General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)

SER137

SOFTWARE ENGINEERING LABORATORY SERIES

SEL-81-012

(NASA-TM-84725) THE RAYLEIGH CURVE AS A MODEL FOR EFFORT DISTRIBUTION OVER THE LIFE OF MEDIUM SCALE SOFTWARE SYSTEMS M.S. Thesis - Maryland Univ. (NASA) 154 p HC A08/MF A01 CSCL 09B G3/61

N83-13840

Unclas 02148

THE RAYLEIGH CURVE AS A MODEL FOR EFFORT DISTRIBUTION OVER THE LIFE OF MEDIUM SCALE SOFTWARE SYSTEMS

DECEMBER 1981

National Aeronautics and Space Administration

Goddard Space Flight Center
Greenbelt, Maryland 20771

NTIF

SOFTWARE ENGINEERING LABORATORY SERIES

SEL-81-012

THE RAYLEIGH CURVE AS A MODEL FOR EFFORT DISTRIBUTION OVER THE LIFE OF MEDIUM SCALE SOFTWARE SYSTEMS

DECEMBER 1981

NASA
National Aeronautics and
Space Administration

Goddard Space Flight Center
Greenbelt, Maryland 20771

FOREWORD

The Software Engineering Laboratory (SEL) is an organization sponsored by the National Aeronautics and Space Administration, Goddard Space Flight Center (NASA/GSFC) and created for the purpose of investigating the effectiveness of software engineering technologies when applied to the development of applications software. The SEL was created in 1977 and has three primary organizational members:

NASA/GSFC (Systems Development and Analysis Branch)
The University of Maryland (Computer Sciences Department)
Computer Sciences Corporation (Flight Systems Operation)

The goals of the SEL are (1) to understand the software development process in the GSFC environment; (2) to measure the effect of various methodologies, tools, and models on this process; and (3) to identify and then to apply successful development practices. The activities, findings, and recommendations of the SEL are recorded in the Software Engineering Laboratory Series, a continuing series of reports that includes this document. A version of this document was originally drafted as a thesis in December 1981, and was also issued as University of Maryland Technical Report TR-1186.

The primary contributor to this document was

Gino O. Piccasso (University of Maryland)

Other contributors include

Victor Basili (University of Maryland)

Single copies of this document can be obtained by writing to

Frank E. McGarry
Code 582.1
NASA/GSFC
Greenbelt, Maryland 20771

Technical Report TR-1186 July 1982
NSG-5123

THE RAYLEIGH CURVE AS A MODEL FOR EFFORT DISTRIBUTION OVER THE LIFE OF MEDIUM SCALE SOFTWARE SYSTEMS*

Gino Picasso

Department of Computer Science
University of Maryland
College Park, MD 20742

Thesis submitted to the Faculty of the Graduate School
of the University of Maryland in partial fulfillment
of the requirements for the degree of
Master of Science
1982

*Research supported in part by National Aeronautics and Space Administration grant NSG-5123 to the University of Maryland. Computer time supported in part through the facilities of the Computer Science Center of the University of Maryland.

ORIGINAL PAGE IS
OF POOR QUALITY

ABSTRACT

Title of Thesis The Rayleigh Curve as a Model of Effort Distribution Over the Life of Medium Scale Software Systems

Putnam has shown that the Rayleigh curve is an adequate model for the life-cycle effort distribution of large scale systems. Previous investigations into the applicability of this model to medium scale software development efforts have met with mixed results. The results of these investigations are confirmed by analyses of runs and smoothing. The reasons for the models' failure are found in the subcycle effort data. There are four contributing factors: uniqueness of the environment studied, the influence of holidays, varying management techniques and differences in the data studied.

CONTENTS

1. INTRODUCTION 1

2. DESCRIPTION OF THE DATA EMPLOYED 4

3. EARLY WORK 7

4. ANALYSIS OF TOTAL WEEKLY EFFORT 13

4.1 Predictions of ta, yd, and K 13

4.2 Curve Fitting and Smoothing of Total Weekly Effort 16

4.3 General Trends in the Total Weekly Effort Distribution 30

5. SUBCYCLE DATA ANALYSIS 39

5.1 Curve Fitting and Smoothing of Subcycle Effort 40

5.2 Comparison of Putnam and SEL data 49

5.3 General Trends in the Total Weekly and Development Distributions 52

5.4 General Trends in the Subcycle Effort Distribution 58

6. USE OF SOFTWARE PARAMETERS TO CLASSIFY SYSTEMS 62

7. ANALYSIS OF COMPONENT DATA 73

8. CONCLUSION 80

9. BIBLIOGRAPHY 84

LIST OF TABLES

Table 1.	Project Data and Rayleigh Predictions	15
Table 2A.	Analysis of Runs for Mapp's Curves	19
Table 2B.	Analysis of Runs for Basili & Beane's Curves	20
Table 3.	Curve Fitting Results for Design, Coding and Test	42
Table 4.	Curve Fits for Coding Phase After Translation	44
Table 5.	Results of Calculations of the Difficulty Parameter	65
Table 6.	Ranking According to D1 Parameter	66
Table 7.	Comparison of Management's Ranking of Projects According to Difficulty and Ranking Obtained from D1	67
Table 8.	Calculations of the Methodology Parameter Csel	69
Table 9.	Ranking Projects According to Methodology Using Csel and Management Estimates	70
Table 10.	Comparison of Management's Ranking of Projects According to Methodology and Ranking Obtained from Csel	71
Table 11.	Total Number Existing Components in the System	75
Table 12.	Components Worked On/ Existing Components	76
Table 13.	Effort/ Components Worked On	77

LIST OF FIGURES

Figure 1.	PROJECT 1 -- Total Weekly Development Effort	23
Figure 2.	PROJECT 1 -- Total Weekly Development Effort Smoothed	24
Figure 3.	PROJECT 2 -- Total Weekly Development Effort	25
Figure 4.	PROJECT 2 -- Total Weekly Development Effort Smoothed	26
Figure 5.	PROJECT 2 -- Total Weekly Development Effort (Programmers)	28
Figure 6.	PROJECT 2 -- Total Weekly Develoment Effort Smoothed (Programmers)	29
Figure 7.	PROJECT 4 -- Total Weekly Develoment Effort	33
Figure 8.	PROJECT 4 -- Total Weekly Develoment Effort Smoothed	34
Figure 9.	PROJECT 3 -- Total Weekly Develoment Effort	36
Figure 10.	PROJECT 3 -- Total Weekly Develoment Effort Smoothed	37
Figure 11.	PROJECT 2 -- Develop Effort -- Translated	45
Figure 12.	PROJECT 2 -- Develop Effort -- Translated and Smoothed	46
Figure 13.	PROJECT 5 -- Develop Effort -- Translated	47
Figure 14.	PROJECT 5 -- Develop Effort -- Translated and Smoothed	48
Figure 15.	Putnam's Life-Cycle Diagram	51
Figure 16.	Sum of Rayleigh Curves Resulting in Rayleigh Curve	54
Figure 17.	Sum of Rayleigh Curves Resulting in two peaks	55

Figure 18. Sum of Rayleigh Curves Resulting in parabola

1. INTRODUCTION

Putnam has claimed that the Rayleigh equation accurately models the software life-cycle effort distribution of large projects. [Putnam1 Putnam2, Putnam3, Putnam4] He uses the derivative form of this equation-- y' = 2*K*A*t*exp(-a*t**2), to predict the man-power distribution over the life of a software system. This equation is fully determined by the parameters K and t. K is the total effort expended and td is the time to development or the time to peak effort (i.e., t = td at the curve peak). a relates to td by the formula a = 1/(2*td**2). The two parameters K and td can be estimated using Bayesian inference on the data gathered from previous programs.

From the Rayleigh curve Putnam derives several project parameters which help classify the system. These include difficulty, the state of technology of a software house (roughly a measure of its ability to do software development), and productivity. When these parameters have been determined for an installation and a particular system, feasibility regions for software development for the installation can be derived. Time-cost tradeoff curves can be drawn which management can use in decision making. Predictions of software size can also be made.

Putnam has also claimed that the individual subcycle curves of design and code, and test follow the Rayleigh curve. Putnam has indicated that the individual subcycle effort distributions when taken together and added result in the a Rayleigh shaped

project profile curve.

The Rayleigh model is very appealing because of its simplicity, management's familiarity with the parameters that determine the equation and the practical aids it provides for decision making. For this reason and because the Rayleigh curve is an adequate model for large scale software effort distributions, the SEL (Software Engineering Laboratory at the University of Maryland) chose to study the Rayleigh curve as a model of medium scale software effort distributions.

Basili and Zelkowitz [B-Z] studied the applicability of the Rayleigh model to medium scale software efforts. They tried using the model to predict total effort, maximum effort and time to acceptance testing. Mapp [M] continued this investigation. In addition, Mapp also compared the Rayleigh curve to other curves to determine whether or not the Rayleigh curve was indeed the underlying man-power curve for medium scale systems. Basili and Beane [BB1] compared the Rayleigh curve to the model proposed by Francis Parr [P] to determine which curve best described the man-power distribution of the smaller systems being studied. Basili and Beane checked to see whether or not the contractor's rule of thumb algorithm for project manning was being followed. [BB1]

All these efforts have given mixed results about the applicability of the Rayleigh curve to medium scale development efforts. Basili and Beane did indicate that the contractor's

algorithm was being used as a rough guideline by the managers. These results do not invalidate the Rayleigh curve as an optimal manning curve, but it cannot be clearly seen whether or not the Rayleigh curve is an adequate model for the man-power distribution of these development efforts.

In what follows, a more thorough investigation of the applicability of the Rayleigh curve is carried out. First the work done by previous investigators will be extended to determine whether or not the supposition that the Rayleigh curve fits the man-power distribution of medium scale systems is true. Trends in the data are studied to explain the deviations from the Rayleigh curve. These are looked into further by studying the effort distribution over subcycles. The possibility of using the Rayleigh curve to classify these systems is explored. Finally, other relations in the data are examined to try to find any invariants which may aid in smoothing and better understand effort distributions. One smoothing technique is used to elucidate the basic trends in the data.

A description of the data used for the previous studies done on medium scale systems is given first. A brief description of the work which has gone before and the conclusions they led to are also given. This is done in order to lay the foundation from which the rest of the study will be conducted.

2. DESCRIPTION OF THE DATA EMPLOYED

It will be helpful at this point to discuss the data used for this paper and in the work done by the previous researchers in SEL. [Basili, et. al.] gives a more thorough explanation of the forms used to collect the data.

The projects studied were primarily attitude control programs ranging in size between 45000 to 112000 lines of code and taking between 10000 to 24000 manhours to develop. The programs were developed by the Computer Science Corporation (CSC) for the National Aeronautics and Space Administration (NASA). The data was collected by the Software Engineering Laboratory (SEL) conducted jointly by NASA, CSC and the University of Maryland Computer Science Department.

Two forms were used to gather the effort data under study. The Resource Summary (RS) form was used for the studies mentioned earlier. It consists primarily of accounting information. The form is filled out at the end of each week by management and represents the actual charges made to the project. It contains the number of hours charged to the project by individual programmers, managers and support personnel for each week of the project. The Component Status Report (CSR) form, from which this study draws much of its information, consists of the actual number of hours spent by programmers on system development. Data is available on a weekly basis by component and phase. Each project is divided into three phases: design, code and test. Each of

these phases is divided into three subphases. Activities which do not fit into these categories are reported under miscellaneous charges which makes up a separate category. The effort data is only available from the start of design through acceptance testing.

The number of hours reported on the CSR forms are generally lower than the number of hours reported in the RS forms. This is to be expected since the data on the CSR forms represent the actual number of hours worked on a project, whereas, RS data represents the number of hours charged to a project. These do not necessarily match because overhead is incurred in hours not directly spent on development activities. The CSR forms probably reflect the actual number of hours programmers spent on a project more accurately than the RS forms. But, the accuracy of the CSR data is somewhat suspect. The CSR forms are filled out by many people (each individual involved on the project) resulting in reporting inconsistencies. The RS forms, on the other hand, were filled out by project managers (only one or two people filling out the form per project) thus making them more consistent. Furthermore, for a couple of projects, the CSR data for the early stages of the project is missing because the forms had not yet been made available.

The RS form, in so much as it consists of budget information, includes the total number of hours charged to the project or the total weekly effort expended by all personnel assigned to the project. The CSR form reports only effort expended in

particular components of the system and represents the number of hours directly expended in development. If the total number of hours reported in the CSR form for each week are added, what is obtained is the total effort expended directly on development activities with little overhead. In this paper the total weekly effort and the total weekly development effort are differentiated. The total weekly effort represents the effort reported on the RS forms and includes all charges made to the project including all overhead. The total weekly development effort is obtained from the CSR forms and represents the effort expended directly on development of a particular components in the system with very little overhead.

The first portion of this paper will center around the analysis of the data from the RS forms, the total weely effort. This is done as a follow on to the work done by previous investigators in SEL. The second portion is a study of individual phase effort distribution and how these relate to the total weekly effort distribution. Also the relation between components and effort is investigated. CSR forms are used to obtain this data.

3. EARLY WORK

The work presented here was done as a continuation of the studies conducted by the SEL at the University of Maryland. Much of the work has focused on three aspects of project manning and effort distribution models. First, effort distribution models have been used to predict the values of three principle parameters: total effort (K), peak effort or maximum man-power requirements (yd), and time to acceptance testing (ta). Secondly, the effort data has been studied to determine the underlying man-power patterns followed when developing medium scale software systems. This consisted in fitting various curve types to the effort data over time and comparing how well these modeled the effort distribution. Most recently the manning algorithm used by management has been checked against the actual effort distribution data. This has been done in order to determine how closely management actually adheres to their own "rule of thumb" for project staffing.

Basili and Zelkowitz were the first to study the applicability of the Rayleigh curve to medium scale development efforts. The data available to them did not match the data studied by Putnam. It did not include the early effort spent on requirements definition and the later effort spent on maintenance. Putnam had observed, however, that for large projects the design/code and test subcycles were Rayleigh in shape and that their sums were also Rayleigh. Basili and Zelkowitz assumed this to be true of medium scale development efforts as well. They reasoned that the

major central portion of the Rayleigh curve for the project profile should fit the design, code and test data well. Design, code and test subcycles are Rayleigh and their sums are too. Using the Rayleigh equation, they derived equations for the three quantities: ta, yd, and K. The equations which they obtained were:

ta = 1.25*K/yd

yd = 1.25*K/ta

K = .80*ta*yd

These three parameters are estimated at the beginning of the project. Taking two of the parameter estimates the third value was calculated using these equations. The predictions of time to acceptance were very good (3% error). This was a better estimate than management had given. Only two projects were used in this study however. The estimates obtained for the other two parameters were not as good.

Mapp derived a separate set of equations from the Rayleigh curve using a shaping factor, a = 1/(td**2). The equations he obtained were:

ta = 1.07*K/yd

yd = 1.07*K/ta

K = .93*ta*yd

Using the same procedure used by Basili and Zelkowitz, he obtained even better estimates for time to acceptance then they had.

Each of these equations determine a Rayleigh curve. When the Rayleigh curves corresponding to these six equations were obtained using management estimates of ta, yd and K, it was found that these did not seem to fit the data well. In fact, these curves were not even the best fitting Rayleigh curves. Since the Rayleigh curves which were responsible for the predictions did not fit the data well, it did not seem that the Rayleigh model was responsible for accurate predictions of time to acceptance.

The estimates obtained from the Parr curve, principally ta, were not any better than the Rayleigh curve predictions. It should also be noted that the parameters that determine the Parr curve are much more difficult to determine than the parameters that determine the Rayleigh curve. The results of these studies were inconclusive.

Attempts to find the underlying curve for man-power distribution were made. These consisted of fitting various curve types to data. Three separate efforts were made. Basili and Zelkowitz linearized the Rayleigh equation and did a least squares fit to the data. Mapp used the least squares method used by Basili and Zelkowitz and a simplified search method using the sum of errors squared as an optimization criteria to fit four curve types. He fitted the Rayleigh curve, a parabola, a trapezoid and a straight

line. Basili and Beane used Newton's method and the search method used by Mapp to fit a three parameter version of the Parr curve, a Rayleigh curve with a horizontal shift, a parabola and a trapezoid.

