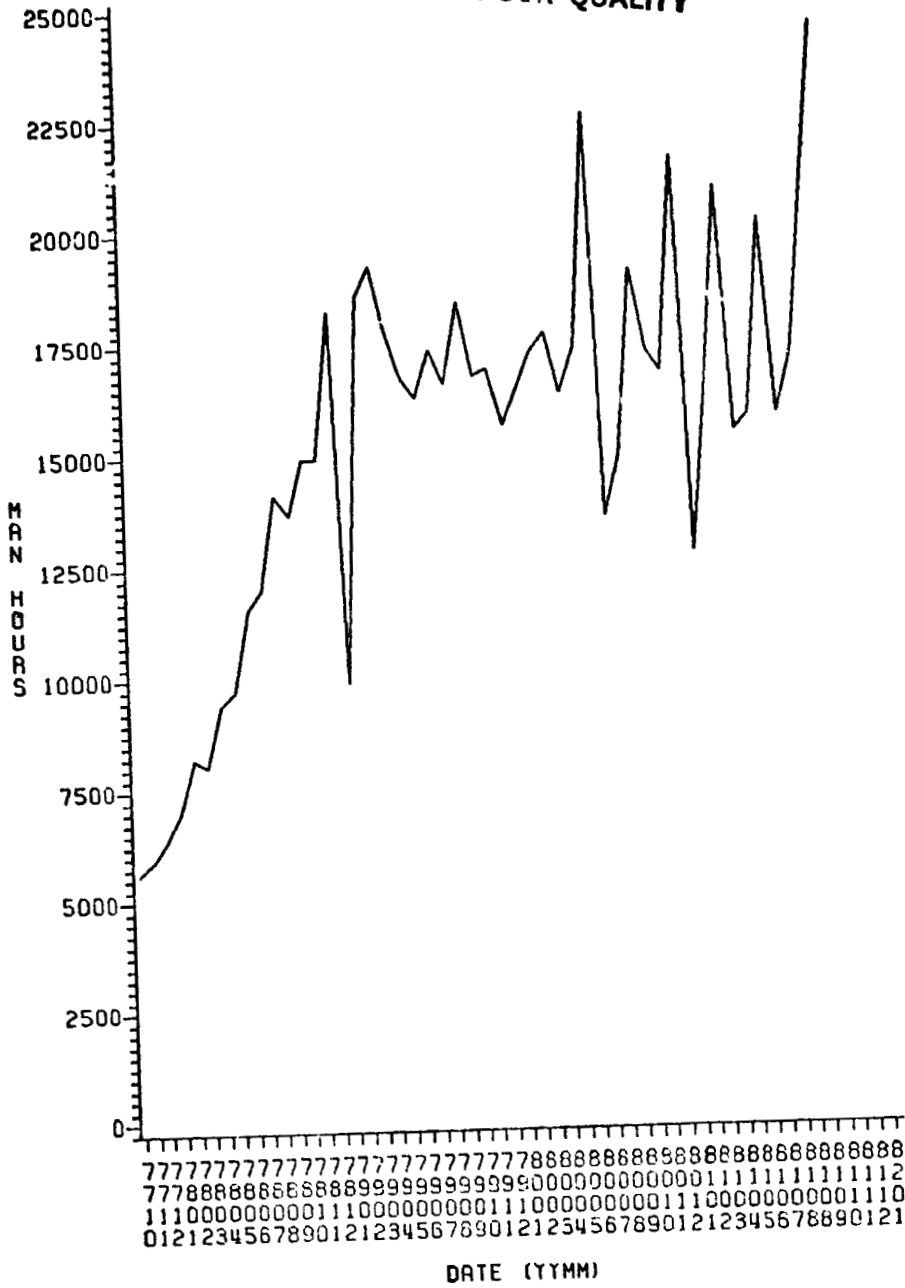


Figure 3: Total hours IBM billed for verification.

ORIGINAL PAGE IS
OF POOR QUALITY

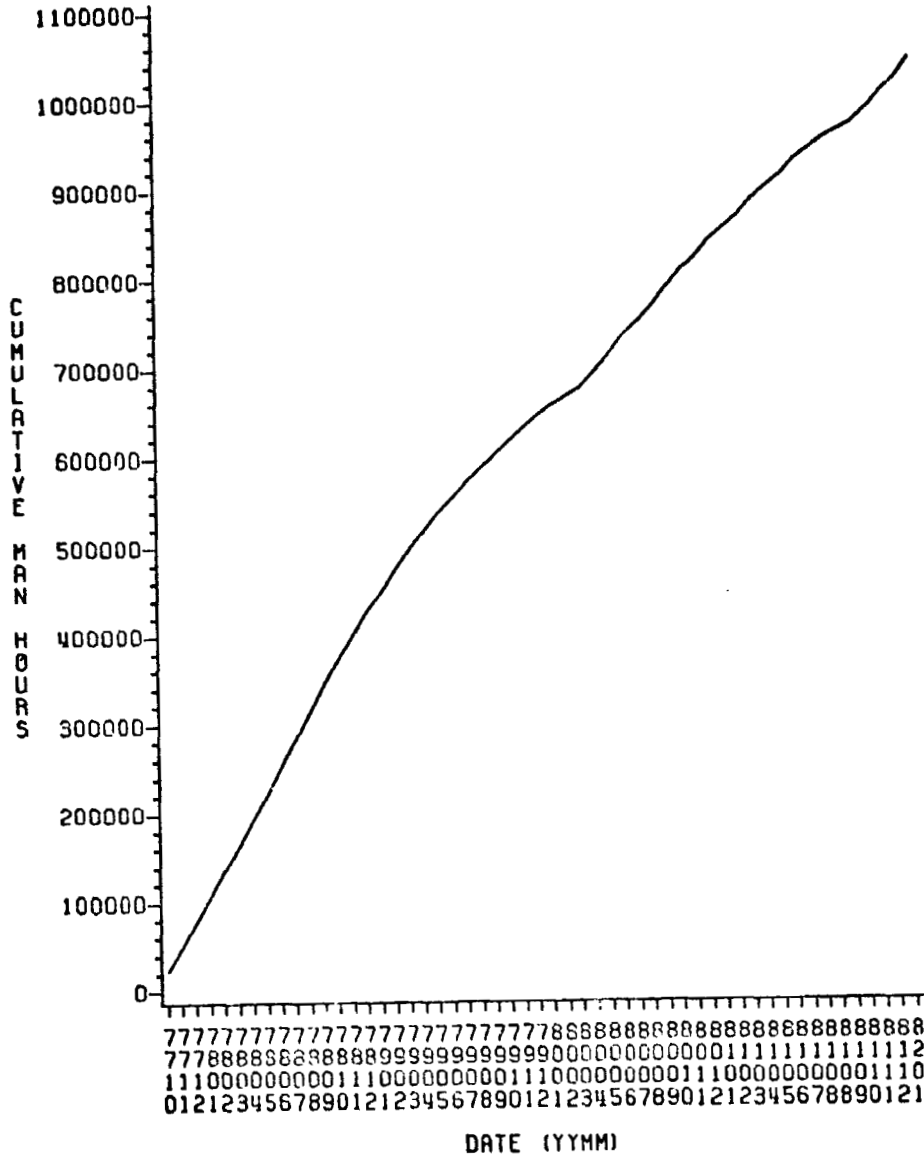


Figure 4: Cumulative total man-hours IBM billed NASA for development.

ORIGINAL PORTION
OF POOR QUALITY

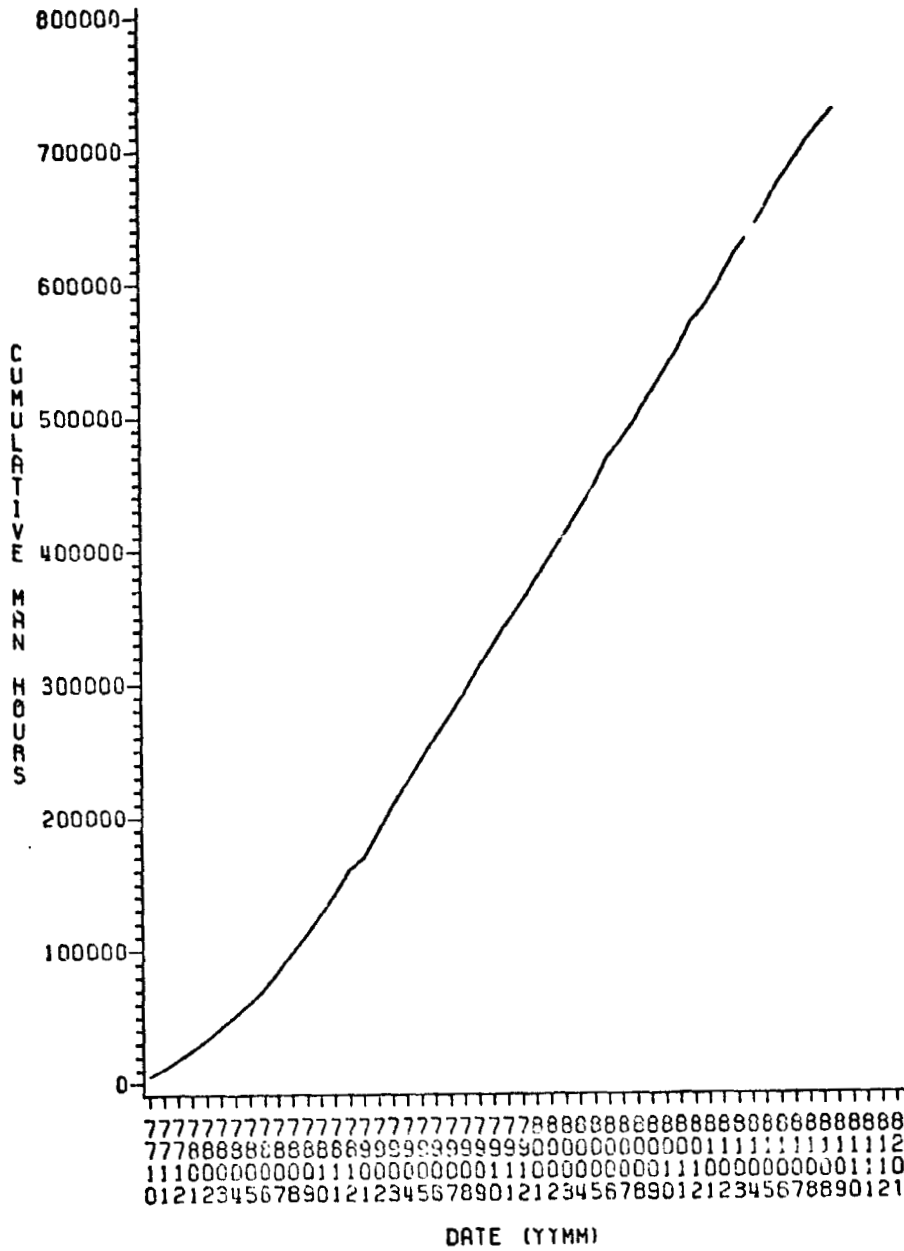


Figure 5. Cumulative total man-hours IBM billed NASA for verification.

OF POOR QUALITY

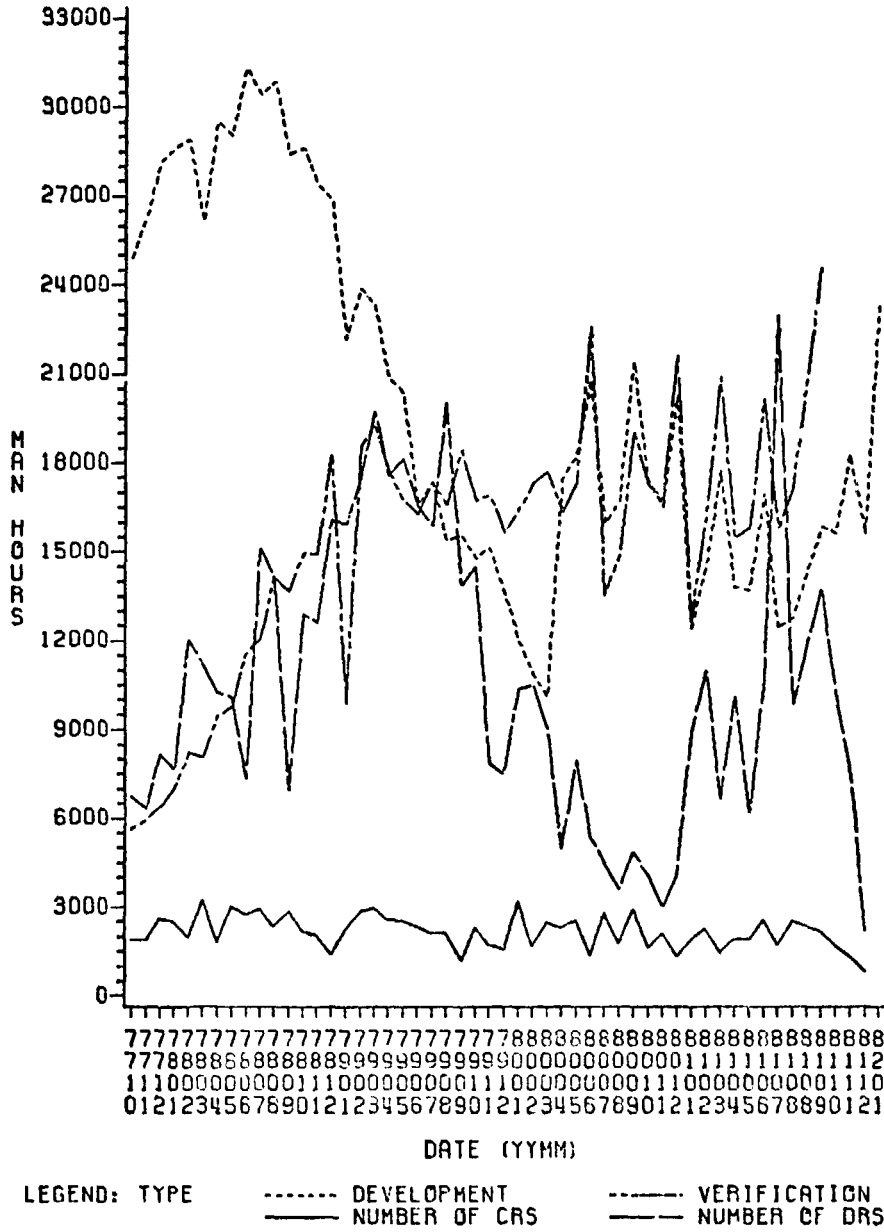


Figure 6: Development and verification cost with scaled change and discrepancy reports frequency.

Appendix N

CHANGE DATA CHARTS

The change data base consisted of change and discrepancy reports. Frequency of selected variables over time and values of variables were plotted in an effort to determine what data were available and their general characteristics. Two or more variables were used in some plots to get a better understanding of the data.

ORIGINAL PAGE IS
OF POOR QUALITY

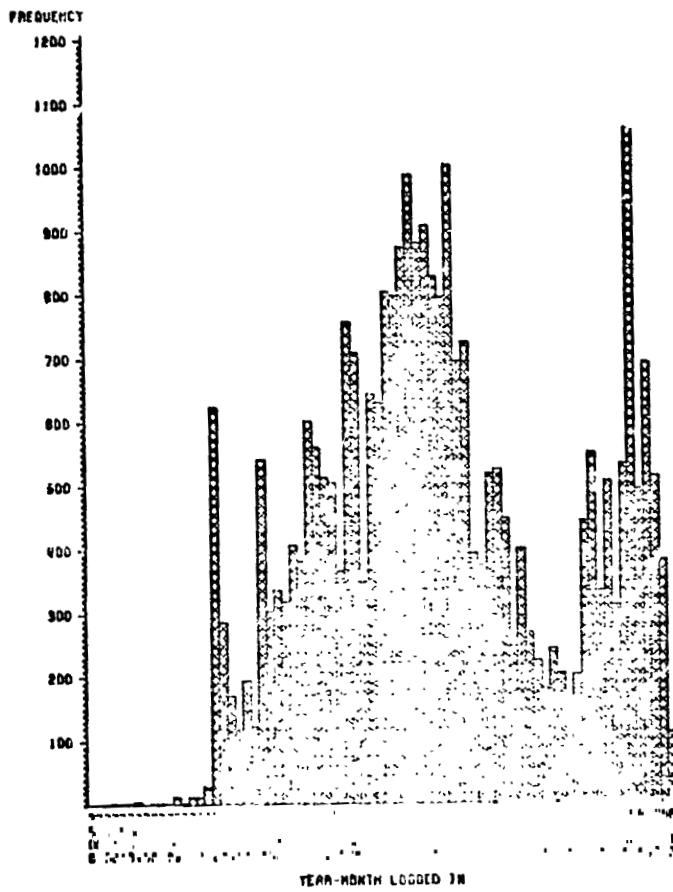


Figure 7: Frequency of DR's logged in by year-month.

ORIGINAL PAGE IS
OF POOR QUALITY

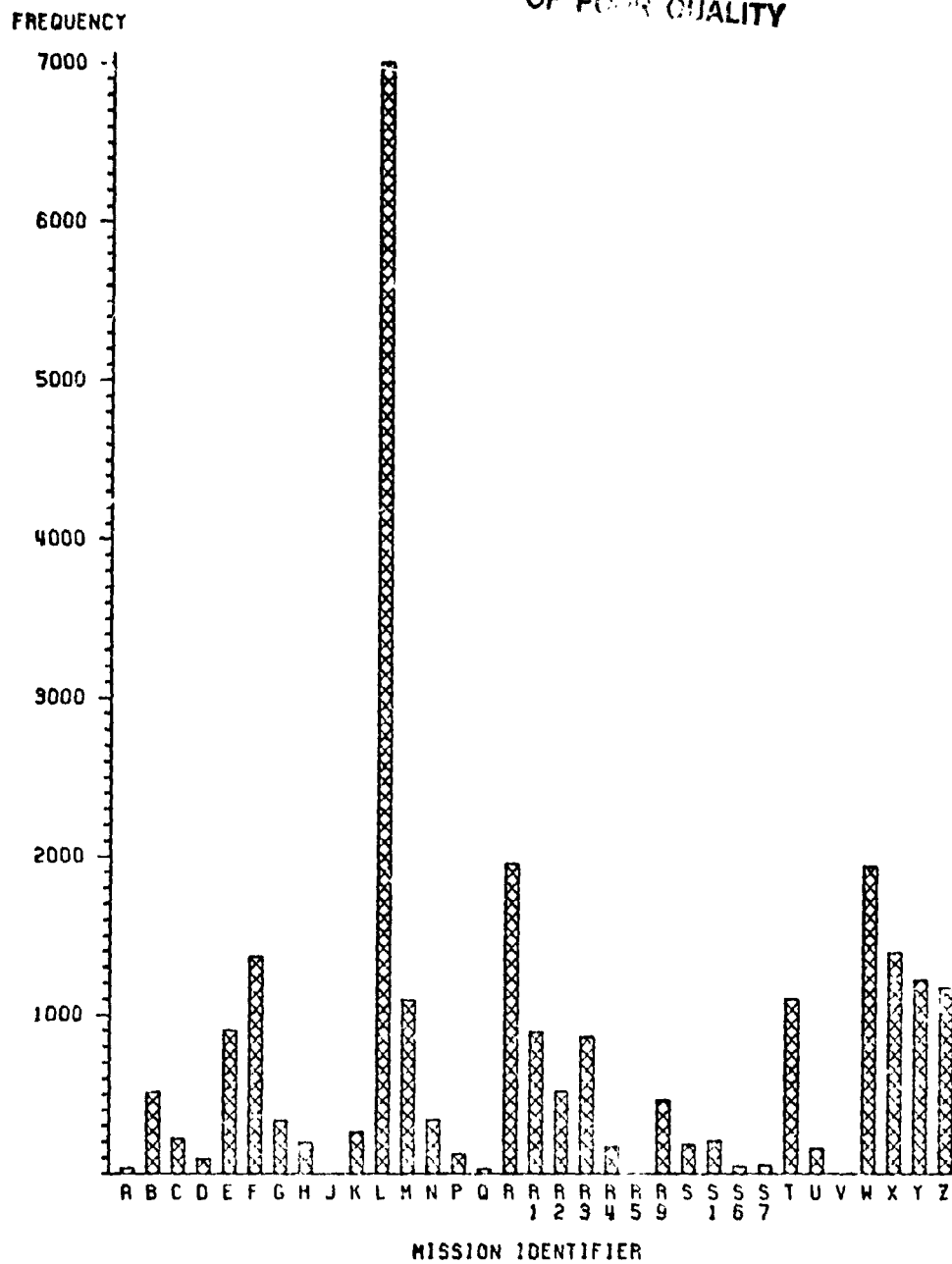


Figure 8: Frequency of DR's for mission identifier (MISSN_ID).

ORIGINAL PRICE
OF FOOD QUALITY

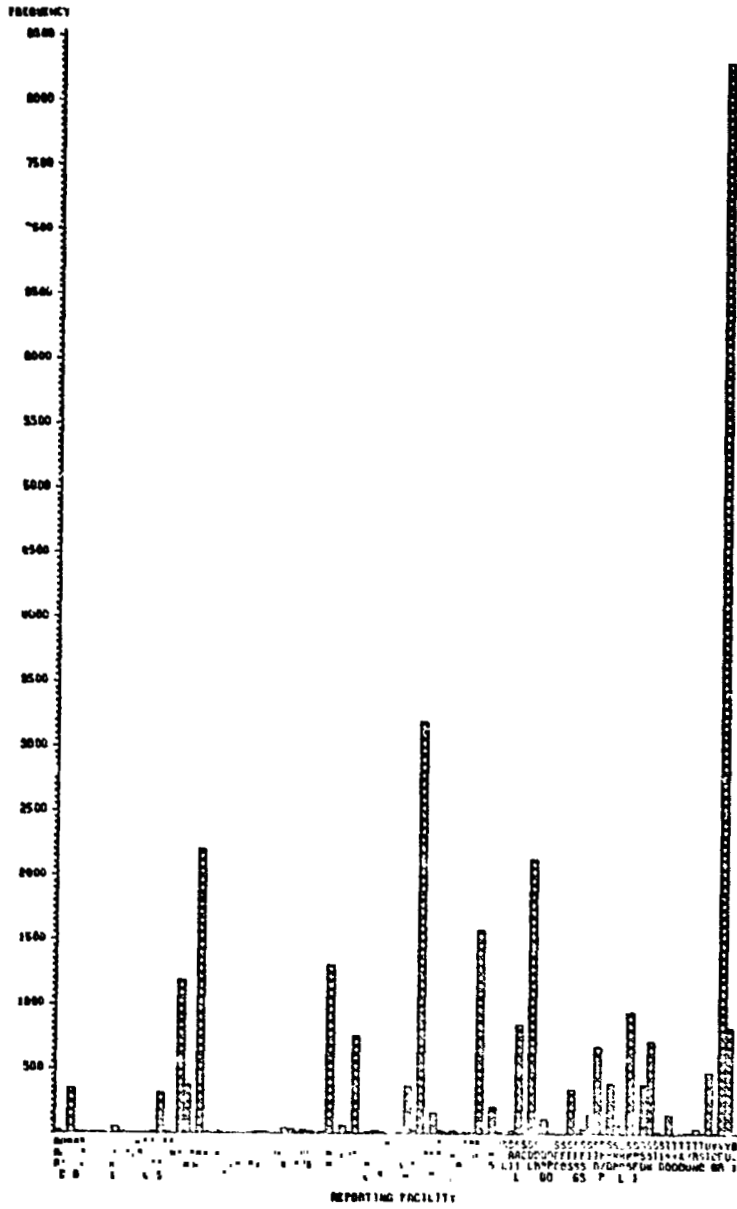


Figure 9: DR frequency for reporting facility (FAC).

ORIGINAL PAGE IS
OF POOR QUALITY

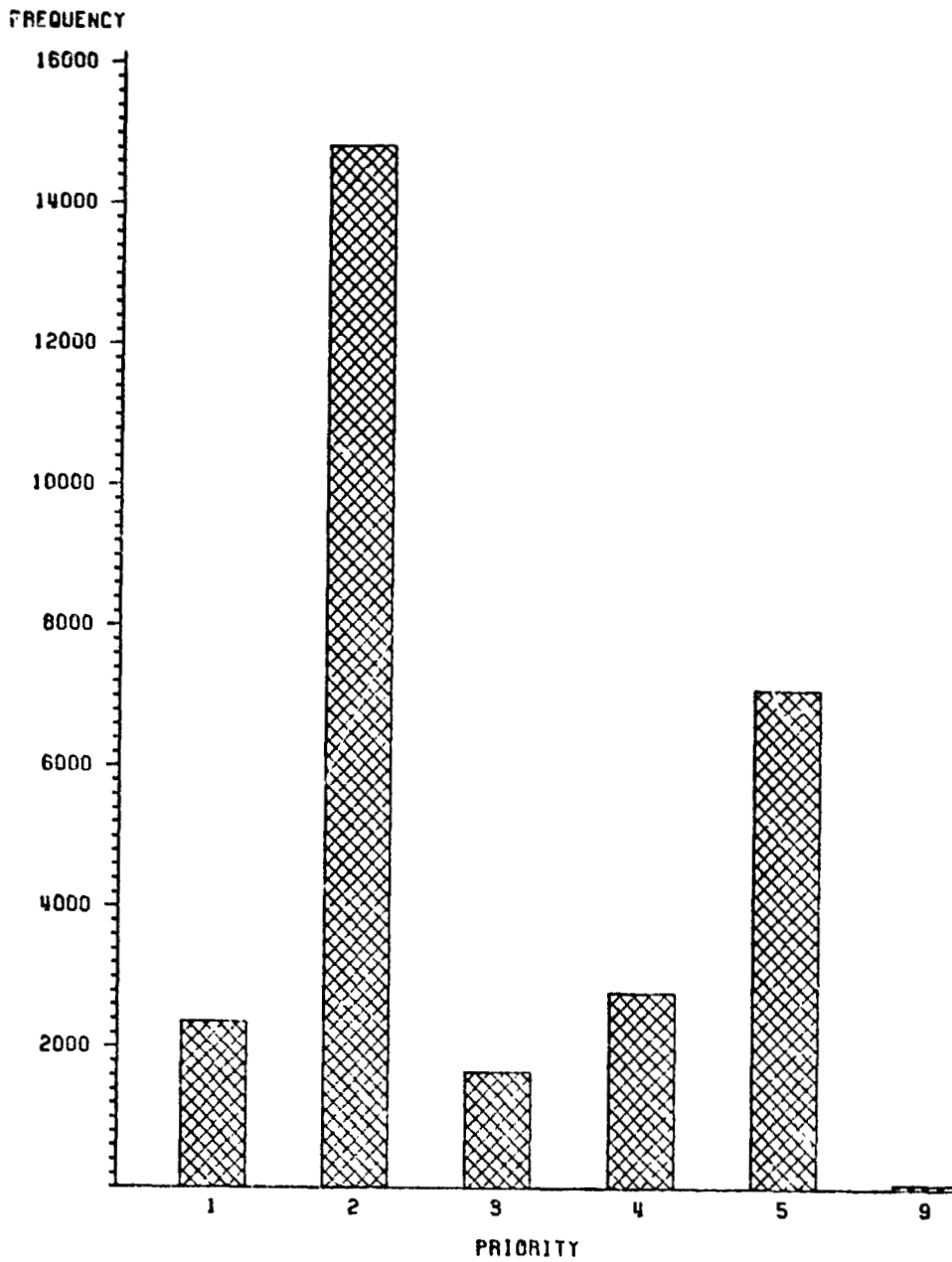


Figure 11: DR frequency for priority (P).

ORIGINAL PARTS
OF POOR QUALITY

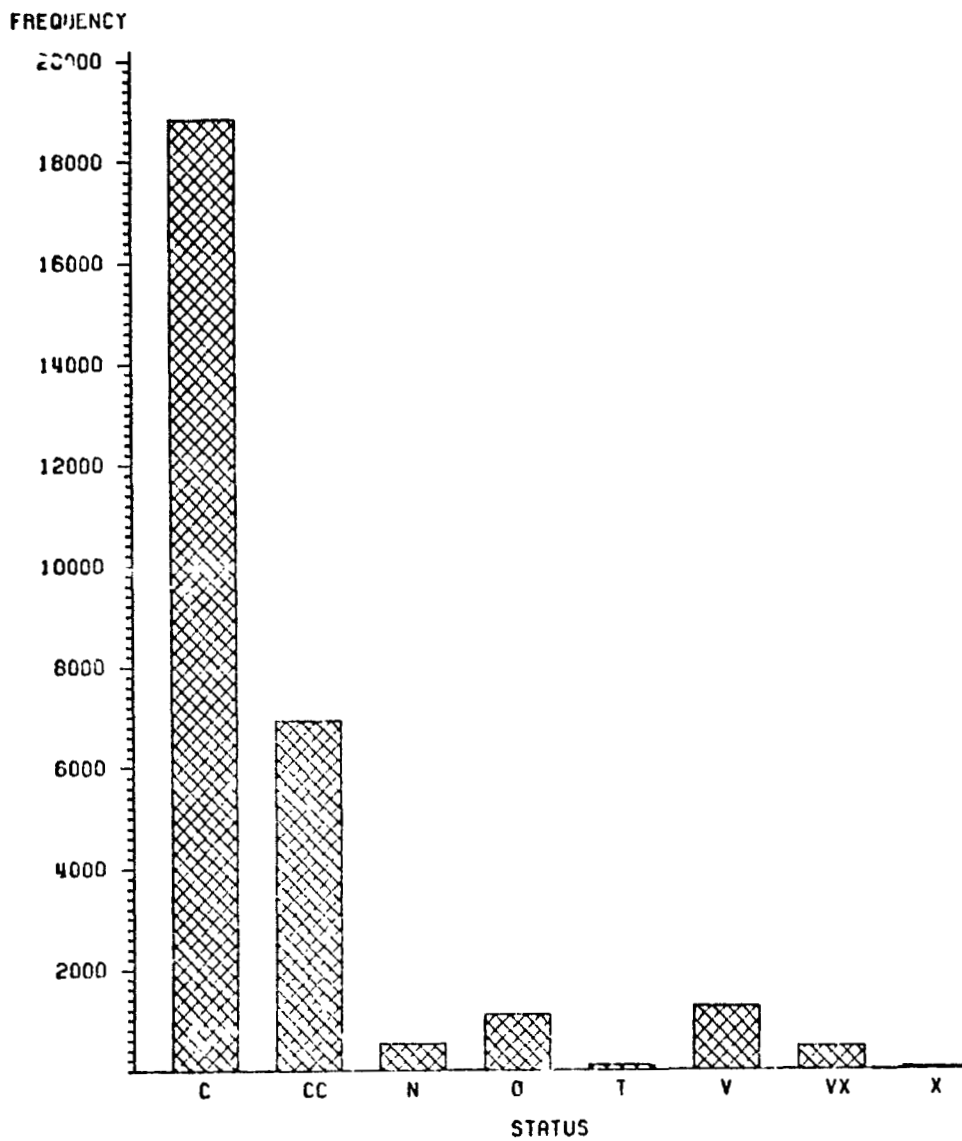


Figure 12: DR frequency for status (ST).

ORIGINAL PAGE IS
OF POOR QUALITY

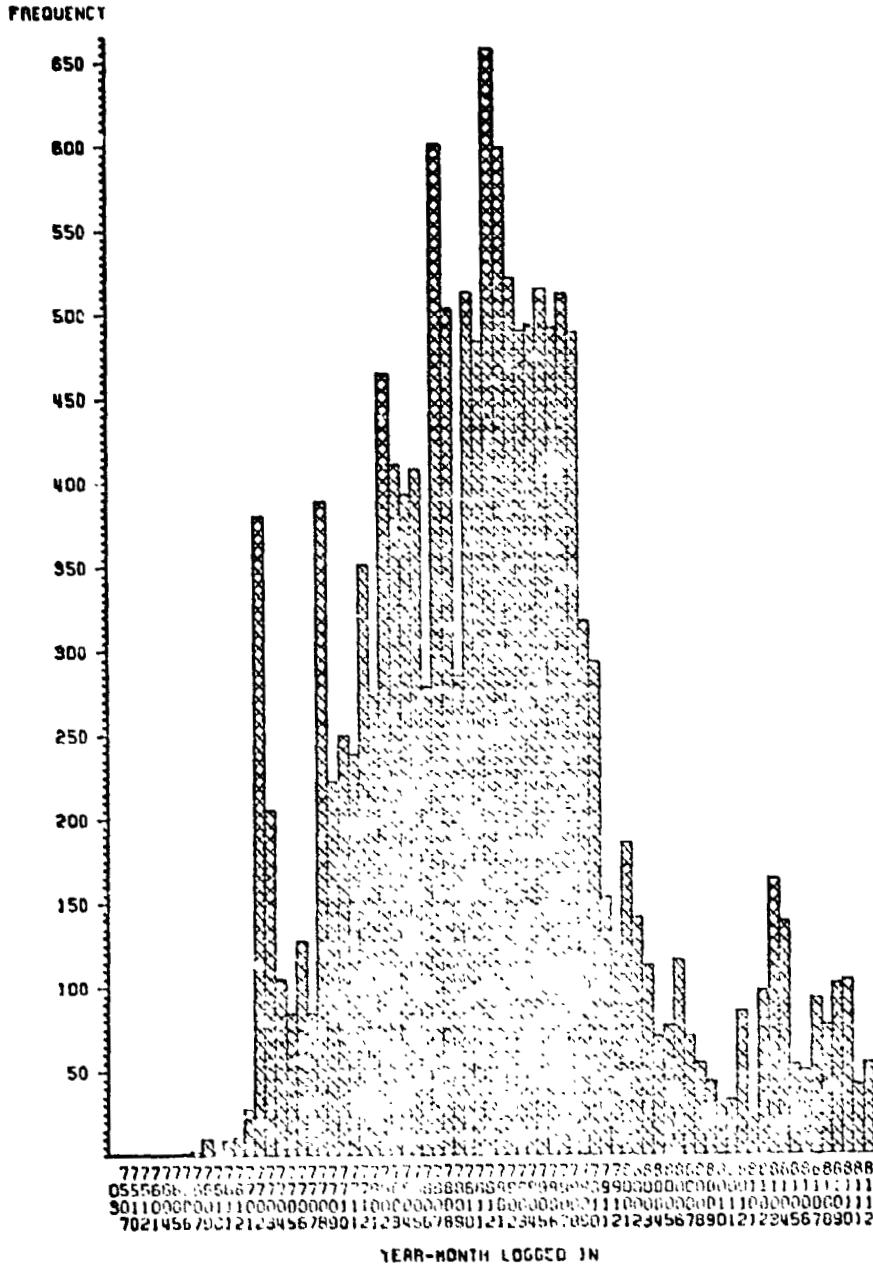


Figure 15: Frequency of priority 2 DR's logged in by year-month (_ODYRMC).