The three parameter Parr curve resulted in the best fit but, it was not significantly better than the other curves. Therefore it could not be concluded that the Parr curve was the best model. Basili and Beane supposed that the fluctuations present in the data made it impossible to determine the best fitting curve. Basili and Zelkowitz had made a similar observation earlier. In addition, Basili and Zelkowitz observed that because medium scale projects assume more of a step function man-loading curve it was difficult to determine where peak effort actually occurred. Where this peak is chosen to be makes a significant difference in the shape of the curve which is obtained.

Basili and Beane conducted a third type of study. They compared a "rule of thumb" staffing algorithm said to be used by management to the actual effort distribution data. This allowed them to determine whether or not management was indeed adhering to their "rule of thumb." The algorithm proposed by management was as follows:

1. At the start of the project assign 1/2 to 3/4 "full staffing" (due to lack of early funding and problems in finding available people).

2. At the end of the design phase, plus or minus a month, build to full staffing.

3. During the coding phase maintain full staffing.

4. During the testing phase:

   a. if the project is on schedule, decrease manning as appropriate.

   b. if the project is behind work, work overtime.

   c. if there are late changes to the user requirements increase manning by an additional 1/3.

As Basili and Beane pointed out, this algorithm would indicate that management has a great deal of flexibility in terms of staffing to handle problems when they arise. When the algorithm was checked against the data, it was seen that the data corresponded fairly well to the algorithm. Basili and Beane concluded that the hypothesis that management was indeed using this staffing algorithm could not be rejected.

These results do not favor adopting the Rayleigh curve as a model for medium scale development effort distribution. But, none of the studies have been conclusive and further investigation is warranted. In the following sections, the results of these investigations will be analyzed and extended. Explanations of the findings will be sought in the subcycle effort data. The assumptions made will also be checked. In addition, the Rayleigh

model will be used to classify projects in terms of difficulty and the results will be compared to management's ranking of these systems. The contractor's algorithm will also be reviewed in light of the Rayleigh curve and the results of the studies conducted.

4. ANALYSIS OF TOTAL WEEKLY EFFORT

In this portion of the paper the work done by previous investigators will be extended to determine whether or not the supposition that the Rayleigh curve fits the total weekly effort distribution is true.

First, the results of using the model to predict project parameters are studied further since previous results have been inconclusive. Factors which support or refute the Rayleigh model are sought. The Rayleigh curve is used to size projects. If the curve can be used in this fashion, this would lend support to the model. Secondly, curve fitting is attempted. If the Rayleigh curve fits the data well and it is a better fit than other curves, it would make it a likely model. Finally, an attempt at finding general trends in the weekly effort distribution is made to see what similarities can be found with the Rayleigh curve.

4.1 Predictions of ta, yd, and K

The equations used by Basili and Zelkowitz and Mapp were accurate in predicting time to acceptance testing (ta). However, even though these equations were derived from the Rayleigh equation it is not necessarily true that the accurate predictions are due to the Rayleigh model. In the first place, the predictions of the other parameters-- maximum manning (yd) and total effort (K), were not as good as ta. Also, the Rayleigh curves resulting from these predictions did not fit the data well and in fact were not the best fitting curves. Further evidence that accurate

predictions are coincidental rather than a product of the Rayleigh model is obtained by comparing the two sets of equations used: Basili's and Mapp's.

First it must be noted that the equations used by Mapp were not directly derived from the Rayleigh equation. Mapp used a shaping factor given by the expression a=1/(td**2). This is not the shaping factor defined by the Rayleigh equation. The shape factor (a) is obtained directly from the Rayleigh equation as follows.

y' = 2*K*a*t*exp(-a*t**2)

y'' = yd'/td = 2*K*a*exp(-a*t**2)*(1-2*a*t**2)

a = 1/(2*td**2)

Since the shape parameter is defined by the Rayleigh curve, the equations used by Mapp are not really derived from the Rayleigh curve. Yet the equations used by Mapp gave predictions of ta at least as good as Basili and Zelkowitz'. The results are summarized in Table 1.

		MANAGEMENT ESTIMATES	BASILI & ZELKOWITZ ESTIMATES	MAPP ESTIMATES	ACTUAL DATA
PROJ 1	ta	48	50	60	60.8
	yd	280	406	349	435
	K	15600	10752	12508	20302
PROJ 2	ta	39	58	50	47.8
	yd	280	417	356	509
	K	13000	8736	10189	16762
PROJ 3	ta	46	63	54	54.5
	yd	240	330	285	340
	K	12133	8832	10228	13288
PROJ 4	ta	48	62	53	60.8
	yd	280	361	310	489
	K	13867	9408	12508	14006
PROJ 8	ta	13	16	14	29
	yd	40	50	43	99
	K	520	384	484	947
PROJ 9	ta	65	63	54	***
	yd	120	117	97	148
	K	6067	5460	7312	4963

Table 1. Project Data and Rayleigh Predictions

ORIGINAL PAGE IS
OF POOR QUALITY

Since the equations derived from the Rayleigh equation do not give better predictions than those that are not, the Rayleigh curve would not seem to be responsible for the accurate predictions of ta. The fact that time to acceptance is predicted accurately would seem coincidental. Whether or not the Rayleigh model is responsible for the results obtained does not invalidate these findings. For the SEL environment these equations seem to work well. However, the model cannot be validated in this manner.

4.2 Curve Fitting and Smoothing of Total Weekly Effort

In this subsection, the work done by Mapp, Basili and Beane to find the best fitting curves for the total weekly effort data is extended to determine if a more definite conclusion can be drawn from their work. The work these researchers have done has been based on the supposition that if the Rayleigh curve is indeed the underlying man-power curve, then it would also be the best fitting curve. Their results have been mixed. They were not able to tell which curve best fit the data. An analysis of runs is used to see whether or not a best fitting curve can be selected from the set studied. Data smoothing is used to evaluate the fits and see whether or not better fits can be obtained.

A time sequence plot of residuals for each of the fitted curves obtained by Mapp and by Basili and Beane was made. An analysis of runs was performed to measure the goodness of fit of the calculated curves. The residuals were obtained by taking the difference between the effort in man-hours expended in week t

(the data) and the distribution curve evaluated at t. The analysis of runs consisted of the following. Assuming that the data is randomly distributed about the fitted curve, then it would be expected that there would be an approximately equal number of positive and negative residuals. For example, if there are 50 observations (data points) and the first 25 residuals are positive and the rest are negative, it is not likely that the data points are randomly distributed about the fitted curve. It is on this concept that the analysis of runs is based. A run is simply a grouping of either positive or negative residuals. In the example just given there are two runs. If the set of residuals exhibit the following pattern of signs, (---++---++--), then there are a total of five runs.

The number of runs should increase as the number of observations increases. If this is not the case, then it is unlikely that the points are randomly distributed about the fitted curve. Therefore the number of runs as a function of number of observations can act as a measure of goodness of fit. The question to be asked is: assuming the data points are randomly distributed about the fitted curve, what is the probability of getting this number of runs given n1+n2 observations, where n1 and n2 are the number of positive and negative residuals.

To obtain this probability a normal approximation to the actual discrete distribution was used. (*) The mean and the

(*) The discrete distribution referred to is defined as follows: given X number of sets of data points, each set containing n1+n2 points randomly distributed about a fitted curve, then

variance of the normal population was used to estimate the actual mean and variance. The mean and the variance of this distribution is given by:

mean = (2*n1*n2)/(n1+n2)+1

var =(2*n1*n2)*(2*n1*n2-n1-n2)/((n1+n2)**2)*(n1+n2-1)

The unit normal deviate (**) as given by:

z = (number of runs - mean+.5)/sqrt(var)

The results of the analysis are tabulated in Tables 2A and 2B. The number of positive and negative residuals, the number of runs and the unit normal deviate for each of the curves obtained by Mapp and by Basili and Beane are given.

---

each set will have some number of runs (Ri, i= 1,X) associated with it. The set of all Ri will form a discrete distribution about the expected value of R (number of runs) given n1+n2 data points. Since the data points are randomly distributed about the fitted curve, the discrete distribution formed by the set of Ri, should be approximated by a normal distribution. This mean of this normal distribution will be approximately equal to expected value (R) of this distribution.

(**) The unit normal deviate is an approximation of the standard deviation for the discrete distribution. The unit normal deviate is calculated using the mean and variance of the normal distribution which only approximate the mean and variance of the actual discrete distribution.

		RAYLEIGH CALCULATED	RAYLEIGH SEARCHED	PARABOLA CALCULATED	PARABOLA SEARCHED
PROJ 1	n1	33	46	32	40
	n2	45	32	46	38
	runs	10	14	7	17
	Z	-6.65	-5.54	-7.19	-4.90
PROJ 2	n1	33	26	31	25
	n2	28	35	30	36
	runs	14	17	16	15
	Z	-4.29	-3.30	-3.74	-3.87
PROJ 3	n1	27	31	28	31
	n2	31	27	30	27
	runs	16	18	11	18
	Z	-3.33	-2.79	-4.64	-2.79
PROJ 4	n1	24	27	24	25
	n2	29	26	29	28
	runs	13	13	10	13
	Z	-3.78	-3.74	-4.62	-3.76

NOTE: Assuming a normal distribution the probability that the value of Z is less than -1 is .16 (i.e., $P(Z < -1) = .16$).

Table 2A. Analysis of Runs for Mapp's Curves

ORIGINAL PAGE IS
OF POOR QUALITY

PROJ 1	n1	42	48	50	24
	n2	36	30	28	54
	runs	26	25	25	26
	Z	-3.04	-2.99	-2.82	-2.07
PROJ 2	n1	28	33	30	23
	n2	33	28	31	38
	runs	14	18	13	14
	Z	-4.37	-3.32	-4.65	-4.17
PROJ 3	n1	25	25	21	27
	n2	32	32	36	30
	runs	17	17	19	19
	Z	-3.14	-3.14	-2.31	-2.66
PROJ 4	n1	27	27	26	27
	n2	31	31	32	31
	runs	12	12	19	13
	Z	-4.62	-4.62	-2.73	-4.36

NOTE: Assuming a normal distribution the probability that the value of Z is less than -1 is .16 (i.e., $P(Z < -1) = .16$).

Table 2B. Analysis of Runs for Basili & Beane's Curves

ORIGINAL PAGE IS
OF POOR QUALITY

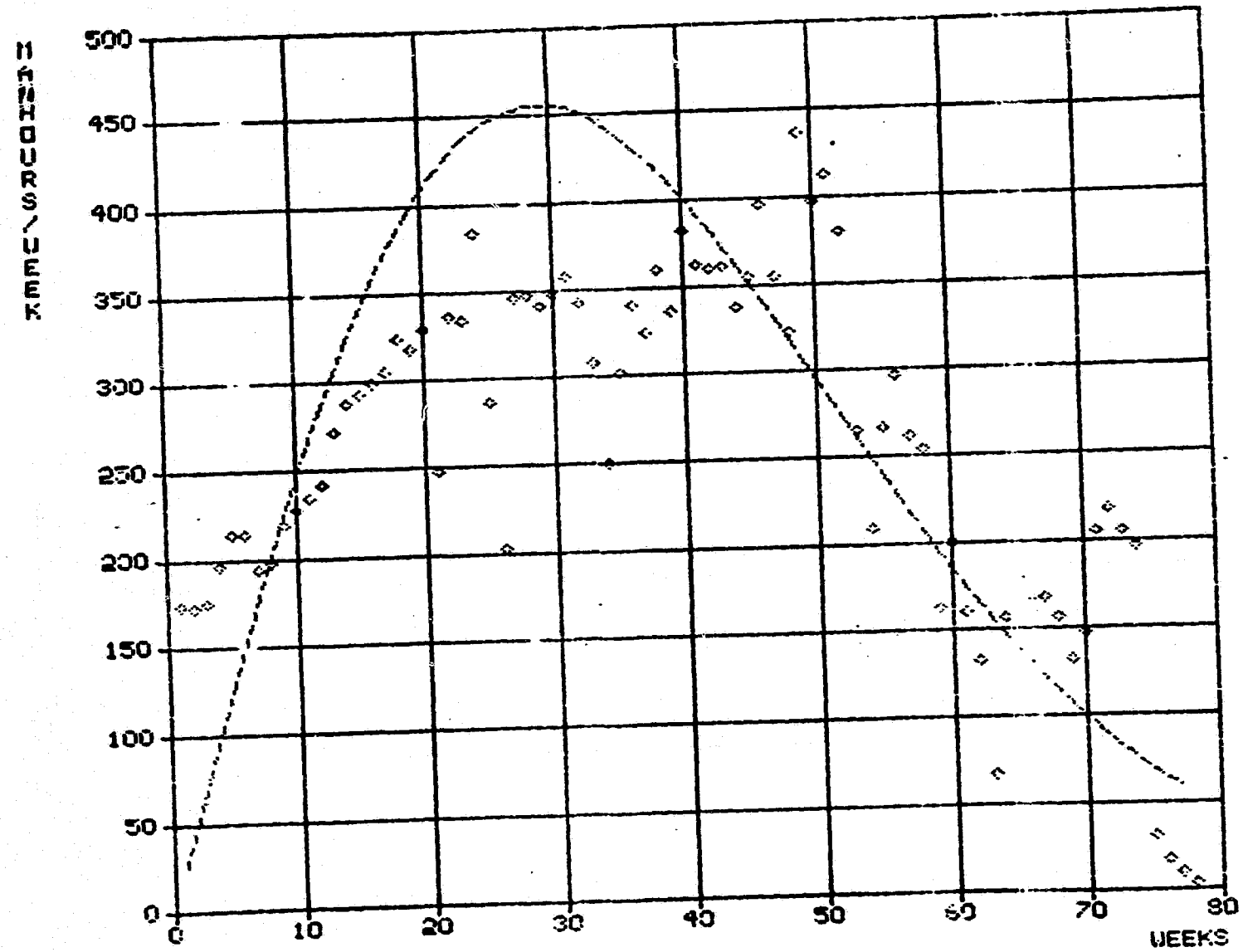
Assuming a normal distribution the probability that the value of Z is less than -1 is .16 ($P(Z<-1)=.16$). Evaluating the results in Tables 2A and 2B we can conclude with some confidence that the pattern of residuals is not random. This indicates that effort data is not randomly distributed about any of the fitted curves. This fact together with the evaluation of the sum of least squares obtained by Basili and Mapp indicates that none of the fitted curves are underlying curves for the total weekly effort data. The plots of the two best fitting Rayleigh curves (Projects 1 and 4) are given in Figures 1 and 3.

The data analyzed included manager and support personnel hours as well as programmer hours. In order to determine whether or not manager and support personnel hours could be responsible for introducing the deviations from the Rayleigh curve shape, programmer hours were isolated, plotted and fitted. The fitted curves did not seem to fit the programmer effort distribution any better.

Basili and Zelkowitz had observed that the total weekly effort data had contained a lot of noise, which was responsible for the deviations. The data was smoothed to determine to what extent noise in the data was responsible for the deviations from the Rayleigh curve. The smoothing was done by calculating a running average of five week intervals; the effort for the two weeks before and the two weeks after were averaged with the effort for the current week. This was done for each of the projects. The plots for projects 1 and 2 for the smoothed data are given in

figures 2 and 4. The best fitting Rayleigh curves were calculated for seven projects.