ORIGINAL PAGE IS
OF POOR QUALITY

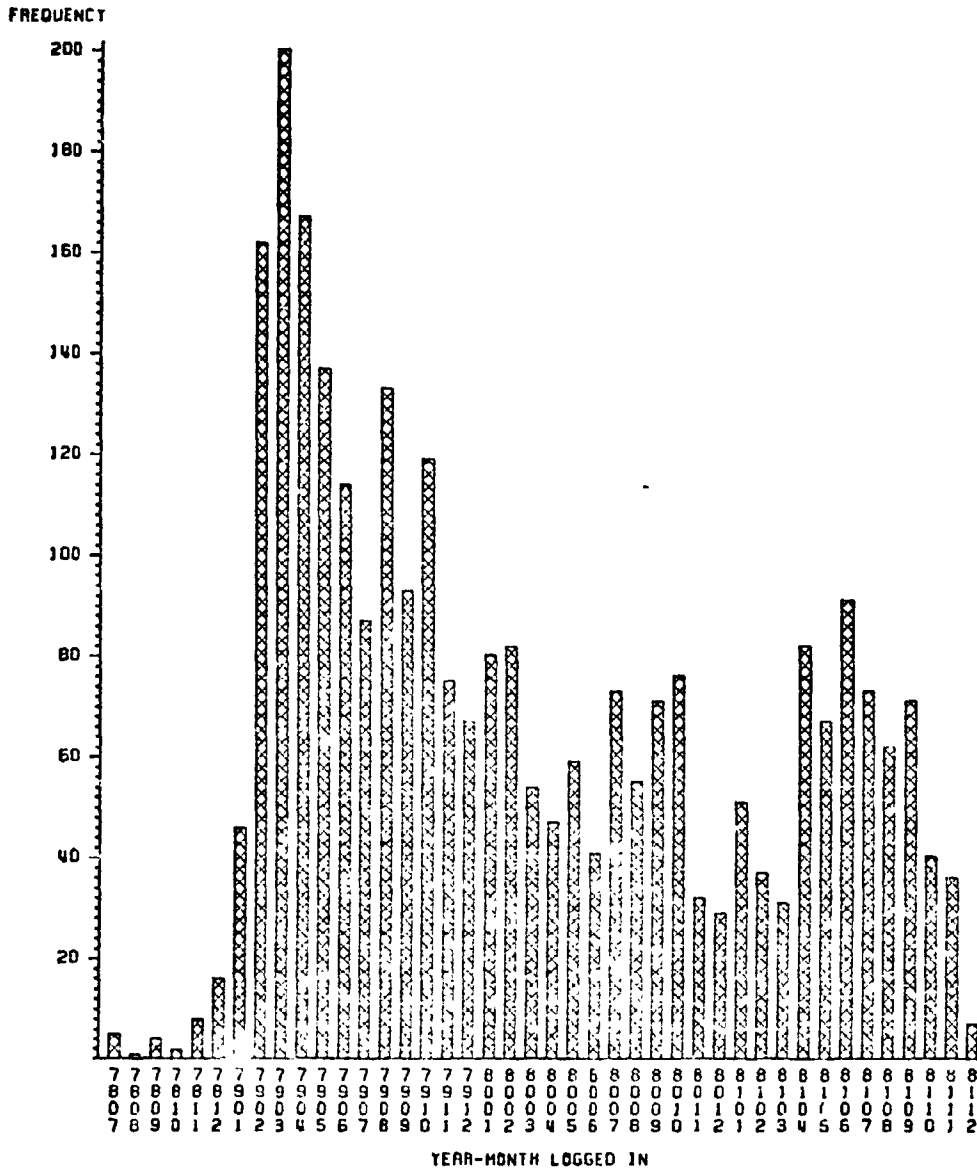


Figure 17: Frequency of priority 4 DR's logged in by year-month (ODYRMO).

ORIGINAL PAGE IS
OF POOR QUALITY

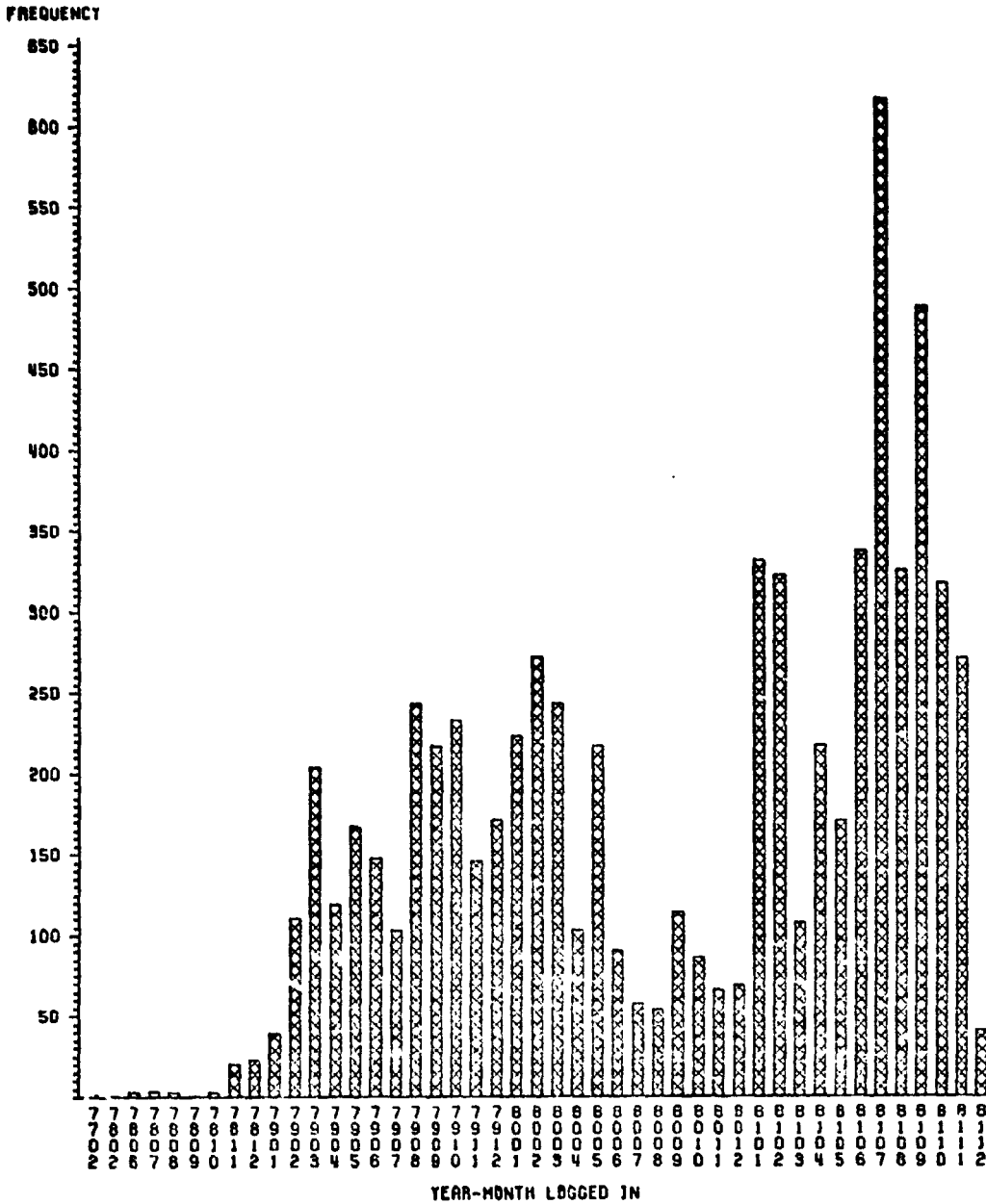


Figure 18: Frequency of priority 5 DR's logged in by year-month (ODYRMO).

ORIGINAL PAGE IS
OF POOR QUALITY

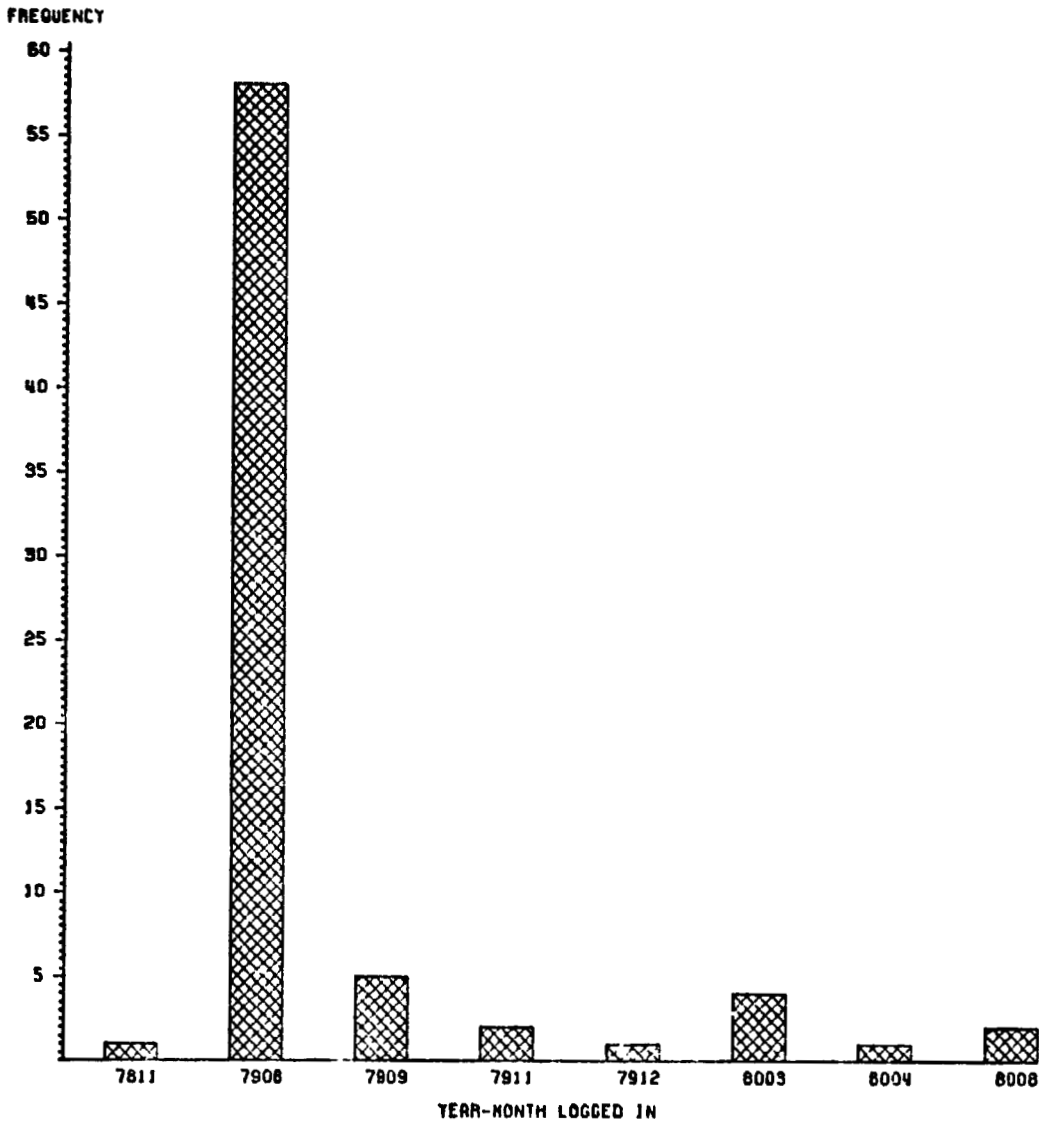


Figure 19: Frequency of priority 9 DR's logged in by year-month (_ODYRMO).

ORIGINAL PAGE IS
OF POOR QUALITY

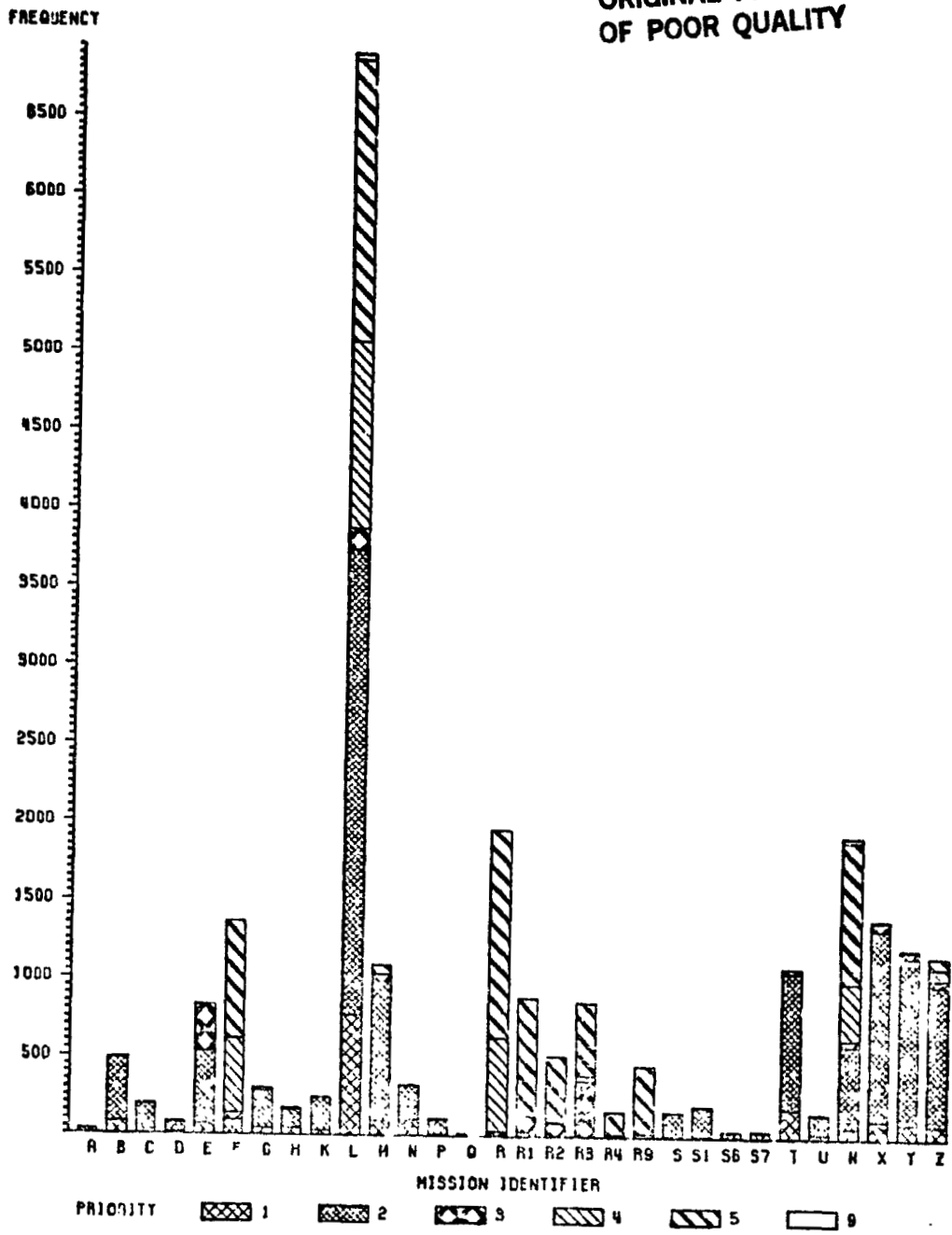


Figure 20: Priority (P) of DR's for mission identifier (MISSN_ID).

ORIGINAL PAGE IS
OF POOR QUALITY

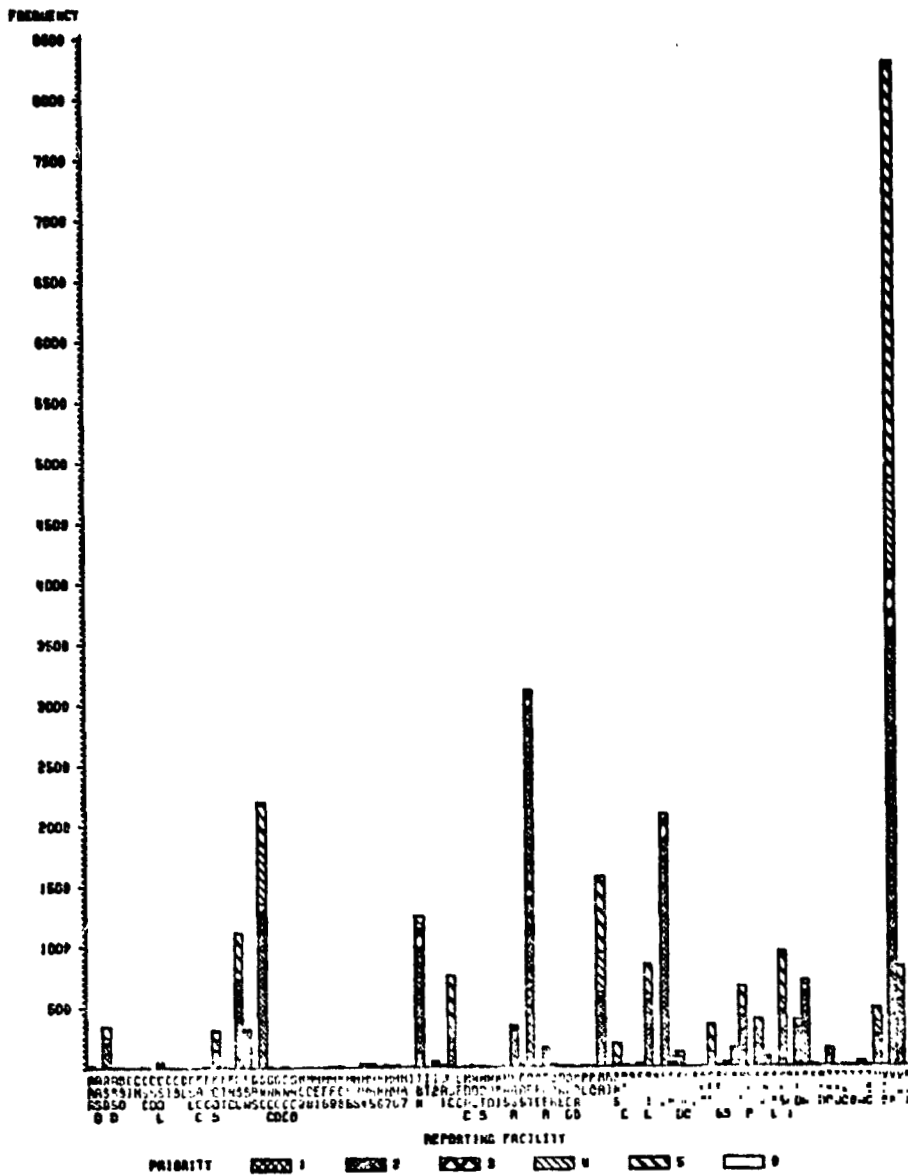


Figure 21: Priority (P) of DR's from a reporting facility (FAC).

ORIGINAL PAGE IS
OF POOR QUALITY

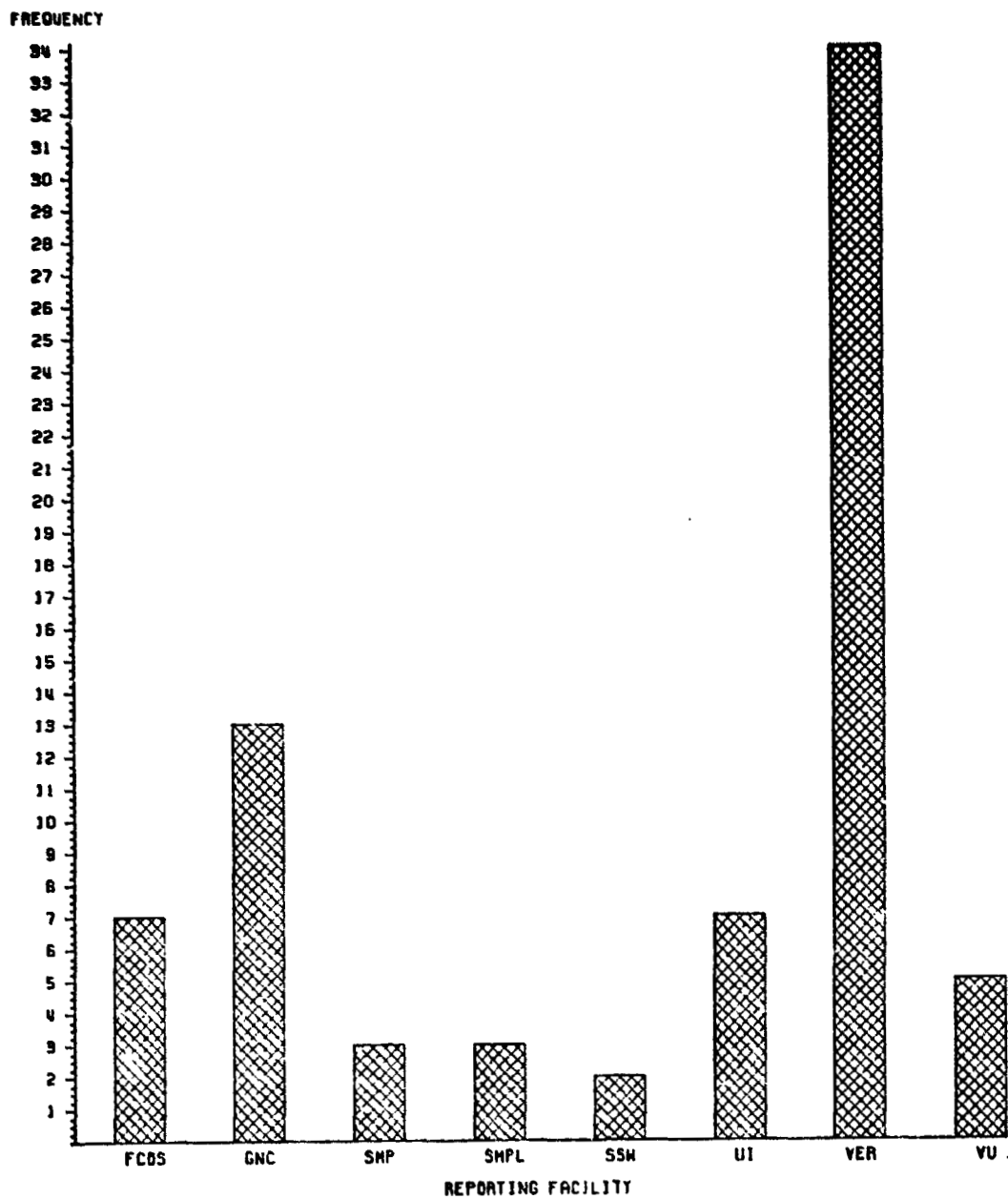


Figure 27: Frequency of priority 9 DR's from a reporting facility (FAC).

ORIGINAL PAGE IS
OF POOR QUALITY

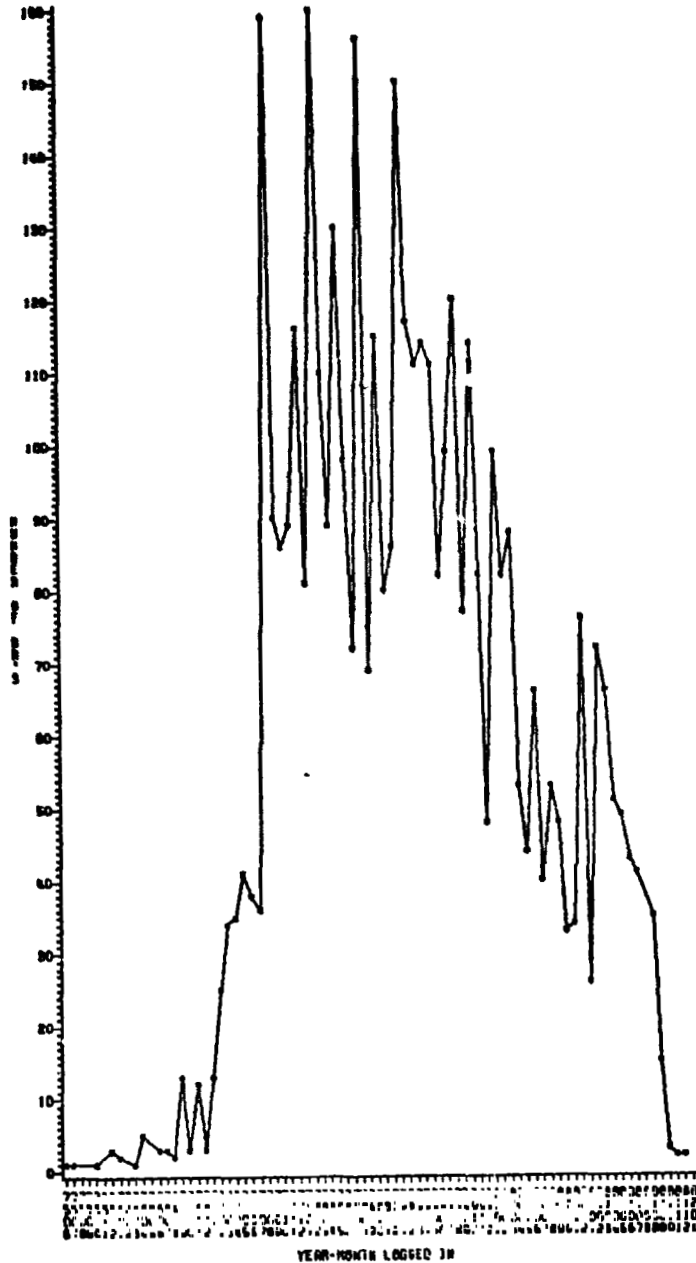


Figure 28: Number of DR's logged without a mission identifier (MISSN_ID).

ORIGINAL PAGE IS
OF POOR QUALITY

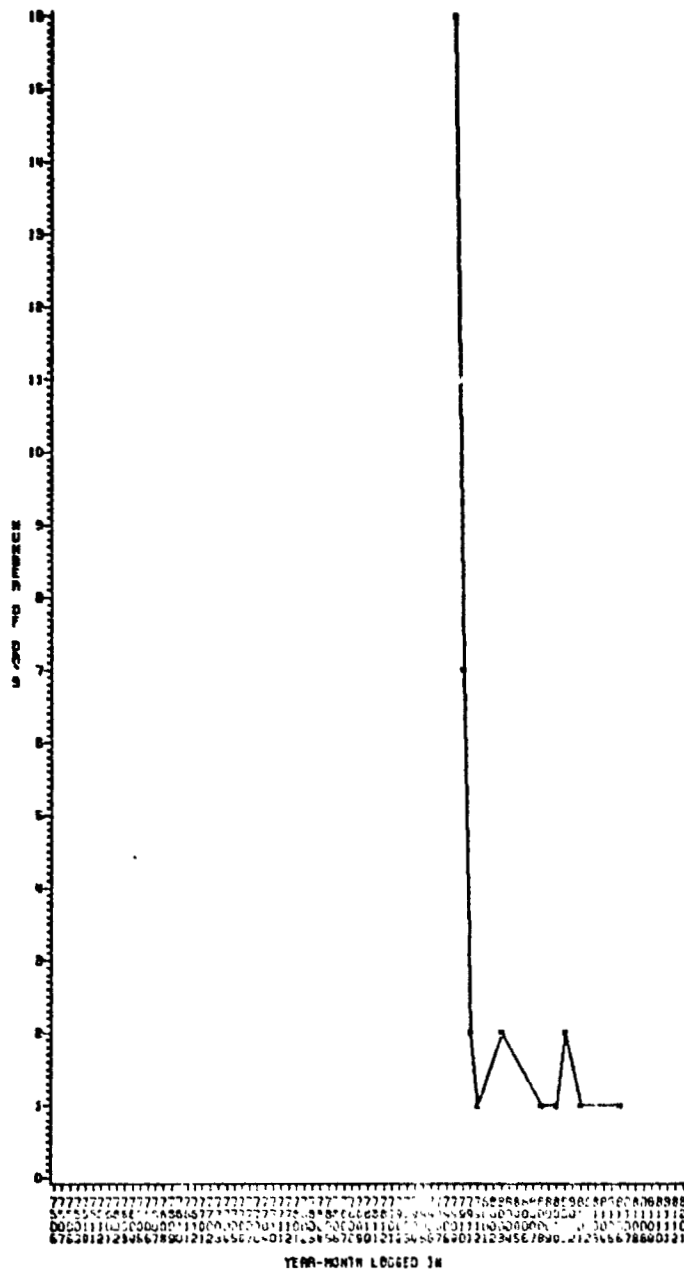


Figure 29: Number of DR's logged for site/test unique patch DR's (MISSN_ID=A).

ORIGINAL PAGE IS
OF POOR QUALITY.

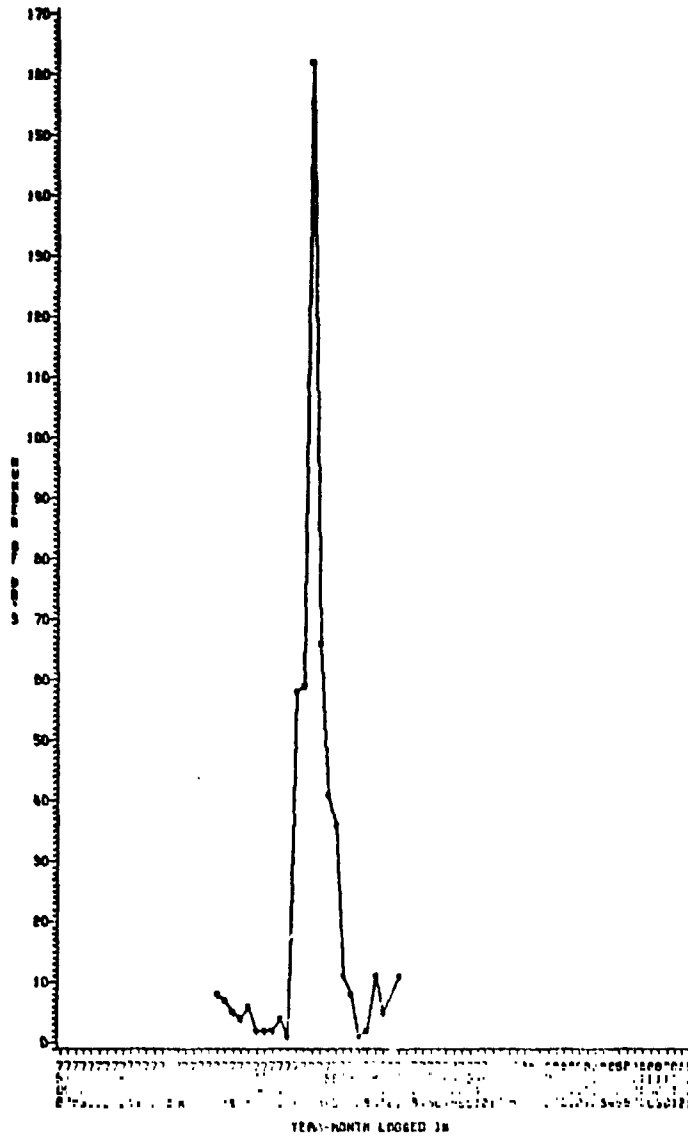


Figure 30: Number of DR's logged for PMD3 (MISSN_ID=B).

ORIGINAL PAGE IS
OF POOR QUALITY

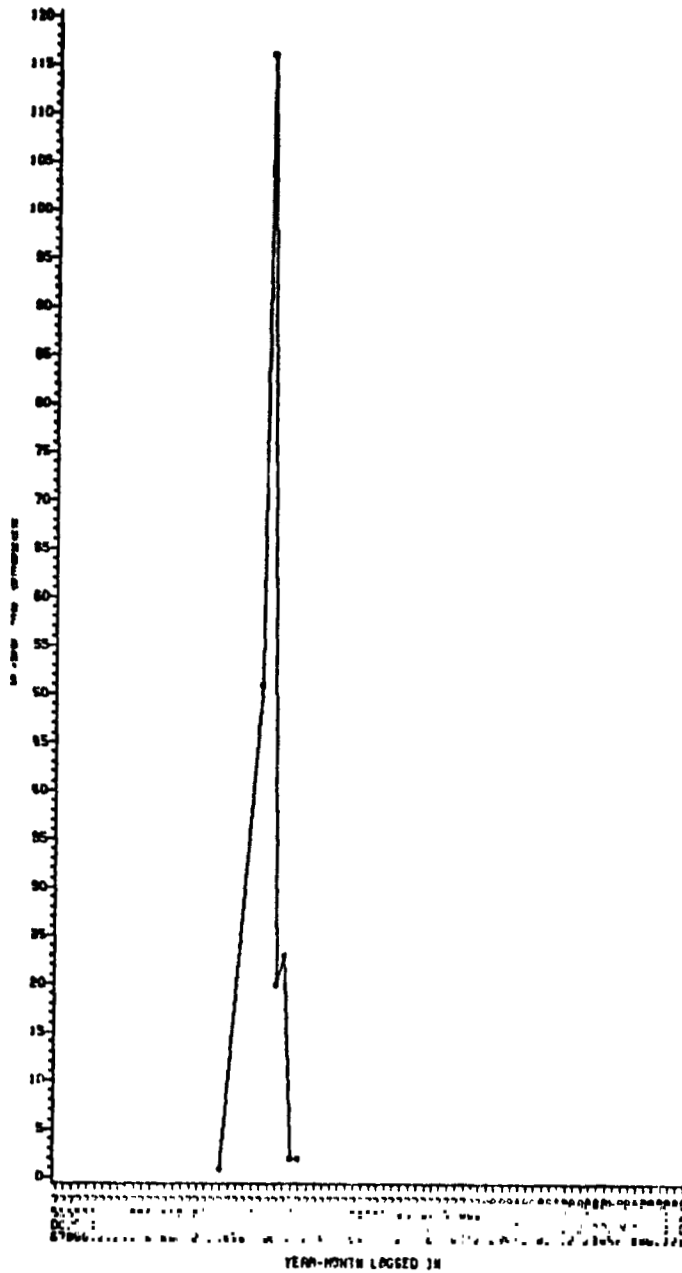


Figure 31: Number of DR's logged for ECL (MISSN_ID=C).