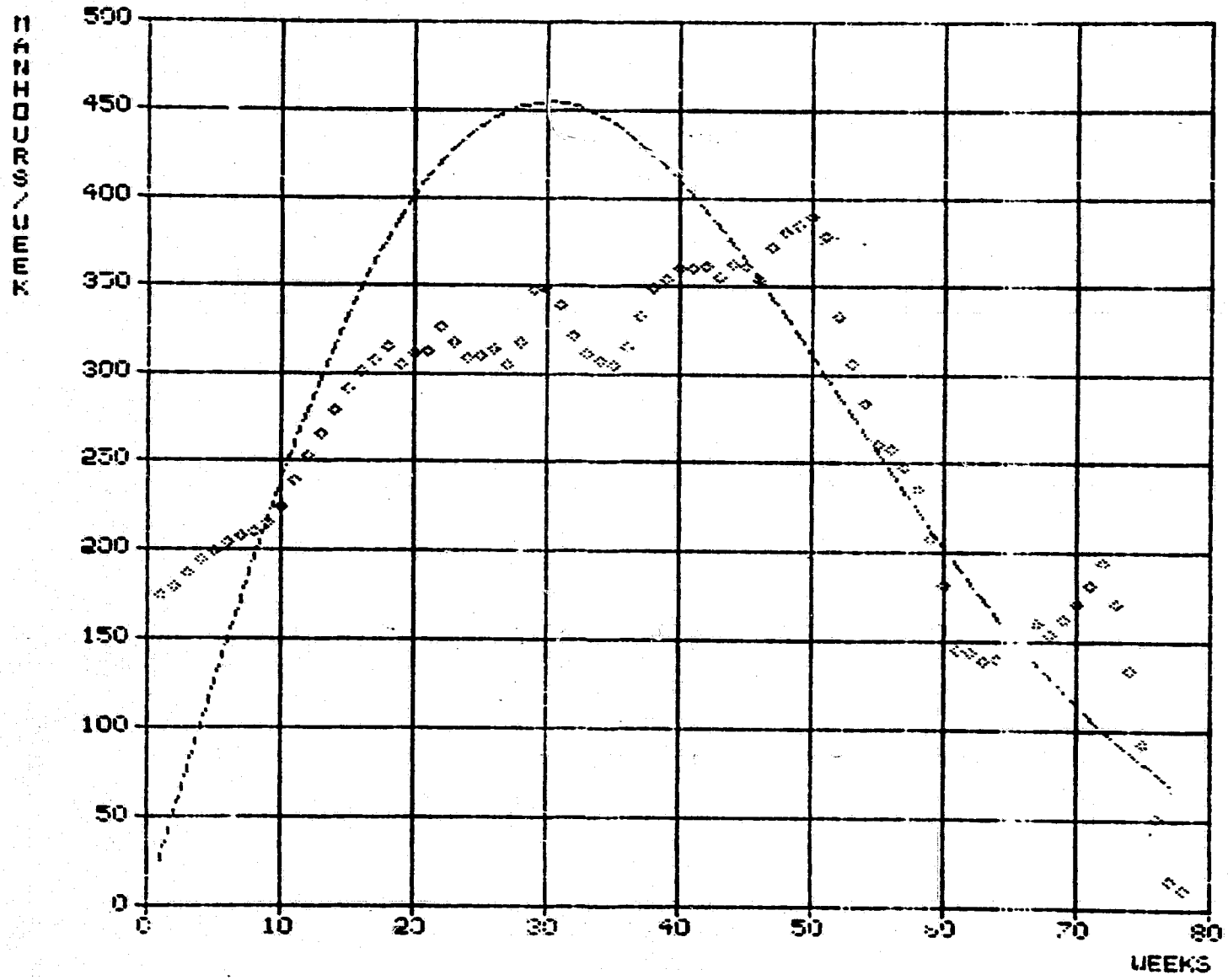
FIGURE 1. PROJECT 1 -- EFFORT VS TIME
WEEKLY TOTALS



ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

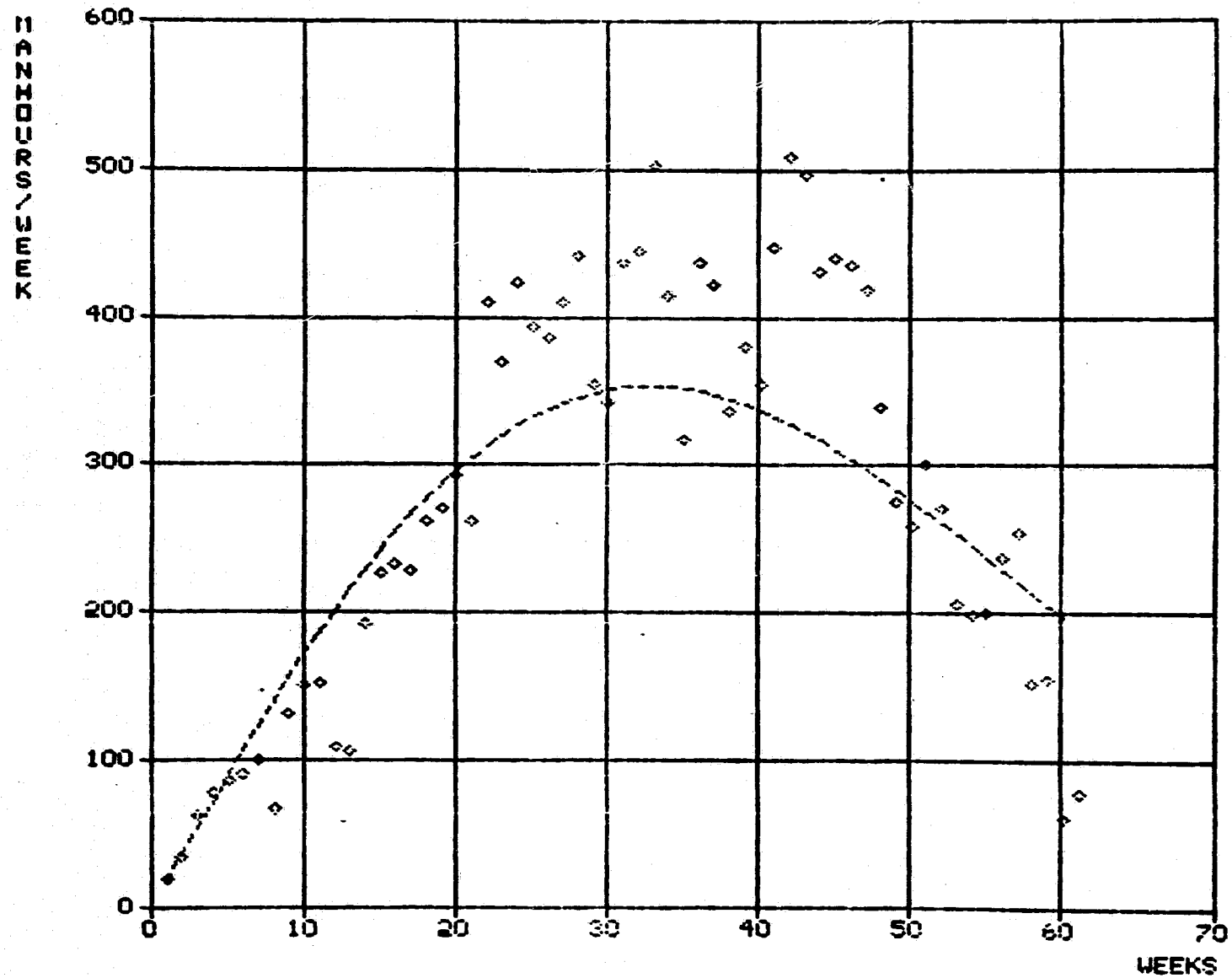
FIGURE 2. PROJECT 1 -- EFFORT VS. TIME
WEEKLY TOTALS -- SMOOTHED



ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE 3. PROJECT 2 -- EFFORT VS. TIME
WEEKLY TOTALS

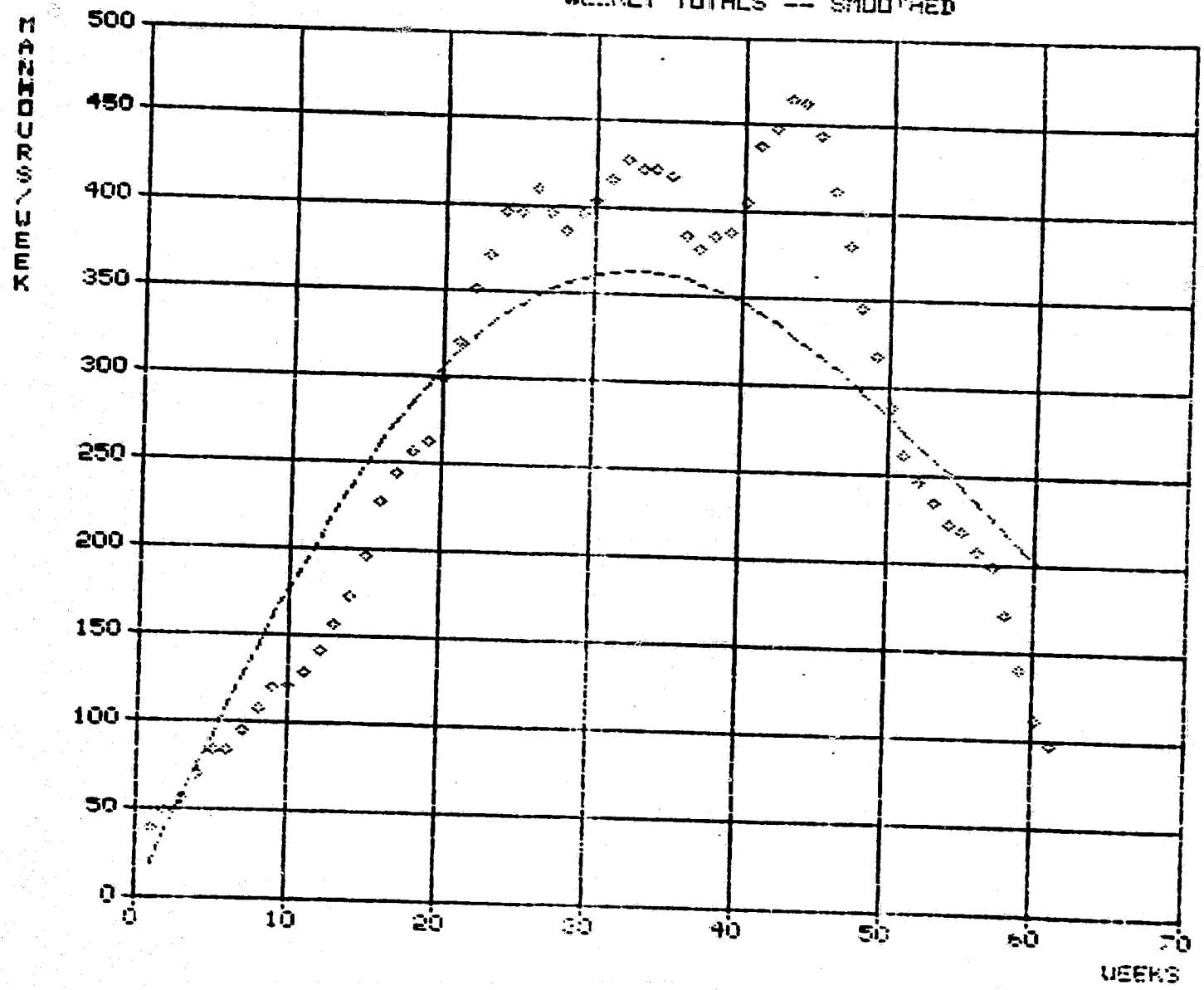


ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE 4.

PROJECT 2 -- EFFORT VS. TIME
WEEKLY TOTALS -- SMOOTHED

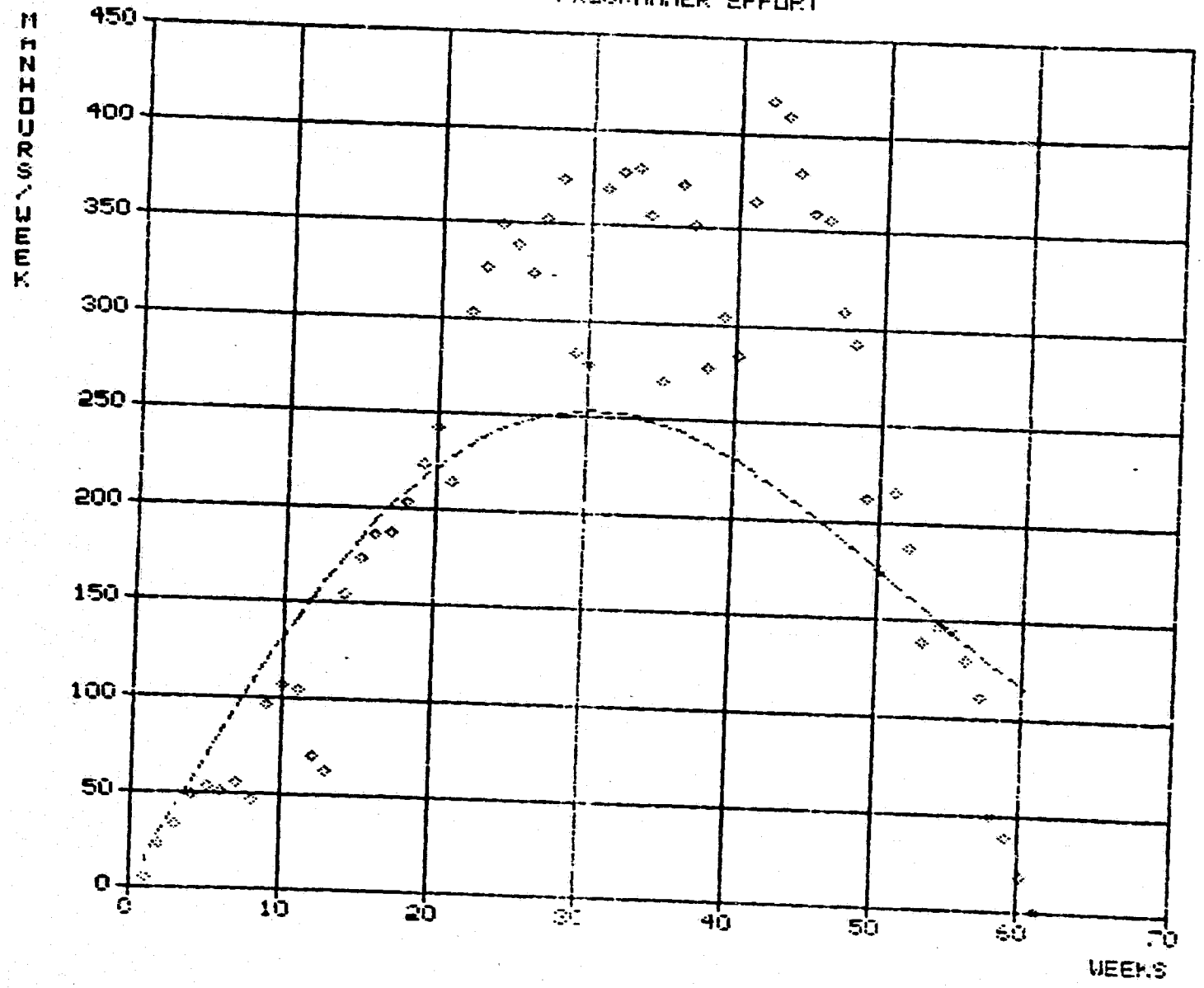


ORIGINAL PAGE IS
OF POOR QUALITY

The smoothing revealed that the data did not follow the Rayleigh curve. A simple visual inspection of the plots demonstrate this. An analysis of runs in this context is not meaningful since the data has undergone transformation and therefore is no longer random.