ORIGINAL PAGE IS
OF POOR QUALITY

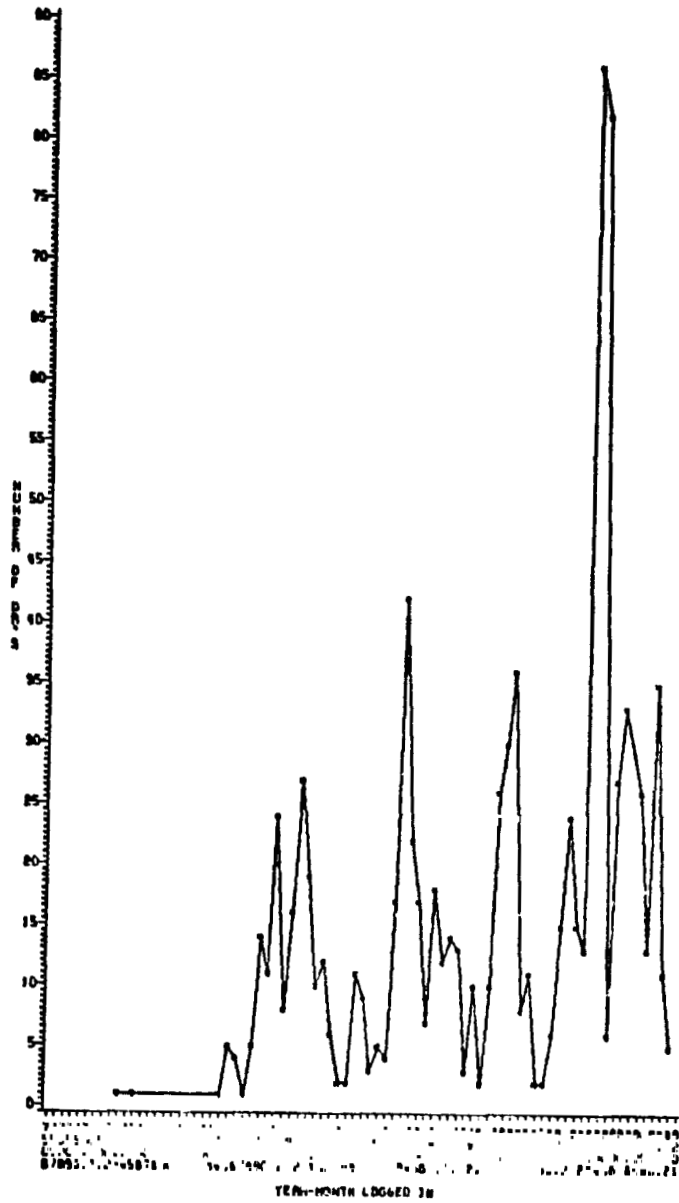


Figure 33: Number of DR's logged for JSC form 1541 (MISSN_ID=E).

ORIGINAL PAGE IS
OF POOR QUALITY

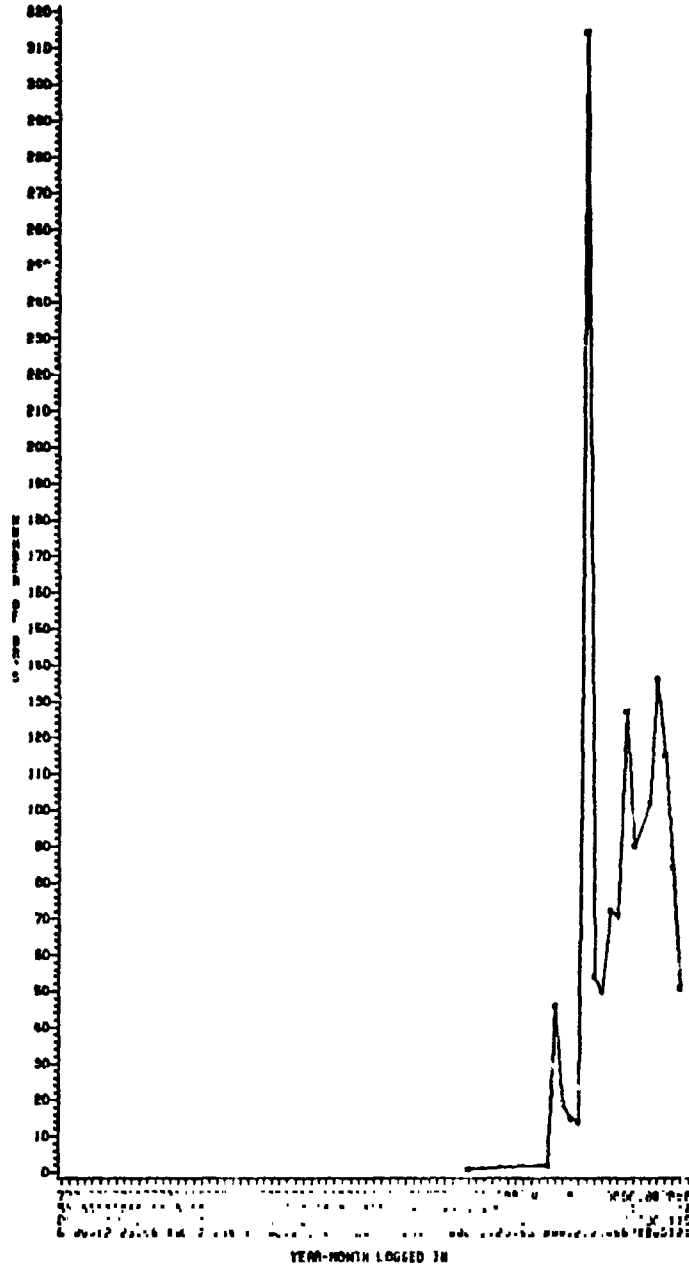


Figure 34: Number of DR's logged for release 19, floor system STS-5 (MISSN_ID=F).

ORIGINAL PAGE IS
OF POOR QUALITY.

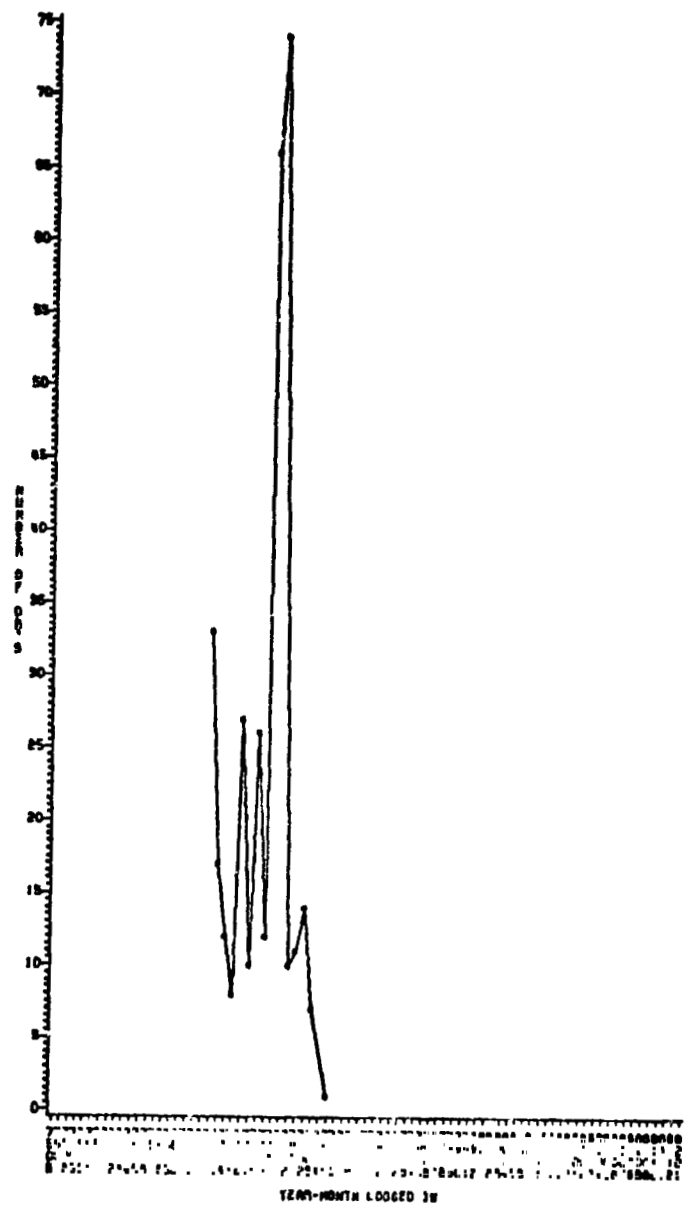


Figure 35: Number of DR's logged for PHD2 (MISSN_ID=G).

ORIGINAL PAGE IS
OF POOR QUALITY

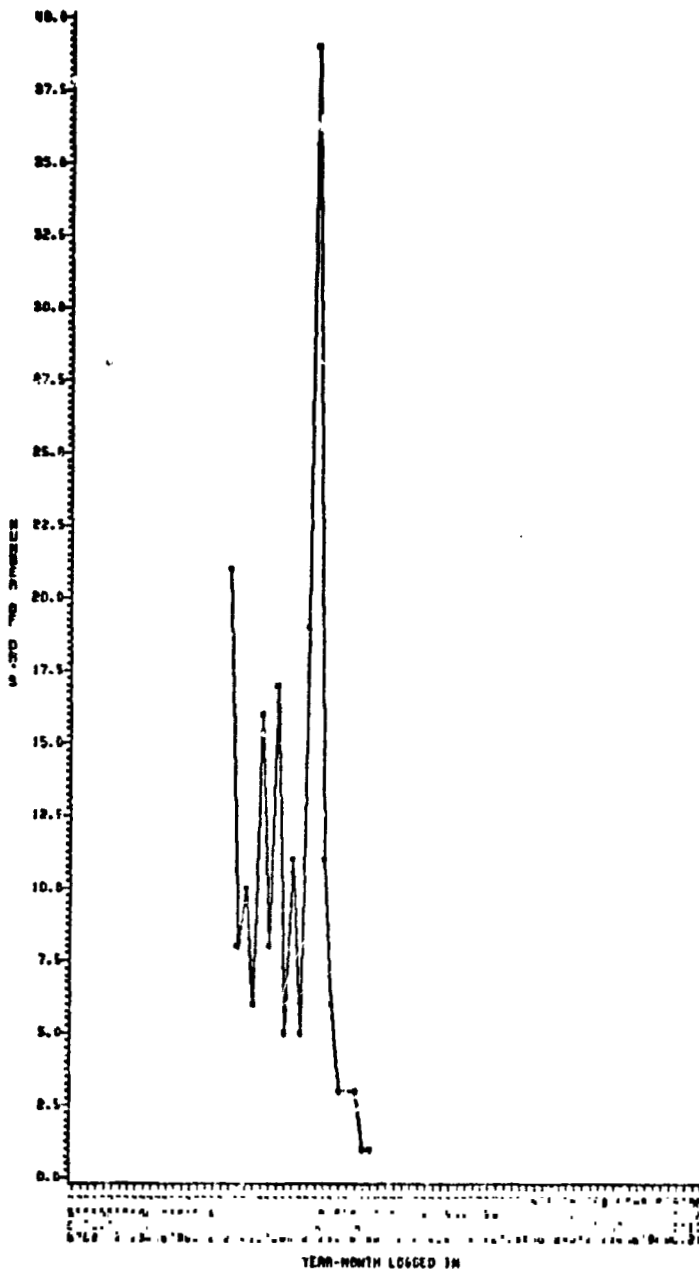


Figure 36: Number of DR's logged for AOA (MISSN_ID=H).

ORIGINAL PAGE IS
OF POOR QUALITY

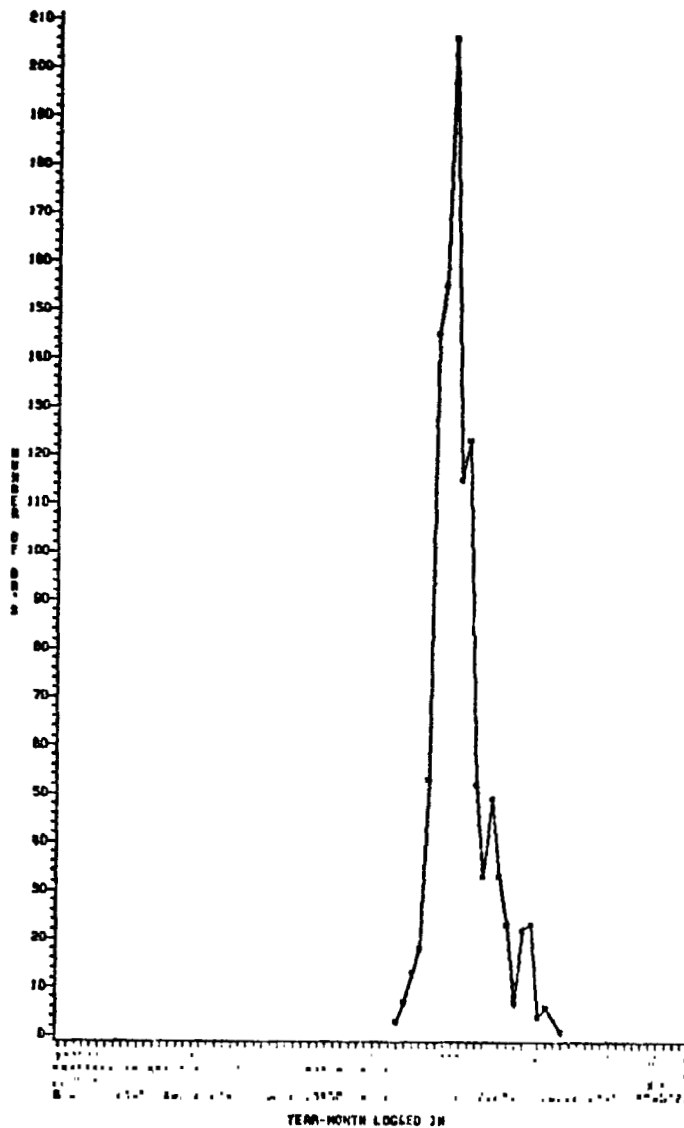


Figure 39: Number of DR's logged for STS-1, FUP4 (MISSN_ID=M).

ORIGINAL PAGE IS
OF POOR QUALITY

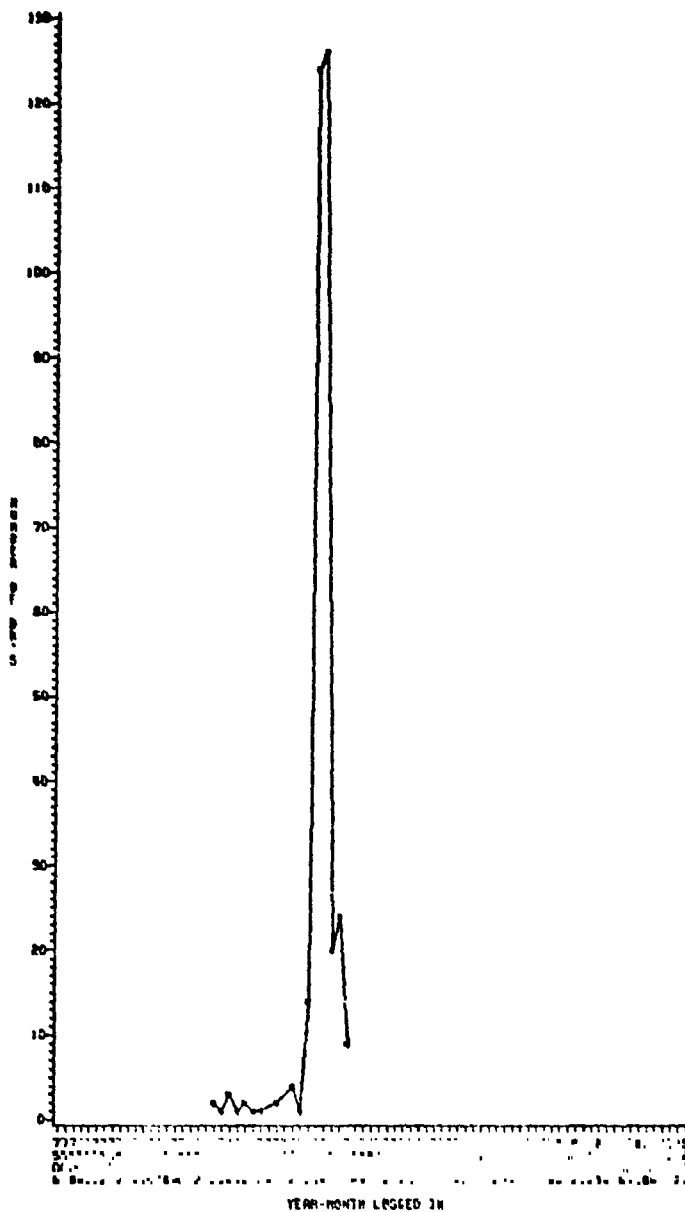


Figure 40: Number of DR's logged for entry (MISSN_ID=N).

ORIGINAL PAGE IS
OF POOR QUALITY

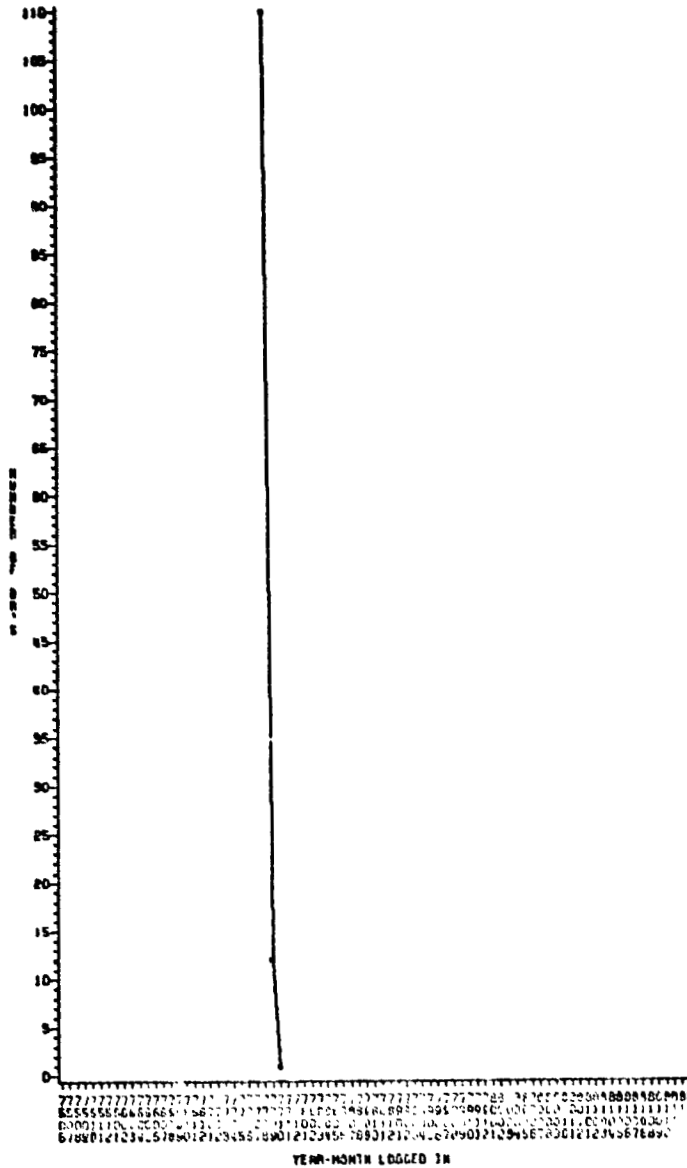


Figure 41: Number of DR's logged for PMD1 (MISSN_ID=P).

ORIGINAL PAGE
OF POOR QUALITY

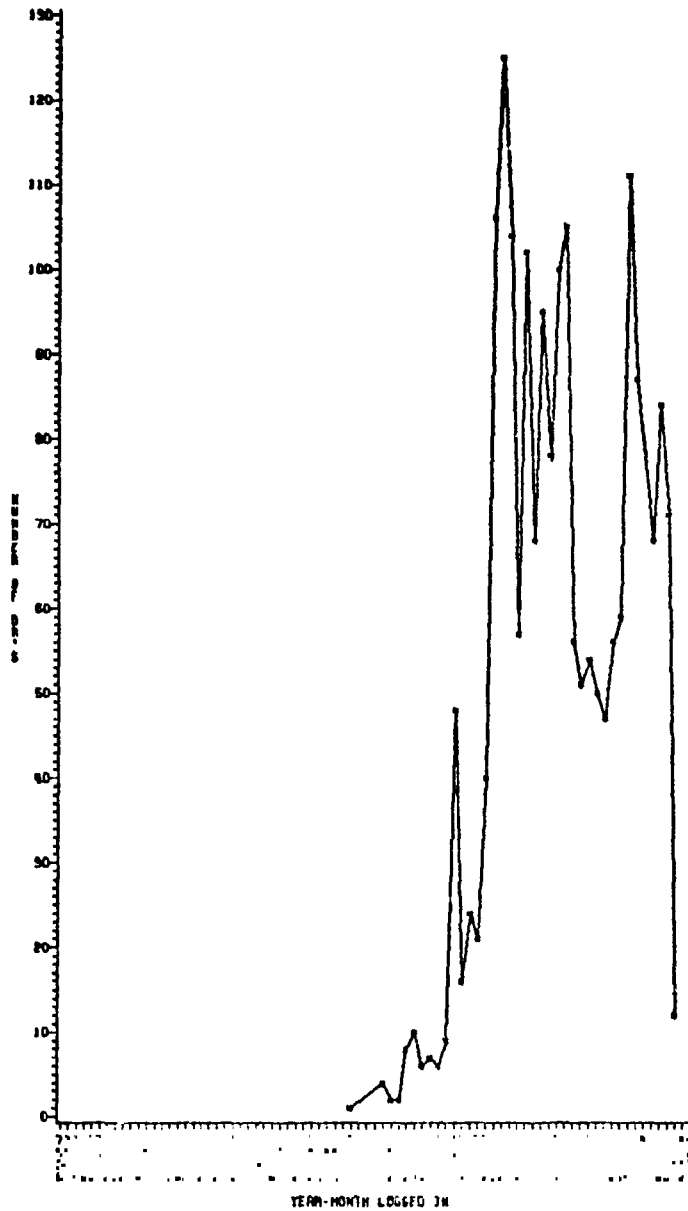


Figure 42: Number of DR's logged for STS-2, R18V21 flight system (MISSN_ID=R).

ORIGINAL PAGE IS
OF POOR QUALITY

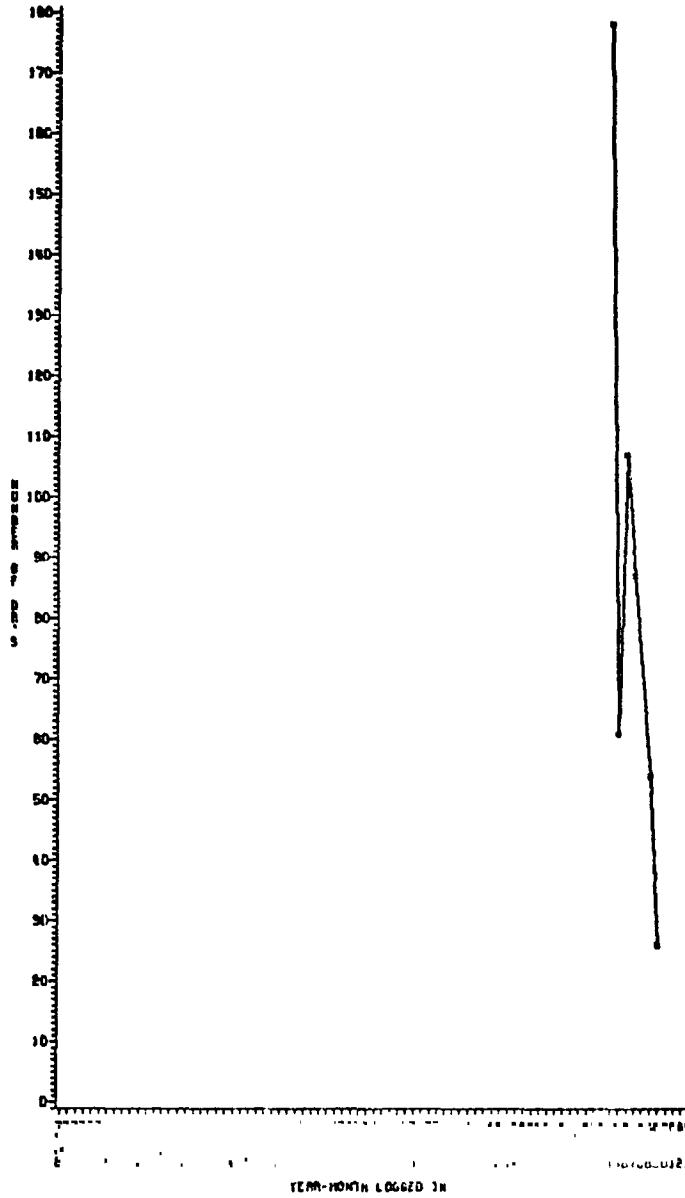


Figure 44: Number of DR's logged for STS-2, R18V12 field system (MISSN_ID=R2).

ORIGINAL PAGE IS
OF POOR QUALITY

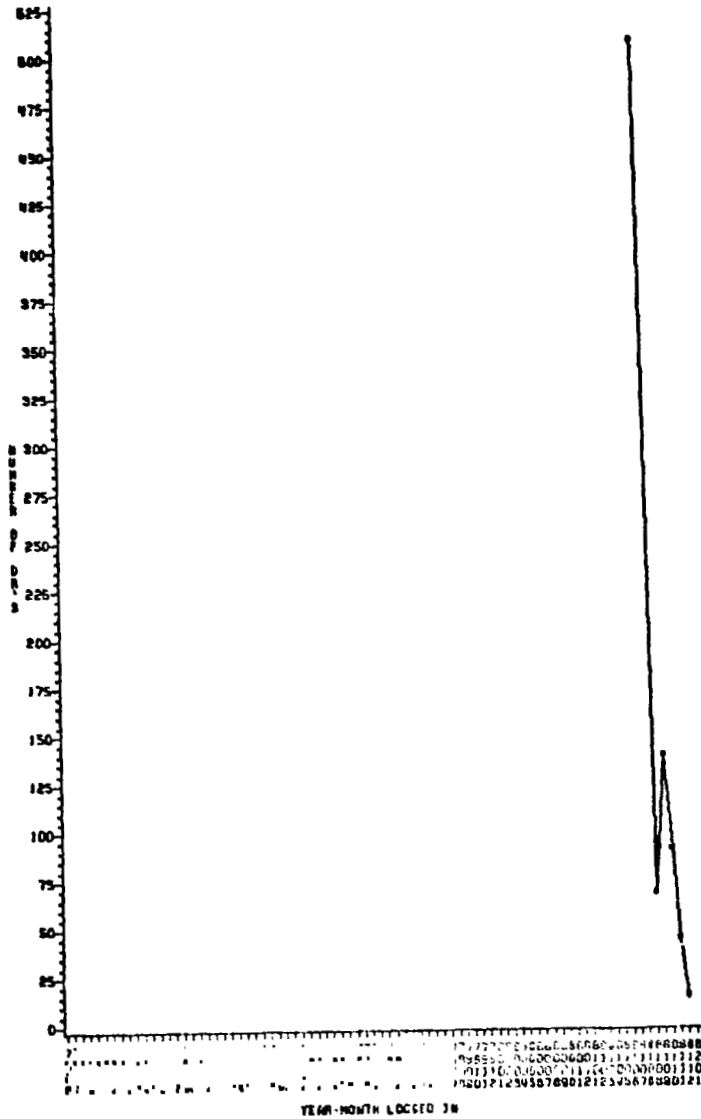


Figure 45: Number of DR's logged for STS-3, R18V30 flight system (MISSION_ID=R3).

ORIGINAL PAGE IS
OF POOR QUALITY

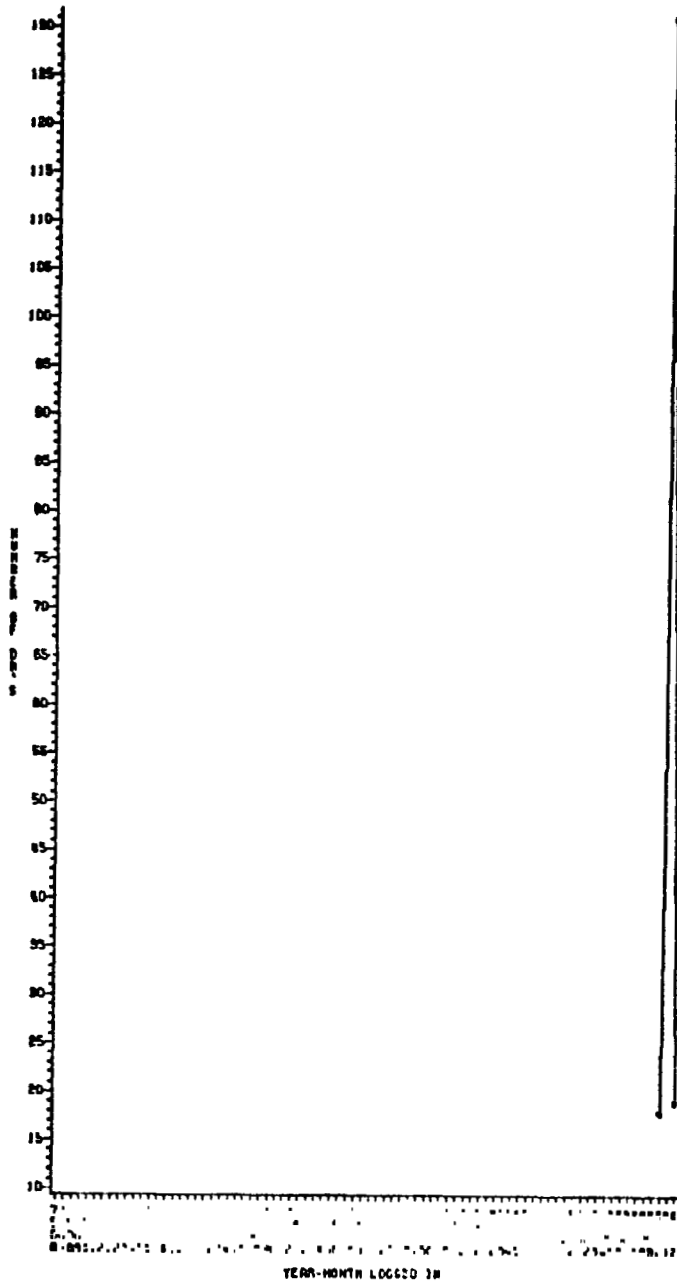


Figure 46: Number of DR's logged for STS-4, R18V45 flight system (MISSN_ID=R4).

ORIGINAL PAGE IS
OF POOR QUALITY

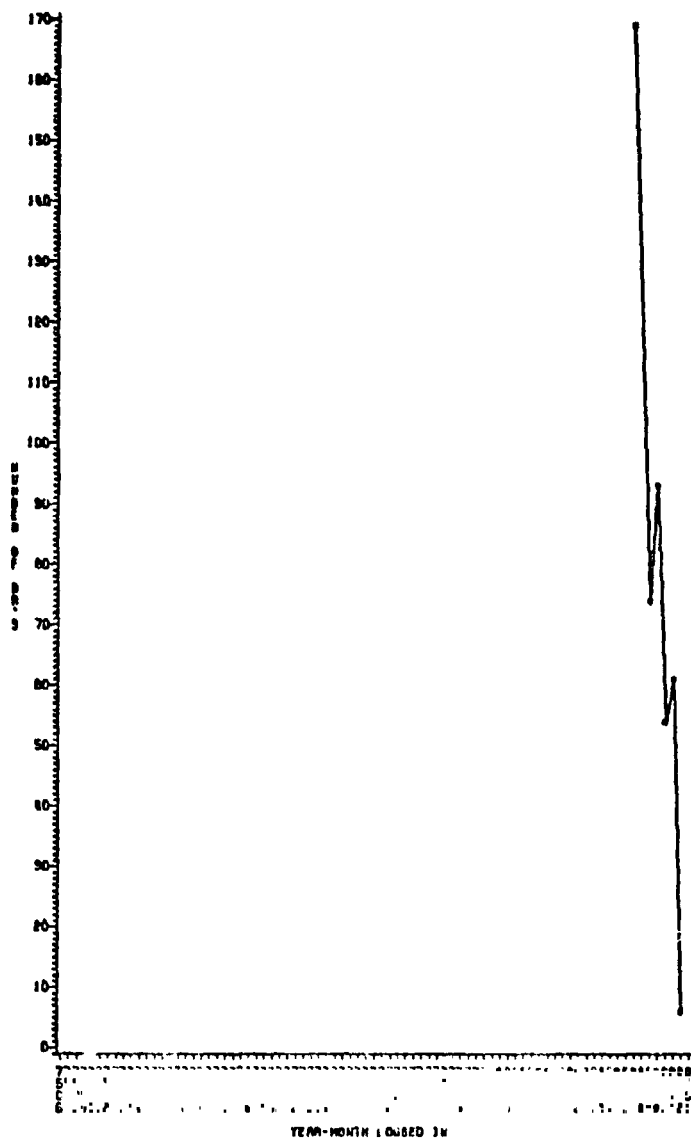


Figure 47: Number of DR's logged for OV-99, R18V90 field Ssstem (MISSN_ID=R9).

ORIGINAL PAGE IS
OF POOR QUALITY

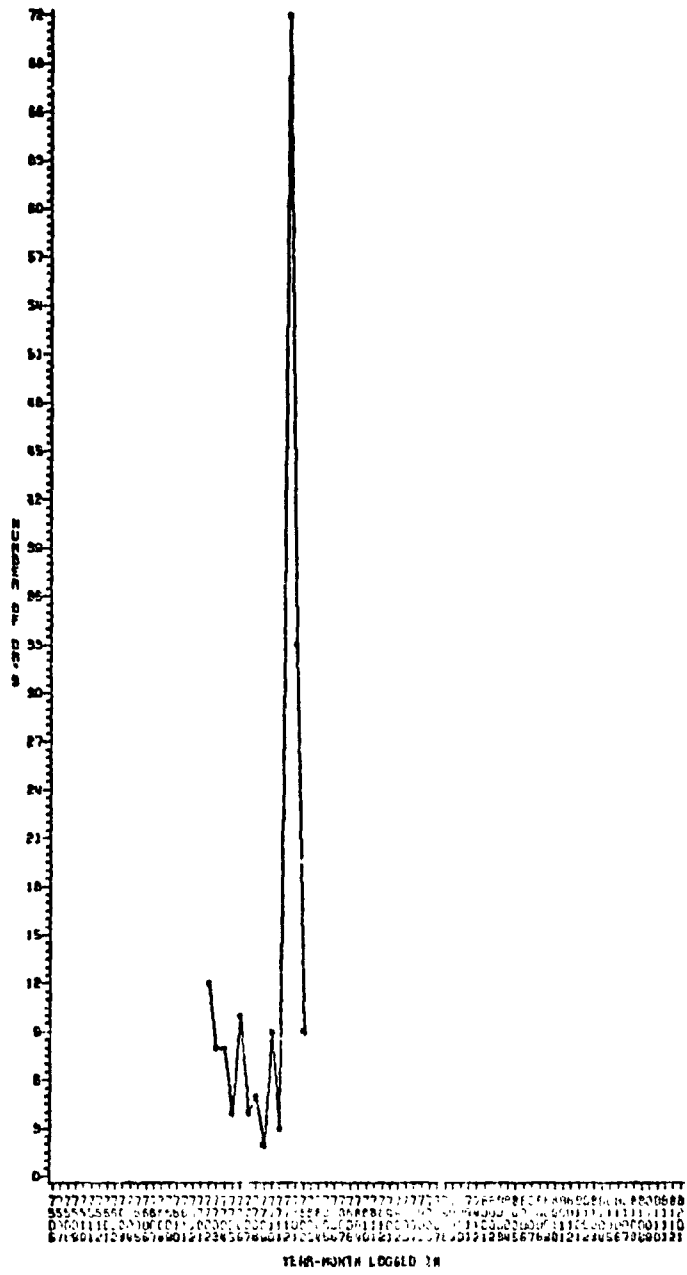


Figure 48: Number of DR's logged for SSW2 (MISSN_ID=S).