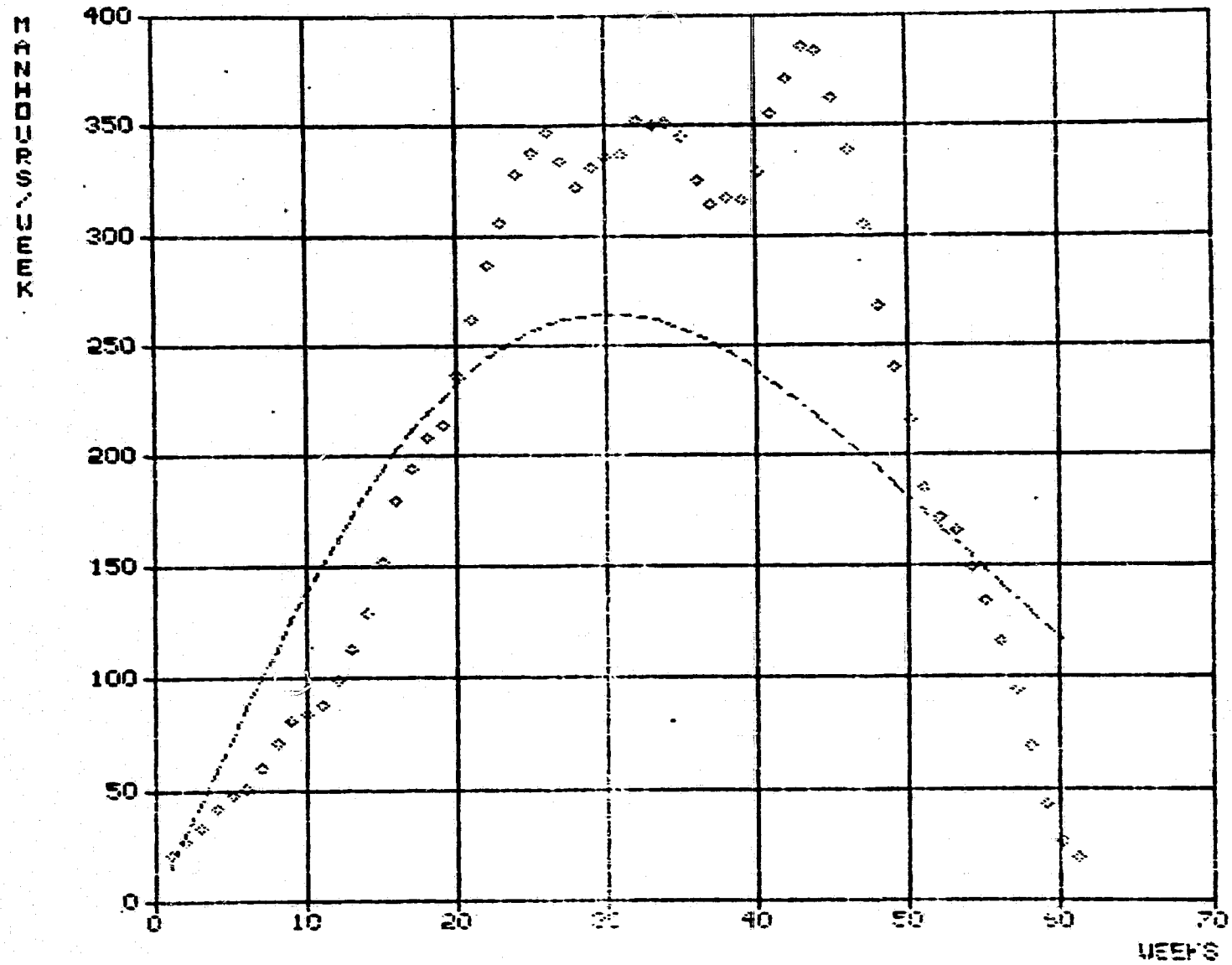
The same analysis was carried out for total weekly programmer hours and similar conclusions were drawn. The plots for project 2 are given in figures 5 and 6.

FIGURE 5. PROJECT 2 -- EFFORT VS. TIME
PROGRAMMER EFFORT



ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE 6. PROJECT 2 -- EFFORT VS. TIME
PROGRAMMER EFFORT -- SMOOTHED



ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

It is seen that a choice among the curves used by Mapp and Basili and Beane is impossible. None of the curves fit the data well enough. Two things give us evidence of this. First, a choice among the curves on the basis of the error criteria used (sum of errors squared) was not possible. This was observed by both of these researchers. Second, the analysis of runs indicate that the data points are not randomly distributed about the fitted curves. The sum of errors squared indicate that the fits are not good and the analysis of runs confirms this result.

The Rayleigh curve is very definitely not an adequate model for the total weekly effort data studied, but no other curve seems to do any better. In fact, it is impossible to state whether or not there is a single curve type which best fits all or even the majority of the projects studied. Attention is now turned to studying any trends in the data.

4.3 General Trends in the Total Weekly Effort Distribution

In this subsection, an attempt is made to find general trends in the data to see what similarities can be found with the Rayleigh curve. If the trends in the data can be explained and the deviations from the Rayleigh curve can be accounted for, then further study of the Rayleigh curve may help explain other behavior in the effort distribution of medium scale software projects. The data is smoothed to make the trends easier to identify.

Two major trends are observed. First, every project exhibited several peaks in total effort expended. This is in contrast

to the single peak representing maximum manning exhibited by the Rayleigh curve. Second, the "more successful" (*) projects all seemed to have the same man-power pattern - a quick rise, a series of peaks, followed by a steady decline. The "less successful" projects did not exhibit this behavior. The Rayleigh curve suggests a rise, peaking and exponential tail off pattern for "successful" projects.

The first of these trends could be explained by the occurrence of holidays. Basili and Zelkowitz had noted that a significant and very noticeable decline in effort occurred during the holidays. For project 2 (figures 3 and 4), Christmas and New Years occurred on weeks 12 and 13, Easter in week 28 and Independence Day fell in week 39. These holidays match up exactly with effort slow downs observed in the data. The same observations were made for project 1 (figures 1 and 2), and for the five other projects studied. The slow downs can therefore be reasonably attributed to employees taking holidays. This acts as noise in the data.

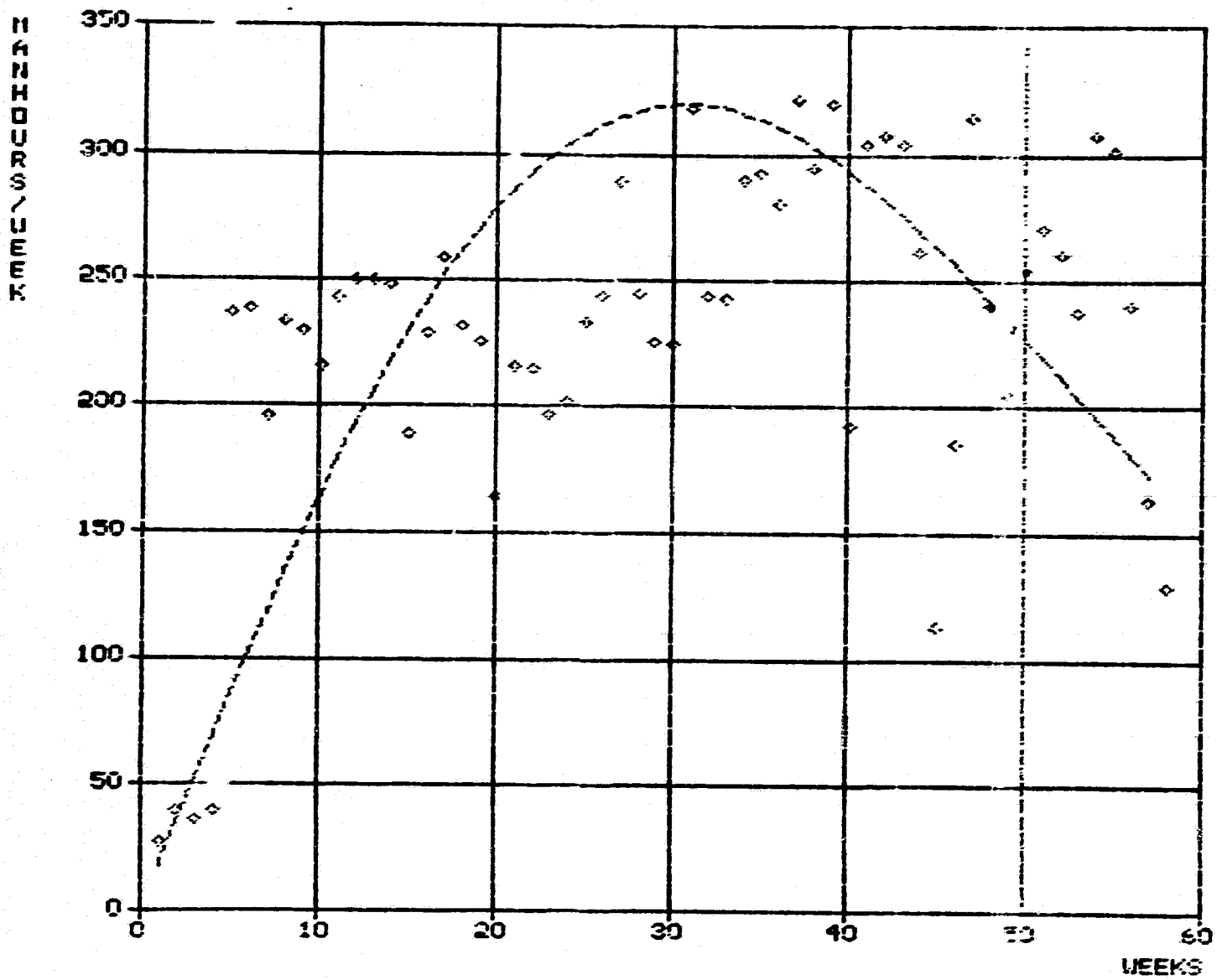
The reason that Putnam did not observe any effect from holidays in the large projects he studied is because in large projects effort expenditure is gathered on a monthly or yearly basis instead of a weekly basis. This causes the effects of the holidays to go unnoticed. Since the presence of multiple peaks in the effort data can be explained by the "noise" due to holidays, it

* Classification of projects as more or less "successful" is based on a subjective evaluation made by management.

can be concluded that the underlying curve for this man-power data should have a single peak.

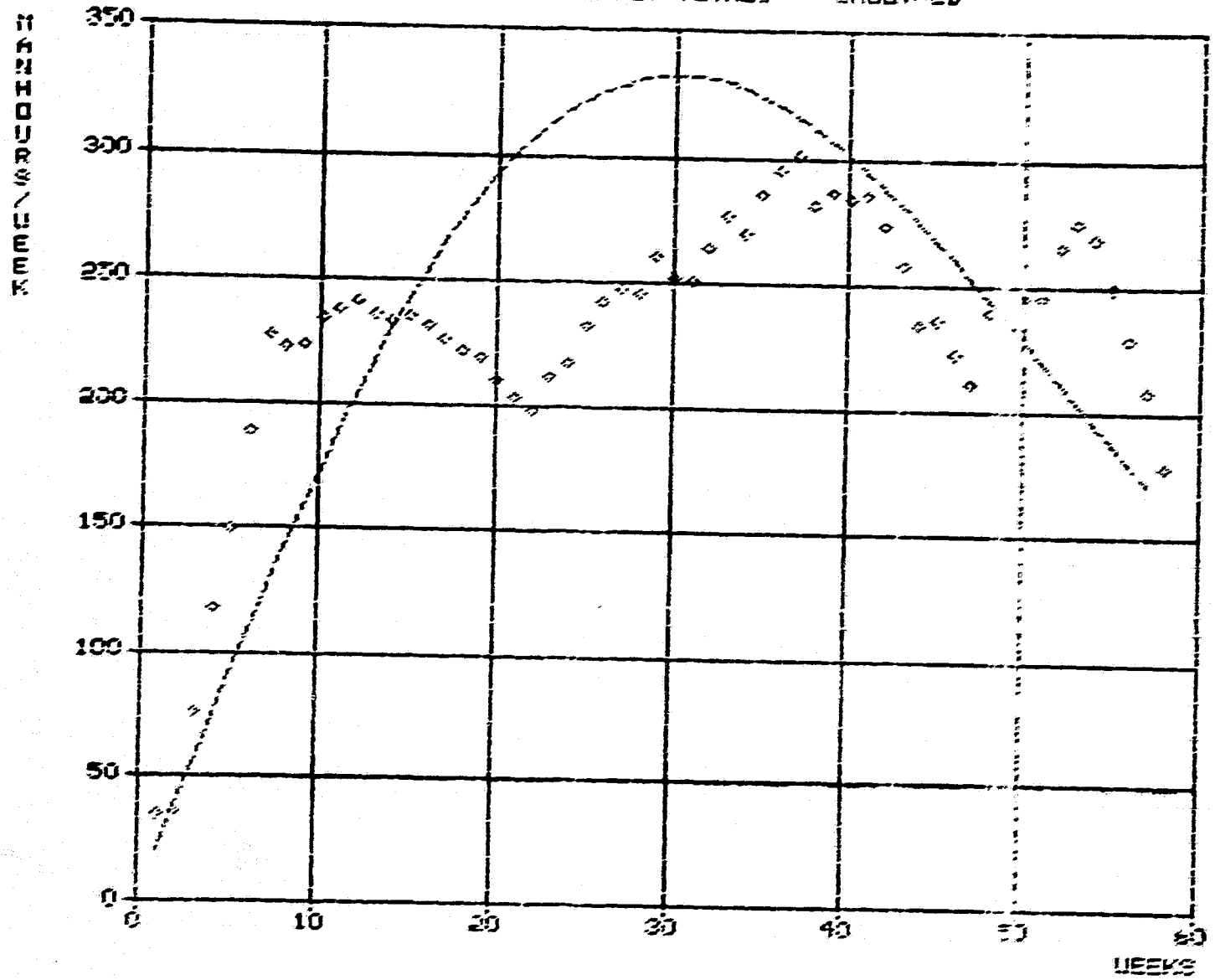
The second observation concerns the general shape of the man-power curves for projects regarded as "more successful" by management. It was observed that the "more successful" projects exhibited man-power patterns characterized by a rise in effort, followed by several peaks and a steady decline. In contrast, a "less successful" project exhibited a slower man-power build-up, peaking very close to delivery, followed by a very sharp decline. (This can be explained by the need to finish quickly. An attempt is made to deliver a project on time by adding more manpower.) If the noise due to holidays is smoothed, the "more successful" projects would exhibit a man-power pattern of rise, peak and decline. The behavior characteristic of the "successful" projects is exhibited by both projects 1 and 2 (figures 1 - 4), while project 4 (figures 7-8) is an example of a "less successful" project.

FIGURE 7. PROJECT 3 -- EFFORT VS TIME
WEEKLY TOTALS



ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE 8. PROJECT 3 -- EFFORT VS. TIME
WEEKLY TOTALS -- SMOOTHED



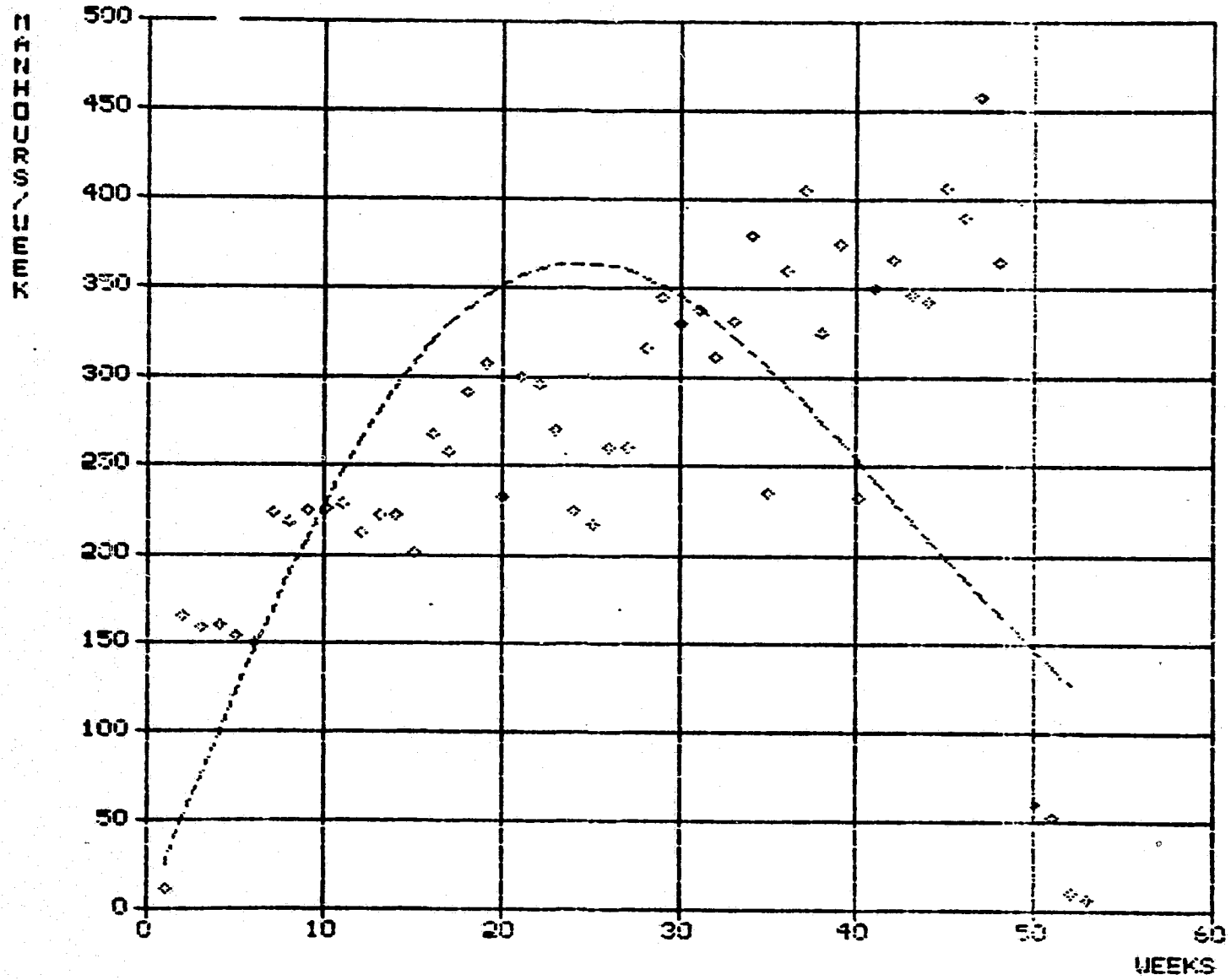
ORIGINAL PAGE IS
OF POOR QUALITY

These examples are insufficient to support any conclusion. But Putnam has made similar observations for the projects he studied. Putnam states,

> "Many of these also exhibited the same basic man-power pattern- a rise, peaking and exponential tail off as a function of time. Not all systems follow this pattern. Some man-power patterns are nearly rectangular; that is, a step increase to peak effort and a nearly steady effort thereafter. There is reason for these differences. It is because man-power is applied and controlled by management. Management may choose to apply it in a manner which is suboptimal or contrary to system requirements. Usually, management adapts to system signals, but generally responds late because the signal is not clear instantaneous with the need." (Putnam1, pg.348)

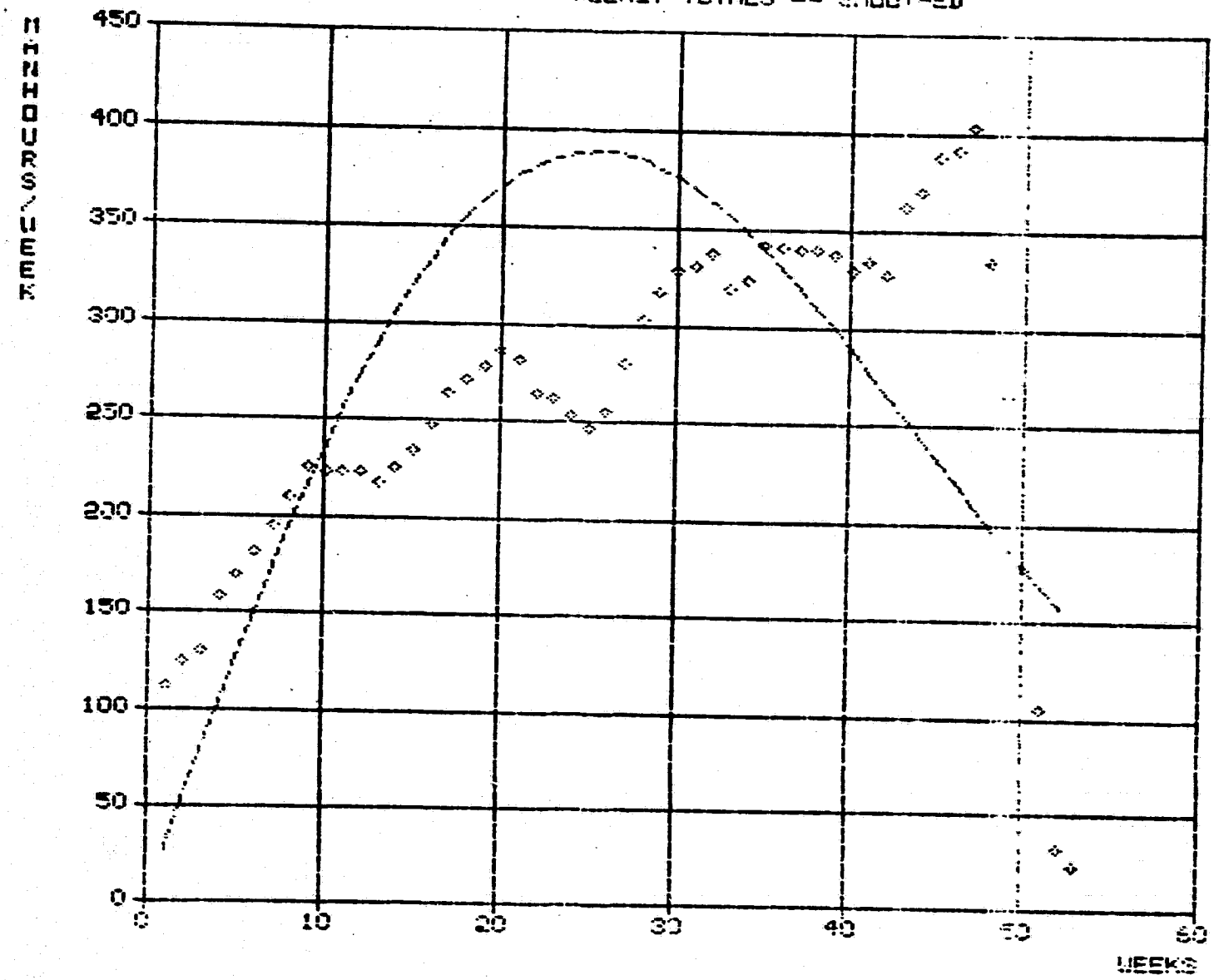
This suggests that the optimal manloading curve follows a pattern similar to that suggested by the "more successful" projects. Project 3 (figures 9 and 10) which was considered a "successful" project exhibits some semblance of a rectangular pattern as Putnam describes. These factors can explain why the Rayleigh curve does not model the data well. Other explanations are given in the next section.

FIGURE 9. PROJECT 4 -- EFFORT VS TIME
WEEKLY TOTALS



ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE 10. PROJECT 4 -- EFFORT VS. TIME
WEEKLY TOTALS -- SMOOTHED



ORIGINAL PAGE IS
OF POOR QUALITY

Thus far it has been shown that the Rayleigh curve is not responsible for the accurate predictions of the time to acceptance testing (ta) obtained by Basili and Zelkowitz and by Mapp. Since the Rayleigh curve did not fit the total weekly effort data well, it cannot be said that the Rayleigh curve is an adequate model for this data. However, there are some explanations for why the Rayleigh curve does not adequately fit the data (the effect of holidays on the effort distribution of small scale software projects). Also, there is some suggestion that the effort distributions for "successful projects" do follow a pattern similar to the Rayleigh curve (a rise, peaking and decline). Therefore, further investigation of the Rayleigh curve may prove helpful in determining some of the characteristics of the effort distribution for medium scale software projects.

ORIGINAL PAGE IS
OF POOR QUALITY

5. SUBCYCLE DATA ANALYSIS

In this section, reasons for the Rayleigh curve's failure to model adequately the total weekly effort data are investigated. Specifically, the effort distribution over individual subcycles is studied to gain further insight into the behavior of man-power distribution curves.

Four assumptions have been made in previous research. Based on Putnam's claims for large projects, it was assumed that the subcycles of design, code and test are Rayleigh in shape and that their sum is also Rayleigh. It was also assumed that the data used by Putnam and the data gathered at SEL differed only in two respects: the size of the projects studied and the phases of the life-cycle effort for which data was gathered. In addition, it was assumed that the subcycles for medium scale projects were distributed in the same fashion as large projects and that the effect of adding these subcycles would result in a similar total effort distribution. Finally, it was implicitly assumed that the manner in which large projects are developed is similar to the development of medium scale systems. These assumptions are checked to determine the adequacy of the Rayleigh curve as a model for this environment.

The subcycle effort distributions are studied to determine whether or not these are Rayleigh in shape. Differences between the two sets of data are studied further to see if the Rayleigh equation is being applied to the type of effort data it was

intended to model. The effect of holidays and of relative milestones are examined to determine how the summing of subcycle effort distributions for medium scale systems is different than that for large scale systems. And, general trends in the subcycle effort distributions are studied in order understand the dynamics of medium scale system development. Possible explanations for the model's failure are set forth.

5.1 Curve Fitting and Smoothing of Subcycle Effort

Basili and Zelkowitz initially hypothesized that since the SEL data only included the effort from design through acceptance testing at least the central portion of the Rayleigh curve should serve as an adequate model for effort distribution. They based this hypothesis on Putnam's claim that for large scale software systems the design/code and testing subcycles are Rayleigh in shape and their sum is also Rayleigh. The hypothesis is tested here in light of this underlying assumption. The design, develop (code) and test subcycles are smoothed and fitted with the Rayleigh curve to determine how well the Rayleigh curve models effort distribution over the subcycle.

The time spent each week on design, coding and test were calculated using the Component Status Report (CSR) data. This data was then smoothed and plotted. Because of the large volumes of the plots resulting for the seven projects used all of them could not be included here. Some sample plots are given in Appendix A. The best fitting curve for all three phases of each pro-

ORIGINAL PAGE IS
OF POOR QUALITY

ject was found using the linear least squares method used by Mapp. Mapp had noticed that the curves obtained using this method were not the best fitting curves. This was due to the fact that when the data is linearized it is distorted. This distortion is made more pronounced when there is a large variance in the magnitudes of the data being fitted. This was not the case for the subcycle effort data. Therefore this method was regarded as adequate for this application. The results of the curve fits and the analysis of runs are given in Table 3.

ORIGINAL PAGE IS
OF POOR QUALITY

	DESIGN	DESIGN SMOOTHED	CODING	CODING SMOOTHED	TEST	TEST SMOOTHED
PROJ 1						
n1	15	20	24	30	28	28
n2	39	34	30	24	26	26
runs	6	3	27	8	25	7
Z	-5.33	-6.64	0.139	-5.15	-0.413	-5.37
PROJ 2						
n1	17	21	19	25	17	20
n2	29	25	27	21	29	26
runs	12	7	11	3	5	3
Z	-3.04	-4.66	-3.54	-5.86	-5.29	-5.92
PROJ 3						
n1	19	27	22	25	18	29
n2	36	28	33	30	37	26
runs	15	7	17	10	19	7
Z	-2.86	-5.58	-2.70	-4.80	-1.70	-5.60
PROJ 4						
n1	11	16	19	24	20	21
n2	29	24	21	16	20	19
runs	11	7	10	5	15	3
Z	-1.82	-4.18	-3.05	-4.84	-1.76	-5.30
PROJ 5						
n1	17	18	13	13	17	12
n2	12	11	16	16	19	17
runs	6	4	7	3	9	5
Z	-3.32	-3.82	-2.87	-4.40	-1.89	-3.71

NOTE: Assuming a normal distribution the probability that the value of Z is less than -1 is .16 (i.e., $P(Z < -1) = .16$).

Table 3. Curve Fitting Results for Design, Coding and Test

The analyses of runs indicate deviations of the data from the Rayleigh curve. Visual inspection of the plots showed that none of the subcycle effort distributions followed a Rayleigh curve with the possible exception of the coding effort data. However, the rayleigh curve that best fit the coding data did not go through the origin. Since the form of the Rayleigh curve used to fit the data had to go through the origin it did not result in the best fitting Rayleigh curve. Therefore, to improve the fits obtained for the coding phase, a coordinate translation was performed and the resulting data was fitted using the same method. Table 4 presents the parameters of the resulting Rayleigh equations and the analysis of runs for five of the projects. No coordinate translation was done for project 4 because it was not necessary. The fits were not available for project 7. Figures 11 through 14 give the plots for two of the projects which exhibited the closest fit.

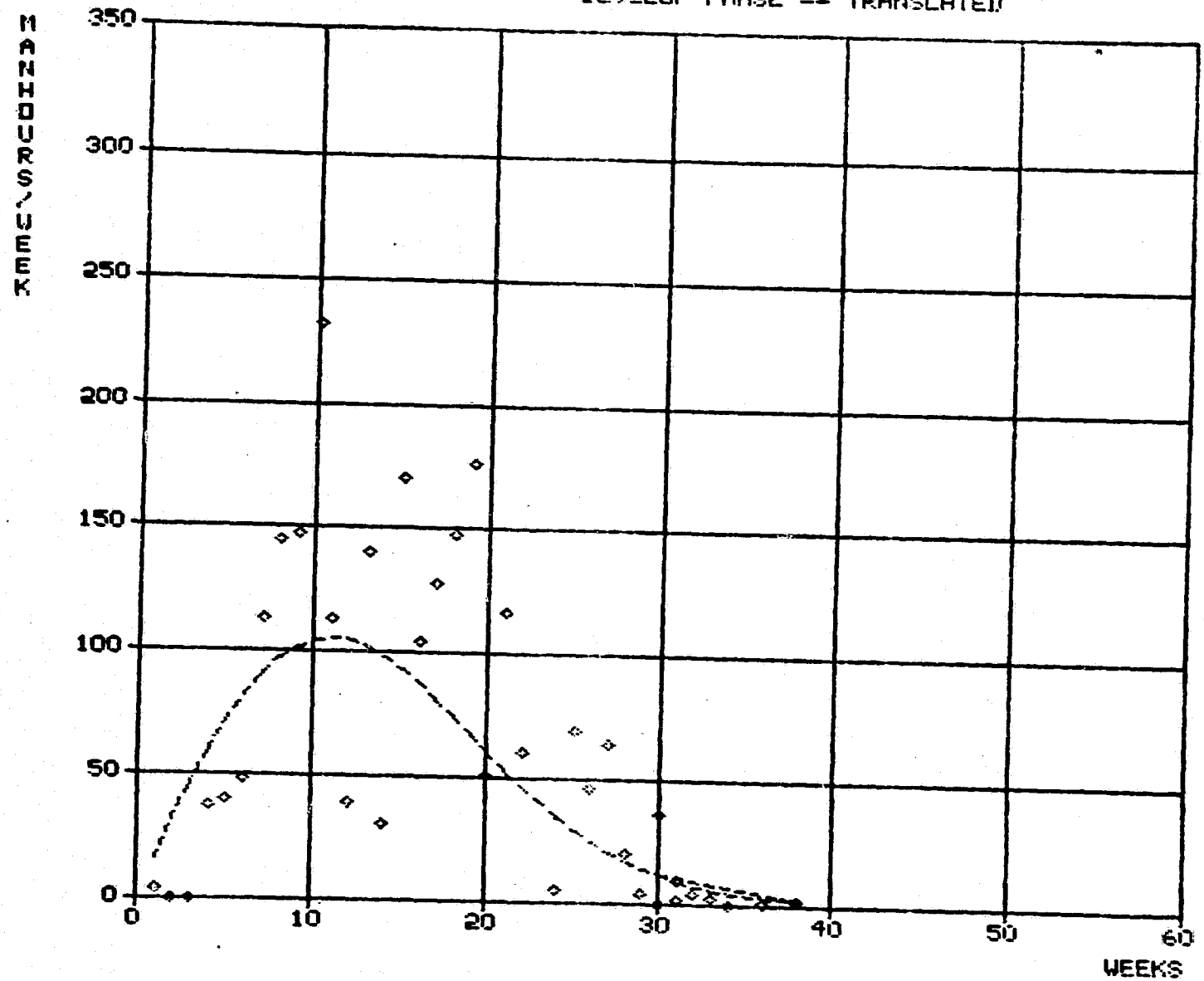
ORIGINAL PAGE IS
OF POOR QUALITY

CODING WITH TRANSLATION		CODING SMOOTHED	CODING SMOOTHED
PROJ 1	n1	12	11
	n2	16	17
	runs	7	3
	Z	-2.56	-4.25
PROJ 2	n1	20	26
	n2	16	10
	runs	15	3
	Z	-0.86	-4.88
PROJ 3	n1	21	21
	n2	22	22
	runs	16	9
	Z	-1.70	-3.86
PROJ 4	n1		
	n2	NOT AVAILABLE	NOT AVAILABLE
	runs		
	Z		
PROJ 5	n1	13	10
	n2	7	10
	runs	12	5
	Z	1.27	-2.52

NOTE: Assuming a normal distribution the probability that the value of Z is less than -1 is .16 (i.e., $P(Z < -1) = .16$).