ORIGINAL PAGE IS
OF POOR QUALITY

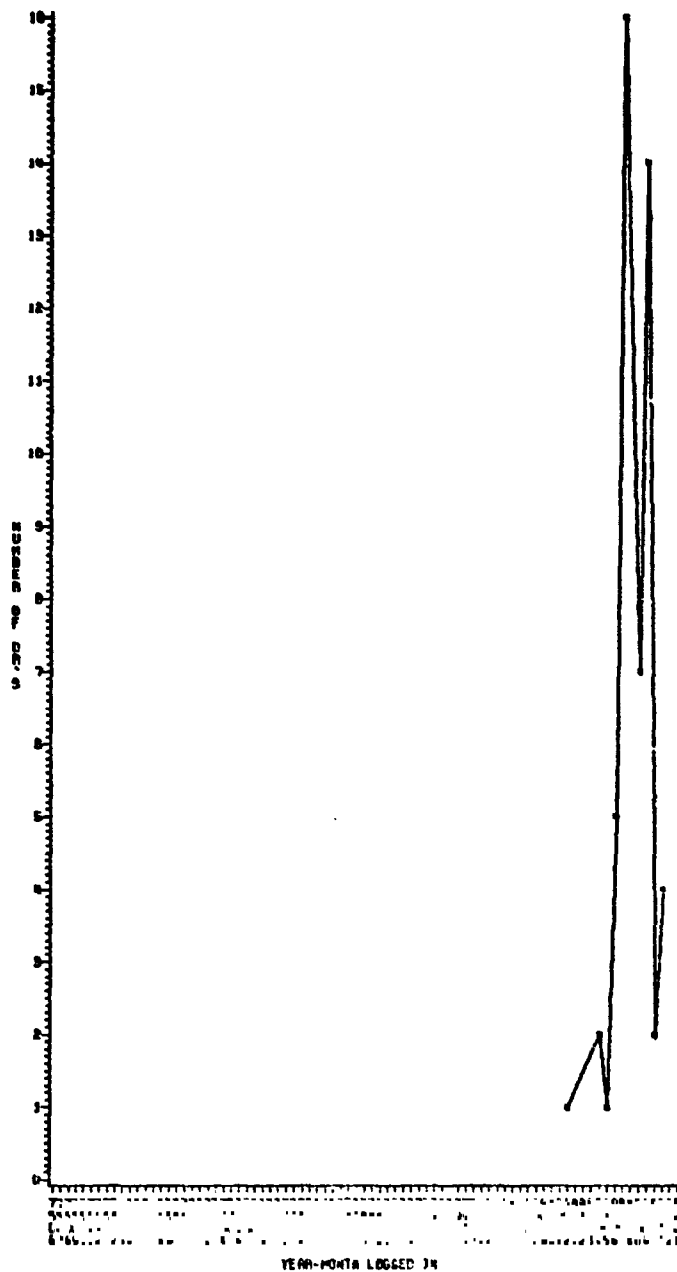


Figure 51: Number of DR's logged for SDL, release 37 (MISSN_ID=S7).

ORIGINAL PAGE IS
OF POOR QUALITY

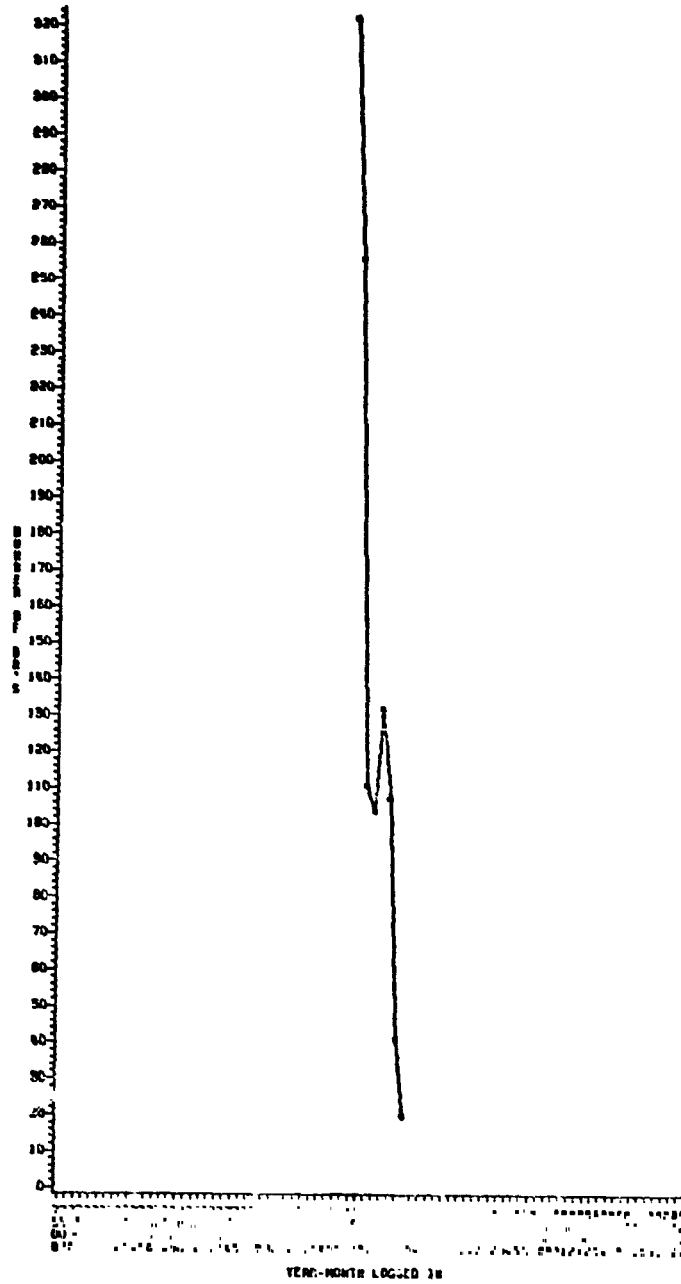


Figure 52: Number of DR's logged for OFT1 (MISSN_ID=T).

ORIGINAL PAGE IS
OF POOR QUALITY

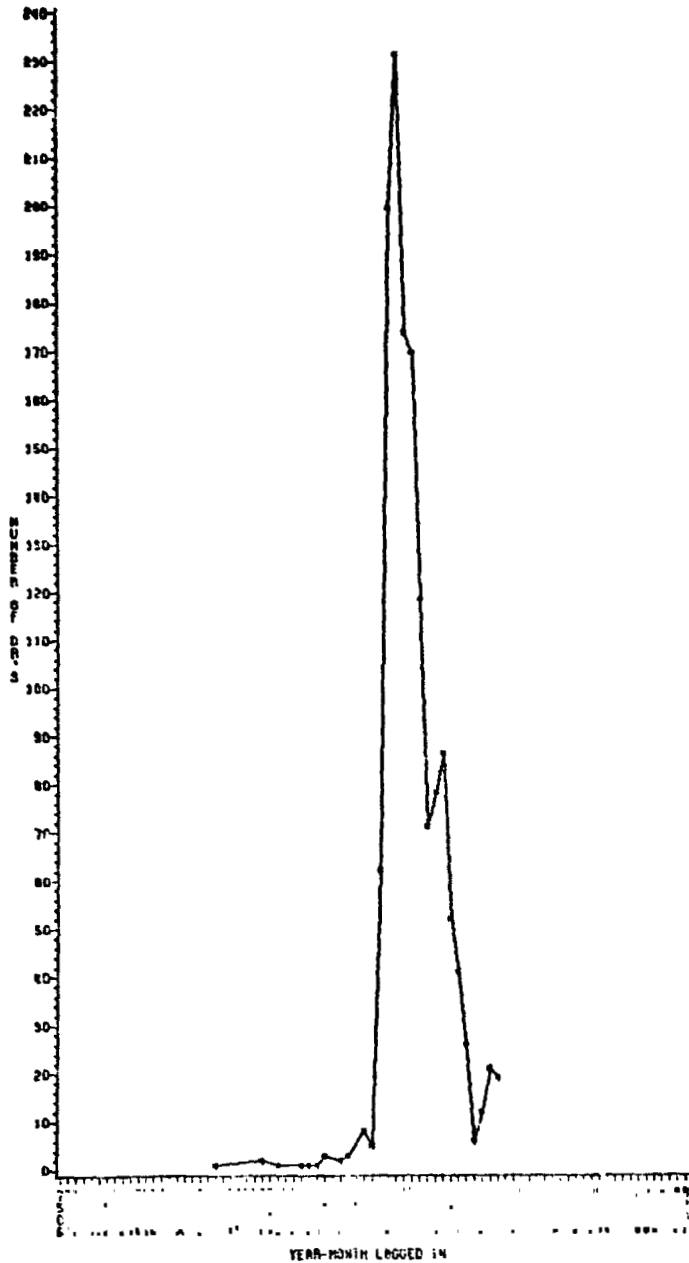


Figure 55: Number of DR's logged for STS-1, FUP1 (MISSN_ID=X).

ORIGINAL PAGE IS
OF POOR QUALITY

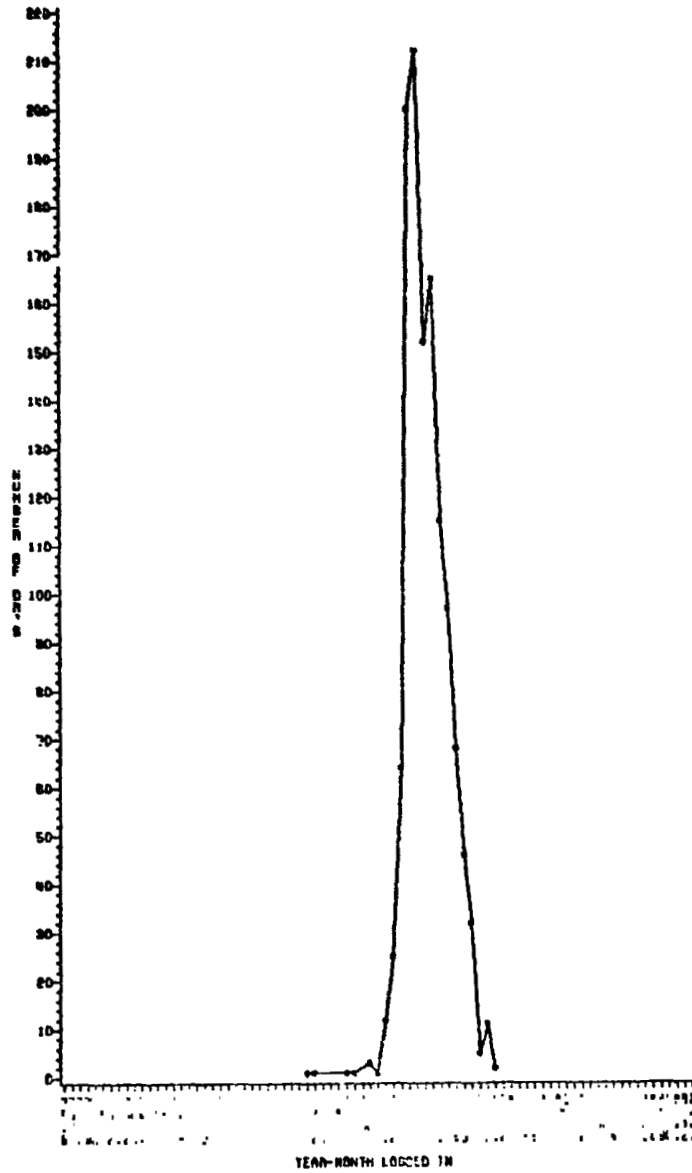


Figure 56: Number of DR's logged for STS-1, FUF2 (MISSN_ID=Y).

ORIGINAL PAGE IS
OF POOR QUALITY

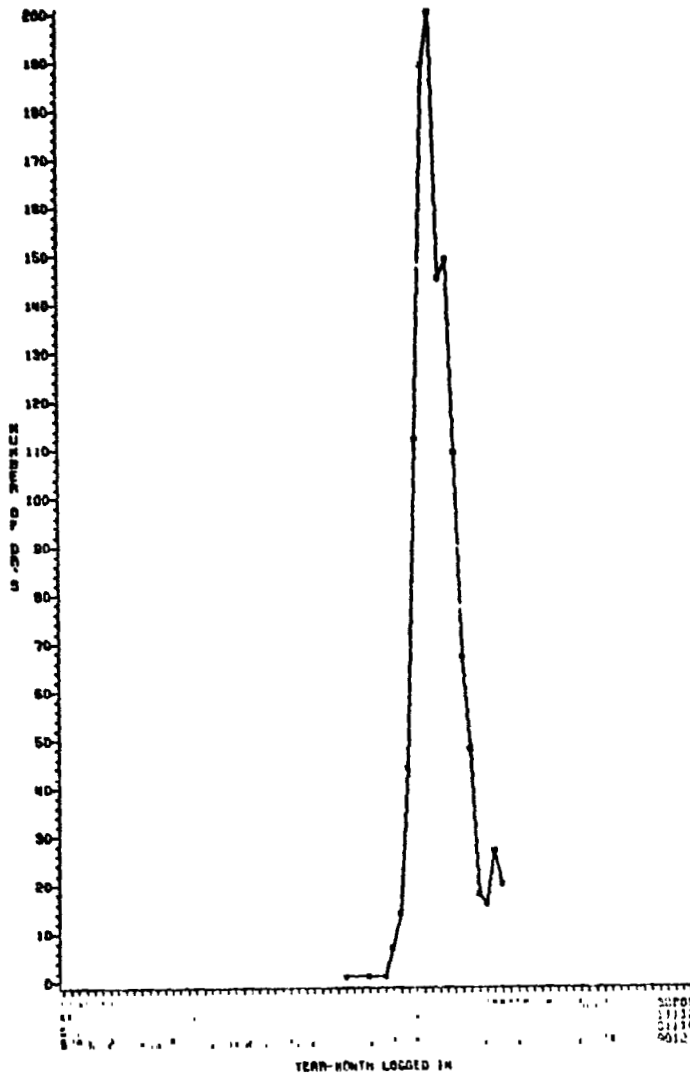


Figure 57: Number of DR's logged for STS-1, FUP3 (MISSN_ID=Z).

ORIGINAL PAGE IS
OF POOR QUALITY

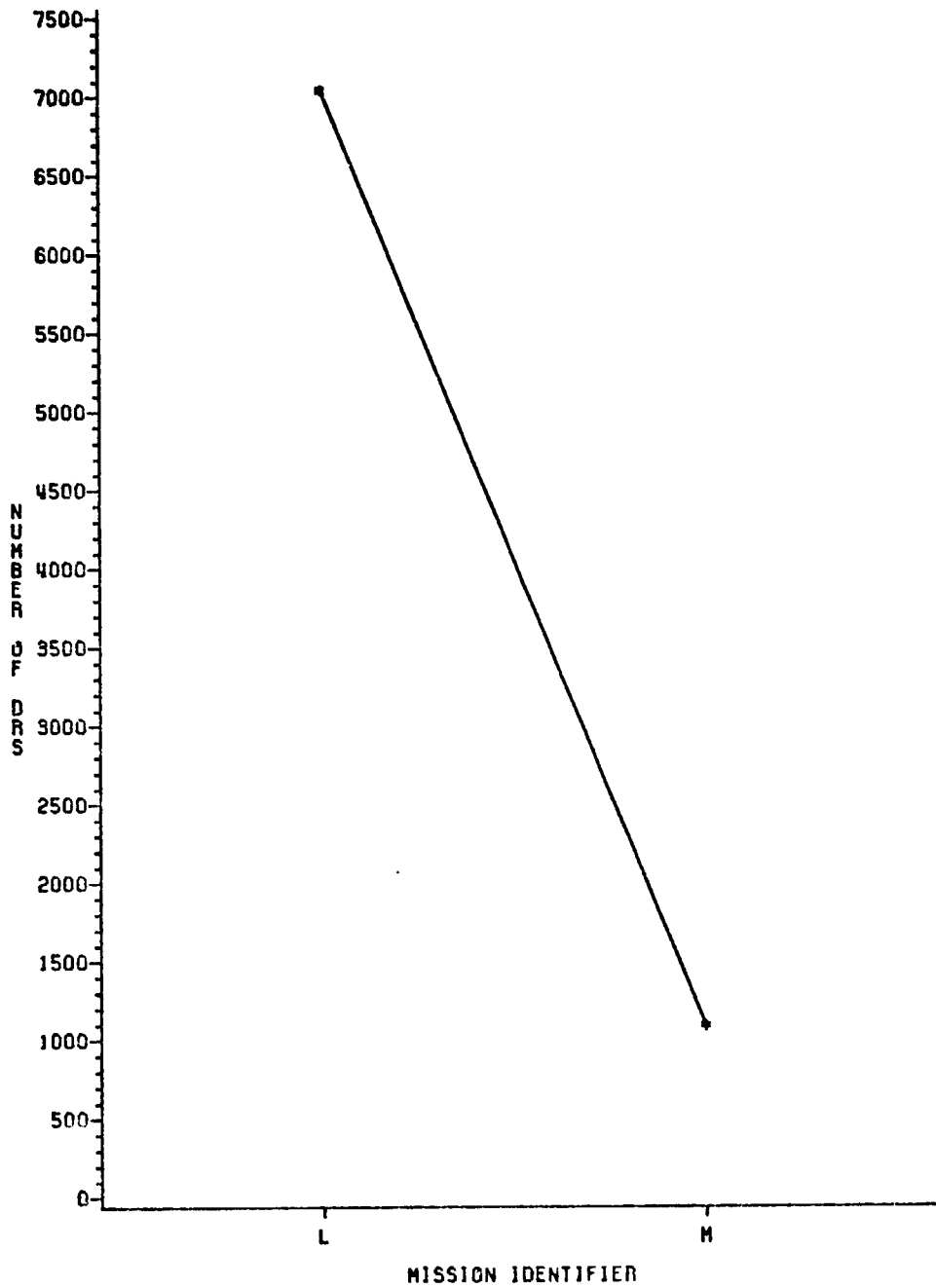


Figure 58: Total number of DR's logged for STS-1, FUP5/FLT (MISSN_ID=L) and STS-1, FUP4 (MISSN_ID=M).

ORIGINAL PAGE IS
OF POOR QUALITY

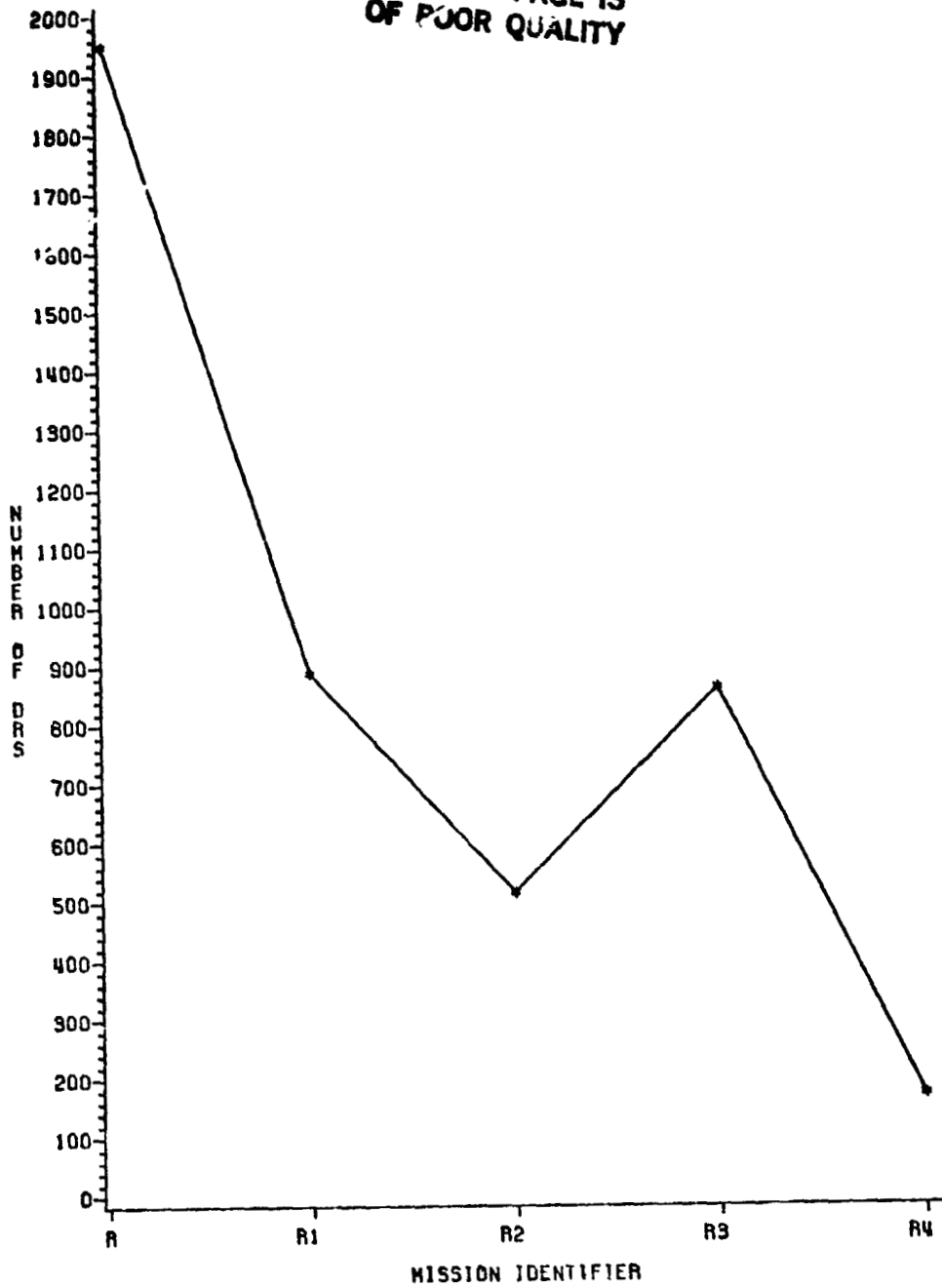


Figure 59. Total number of DR's logged for STS-2 (MISSN_ID=R), STS-2 (MISSN_ID=R1), STS-2 (MISSN_ID=R2), STS-3 (MISSN_ID=R3), and STS-4 (MISSN_ID=R4).

ORIGINAL PAGE IS
OF POOR QUALITY

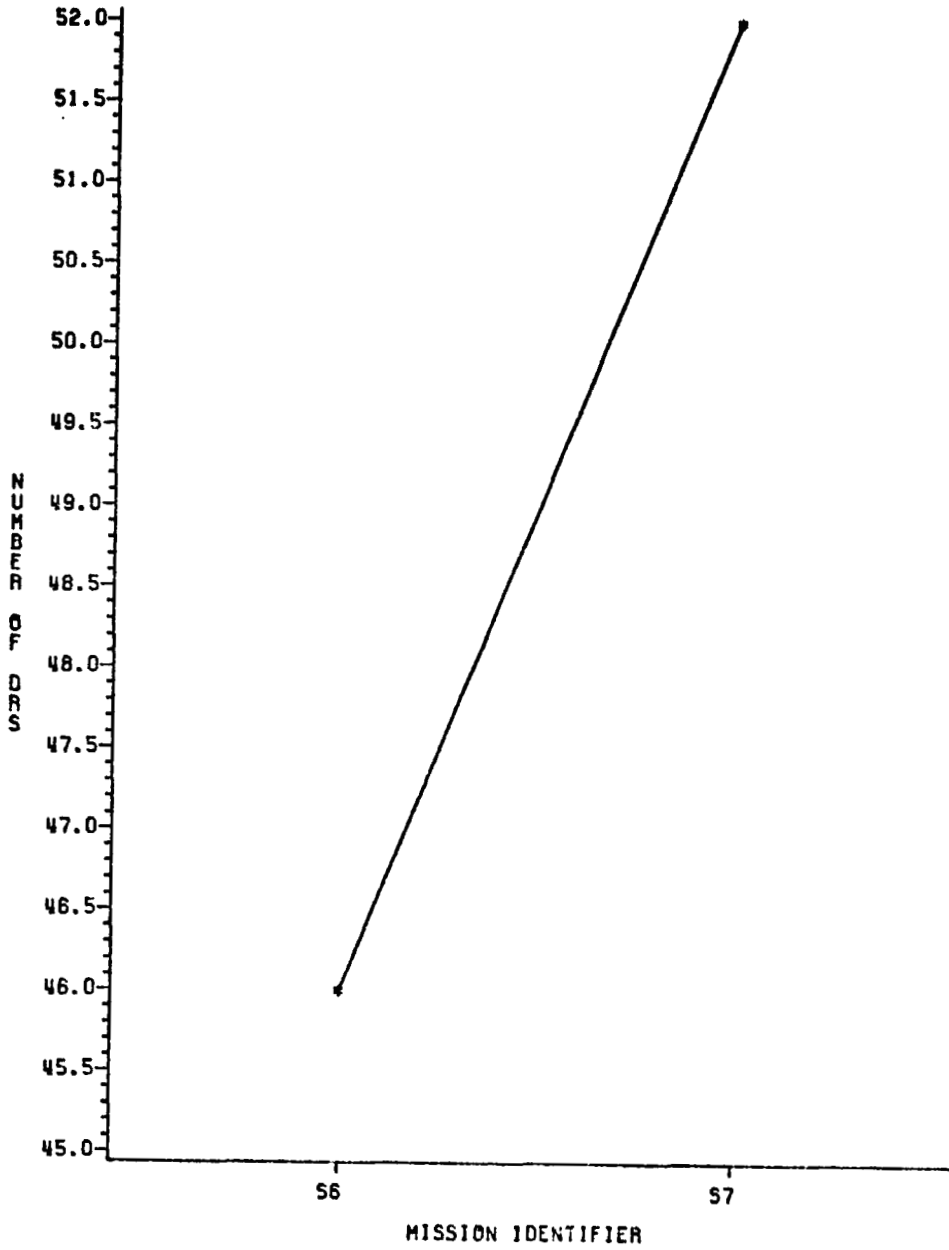


Figure 60: Total number of DR's logged for SDL releases 36 and 37 (MISSION_ID=S6 and S7).

ORIGINAL PAGE IS
OF POOR QUALITY

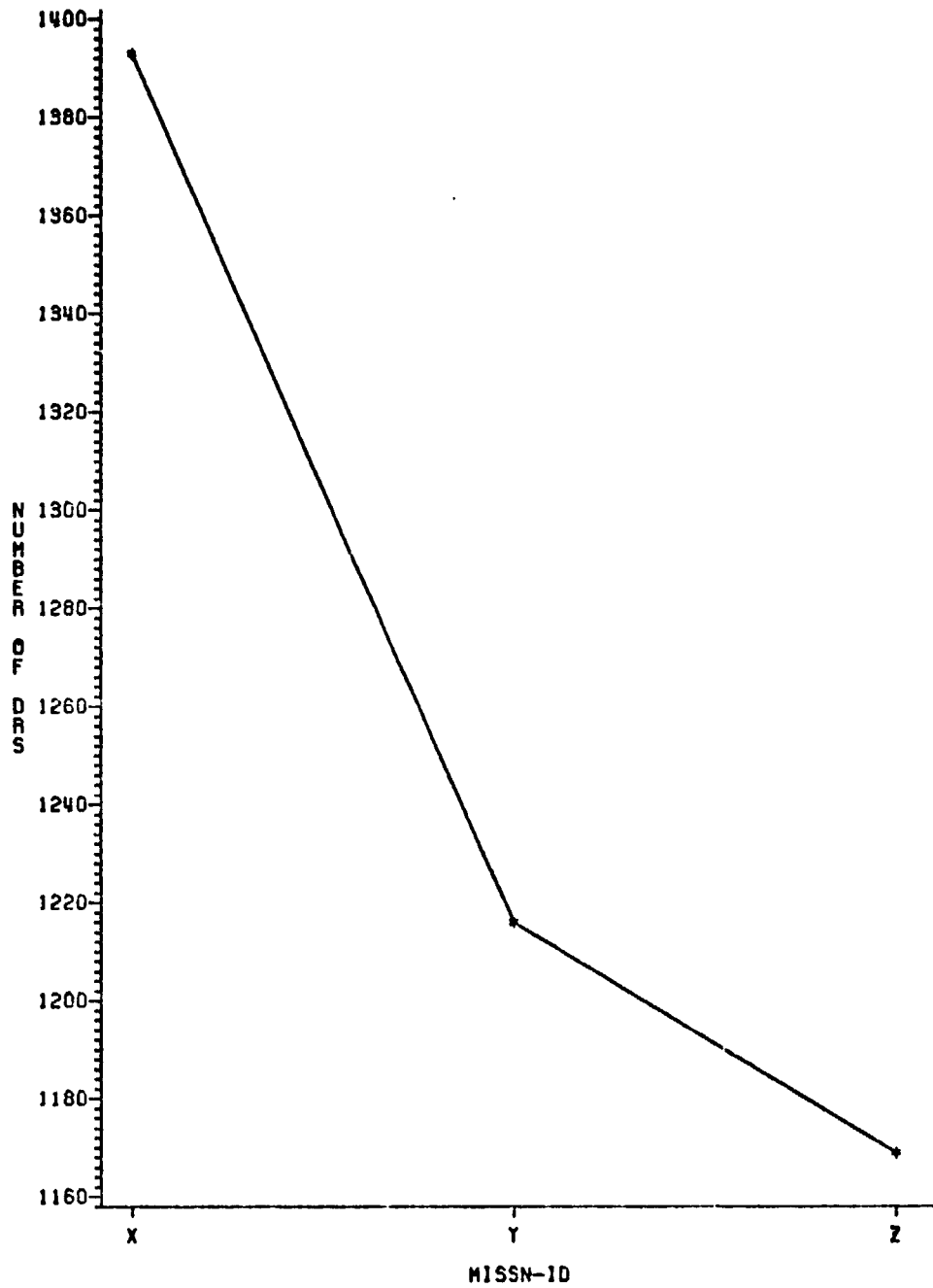


Figure 61: Total number of DR's logged for STS-1: FUP1 (MISSN_ID=X), FUP2 (MISSN_ID=Y), and FUP3 (MISSN_ID=Z).

ORIGINAL PAGE IS
OF POOR QUALITY

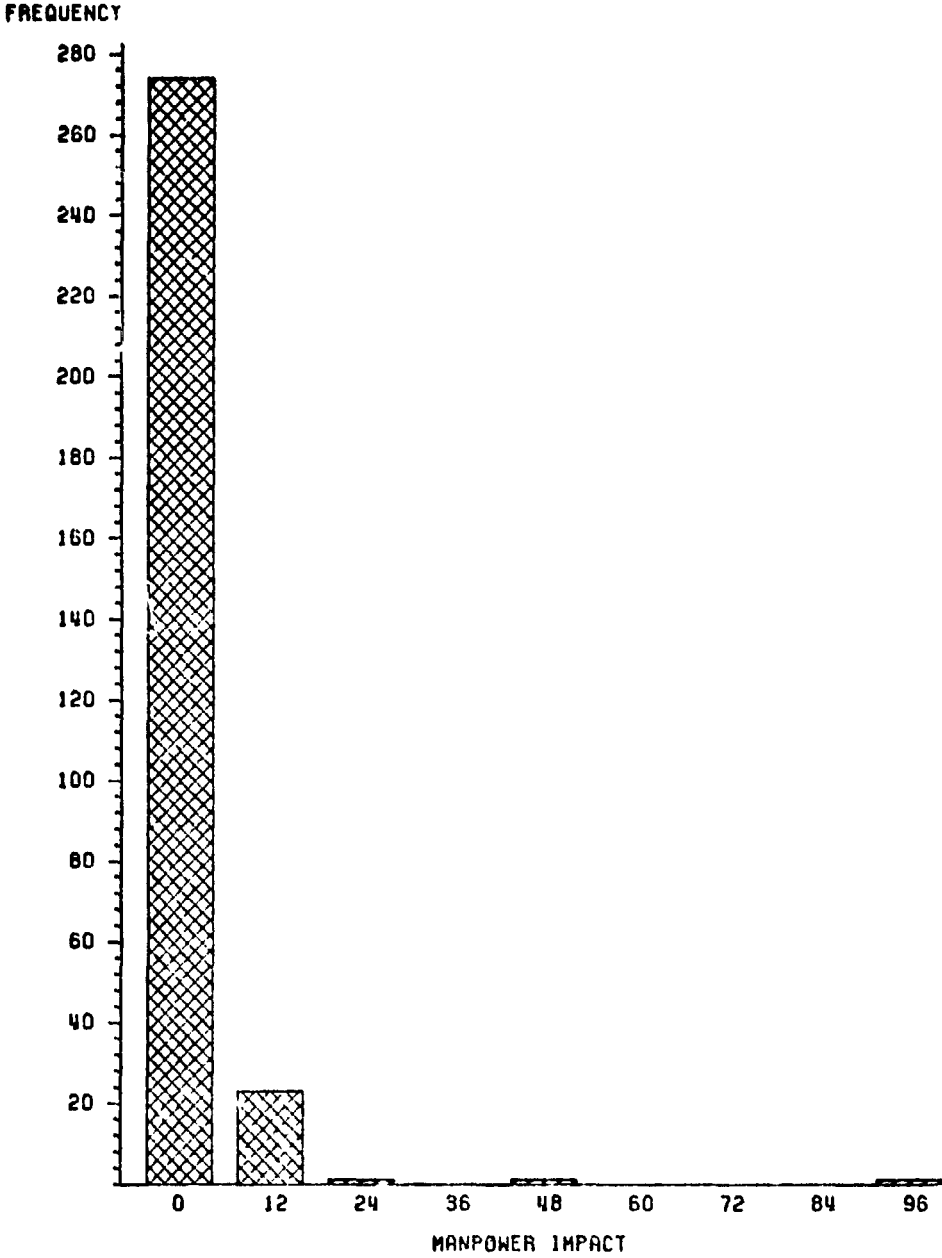


Figure 62: CR frequency for department 1 manpower impact, year 1 (DIIMPYR1).

ORIGINAL PAGE IS
OF POOR QUALITY



Figure 63: CR frequency for department 1 manpower impact, year 2 (D1IMPYR2).

ORIGINAL PAGE IS
OF POOR QUALITY

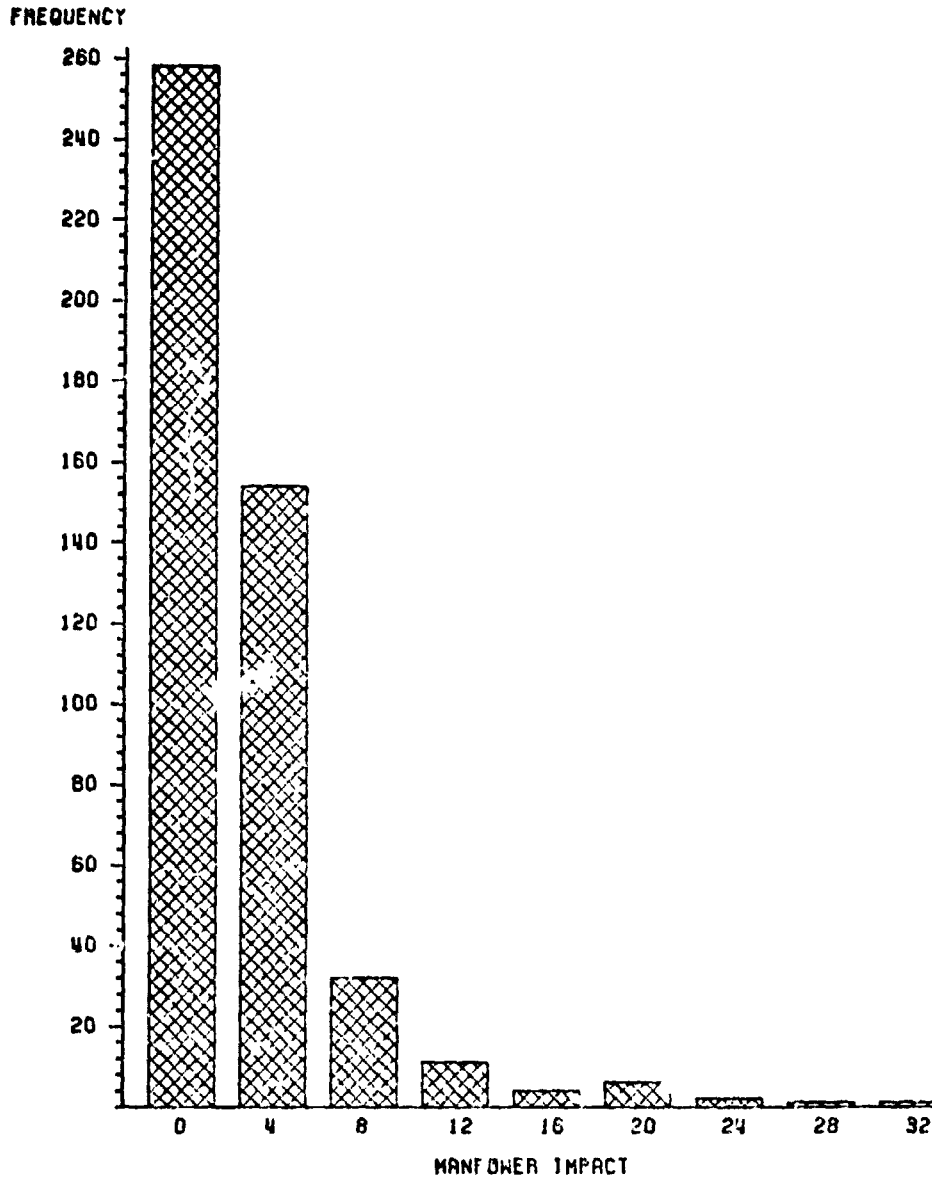


Figure 64: CR frequency for department 1 manpower impact, year 3 (D1IMPYR3).

ORIGINAL PAGE IS
OF POOR QUALITY

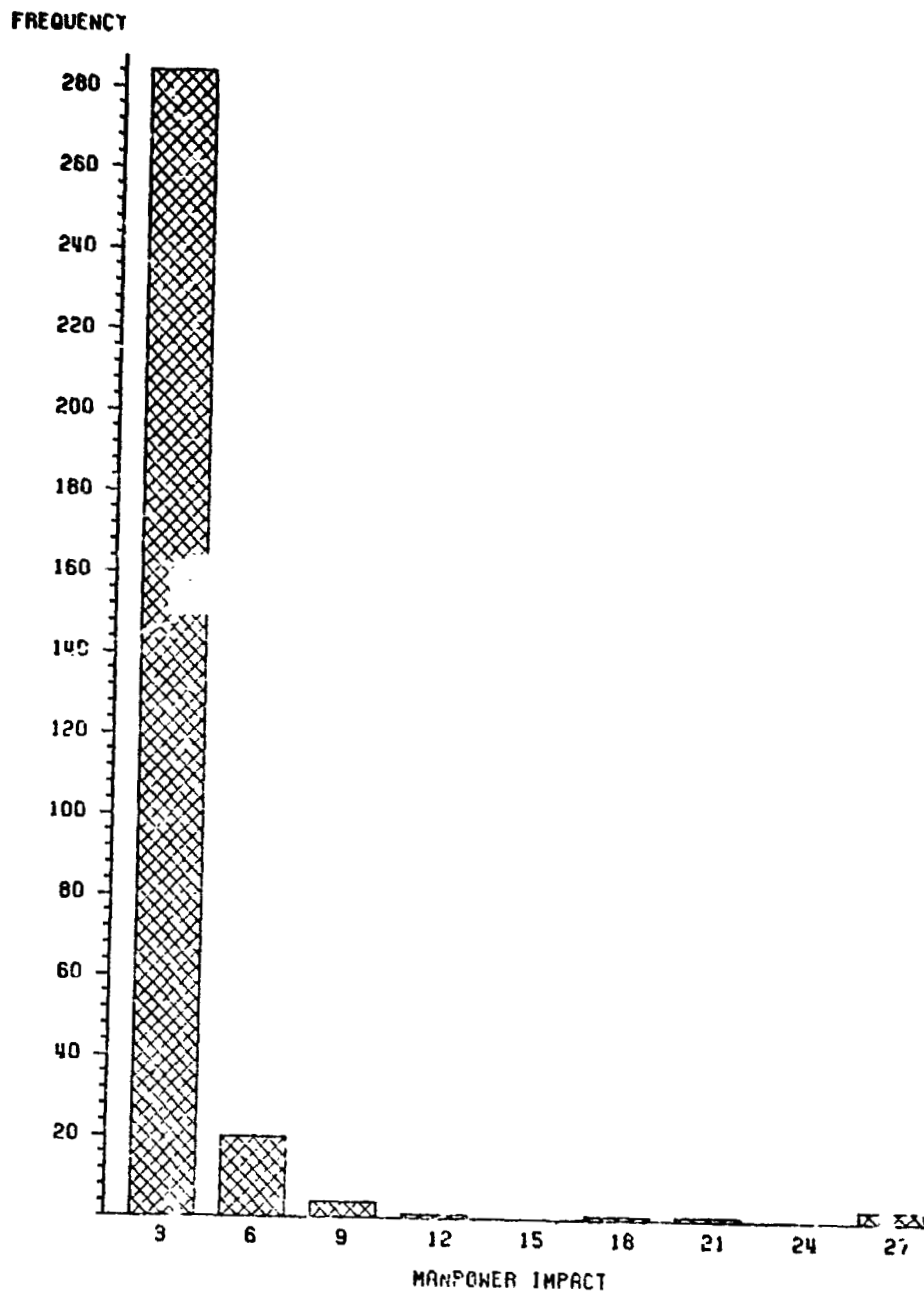


Figure 65: CR frequency for department 1 manpower impact, year 4
(D:EMPYR4).

ORIGINAL PAGE IS
OF POOR QUALITY

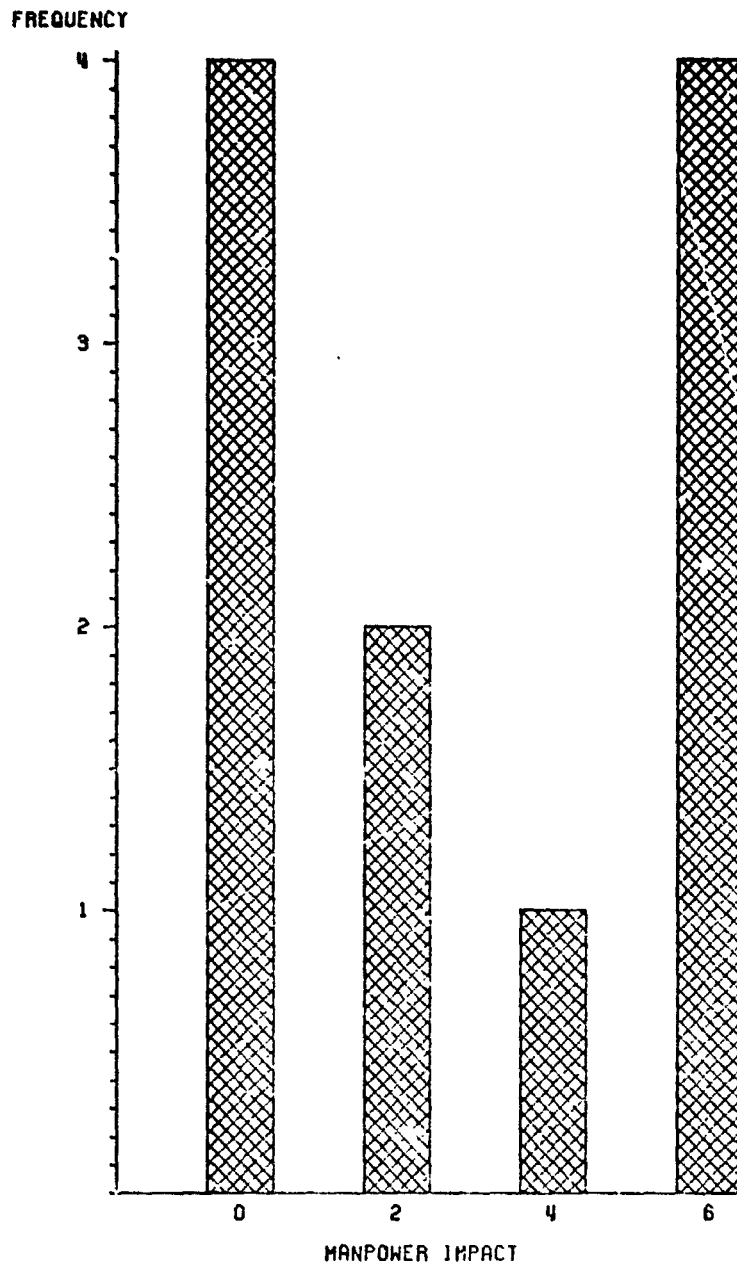


Figure 66: CR frequency for department 2 manpower impact, year 1 (D?IMPYR1).

ORIGINAL PAGE IS
OF POOR QUALITY.

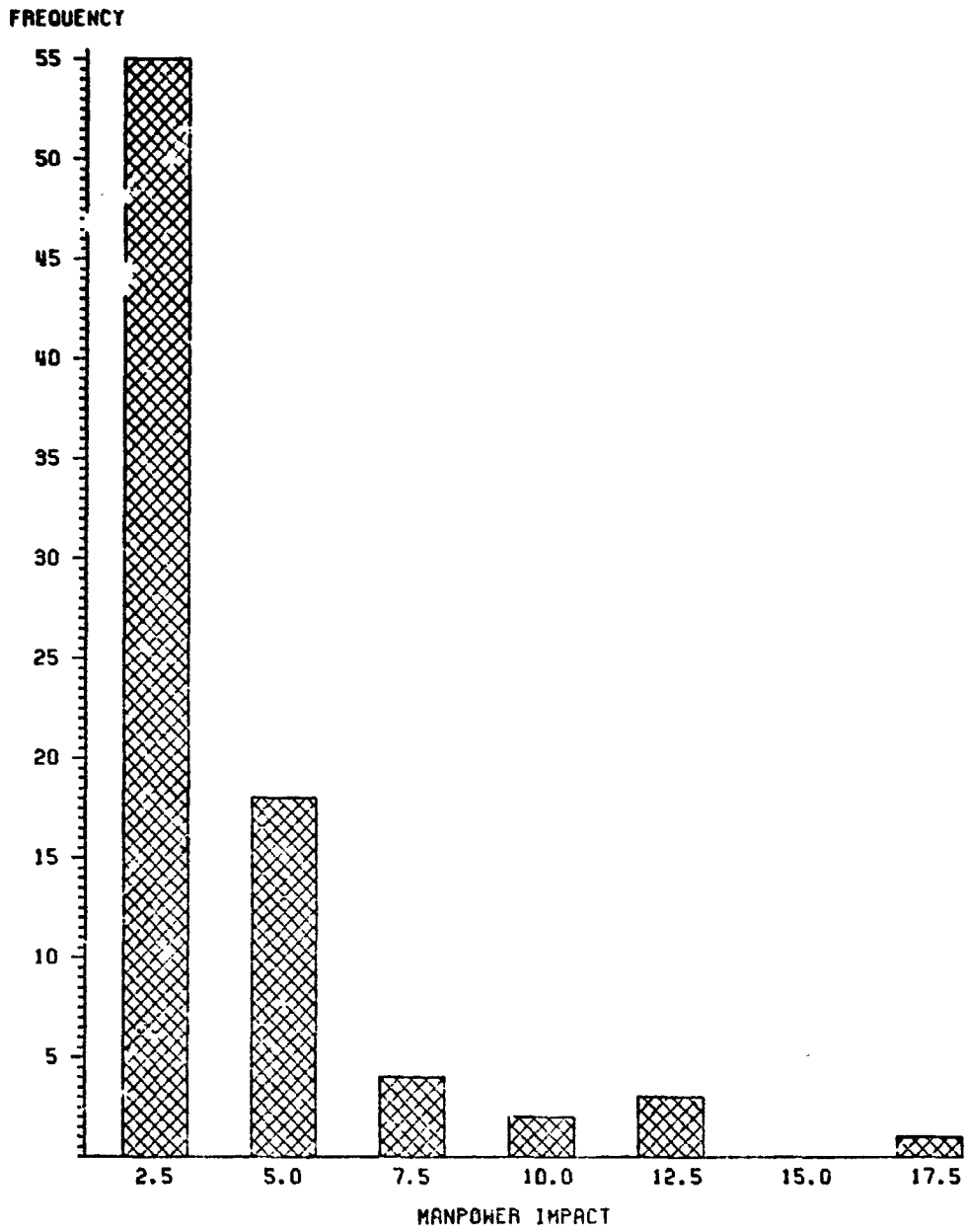


Figure 67: CR frequency for department 2 manpower impact, year 2 (D21MPYR2).

ORIGINAL PAGE IS
OF POOR QUALITY

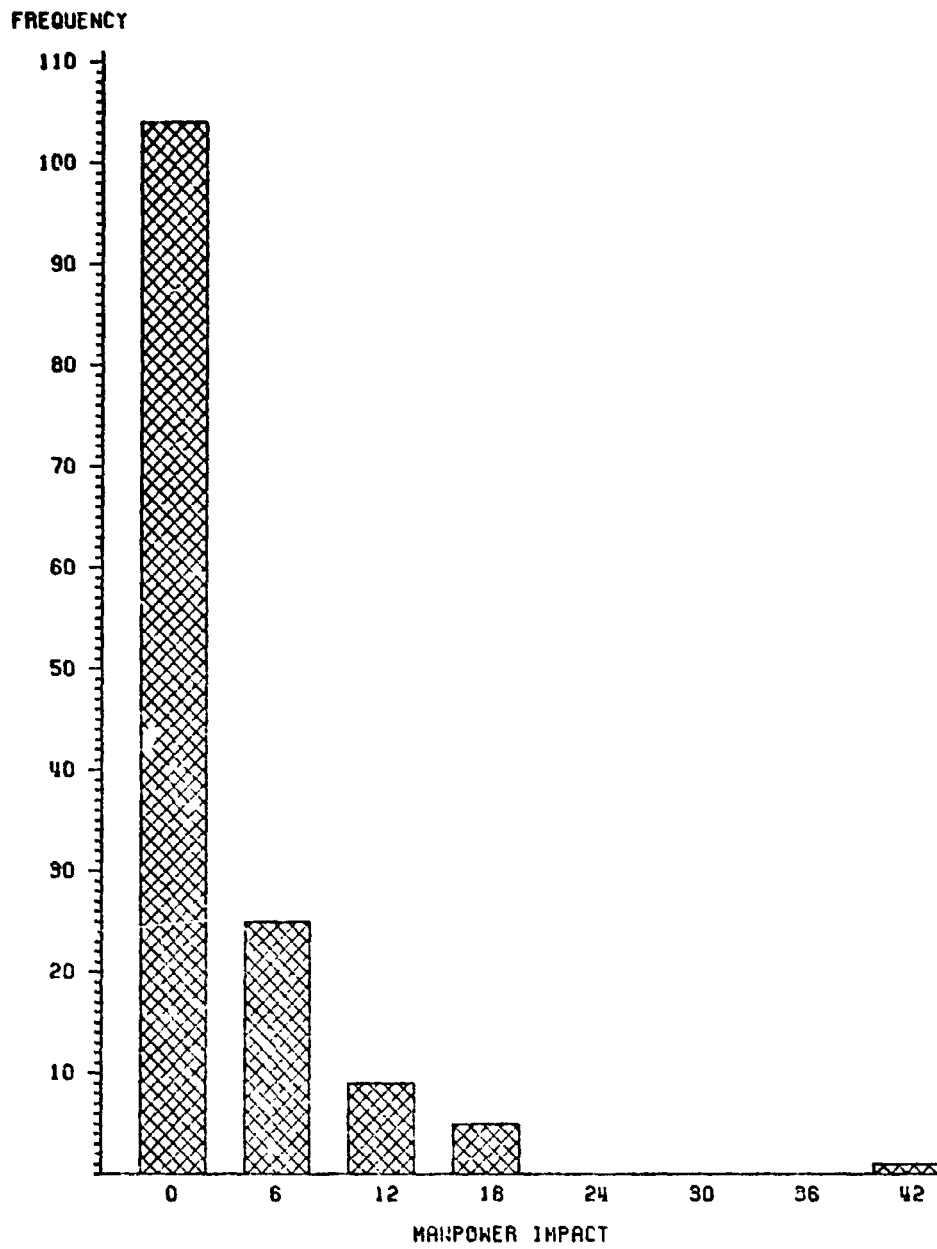


Figure 68: CR frequency for department 2 manpower impact, year 3 (D2IMPYR3).

ORIGINAL PAGE IS
OF POOR QUALITY

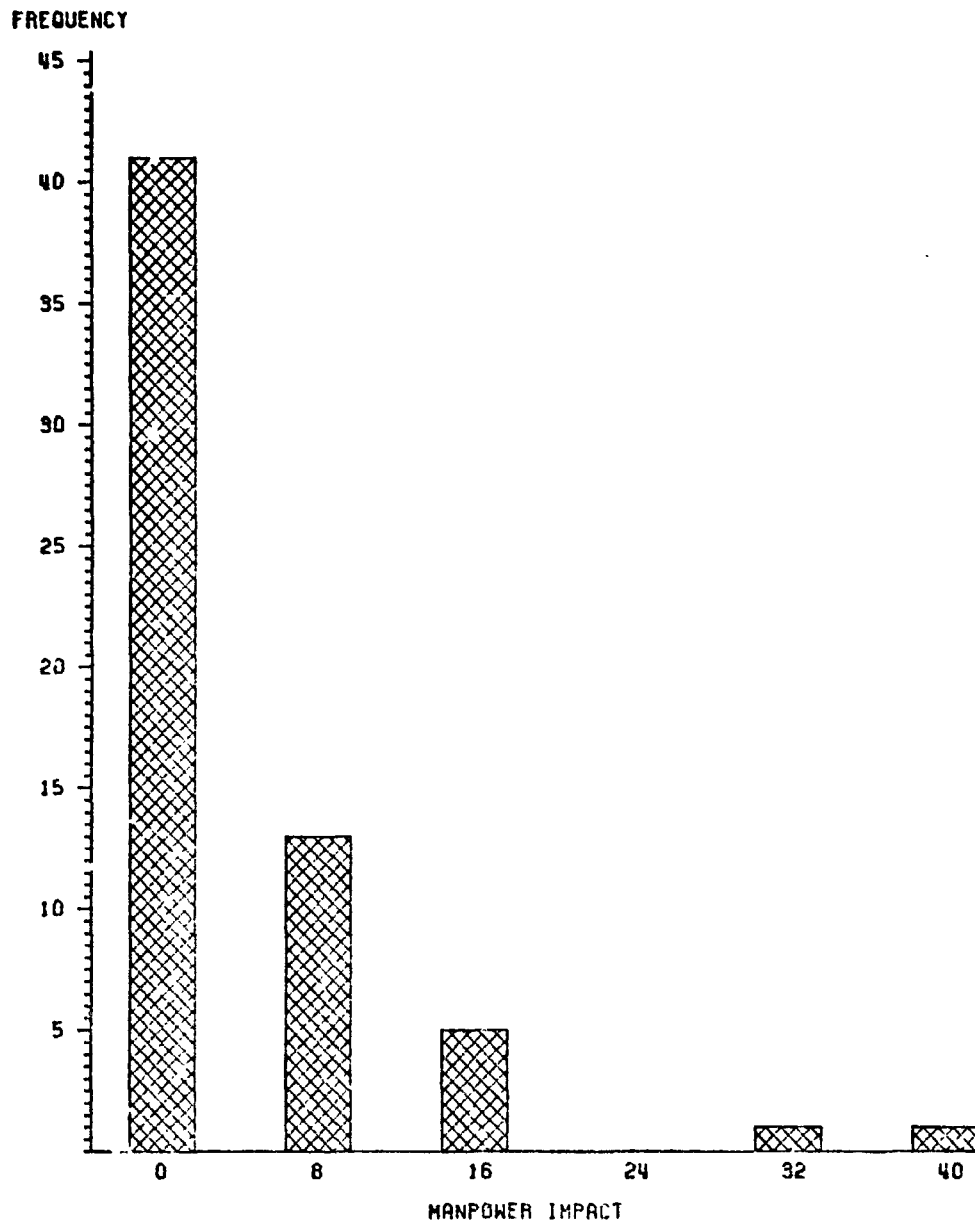


Figure 69: CR frequency for department 2 manpower impact, year 4 (D2IMPYR4).

ORIGINAL PAGE IS
OF POOR QUALITY

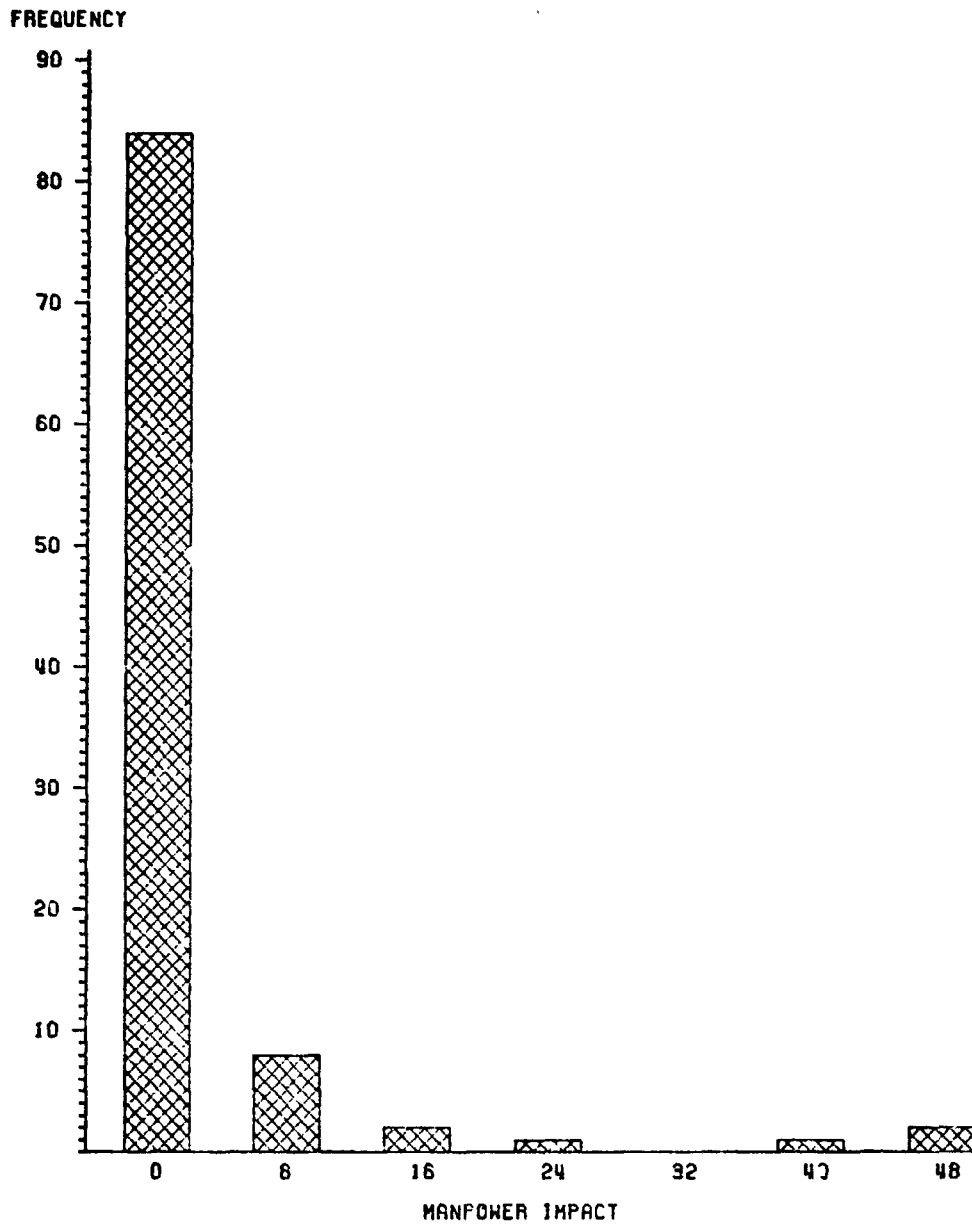


Figure 70: CR frequency for department 3 manpower impact, year 1 (D3IMPYR1).

ORIGINAL PAGE IS
OF POOR QUALITY

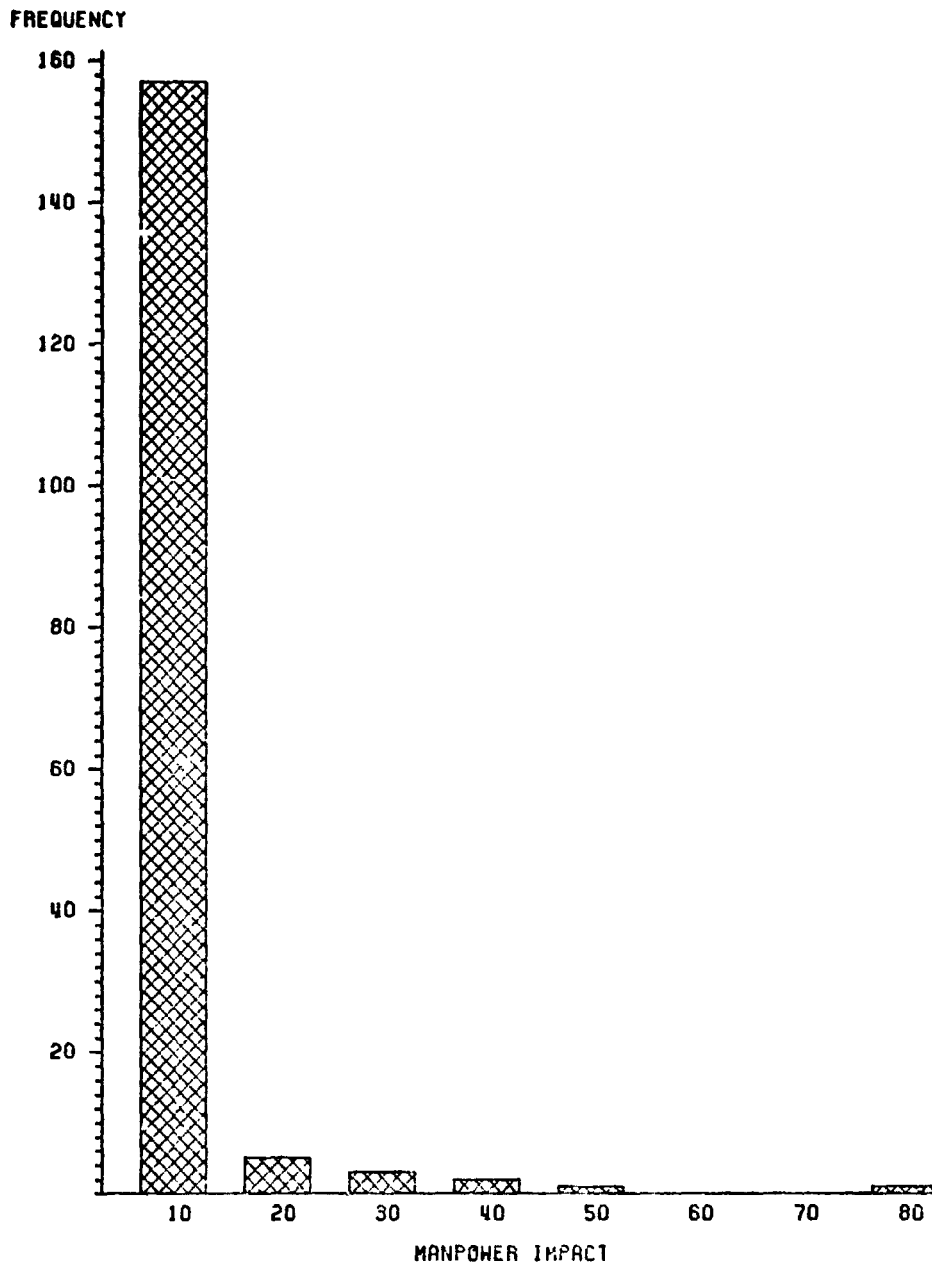


Figure 71: CR frequency for department 3 manpower impact, year 2 (D3IMPYR2).

ORIGINAL PAGE IS
OF POOR QUALITY

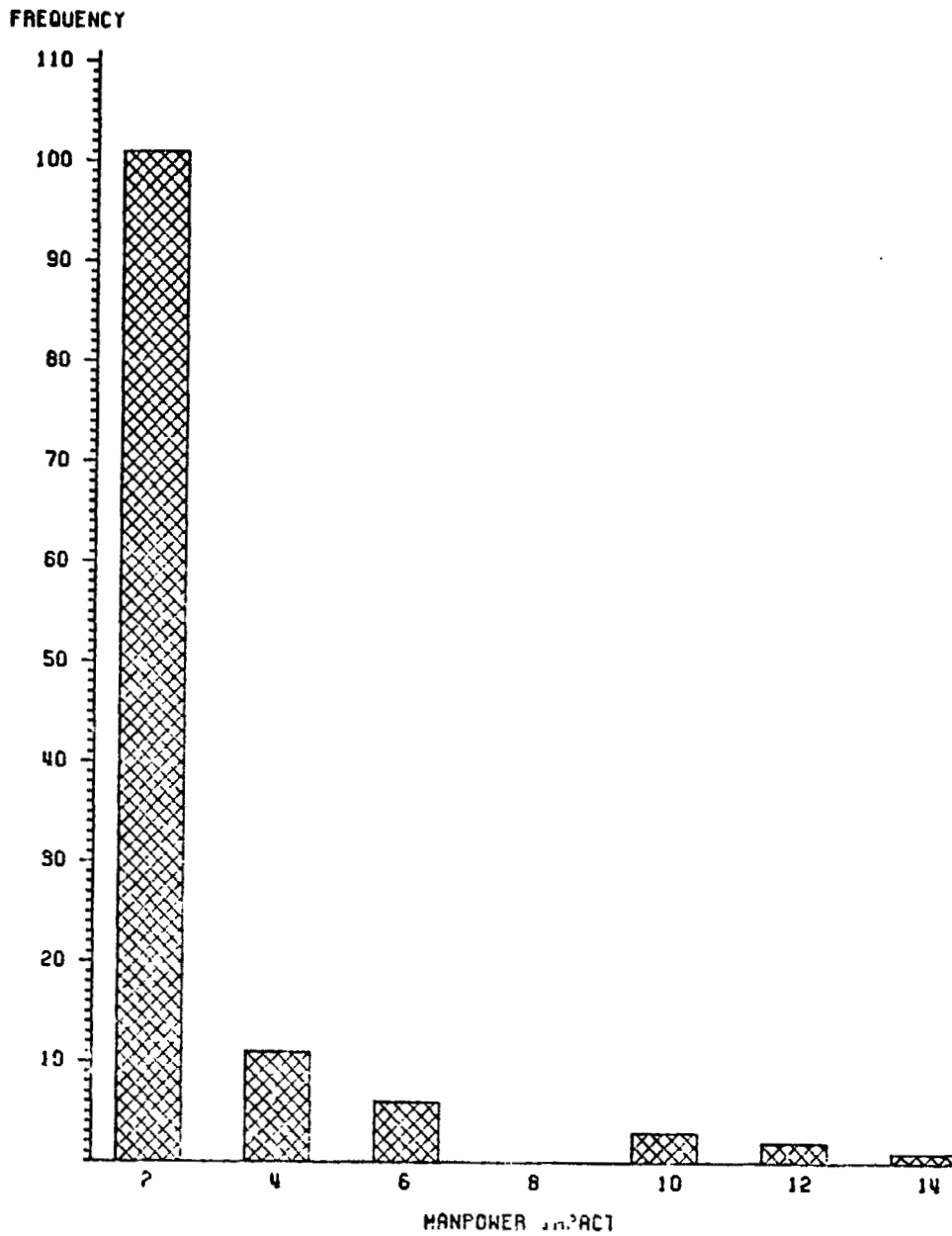


Figure 72: CR frequency for department 3 manpower impact, year 3 (D3IMPYR3).