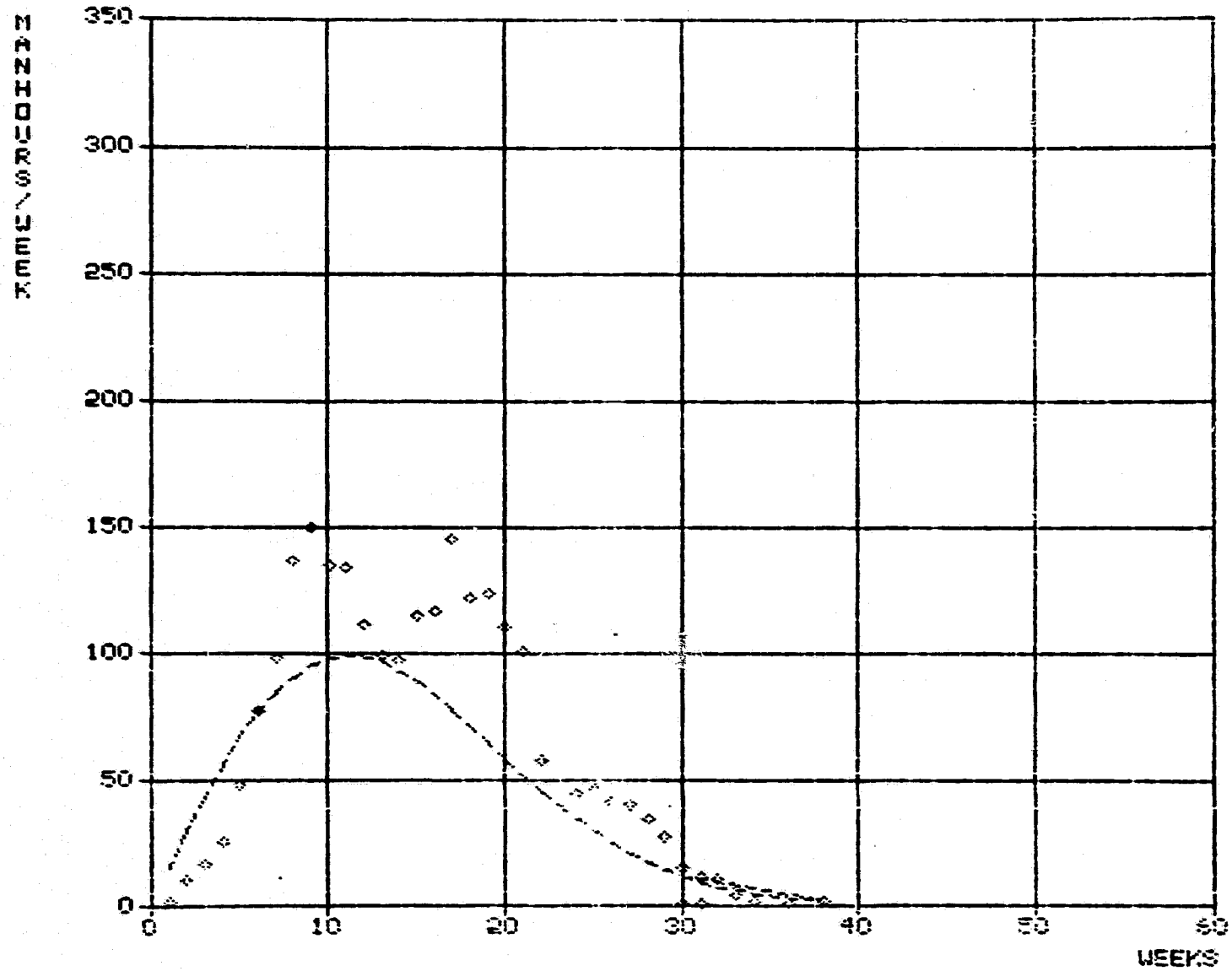
Table 4. Curve Fits for Coding Phase After Translation

FIGURE 11. PROJECT 2 -- EFFORT VS. TIME
DEVELOP PHASE -- TRANSLATED



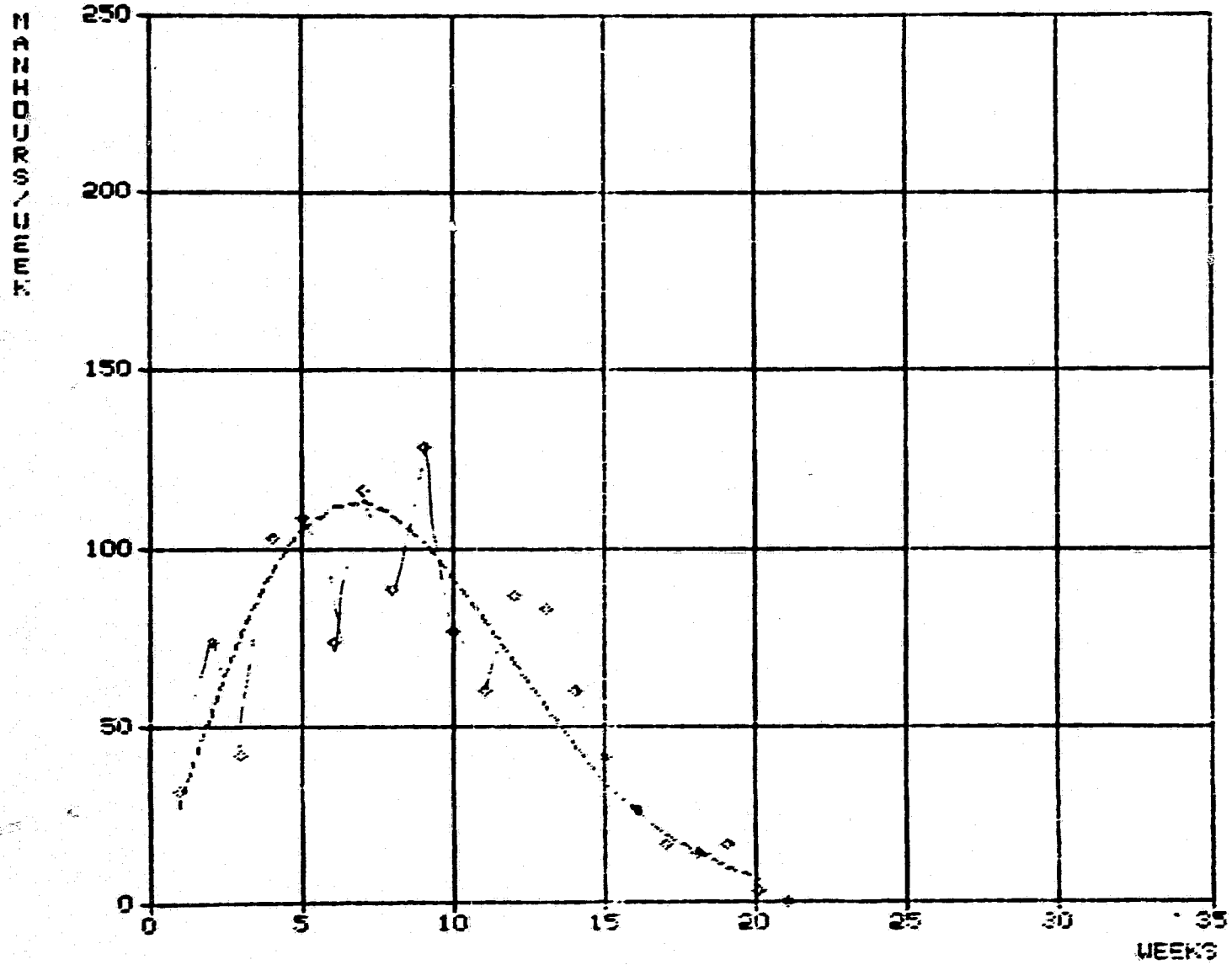
ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE 12. PROJECT 2 -- EFFORT VS. TIME
DEVELOP PHASE -- TRANSLATED AND SMOOTHED



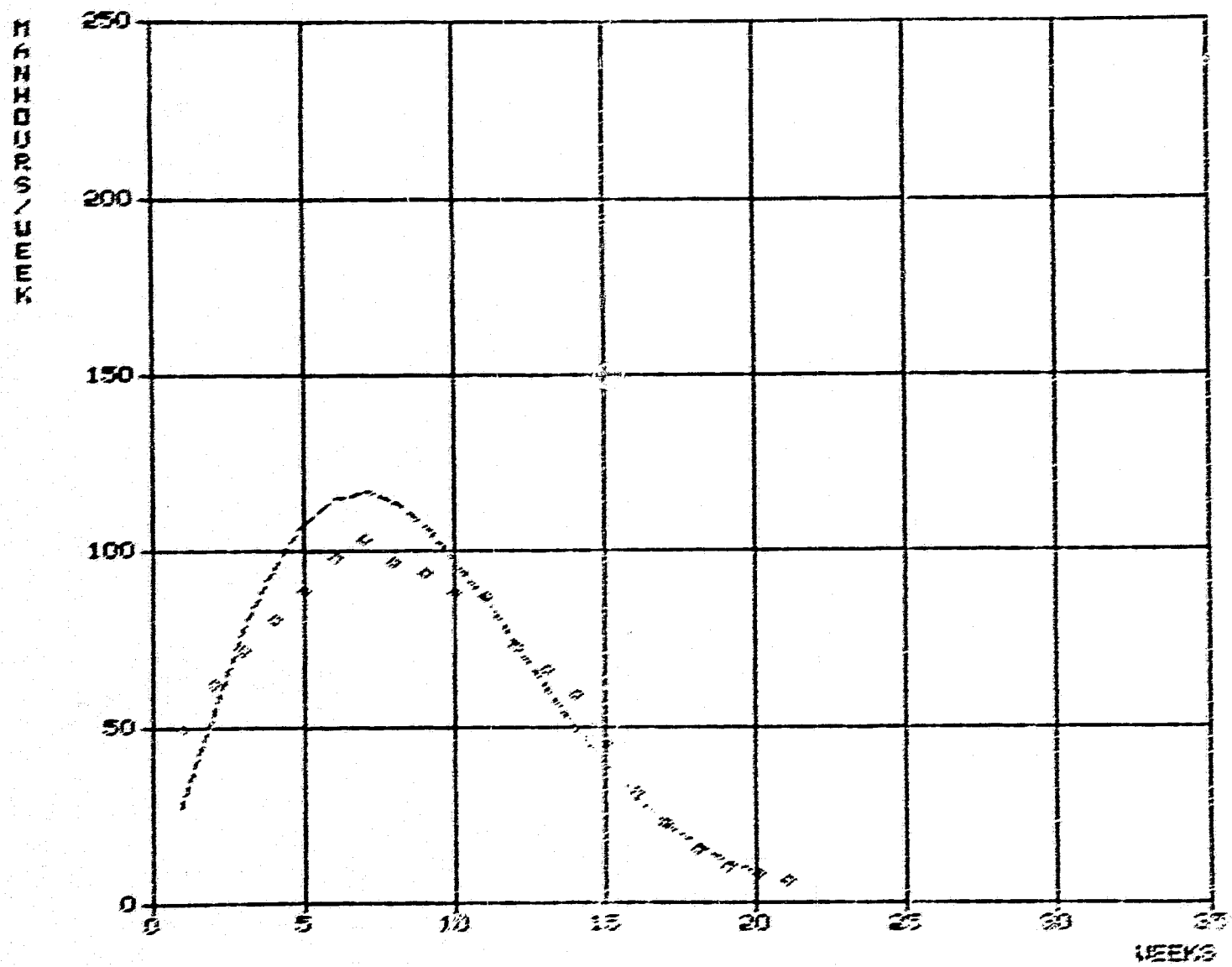
ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE 13. PROJECT 5 -- EFFORT VS. TIME
DEVELOP PHASE -- TRANSLATED



ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE 14. PROJECT 5 -- EFFORT VS. TIME
DEVELOP PHASE -- TRANSLATED AND SMOOTHED



ORIGINAL PAGE IS
OF POOR QUALITY

Except for project 1, the fits for the coding phase curve were improved by the coordinate translation. This is because the translation used for project 1 cut off the beginning portion of the man-power curve. Overall the curves fit the data considerably better after translation.

From this analysis it is seen that none of the subcycles can truly be said to follow a Rayleigh man-power distribution with the exception of the coding subcycle. The assumption that the design, code and test subcycles are Rayleigh in shape is not true. Therefore it is unlikely that the total weekly effort would assume a Rayleigh shape either. But, the fact that the code effort is approximated by the Rayleigh curve is significant and cannot be ignored.

5.2 Comparison of Putnam and SEL data

In this subsection, the difference between the data studied by Putnam and that used for this study are looked at more closely. The actual data used by Putnam was not available for the purposes of this study. However, this data was not needed to conduct this comparison. The purpose of this comparison is to determine what type of data was included in Putnam's study. This information can be gotten from the literature.

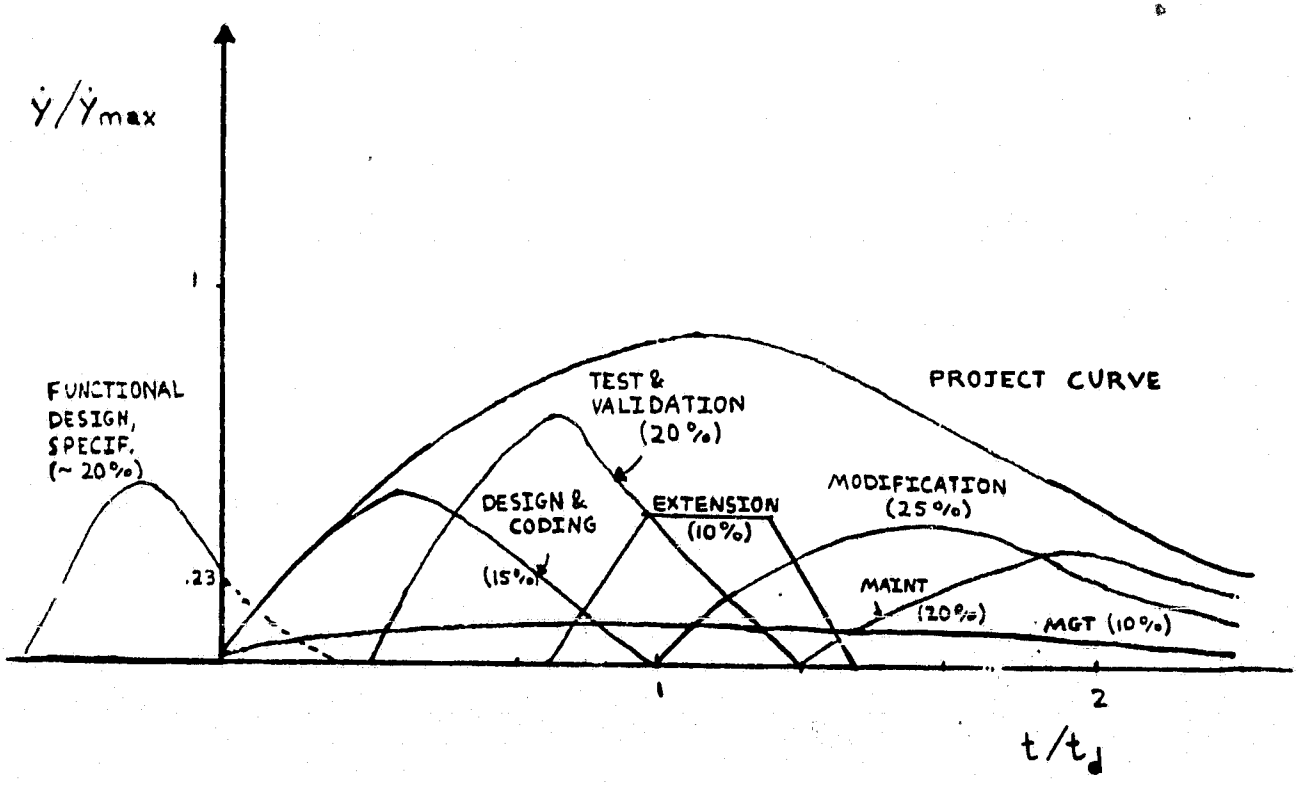
As has been mentioned earlier, there are two types of weekly effort which are reported in the SEL environment. What was studied in the previous section was called the total weekly effort and consisted of the total effort expended on the project by all

personnel. This data is gathered in the RS form. When the data from the CSR form is totaled by week what is obtained is the total effort expended in development activities. This is given the name total weekly development effort here.