ORIGINAL PAGE IS
OF POOR QUALITY

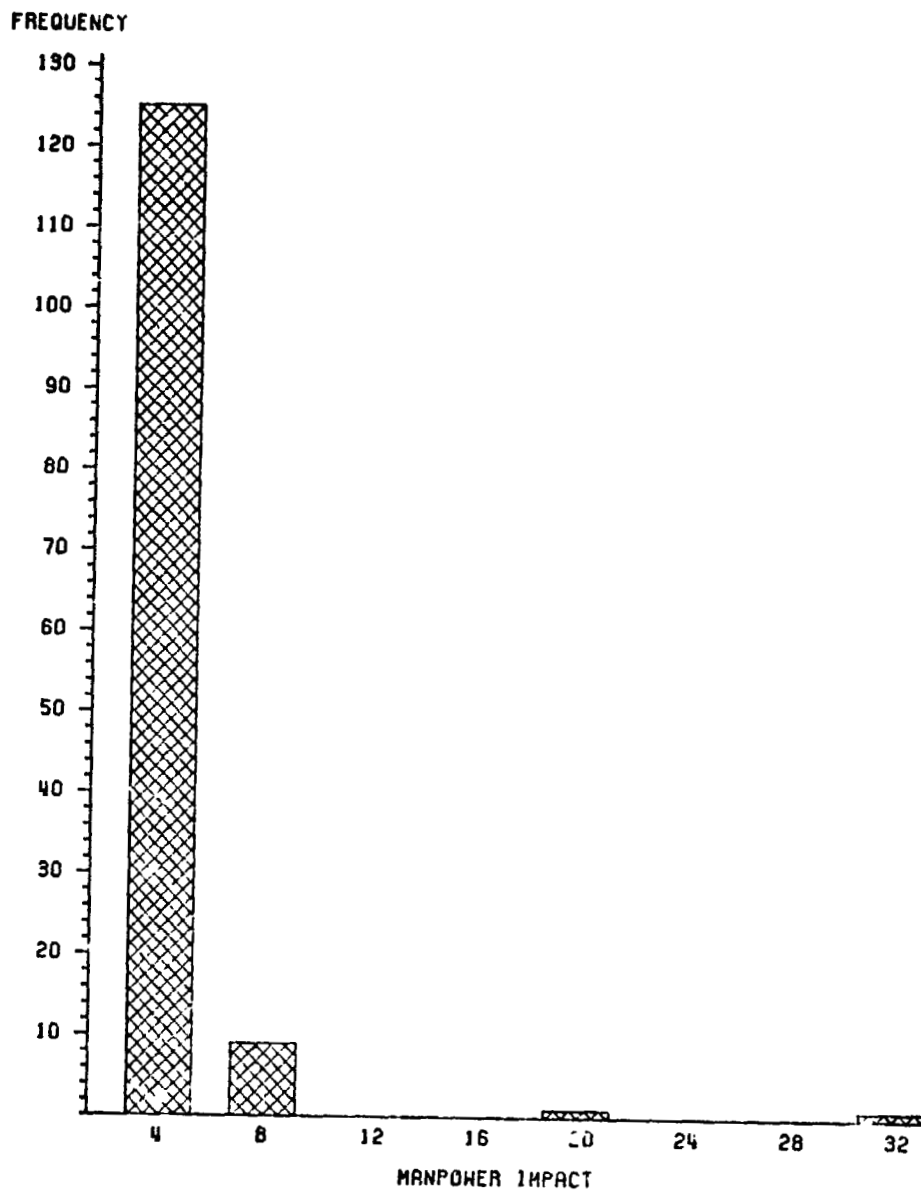


Figure 73: CR frequency for department 3 manpower impact, year 4 (D3IMPYR4).

ORIGINAL PAGE IS
OF POOR QUALITY

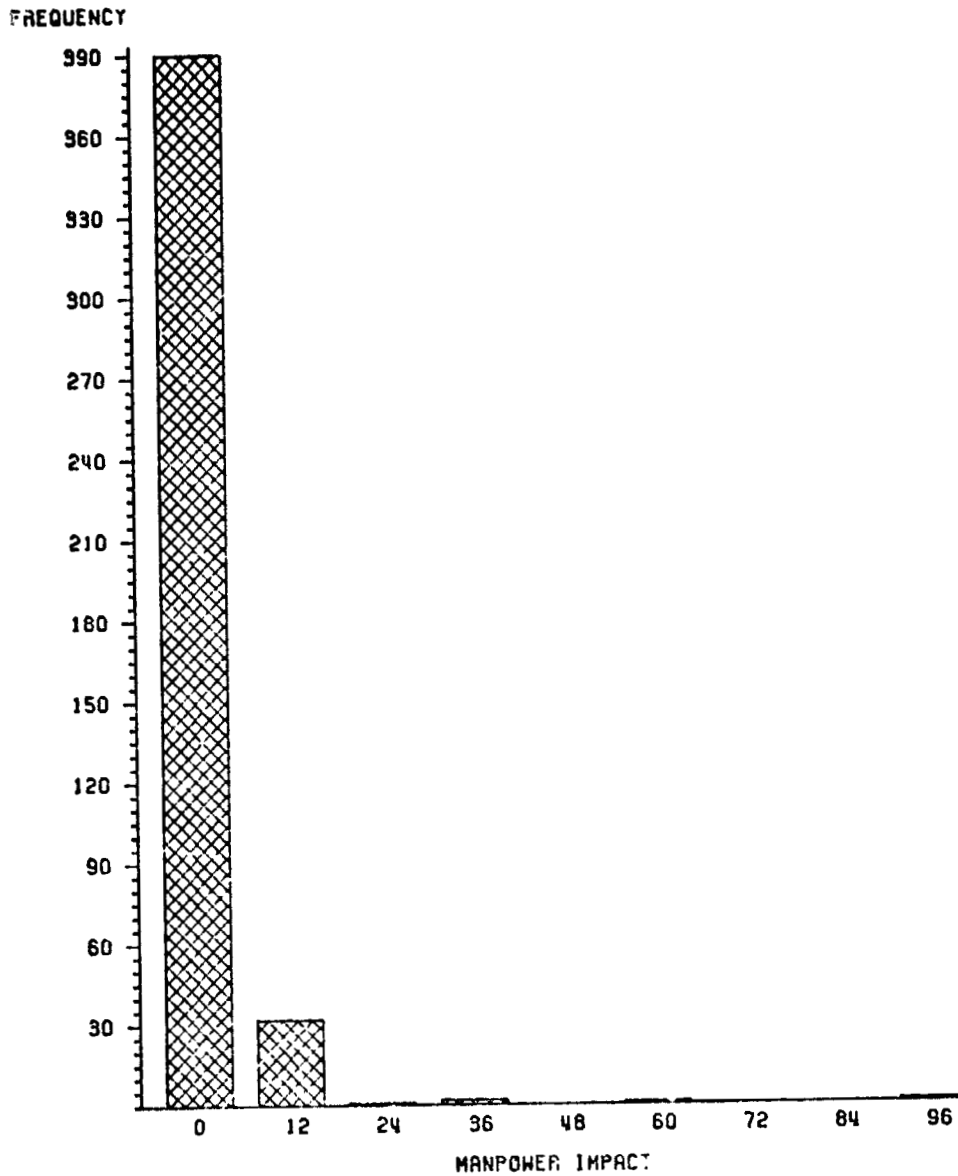


Figure 74: CR frequency for department 4 manpower impact, year 1 (D4IMPYR1).

ORIGINAL PAGE IS
OF POOR QUALITY

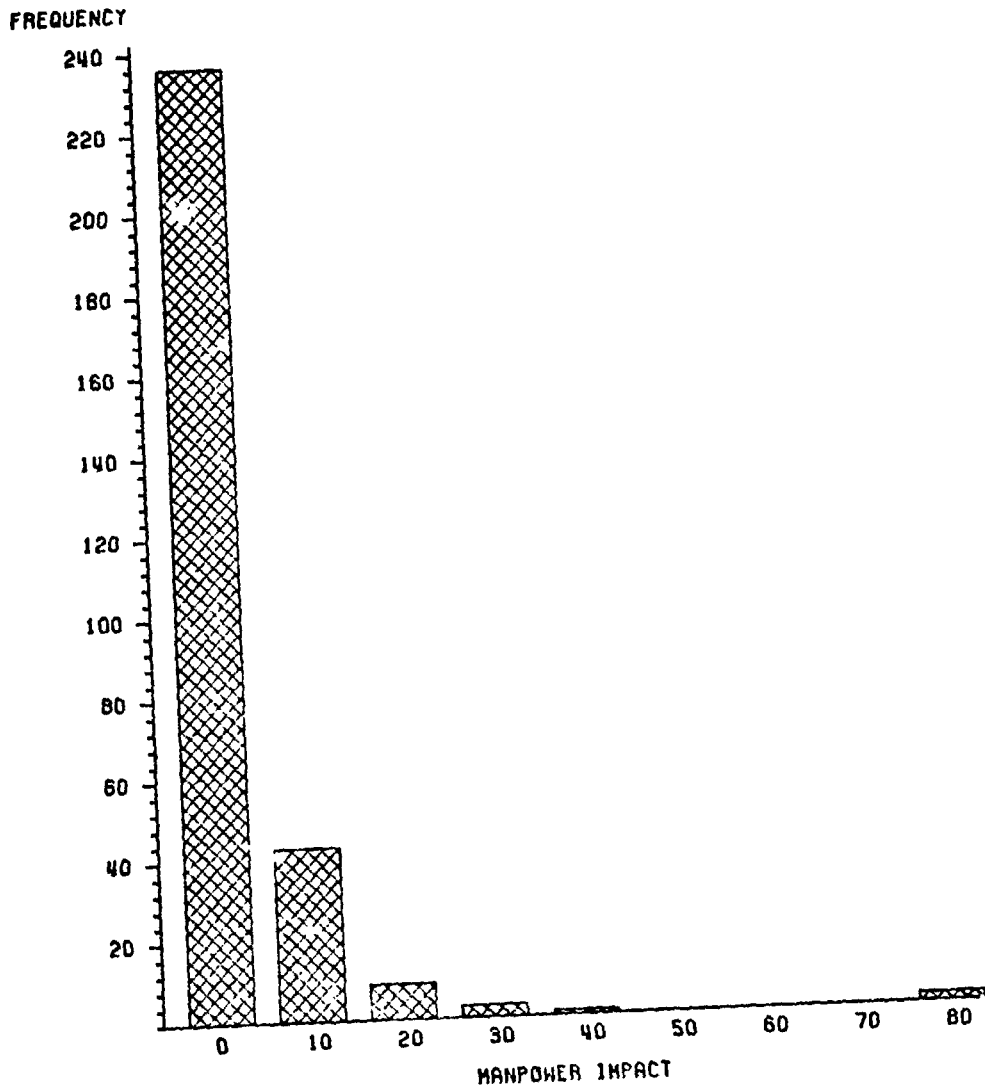


Figure 75: CR frequency for department 4 manpower impact. car 2 (D4IMPYR2).

ORIGINAL PAGE IS
OF POOR QUALITY

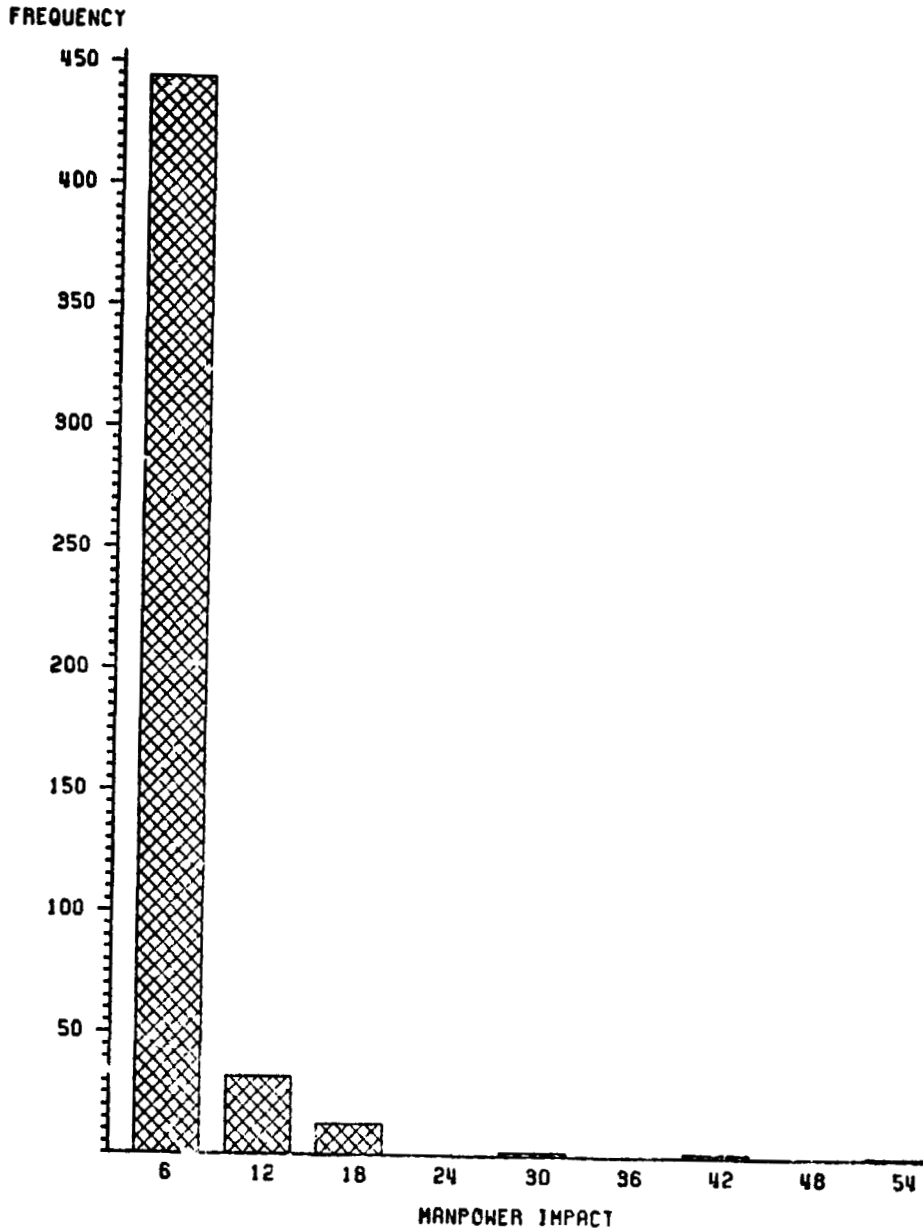


Figure 76: CR frequency for department 4 manpower impact, year 3 (D4IMPYR3).

ORIGINAL PAGE IS
OF POOR QUALITY

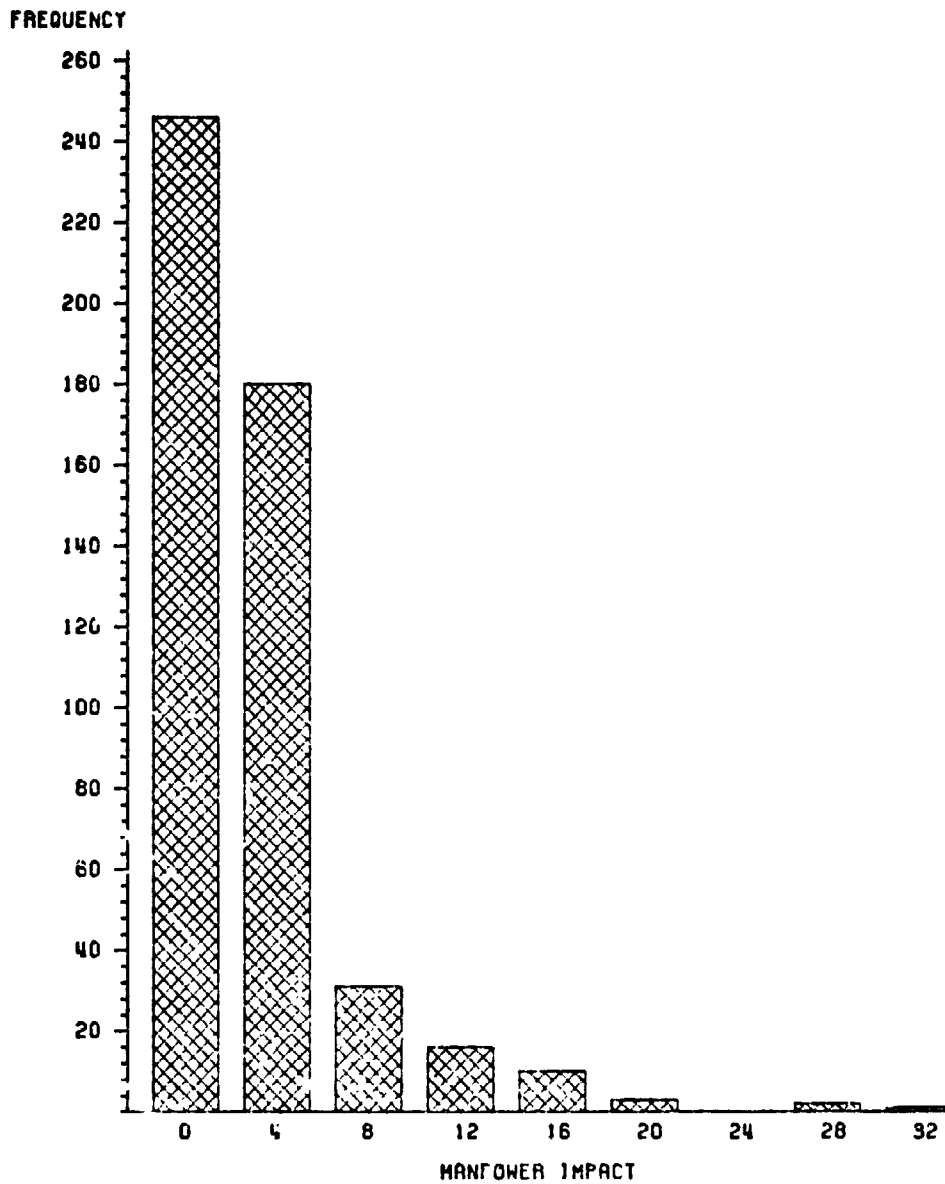


Figure 77: CR frequency for department 4 manpower impact, year 4 (D4IMPYR4).

ORIGINAL PAGE IS
OF POOR QUALITY

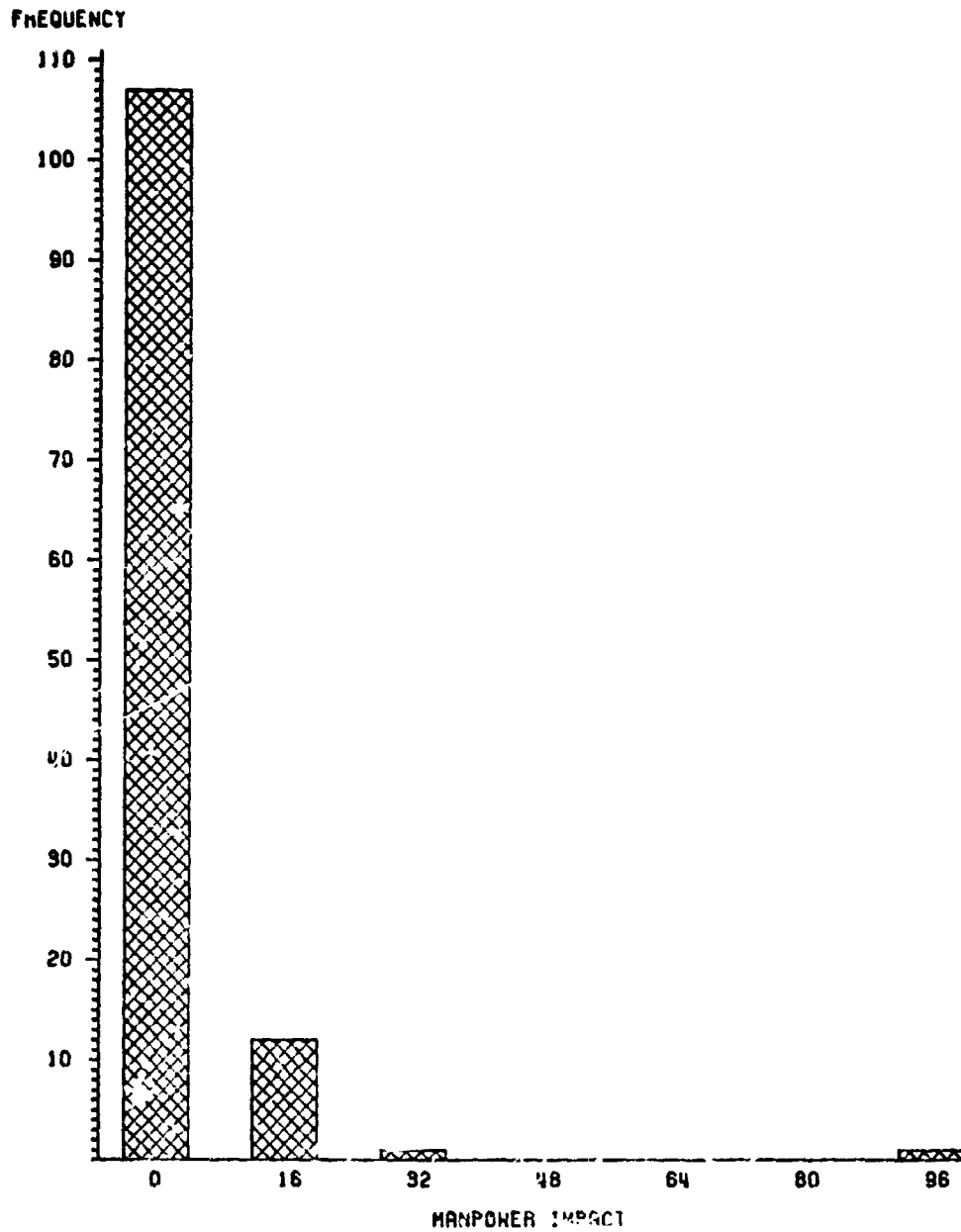


Figure 78: CR frequency for department 5 manpower impact, year 1 (DSIMPYR1).

ORIGINAL PAGE IS
OF POOR QUALITY

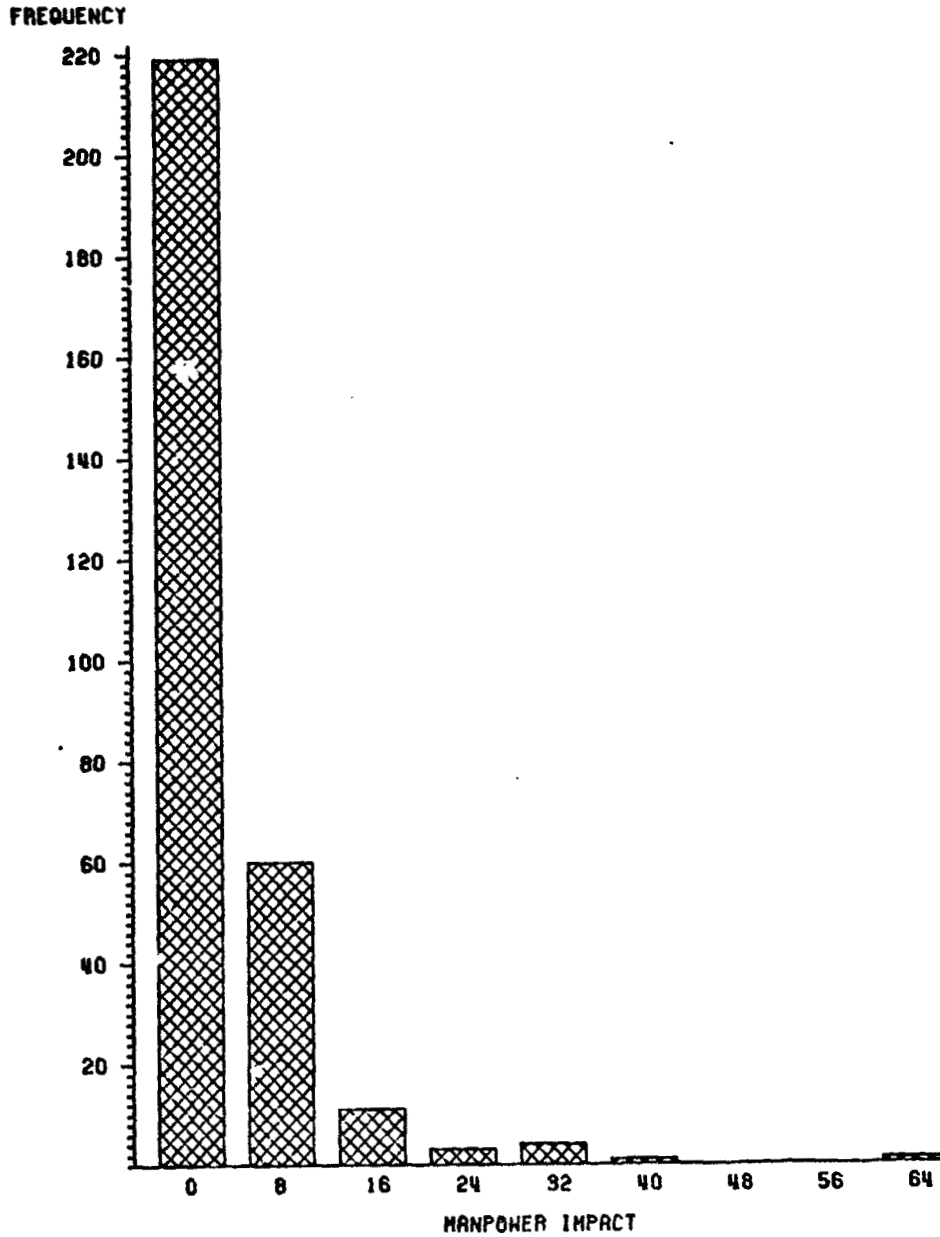


Figure 79: CR frequency for department 5 manpower impact, year 2 (DSIMPYR2).

ORIGINAL PAGE IS
OF POOR QUALITY

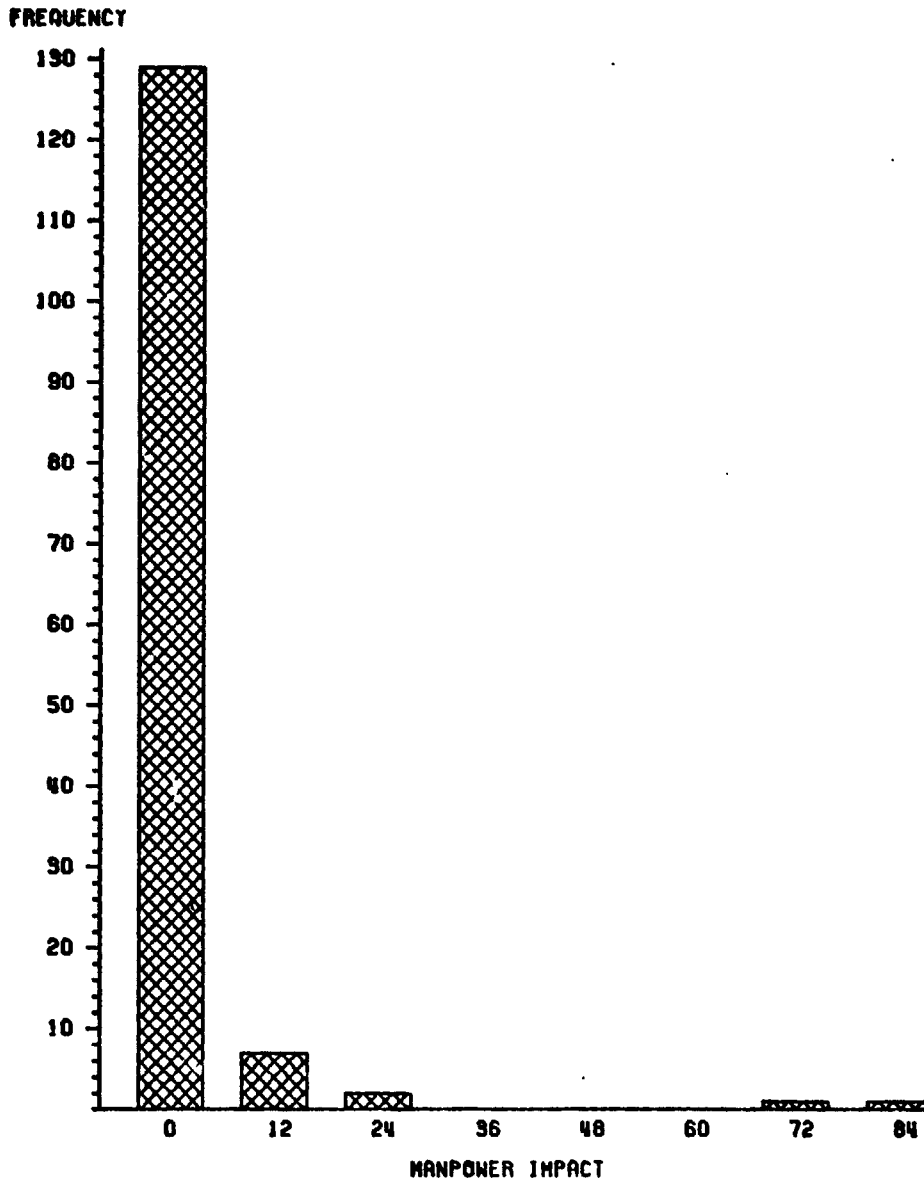


Figure 80: CR frequency for department 5 manpower impact, year 3 (D5IMPYR3).

ORIGINAL PAGE IS
OF POOR QUALITY

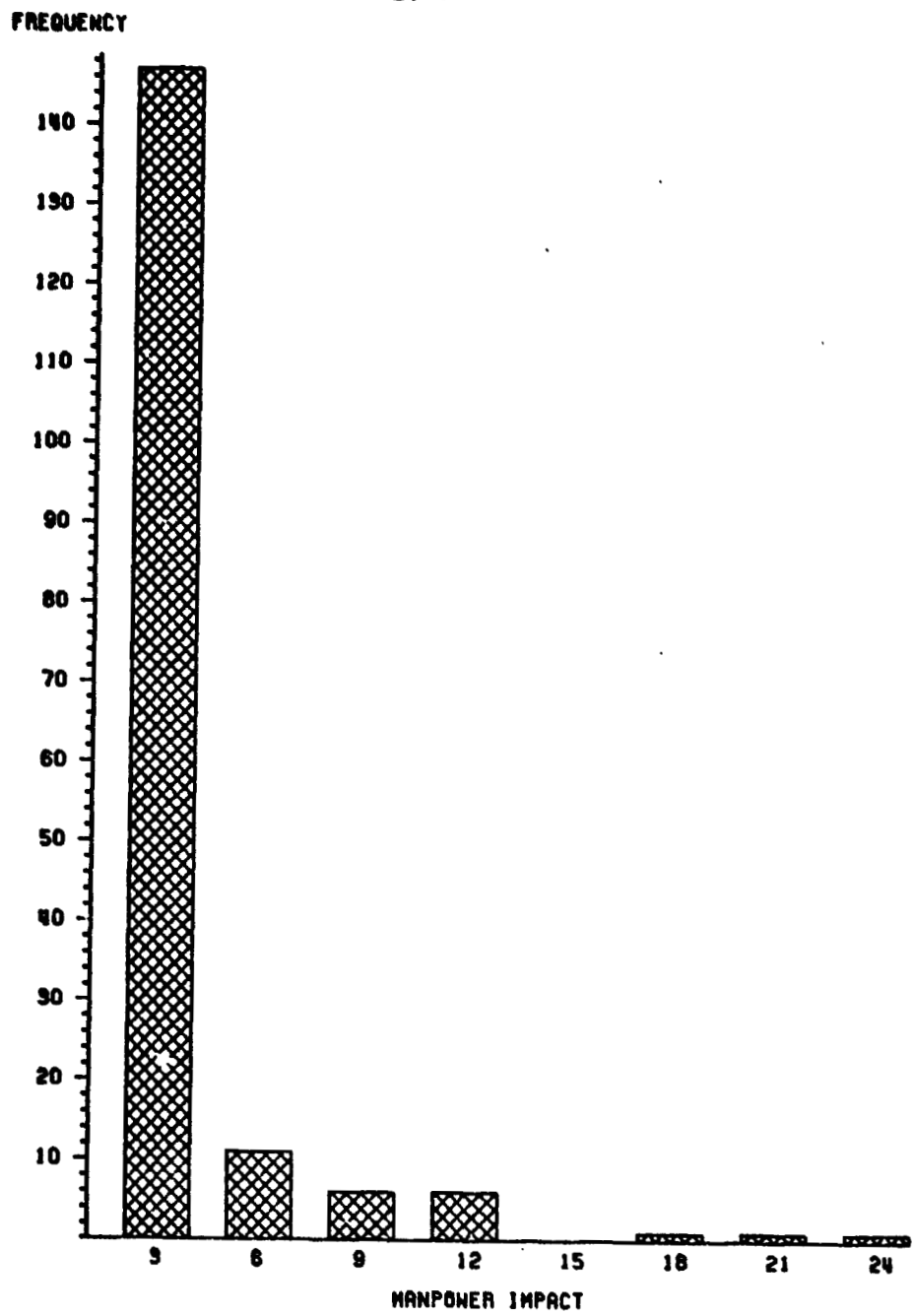


Figure 81: CR frequency for department 5 manpower impact, year 4 (D5IMPYR4).

ORIGINAL PAGE IS
OF POOR QUALITY

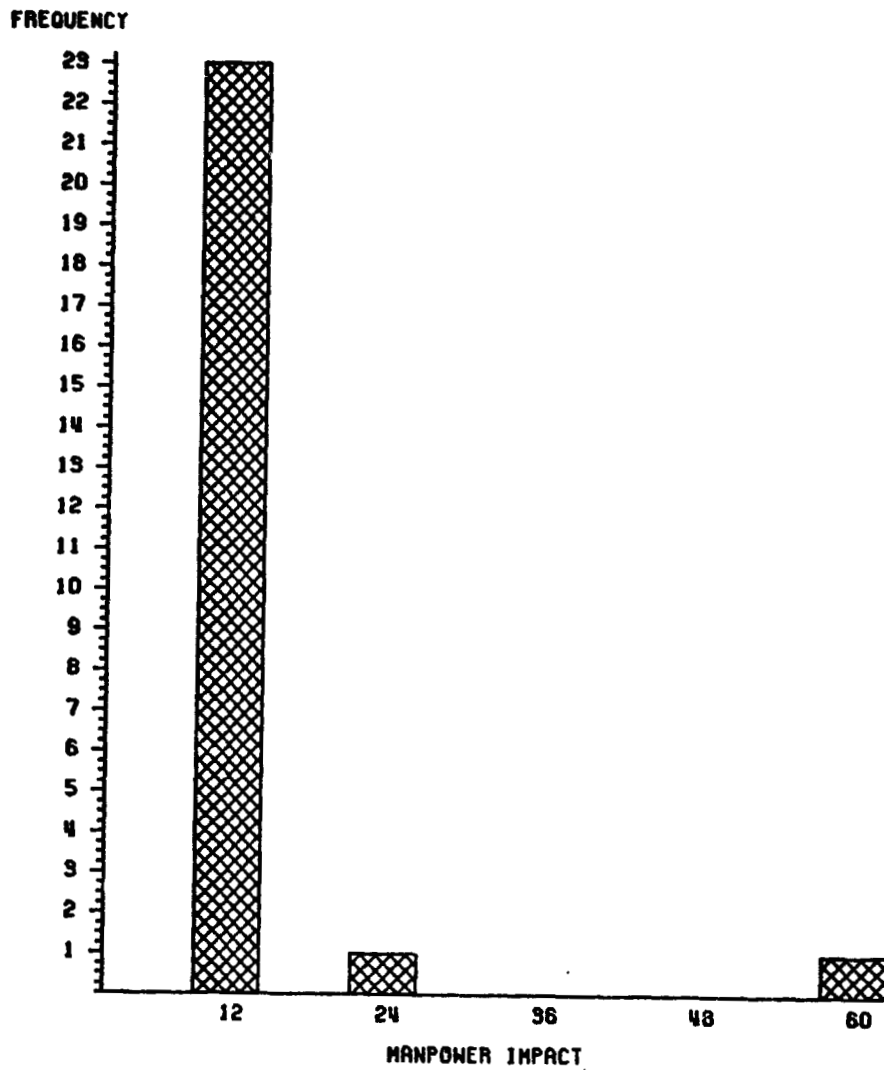


Figure 82: CR frequency for department 6 manpower impact, year 1 (D6IMPYR1).

ORIGINAL PAGE IS
OF POOR QUALITY

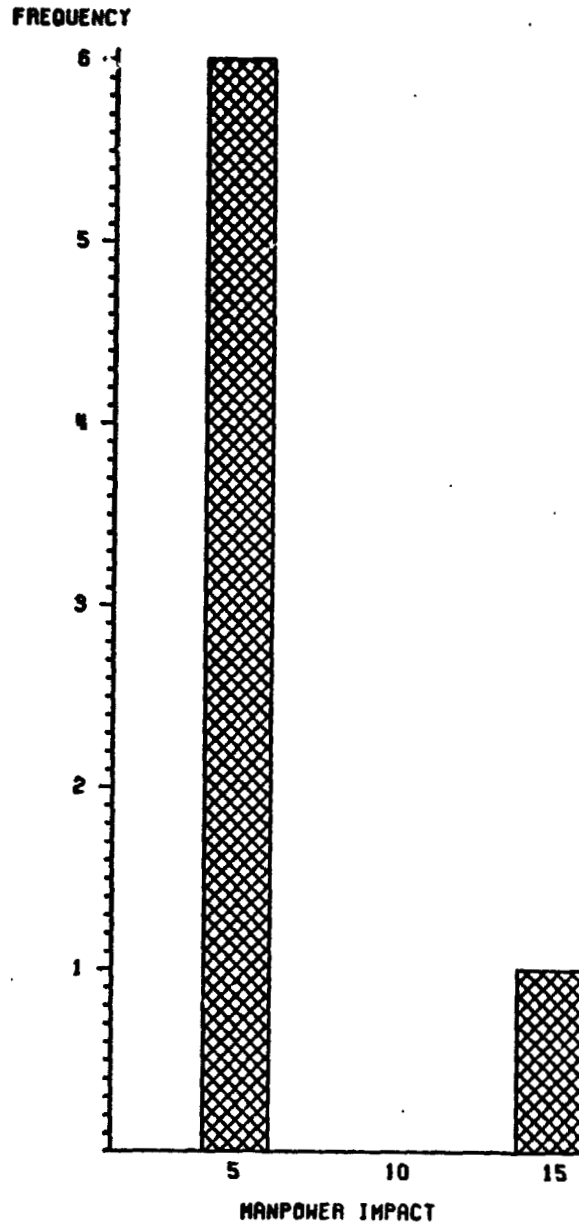


Figure 83: CR frequency for department 6 manpower impact, year 2 (D6IMPYR2).

ORIGINAL PAGE IS
OF POOR QUALITY

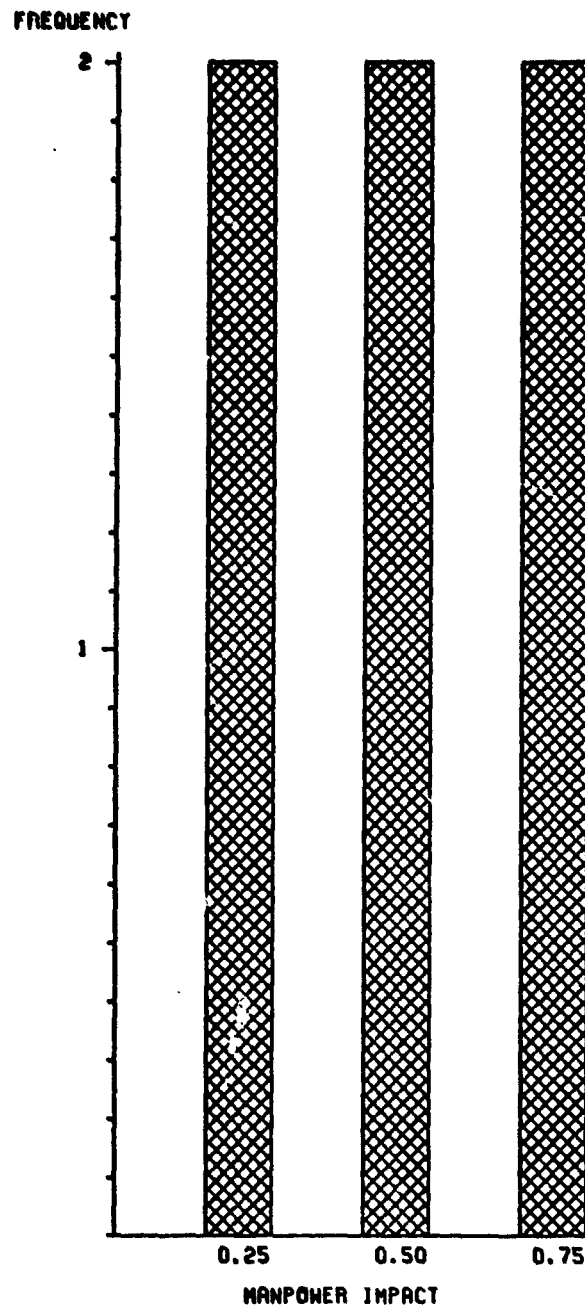


Figure 84: CR frequency for department 6 manpower impact, year 3 (D6IMPYR3).

ORIGINAL PAGE IS
OF POOR QUALITY

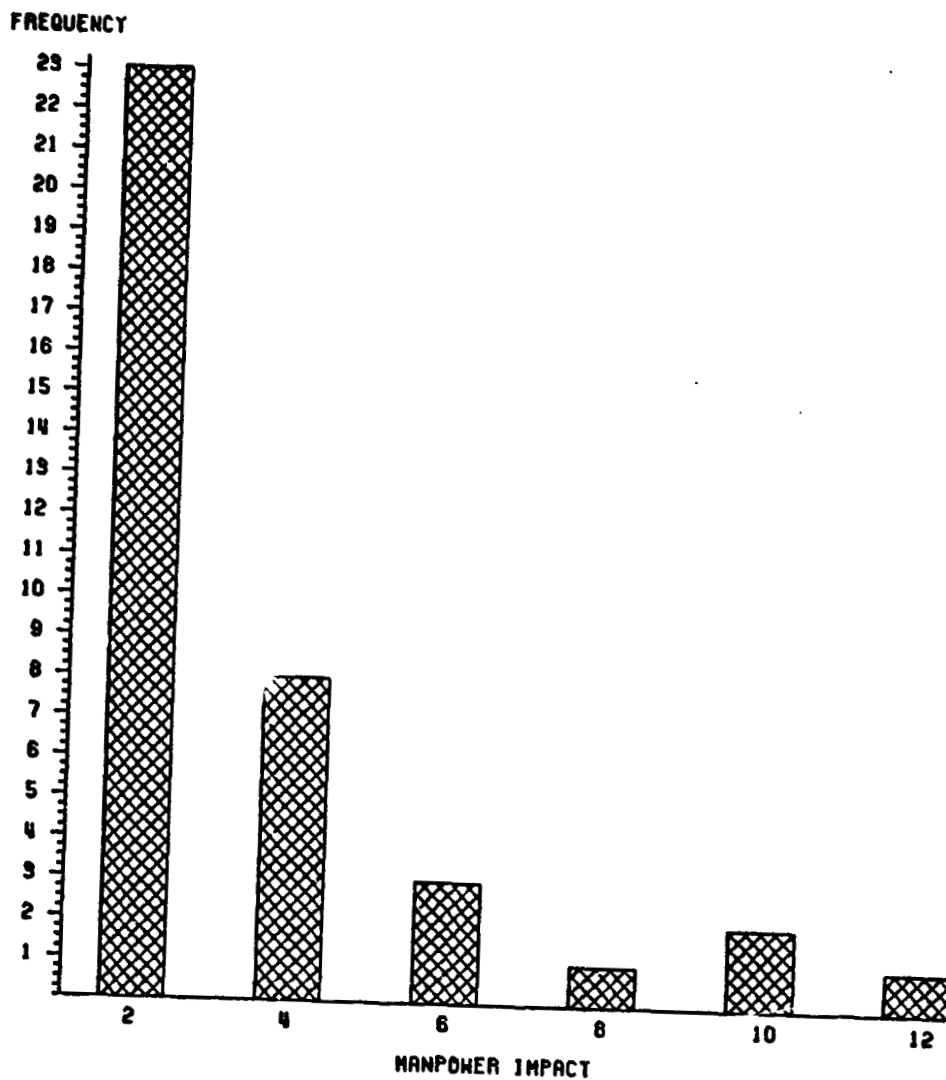


Figure 85: CR frequency for department 6 manpower impact, year 4 (D6IMPYR4).

ORIGINAL PAGE 18
OF POOR QUALITY

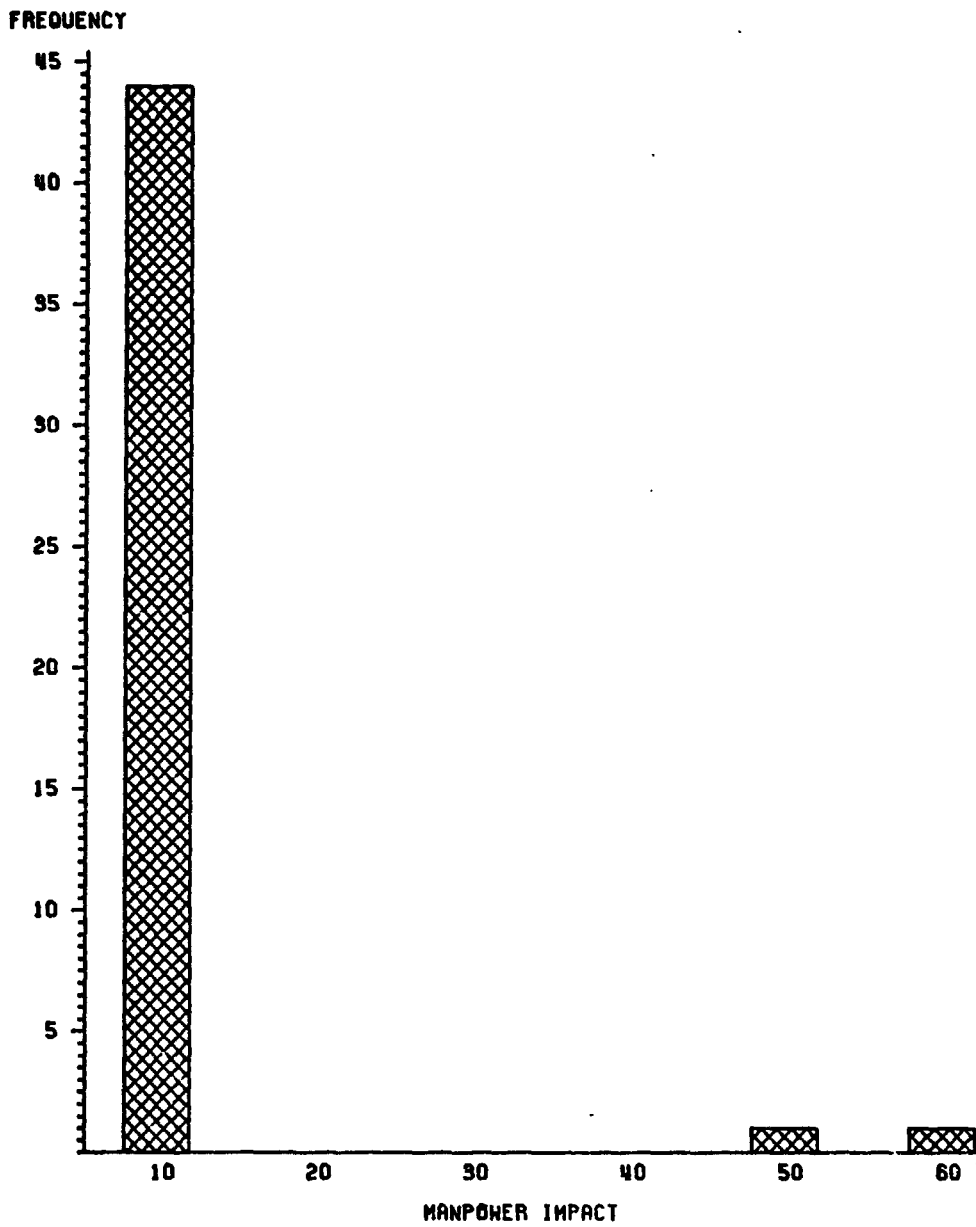


Figure 86: CR frequency for department 7 manpower impact, year 1 (D7IMPYR1).

ORIGINAL PAGE IS
OF POOR QUALITY

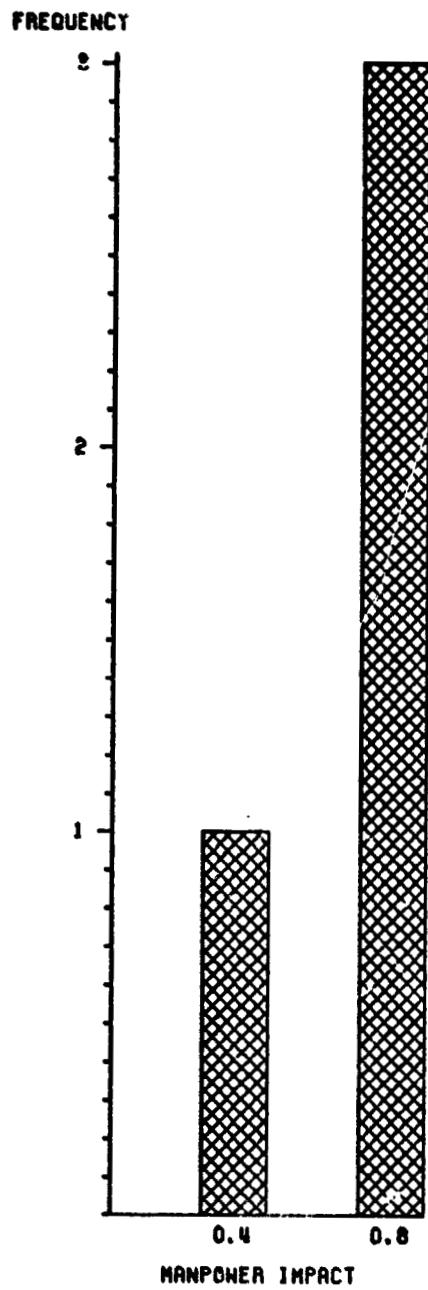


Figure 87: CR frequency for department 7 manpower impact, year 3 (D7IMPYR3).

ORIGINAL PAGE IS
OF POOR QUALITY

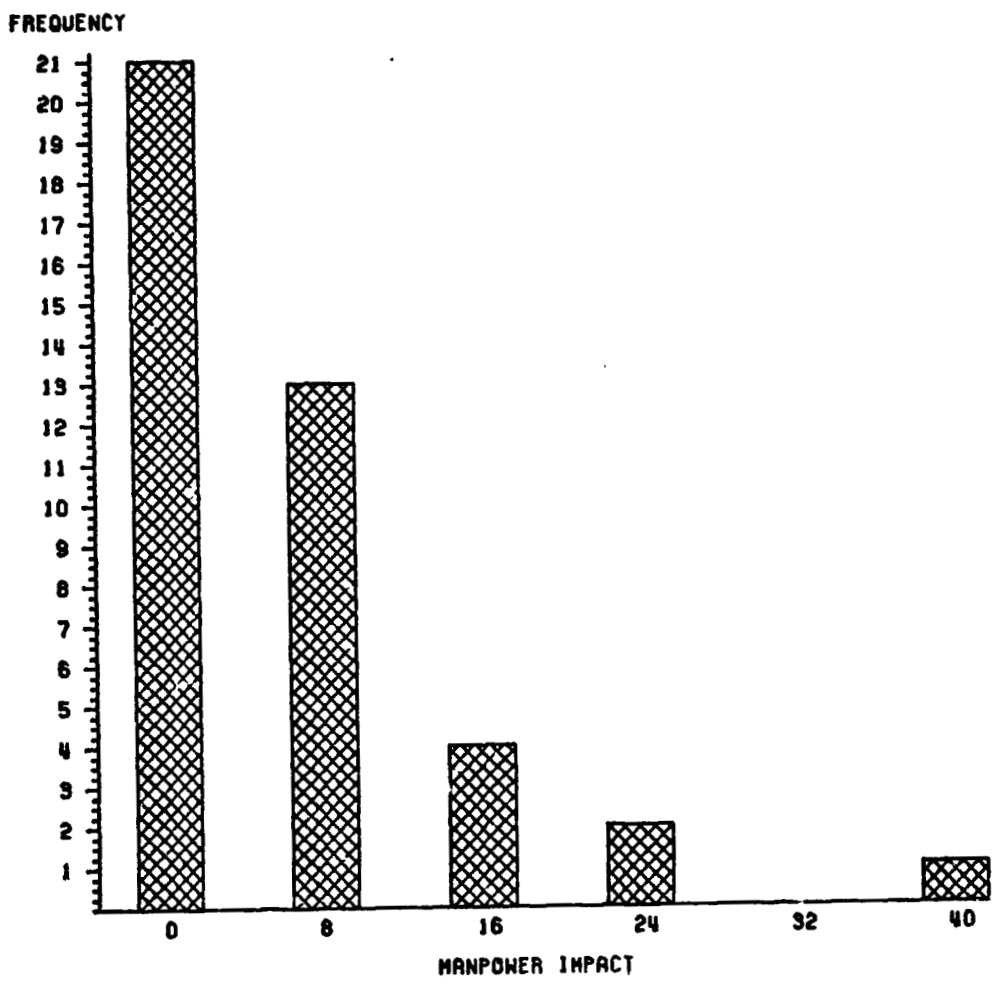


Figure 88: CR frequency for department 7 manpower impact, year 4 (D7IMPYR4).

ORIGINAL PAGE IS
OF POOR QUALITY

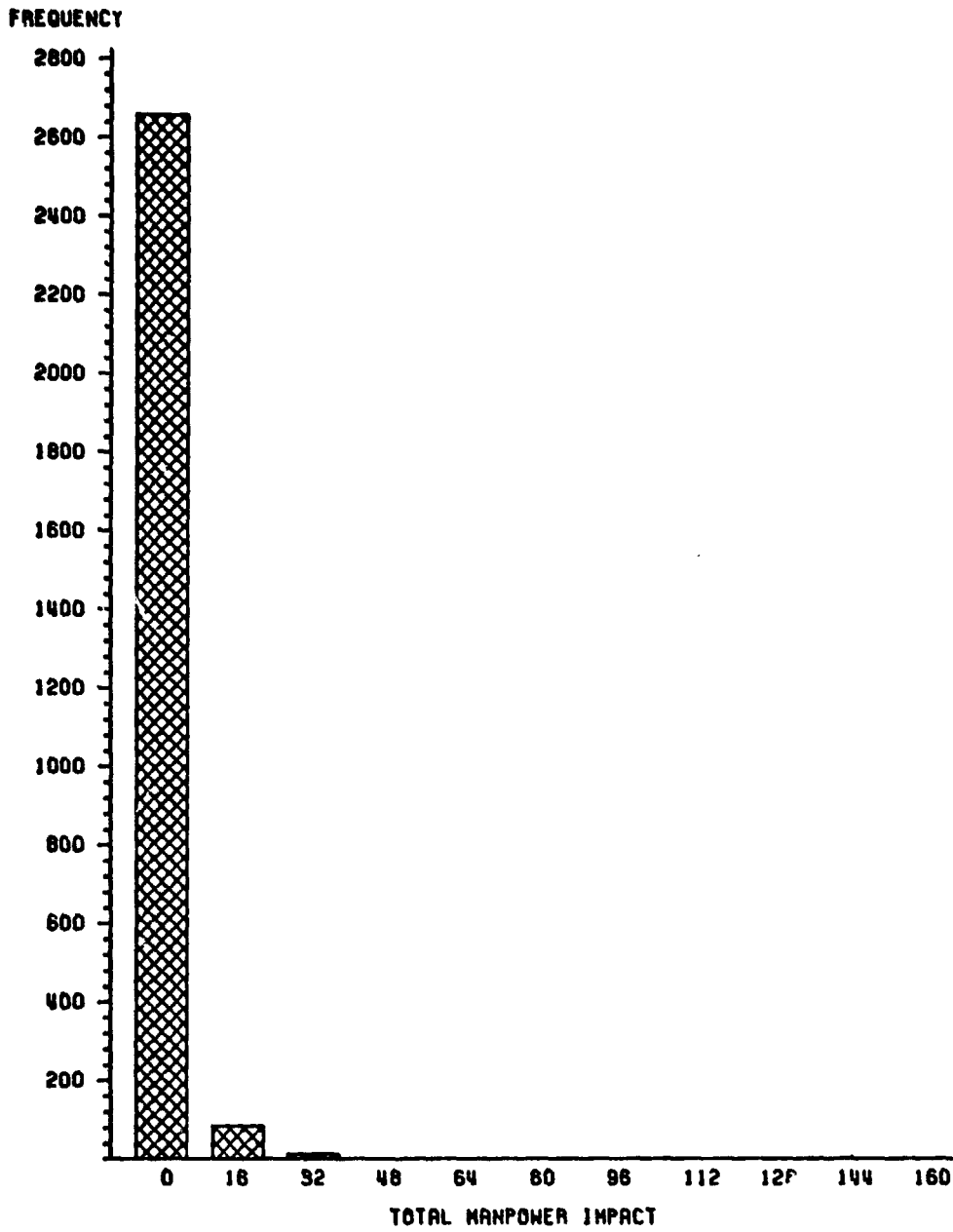


Figure 89: CR frequency for department 1 total manpower impact (DITOTIMP).

ORIGINAL PAGE IS
OF POOR QUALITY

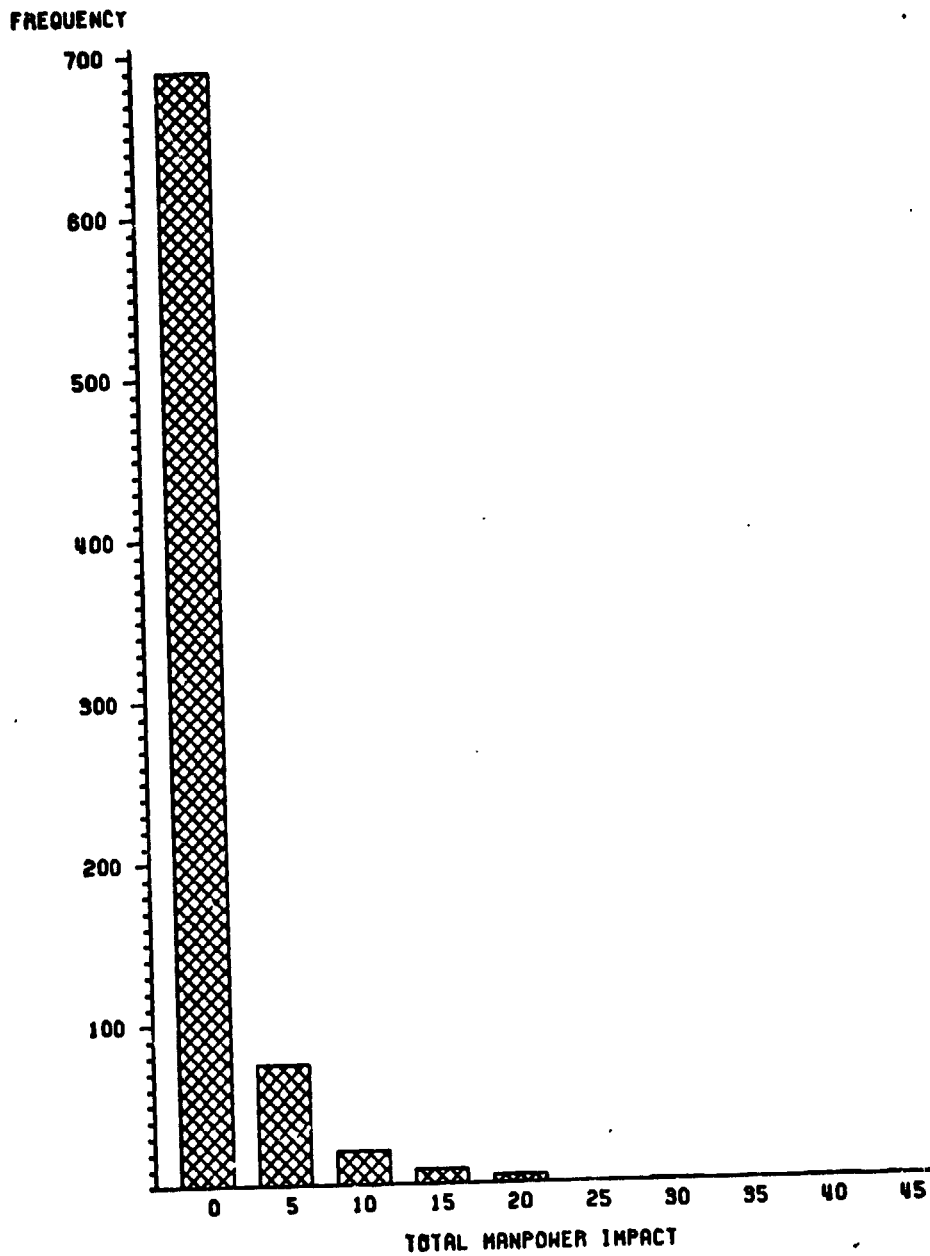


Figure 90: CR frequency for department 2 total manpower impact (D2TOTIMP).

ORIGINAL PAGE IS
OF POOR QUALITY

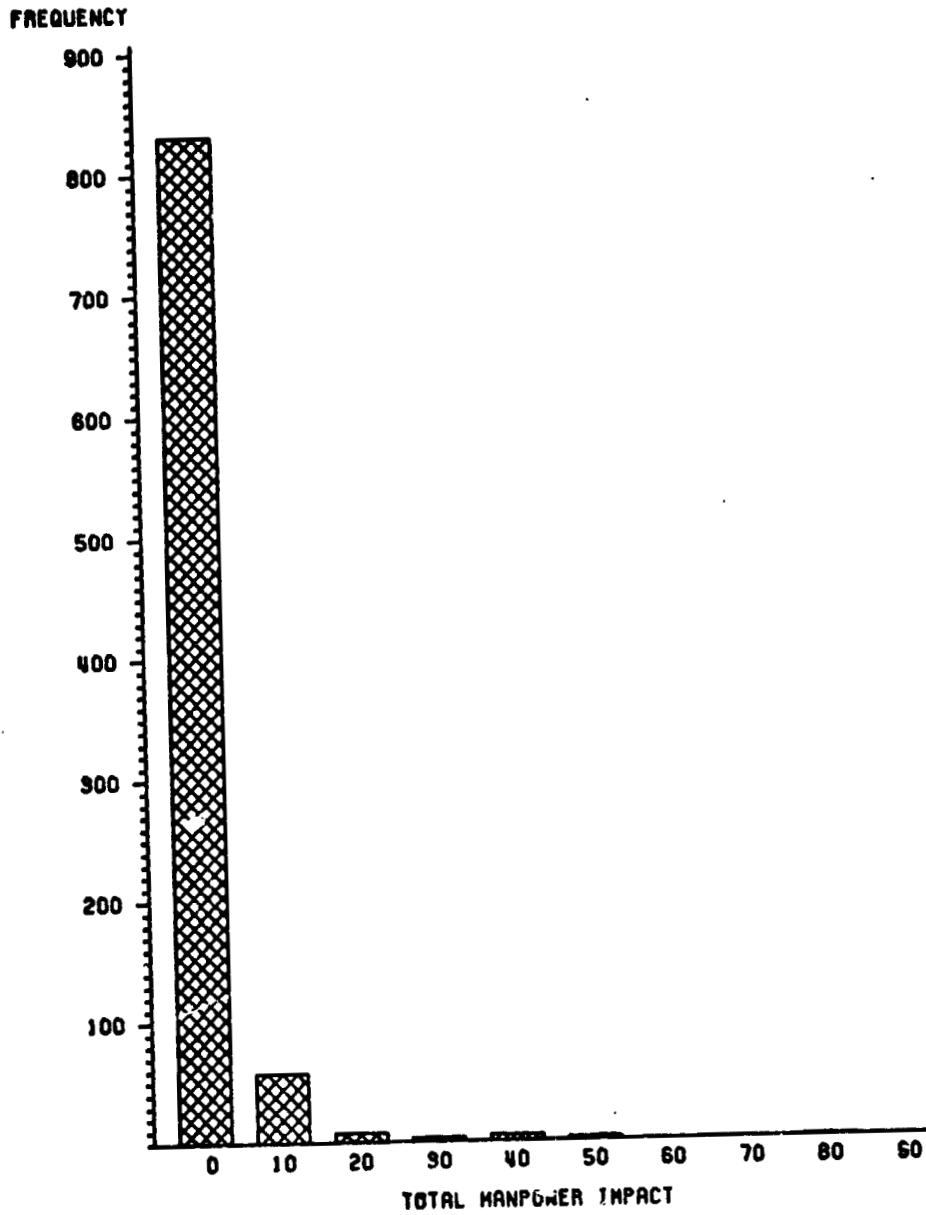


Figure 91: CR frequency for department 3 total manpower impact (D3TOTIMP).

ORIGINAL PAGE IS
OF POOR QUALITY

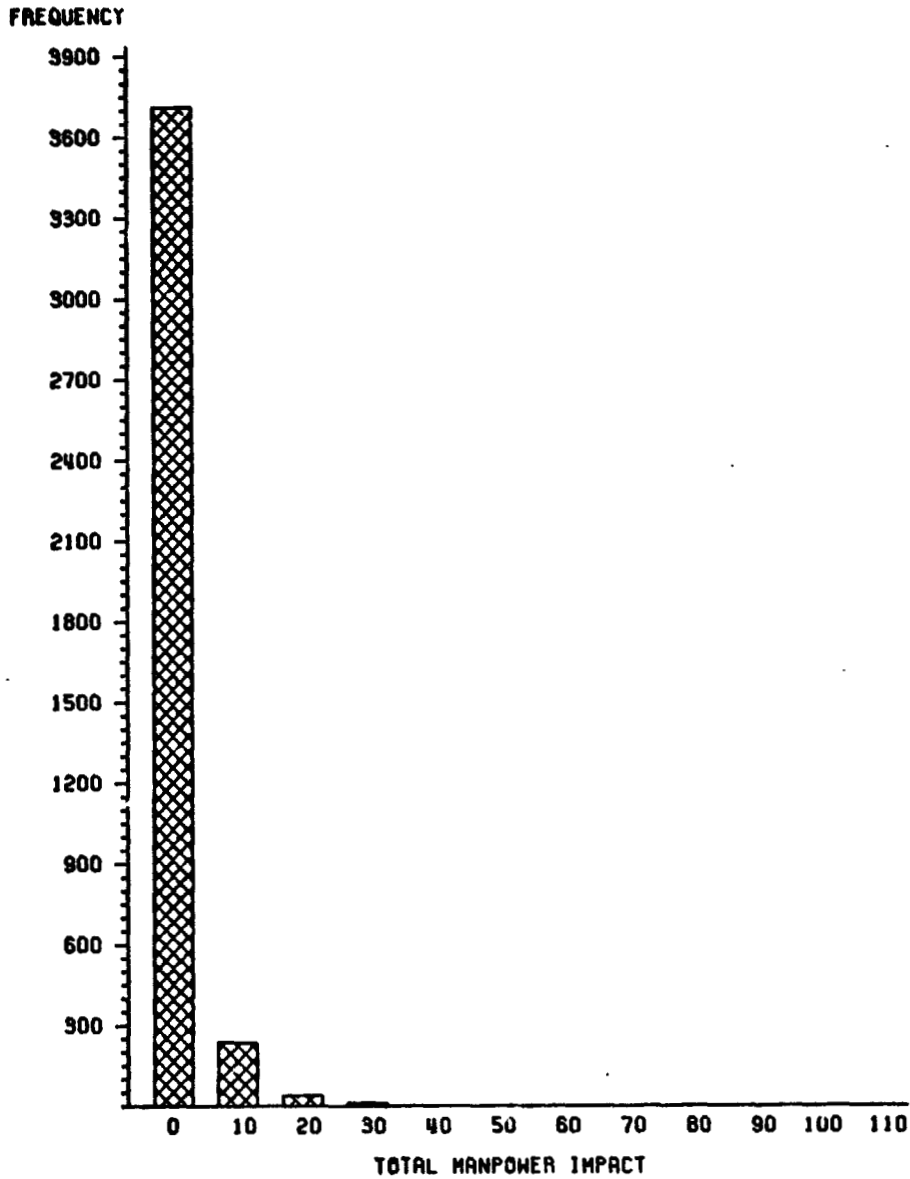


Figure 92: CR frequency for department 4 total manpower impact (D4TOTIMP).