In Putnam's life cycle diagrams (figure 15), management effort is given a separate curve. SEL's total weekly effort includes management effort and other overhead charges, whereas the total weekly development effort does not. Therefore the total weekly development effort data more closely matches the effort data regarded by Putnam to properly belong to the subcycles of design/code and testing. When the total weekly development effort data was smoothed and fitted, the Rayleigh curve fit the weekly development effort better than it did the total weekly effort. (Examples can be seen in Appendix A in graphs labeled WEEKLY CHARGES). This perhaps indicates that the Rayleigh curve should be regarded as a model for development effort and not as an estimate for the budget type data which makes up the total weekly effort. The fits obtained were still not very good however, and no definite conclusion can be drawn. Furthermore this observation is confined to the SEL environment.

Figure 15. Putnam's Life-Cycle Diagram

ORIGINAL PAGE IS
OF POOR QUALITY



5.3 General Trends in the Total Weekly and Development Distributions

In this subsection, the effect that holidays have on subcycle effort distribution and how these relate to the disturbances observed in the total weekly effort distribution are studied. Also, the effect of changing the relative start times of each phase of a project are studied in light of how shifting relative start times influences the shape of the overall profile curve. What is being sought is possible factors which may cause a project's effort distribution not to be Rayleigh.

It is assumed that the forces acting on the total weekly effort distribution are similar to the forces acting on the total weekly development effort. Therefore studying the subcycle data will serve to explain both observations made about the total weekly effort as well as the total development effort.

The subcycle data was first studied to see what kind of effect holidays had on effort expended. It was found that the occurrence of holidays could not be associated with any of the major effort slow downs observed in the individual subcycle data. When the development effort was inspected as a sum (that is, when these subcycles were summed together), the holidays could be seen to correspond to all major slow downs. In other words, the holidays did not cause any noticeable noise at the subcycle level, but the cumulative effect of adding the subcycles to obtain the total weekly development effort made them apparent. This

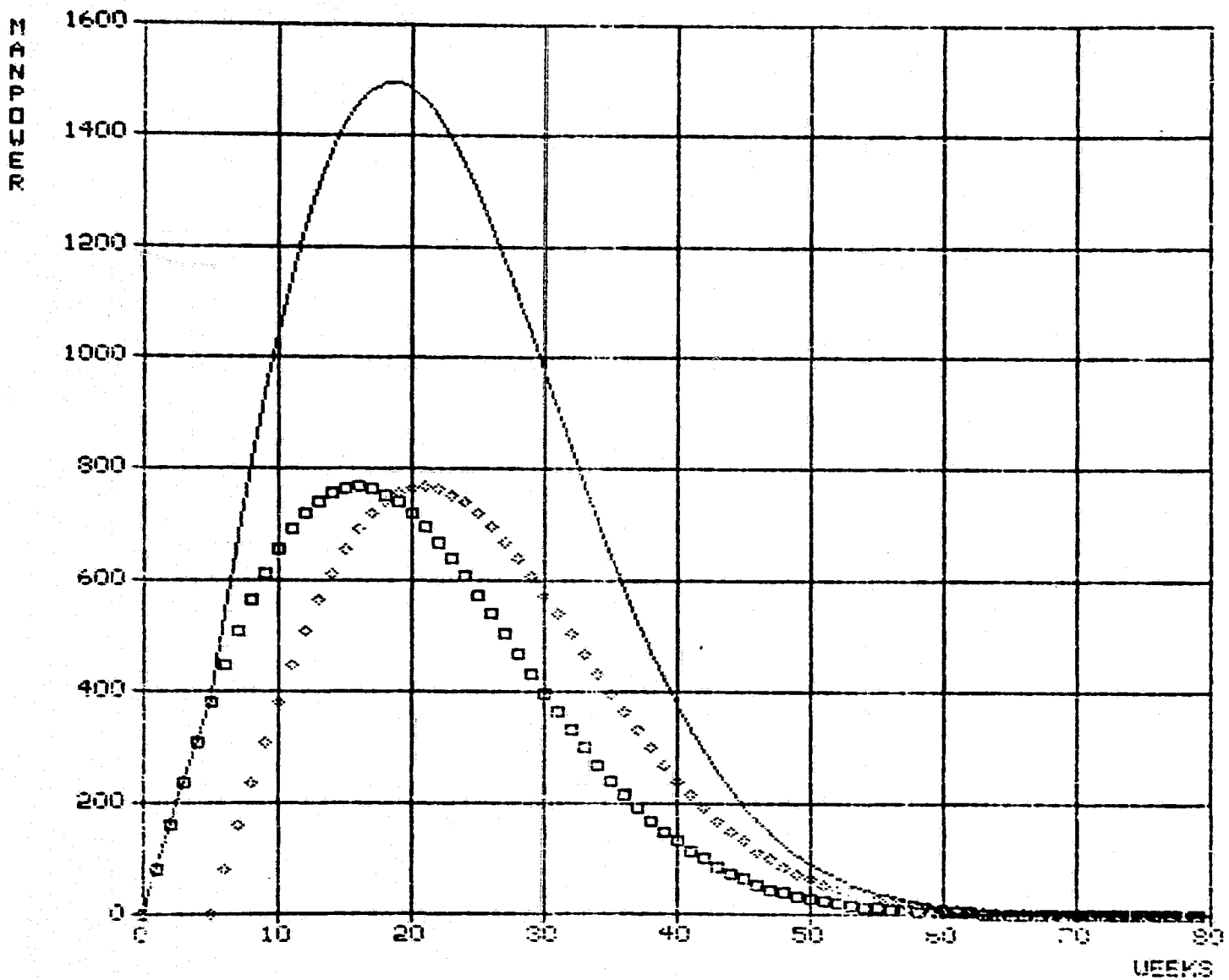
corresponds to the observation made earlier of the effects of holidays on total weekly effort. Other noise in the data seems to cancel out because it occurs randomly. For example, not everybody gets sick at the same time. The noise due to holidays is not random - most people like to take vacations around holidays. Therefore, the effects do not cancel but are reinforced when the subcycles are added together.

It can be said that the reason holidays have such an impact on the total weekly effort is because of the cumulative effects introduced by these non-random disturbances. For large scale systems these disturbances are insignificant and therefore are not noticeable. This gives one key as to why the total weekly effort for medium scale systems may not be Rayleigh in shape.

Another reason can be given by making some observations about the relative start dates of each phase of a project. Putnam observed that for large projects the relative dates for milestones (the start dates for different phases of a project) were similar. What would be the effect of changing the project milestones in the overall curve?

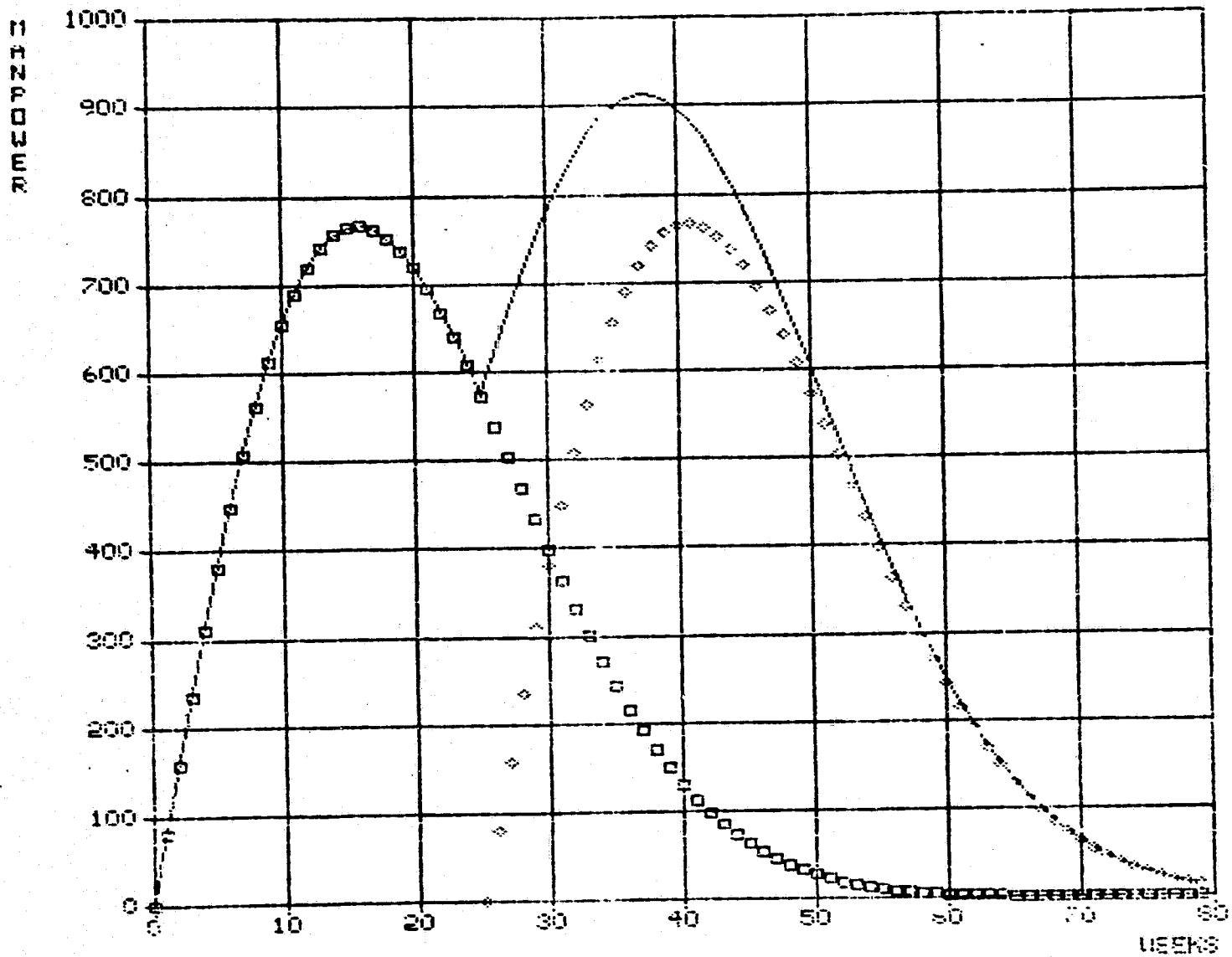
To answer this question the sum of pairs of Rayleigh curves were considered. They are illustrated in figures 16 through 18.

FIGURE 16. SUM OF RAYLEIGH CURVES
SUM IS RAYLEIGH



ORIGINAL PAGE IS
OF POOR QUALITY

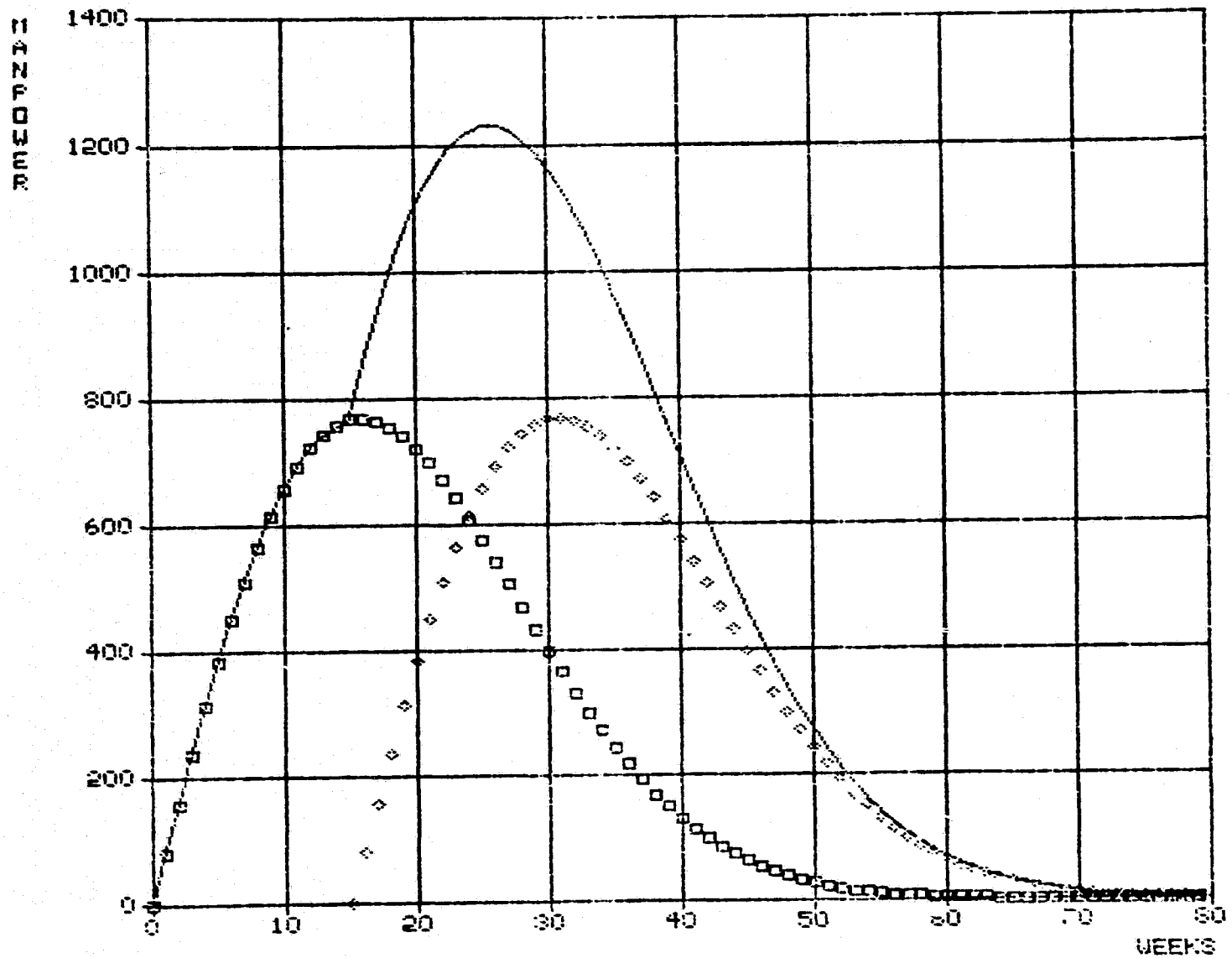
FIGURE 17. SUMOF RAYLEIGH CURVES
SUM WITH TWO PEAKS



ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE 18. SUM OF RAYLEIGH CURVES
SUM IS PARABOLIC



ORIGINAL PAGE IS
OF POOR QUALITY

Depending on how these curves were shifted, illustrating the changing of milestone dates, the sum of these curves varied. The resulting sums were a curve which looked like a Rayleigh curve with some "noise" (figure 16), a curve which has two distinct peaks (figure 17), and a curve which could be modeled by a parabola if its tail end were eliminated (figure 18).