ORIGINAL PAGE IS
OF POOR QUALITY

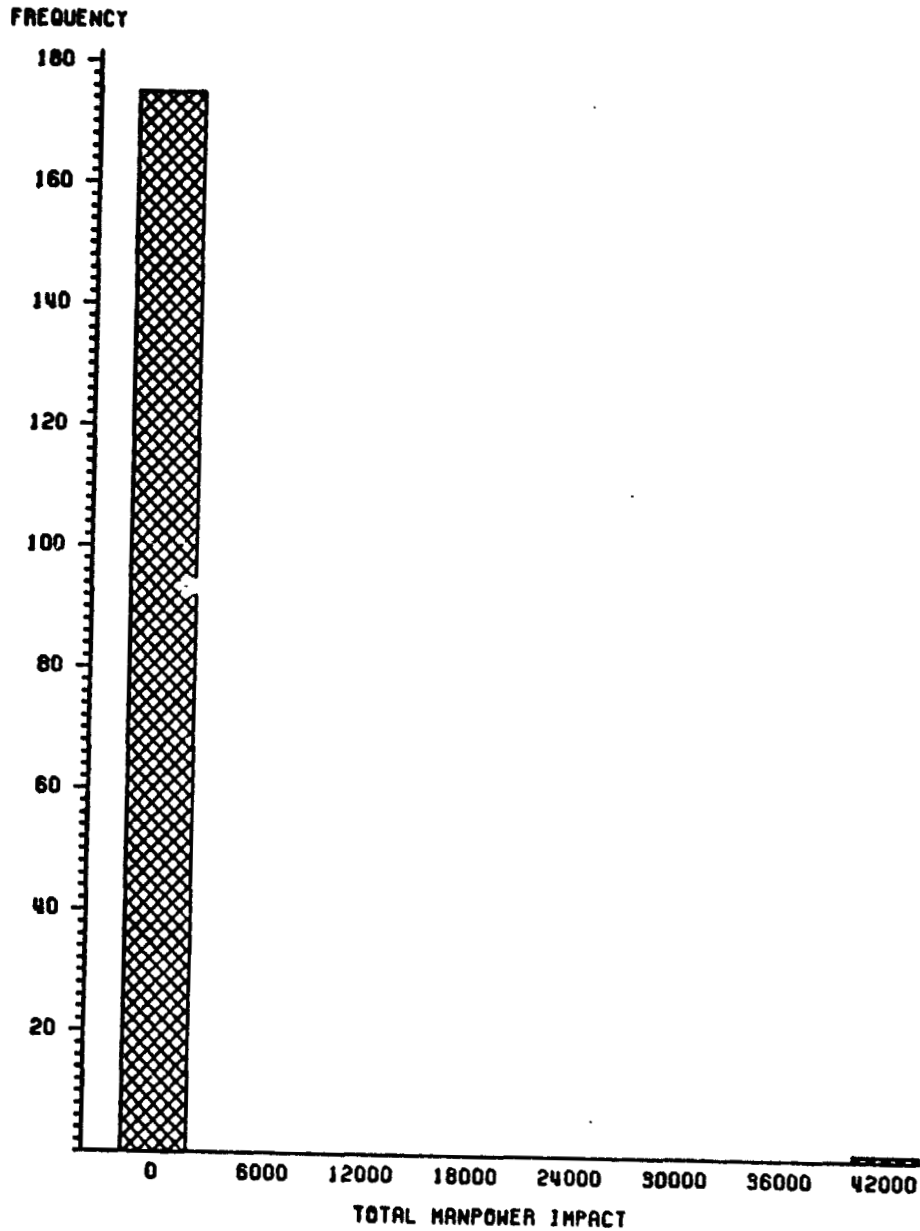


Figure 93: CR frequency for department 5 total manpower impact (DSTOTIMP).

ORIGINAL PAGE IS
OF POOR QUALITY

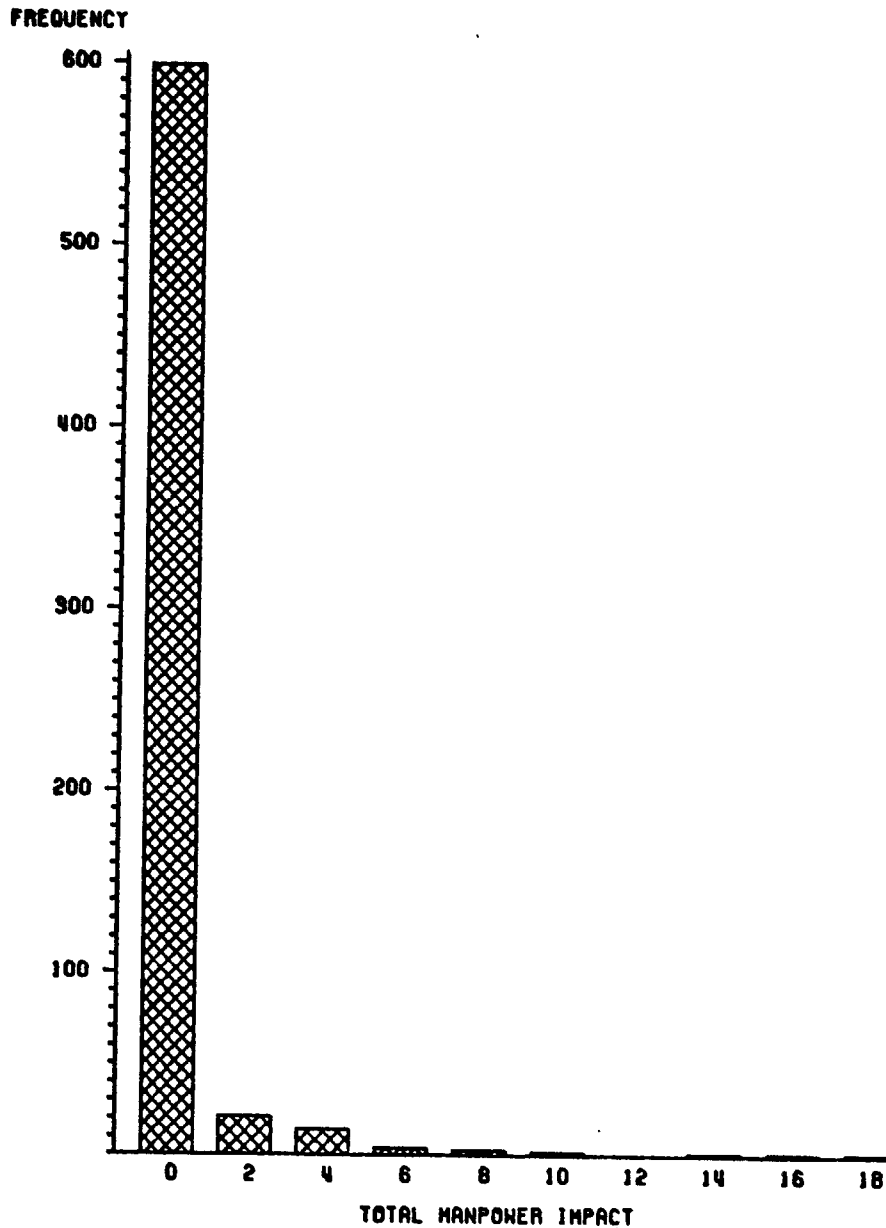


Figure 94: CR frequency for department 6 total manpower impact (D6TOTIMP).

ORIGINAL PAGE IS
OF POOR QUALITY

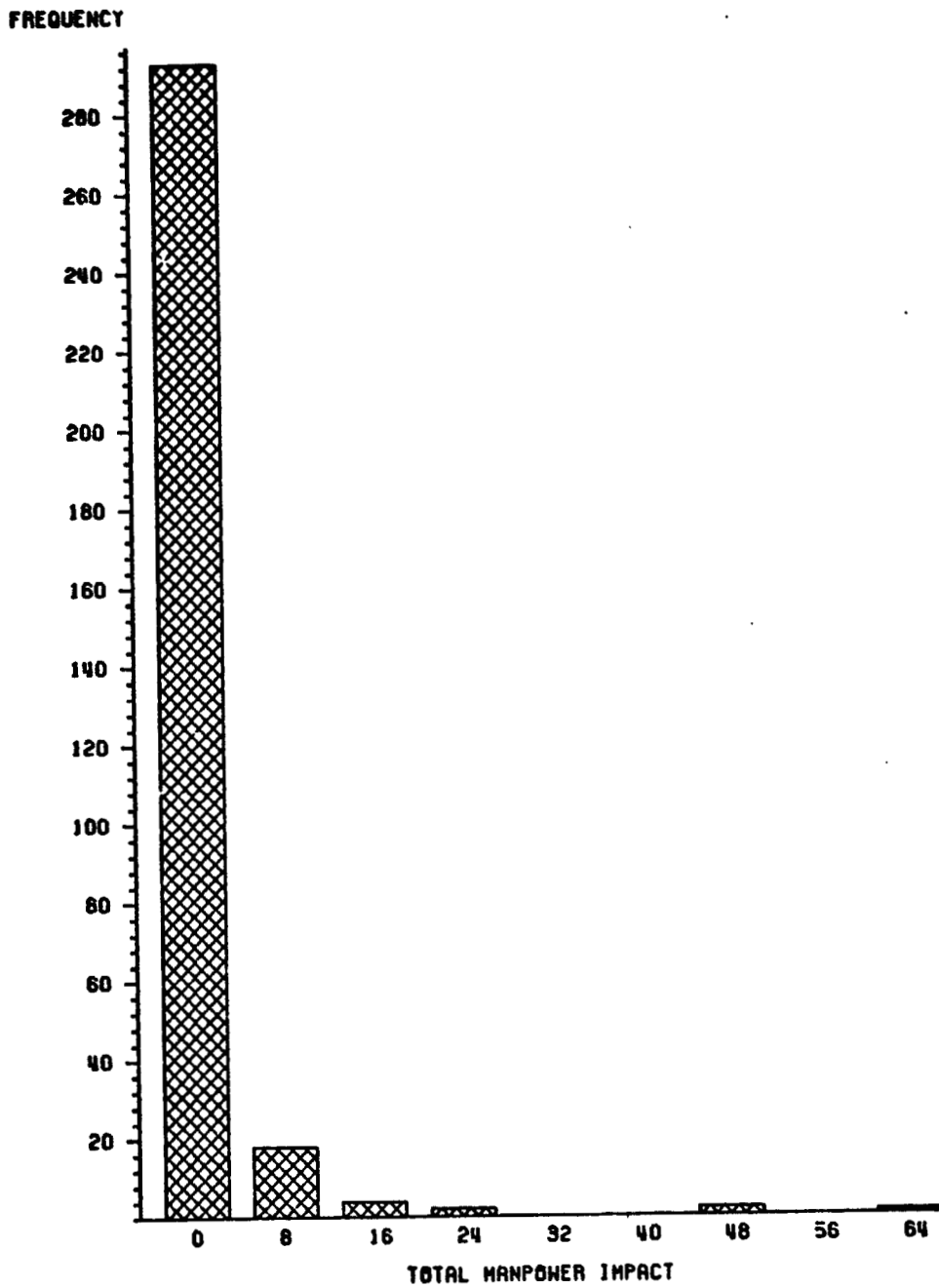


Figure 95: CR frequency for department 7 total manpower impact (D7TOTIMP).

ORIGINAL PAGE IS
OF POOR QUALITY

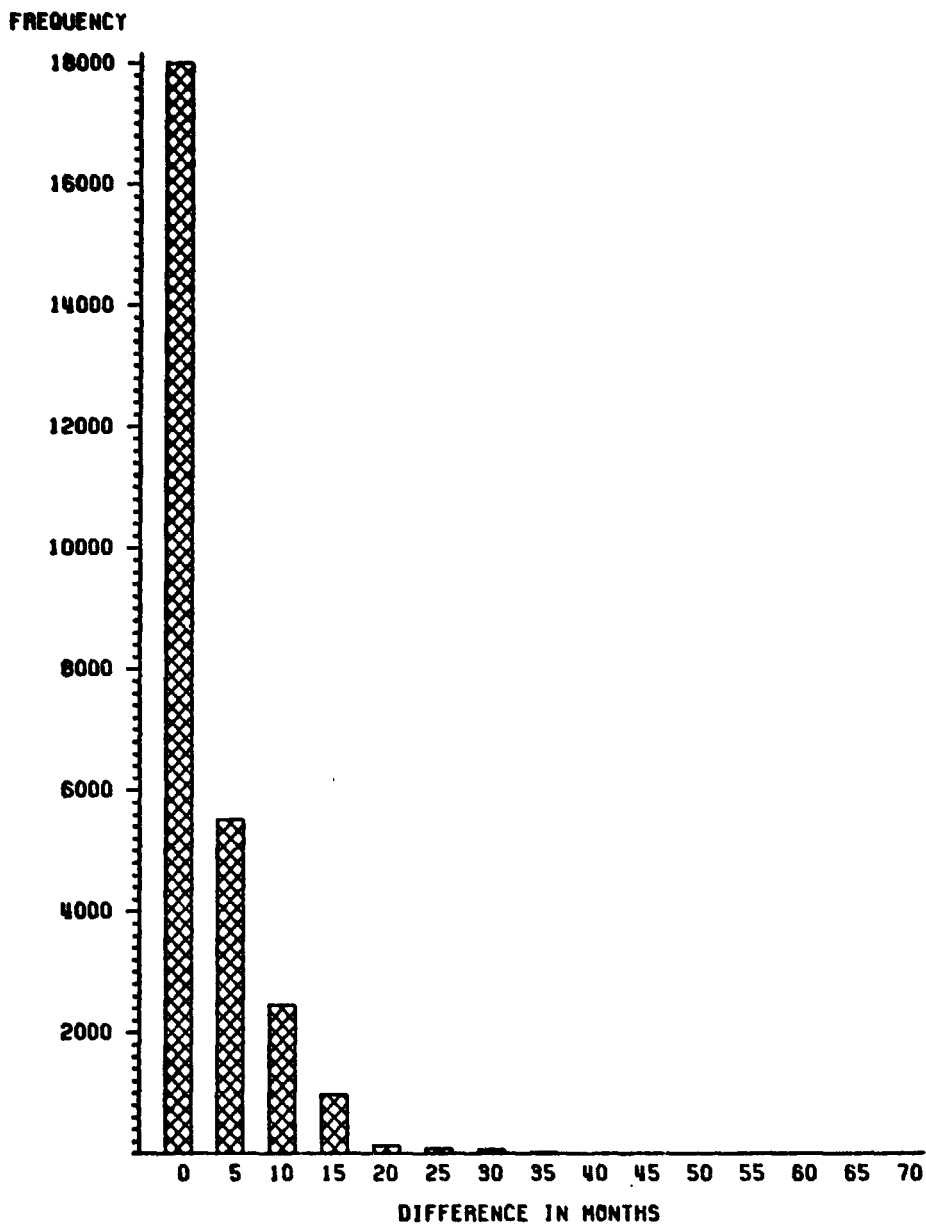


Figure 96: DR frequency of number of months between date logged in (_OD) and development close date (BD).

ORIGINAL PAGE IS
OF POOR QUALITY

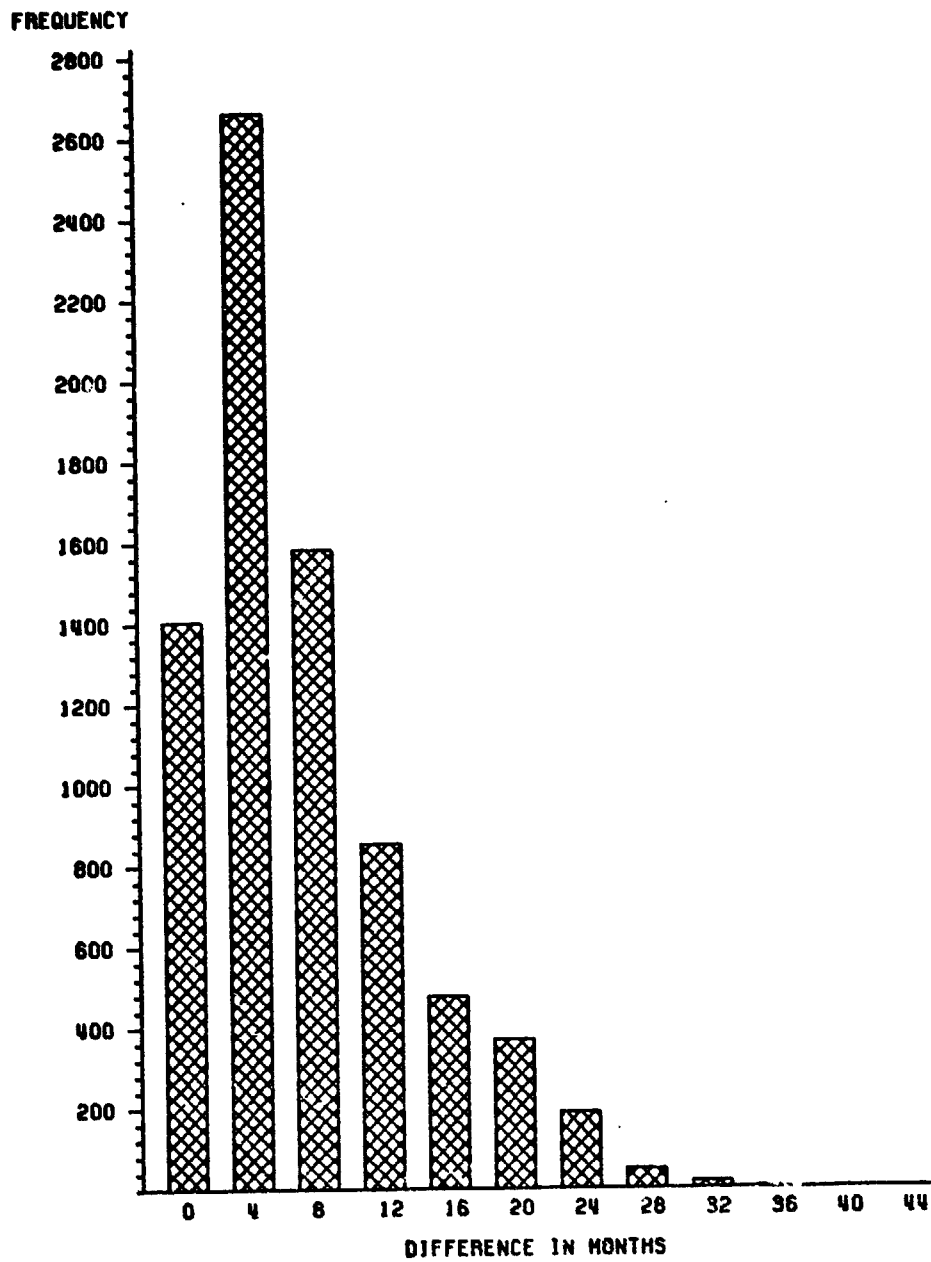


Figure 97: DR frequency of number of months between date logged in (OD) and verification date (VC).

ORIGINAL PAGE IS
OF POOR QUALITY

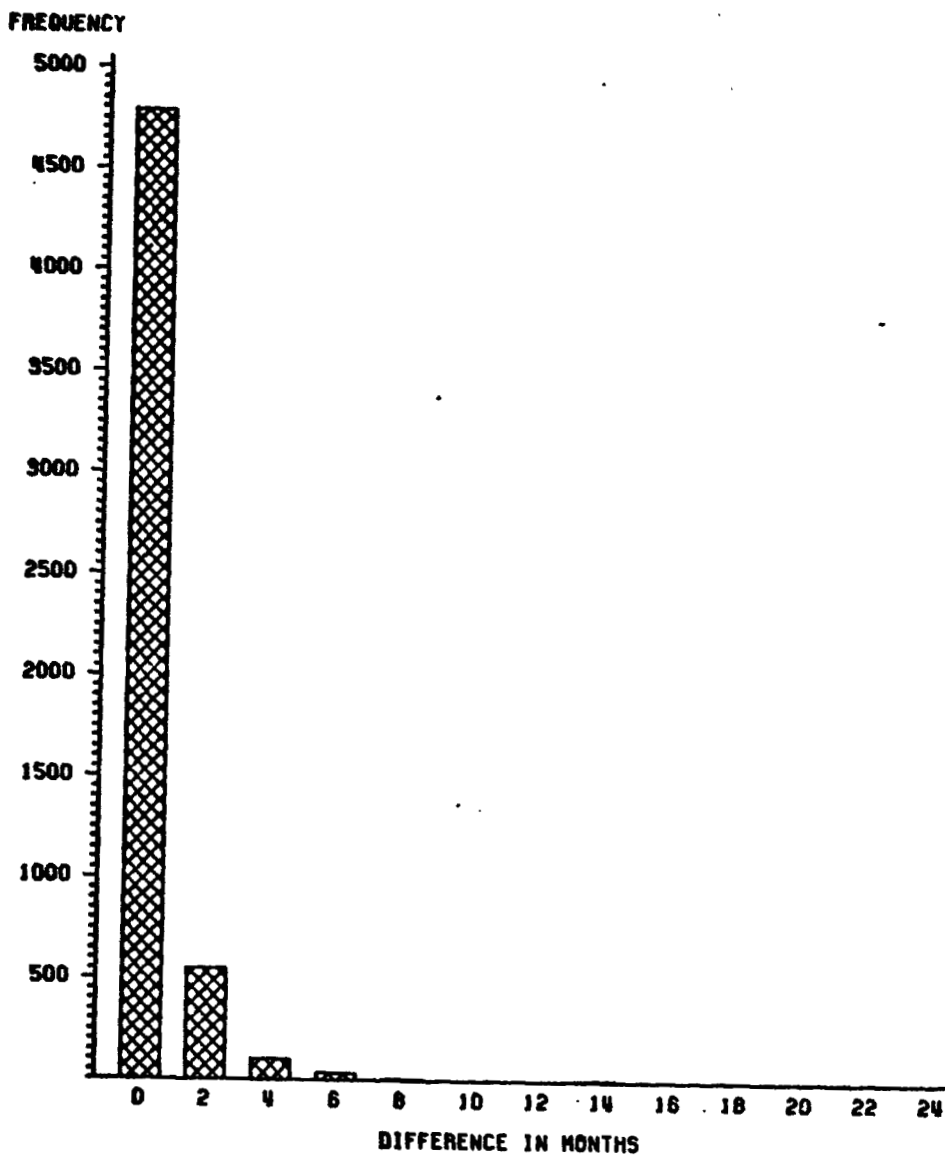


Figure 98: CR frequency of number of months between date received (DTRECD) and RRB disposition date (RRBDT).

ORIGINAL PAGE IS
OF POOR QUALITY

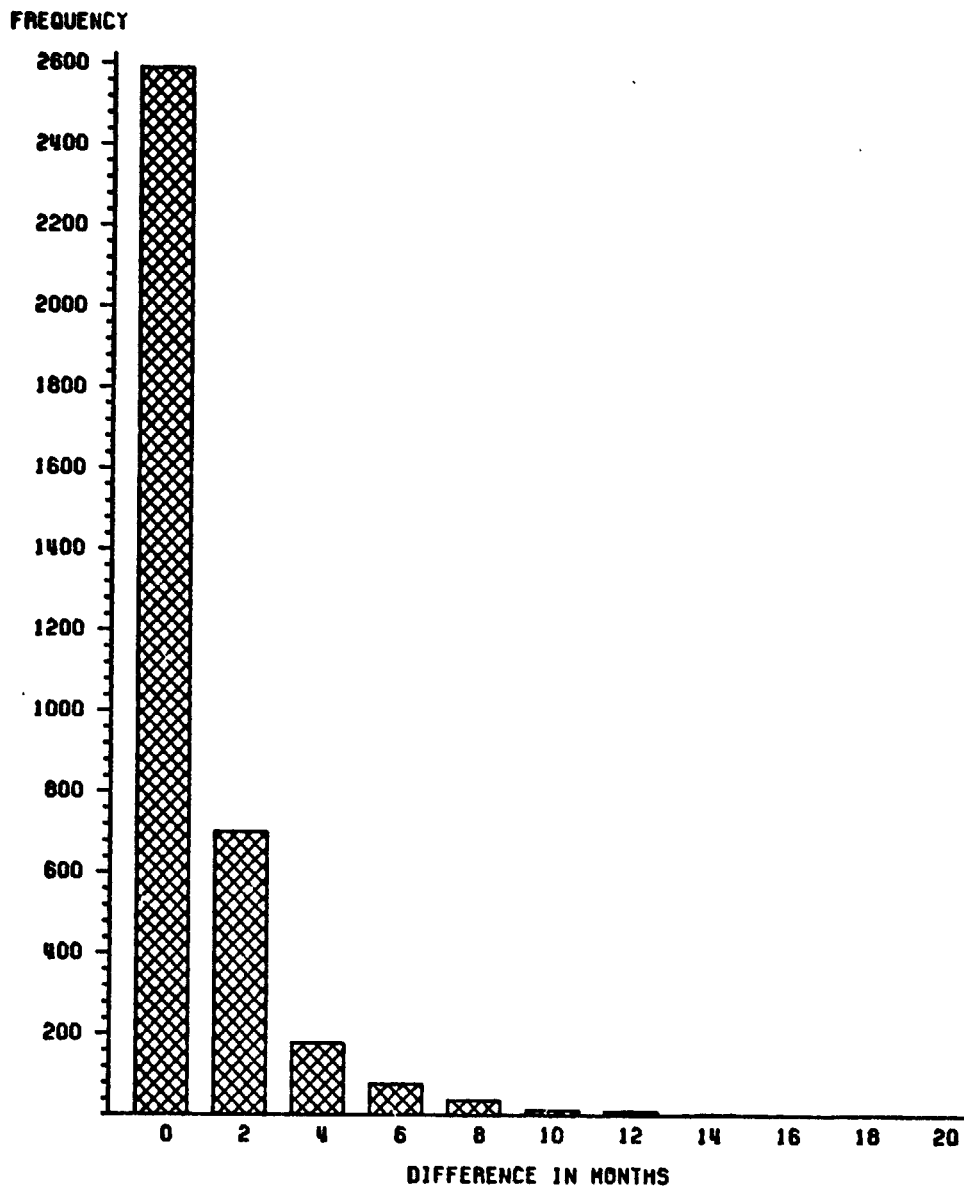


Figure 99: CR frequency of number of months between date received (DTRECD) and OASCB disposition date (OASCBDT).

ORIGINAL PAGE IS
OF POOR QUALITY

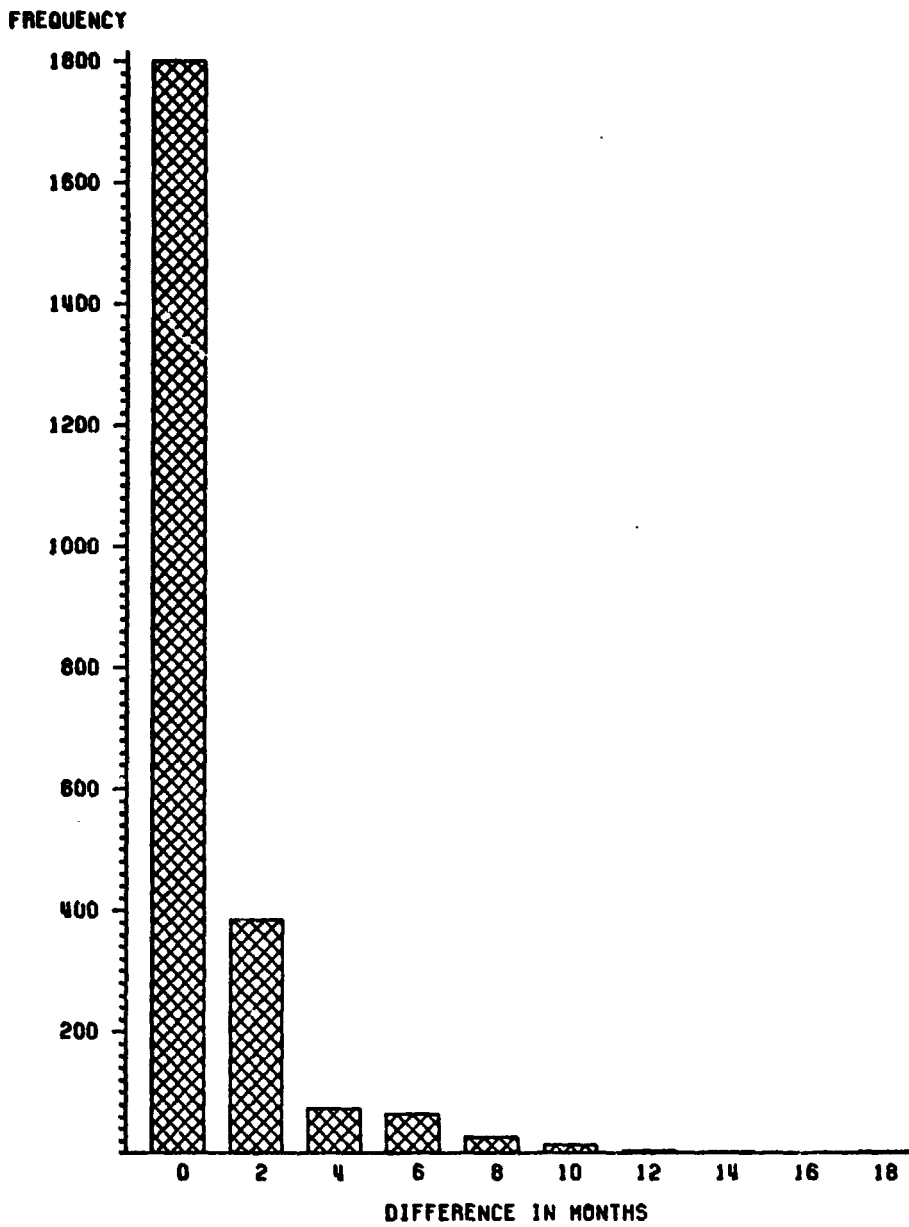


Figure 100: CR frequency of number of months between RRB disposition date (RRBDT) and OASCB disposition date (CASCBDT).

BIBLIOGRAPHY

1. Baker, Albert L. and Stuart H. Zweben, "A Comparison of Measures of Control Flow Complexity," IEEE Transactions on Software Engineering, vol. SE-6, no. 6 (November 1980), pp. 110-120.
2. Belady, L. A., "Complexity of Large Systems," In Alan Perlis, Frederick Sayward and Mary Shaw, editors, Software Metrics: An Analysis and Evaluation, MIT Press, Cambridge, Massachusetts, 1981, pp. 225-234.
3. Boehm, B. W., J. R. Brown, H. Kasper, M. Lipow, G. J. McLeod and M. J. Merritt, Characteristics of Software Quality, North-Holland Publishing Company, 1978.
4. Browne, J. C. and Mary Shaw, "Toward a Scientific Basis for Software Evaluation," In Alan J. Perlis, Frederick G Sayward, and Mary Shaw, editors, Software Metrics: An Analysis and Evaluation, MIT Press, Cambridge, Massachusetts, 1981, pp. 19-42.
5. Carver, Doris L. "The Effects of Complexity on COBOL Program Changes," Ph.D. dissertation, Computer Science Department, Texas A&M University, 1981.
6. Curtis, B., "Measurement and Experimentation in Software Engineering," Proceedings of the IEEE, September 1980.
7. Dunsmore, H. E. and J. D. Gannon, "Data Referencing: An Empirical Investigation," Computer, vol. 12, no. 12 (December 1979), pp. 50-59.
8. Duran, Joe W. and John J. Wiorkowski, "Capture - Recapture Sampling for Estimating Software Error Content," IEEE Transactions on Software Engineering, vol. SE-7, no. 1 (January 1981), pp. 147-148.
9. Gannon, J. D., "An Experimental Evaluation of Data Types on Programming Reliability," Special issue of Communications of the ACM on Language Design for Reliable Software, March 1977.
10. Halstead, M. H., Elements of Software Science, Elsevier Scientific Publishing Co., New York, 1977.
11. Henry, Sallie and D. Kafura, "Software Structure Metrics Based on Information Flow," IEEE Transactions on Software Engineering, vol. SE-7, no. 5 (September 1981), pp. 510-518.

12. Lehman, M. M., "Programs, Programming and the Software Life Cycle," Proceedings of the IEEE, September 1980.
13. McCabe, T. J., "A Complexity Measure," IEEE Transactions on Software Engineering, vol. SE-2, no. 4 (December 1976), pp. 308-320.
14. Mendis, Kenneth S., "Quantifying Software Quality," Quality Progress, vol. XV, no. 5 (May 1982), pp. 18-22.
15. Motley, R. M. and W. D. Brooks, "Statistical Prediction of Programming Errors," Rome Air Development Center, RADC-TR-77-175, May 1977.
16. Musa, John D., "Software Reliability Measures Applied to Systems Engineering," Proceedings of AFIPS 1979 National Computer Conference, (June 1979) pp. 941-946.
17. Myers, G. J., "An Extension to the Cyclomatic Measure of Program Complexity," SIGPLAN Notices, vol. 12, no. 10 (October 1977), pp. 61-64.
18. Ottenstein, L., "Quantitative Estimates of Debugging Requirements," IEEE Transactions on Software Engineering, vol. SE-5, no.3 (September 1979), pp. 504-514.
19. Perlis, Alan J., Frederick G Sayward, and Mary Shaw, editors, Software Metrics: An Analysis and Evaluation, MIT Press, Cambridge, Massachusetts, 1981.
20. Sammet, Jean E., "High Level Language Metrics," In Alan J. Perlis, Frederick G Sayward, and Mary Shaw, editors, Software Metrics: An Analysis and Evaluation, MIT Press, Cambridge, Massachusetts, 1981, pp. 131-142.
21. Schneidewind, N. F. and H. M. Hoffman, "An Experiment in Software Error Data Collection and Analysis," IEEE Transactions on Software Engineering, vol. SE-5, no. 3 (May 1979), pp. 276-286.
22. Shaw, Mary, G. T. Almes, J. M. Newcomer, B. K. Reid and W. H. Wolf, "A Comparison of Programming Languages for Software Engineering," Software -- Practice and Experience, 1980.
23. Shick, G. J. and R. W. Wolverton, "An Analysis of Competing Software Reliability Models," IEEE Transactions on Software Engineering, vol. SE-4, no. 3 (March 1978), pp. 104-120.
24. Thayer, Thomas A., Myron Lipow and Eldred C. Nelson, Software Reliability: A Study of Large Project Reality, North-Holland Publishing Company, Amsterdam, 1978.

25. Woodfield, Scott N., "An Experiment on Unit Increase in Problem Complexity," IEEE Transactions on Software Engineering, vol SE-5, no. 2 (March 1979), pp. 76-79.
26. Woodward, M. R., M. A. Hennell and David Hedley, "A Measure of Control Flow Complexity in Program Text," IEEE Transactions on Software Engineering, vol. SE-5, no. 1 (January 1979), pp. 45-50.
27. Zolnowski, Jean C., "A System for Measuring Complexity," Ph.D. dissertation, Computer Science Department, Texas A&M University, 1976.
28. Zolnowski, Jean. C. and Dick B. Simmons, "Measuring Program Complexity", Digest of Papers of Fall COMPCON '77, IEEE CAT. NO.77CH1258-3C. pp 336-340.