It follows therefore that one of the reasons that large scale projects have Rayleigh shaped effort distributions is because of the particular arrangement of milestones among this size system. One of the reasons that the smaller projects studied here do not exhibit a Rayleigh shape is because of the differences in relative milestone dates. Further evidence for this difference lies in Putnam's observation about the time to reach peak effort. Putnam has indicated(*) that for smaller projects the time to peak effort is half the time it takes to complete development. This is unlike large projects where peak effort occurs at the end of development. This suggests that the relative start dates for each phase in a small project are different than those for a large project.

These two factors-- holidays and shifting milestones, affect the shape of the overall effort distribution because of their cumulative effects on the project profile curve. These observations help give some explanation about why the Rayleigh curve may not model the total weekly effort distribution for projects with

* This information was given to the SEL in a private communication.

Rayleigh shaped subcycle effort distributions. However, the projects studied here do not have Rayleigh shaped subcycle curves. Explanations for this latter phenomena are sought in the next subsection.

5.4 General Trends in the Subcycle Effort Distribution

Thus far it has been seen that the subcycle man-power for design and testing were not Rayleigh in shape. Only the coding man-power curve seemed to be modeled by the Rayleigh curve. This has helped explain why the overall project curves were not Rayleigh but it still leaves us with the questions why are not all three of these subcycle curves Rayleigh in shape as Putnam proposed and why is it that the coding man-power curve seems to be Rayleigh?

We can explain these phenomena by taking a closer look at the SEL environment itself. Basili and Beane pointed out that the SEL environment was not typical because of the contractor's intimate familiarity with the problem area and because of the similarity of the programs. This has a great deal of bearing on the shape of the man-power curves. First we must point out what is so unique about this environment however.

At the SEL, managers use a heuristic algorithm (this algorithm was given earlier). When Basili and Beane examined this algorithm they found that managers were indeed making use of it. What is unusual is that managers could seemingly apply personnel at any point of the project without having any major adverse effects on the development process. This is because the projects

ORIGINAL PAGE IS
OF POOR QUALITY

were so similar in nature and the personnel was so familiar with the application area that it did not take them very long to come up to speed on a new project. Their learning curve was significantly reduced. This is not a typical situation.

The significance of this is the effect this has on project development and as a consequence the effect it has on man-power curves. Seeing that the SEL environment is very different than those studied by Putnam, it does not come as a surprise that the effort distribution curves do not seem to match. We could leave our explanation at that were it not for the man-power curve for the coding subcycle. The coding curve seems to be modeled well by the Rayleigh curve. This needs to be examined further.

Because of the contractor's familiarity with the application area we would expect that the effort expended on new projects would be considerably less than if the problem area were unfamiliar to the people working on the project. Furthermore, we would also expect that the effort expended would be applied optimally or nearly so since they had done this sort of thing before.

The fact that the problem space is reduced significantly impacts the design effort since it is in the design phase that many of the problems need to be solved. In the SEL environment many of the problems are solved even before the project begins. This is what allows management to allocate as much as 1/2 to 3/4 "full staffing" at the very beginning of the project. This is considerably different from what is suggested by the Rayleigh

ORIGINAL PAGE IS
OF POOR QUALITY

curve. The Rayleigh curve suggests that the design curve should go through the origin. Not only is there a shifting of the man-power curve but also the very shape of the man-power curve is affected. Problems will be handled in a very different manner. More problems will be done in parallel. Also the learning curve which is what determines the man-power curve is almost non-existent. These factors cause the man-power curve to exhibit a different shape. The design curve in the SEL environment is not Rayleigh because the environment is significantly different.

Unlike the design man-power curve, the man-power curves for coding and testing should not be as affected by these particular differences in environment. This is because the coding and testing phases are not as greatly affected by the personnel's familiarity with the problem area except perhaps in doing things more efficiently. The basic problem of coding from a design specification remains unchanged. Except for a possible reduction of some types of coding errors and lifting some code directly from a previous project the problems encountered will be the same and there will be just as many of them. Modifying code, by the contractor's own account, is difficult and brings its own set of problems so no great advantage is gained by this. Because the problem set is not significantly changed, the problem of coding - translating a design specification into a programming language, remains unchanged. This of course means that there will be little if no impact on the overall shape of the man-power curve, except it might reach peak effort at an earlier date. Furthermore since the

contractor has a lot of experience with the application his effort will be expended in almost optimal manner. This means that the man-power curve for the coding subcycle will be very close to the optimal man-power curve for coding for medium scale systems. Since this curve seems to be Rayleigh in shape, this suggests that the Rayleigh curve might be the optimal manloading curve for coding in a typical environment. This agrees with Putnam's observations.

We are still left with one problem. The testing man-power curve is not Rayleigh in shape. Also the factors causing the design curve to be different do not have a significant impact on the testing curve for the same reasons that they did not influence the coding curve. But, it was noted previously that the testing subcycle was most likely made up of two phases: module testing and integration testing. The man-power curves for these two phases taken separately may very well be Rayleigh in shape. There is no way of telling whether this is true or not from the SEL data however.

The implications of these results are important even though they may not be conclusive. The Rayleigh model continues to be a good candidate man-power curve for medium scale environments. It must be noted that Parr has offered a different explanation for why the design curve does not go through the origin. The Parr model therefore cannot be discarded as a possible candidate either. Further work is definitely warranted.

ORIGINAL PAGE IS
OF POOR QUALITY

6. USE OF SOFTWARE PARAMETERS TO CLASSIFY SYSTEMS

Putnam has derived relations for difficulty, system size and a measure of the state of technology. In this section, the possibility of using these relations to classify medium scale projects are studied.

Putnam observed that for large projects the relation D=K/td**2, acted as a measure of difficulty in terms of the programming effort and the time to produce the system. He also derived a relation between the number of source lines of code, the effort and the time to produce it. This is given by the equation: Ss=Ck*(K**1/3)*(td**4/3), where Ss is the number of source lines, K is the total effort, td is the time to reach peak effort and Ck is the state of technology. Putnam observes that Ck "seems to relate to machine throughput (or programmer turnaround, available test time, etc.) and other technological improvements like Chief Programmer Team, Top Down Structured Programming, on-line interactive job submission, etc." Since the data studied by Putnam differs from the data studied here, these relations cannot be applied directly. A new set of equations is derived using Putnam's techniques.

The fundamental difference that must be considered is that SEL data includes effort expended through acceptance testing only, while the data studied by Putnam includes the entire lifecycle through maintenance. Because of this the time to reach peak effort reported in the SEL data does not correspond to td, the

time to development, nor does the total effort reported in the SEL data correspond to K, the total effort through maintenance. The time to reach peak effort reported in SEL is called tm. The total effort reported in the SEL data through acceptance testing, is called Ka. tm and Ka are substituted for td and K

The resulting equations are identical in form to those given by Putnam. The shaping parameter is given by 1/(2*tm**2). Putnam's difficulty parameter D, given by 2*K*a, is replaced by the expression Dl=2*Ka*al, where al=1/(2*tm**2). The equation for number of source lines is given by, Ssel=Csel*Dl**2/3*Ka where Ssel and Csel replace Ss and Cn respectively(*). Note that although these equations are of the same form as Putnam's equations they have significantly different meaning.

These equations were applied to the SEL data. Dl was calculated using the parameters of the curves fitted to total weekly effort, total weekly development effort and total weekly effort spent on coding. The values obtained for Dl are compared for consistency with a subjective measure of difficulty. This measure is given by management. Projects are ranked according to these measures and the two rankings are then compared. Table 5 summarizes the values of Dl and Table 6 shows a rank ordering of projects according to Dl. The values of Dl were obtained by using the values for the total effort and the values of the shaping parameters were obtained from the least squares fit of the effort data.

* Ss = Cn*(D**2/3)*K

Ss = Ck*(K**1/3)*(td**4/3)

A rank of one (1) is given to the project with lowest value of D1, and a value of five (5) to the highest value.

ORIGINAL PAGE IS
OF POOR QUALITY

	D1 FOR TOTAL EFFORT (CALCULATED)	D1 FOR TOTAL EFFORT (SEARCHED)	D1 FOR DEVELOPMENT EFFORT	D1 FOR CODING EFFORT
PROJ 1	27.7	19.4	17.6	17.6
PROJ 2	18.2	19.6	14.0	14.0
PROJ 3	15.9	14.9	6.73	6.73
PROJ 4	27.6	19.4	12.1	12.1
PROJ 5	33.2		24.2	24.2

Table 5. Results of Calculations of the Difficulty Parameter

ORIGINAL PAGE IS
OF POOR QUALITY

	PROJ 1	PROJ 2	PROJ 3	PROJ 4	PROJ 5
Total Weekly Effort Calc	4	2	1	3	5
Total Weekly Effort Searched	2	4	1	2	
Total Weekly Develop Effort	4	3	1	2	5
Total Weekly Coding Effort	3	4	1	2	5
Average Value	3.25	3.25	1	2.25	5

Table 6. Ranking According to D1 Parameter

ORIGINAL PAGE IS
OF POOR QUALITY

The projects were evaluated by management using 42 categories of difficulty which were divided into three groupings: complexity, internal and external influences. These all give some indication of how difficult it was to develop a project. For each project a value from zero (lowest difficulty) to fifty (highest difficulty) was assigned to each category. The values for the categories under each grouping were added and totals for each project were obtained. The projects were then ranked in the same way as with D1. The rankings are compared in Table 7.

	D1	Difficulty
least	PROJ 3	PROJ 5
	4	2
	2	3
	1	1
greatest	5	4

Table 7. Comparison of Management's Ranking of Projects According to Difficulty and Ranking Obtained from D1

As can be seen there is no correspondence between management's perception of difficulty and D1. Comparing each one of the groupings of difficulty factors: complexity, external and internal influences, separately does not improve the results.

ORIGINAL PAGE IS
OF POOR QUALITY

The next effort was to determine the value of the constant Csel for each project to see whether projects could somehow be ordered according to technology or methodology. Management appraisals of the methodology were used as a basis of comparison.

The equation for Ssel is used. The number of source lines was defined as the total number of source lines including comments. The values of Dl were taken from Table 5. [BB2] Table 8 summarizes the results. The values used for Ka and Dl are obtained from Table 5.

The systems were ranked from lowest to highest value of Csel. Management's evaluation of the methodology employed on these projects was also used to rank the projects. All these values are summarized in Table 9.

ORIGINAL PAGE IS
OF POOR QUALITY

	CSEL FOR TOTAL EFFORT (CALCULATED)	CSEL FOR TOTAL EFFORT (SEARCHED)	CSEL FOR DEVELOPMENT EFFORT	CSEL FOR CODING EFFORT
PROJ 1	4.62	3.84	11.9	38.4
PROJ 2	2.06	1.89	6.23	18.0
PROJ 3	1.99	1.91	1.44	6.76
PROJ 4	4.93	3.14	7.56	11.4
PROJ 5	10.2		11.0	54.1

Table 8. Calculations of the Methodology Parameter Csel

ORIGINAL PAGE IS
OF POOR QUALITY

	PROJ 1	PROJ 2	PROJ 3	PROJ 4	PROJ 5
Total Weekly Effort Calc	3	2	1	4	5
Total Weekly Effort Searched	4	1	2	3	
Total Weekly Develop Effort	4	2	1	3	5
Total Weekly Coding Effort	4	3	1	2	5
Average Value	3.75	2	1.25	3	5

Table 9. Ranking Projects According to Methodology Using Csel ans Management Estimates

ORIGINAL PAGE IS
OF POOR QUALITY

The projects were ranked using the same ranking scheme as above. The rankings are compared in Table 10.

	Csel	Methodology
least	PROJ 3	PROJ 3
	2	2
	4	4
	1	1
greatest	5	5

Table 10. Comparison of Management's Ranking of Projects According to Methodology and Ranking Obtained from Csel.

It is seen that the ranking obtained using Csel seems to match the ranking given by management. It is not clear whether or not Csel is a product of the Rayleigh model however. Like the estimates for ta, yd, and Ka, Csel is somewhat suspect. The link to the Rayleigh model is made through the difficulty measure Dl. Csel relates to this measure and to total effort. Dl is not a very good measure for difficulty at SEL and therefore provides a weak link between the number of source lines and technology. The constant Csel used by Putnam arose empirically when productivity was plotted against difficulty. This seems like a rather loose connection with the Rayleigh curve. Therefore it is not clear whether or not the ranking can be credited to the use of the Rayleigh model. However, even though there does not seem to be any

ORIGINAL PAGE IS
OF POOR QUALITY

theoretical support for these results at present, it does not reduce their significance. These equations can still be used to give management a quick estimate of how well things were done on a project in rankings which correspond to rankings he would have given.

From these results it cannot be decided whether or not the Rayleigh model itself could be used to classify medium scale projects according to difficulty or methodology. If there were a way in which to estimate the constant Csel a priori, either by use of historical data or some evaluation method, it may be possible to estimate the number of source lines by using the equation for Ssel. This was not attempted because there were not a sufficient number of projects to estimate a value for Csel. How the Rayleigh model can be used to classify or size medium scale systems is uncertain, at least for the SEL environment.

7. ANALYSIS OF COMPONENT DATA

In this last section, attention is turned away from man-power distribution to other relations in the data. The object is two-fold. To try to find invariants which may allow the data to be smoothed and analyzed further, and, to find relations which will facilitate the understanding of effort distributions.

Attention is centered around the relations between components and effort. It was reasoned that once the requirements had been defined, the problem of determining the number of components would be much more tractable than estimating the number of source lines of code. If a relation between the number of components and effort could be established, it would make the problem of estimation much simpler than the traditional approach of estimating lines of code.

The relation between the number of components and effort was studied to determine whether or not the effort distribution could be obtained by determining the number of components worked on. To do this the total number of components in existence in the system in any given week was gotten from the CSR form and plotted. Components are defined as any named portion of the system. The weekly ratio of components worked on in a given week to the total number of components in existence that week was also plotted along with the ratio between the number of hours worked in a given week to the number of components worked on that week. The ratios were computed for each week. Multiplying each of the

ratios gives the weekly total effort.

effort/week = (existing components/week)
X (components worked on/existing components)/week
X (effort/components worked on)/week

These ratios were fitted with six different two parameter curves. Table 11 gives the parameters for the best fitting curves for existing components/week and the corresponding correlation for four of the projects. (The fits could not be performed on the other three projects.) Table 12 and 13 give the same information for the other two ratios.

ORIGINAL PAGE IS
OF POOR QUALITY

		PROJ 1	PROJ 2	PROJ 3	PROJ 5
y = a + b*x	a	3.58	12.5	-33.9	2.41
	b	2.51	0.66	0.46	0.652
	r	6.52e-2	0.91	0.94	0.96
y=a*exp(b*x)	a	3.40	9.73e-3	9.38	9.78
	b	1.66	15.8	7.23e-3	1.53e-2
	r	.03	.70	.94	.78
y=a*(x**b)	a	5.41	1.89	4.52	2.75
	b	-3.0e-2	.799	.483	.682
	r	1.61e-3	.81	.40	.82
y=a+(b/x)	a	8.31	128	82.7	73.3
	b	.339	-13.1	-7.05	-7.35
	r	.001	.068	.028	.097
y=1/(a+b*x)	a	.292	.104	.066	.137
	b	1.34e-4	-4.1e-4	-1.6e-4	-8.6e-4
	r	.003	.29	.88	.34
y=x/(a+b*x)	a	-2.4e-2	4.8e-2	5.3e-3	4.7e-2
	b	.323	.024	.024	.031
	r	.011	.75	.11	.81

Table 11. Total Number Existing Components in the System

ORIGINAL PAGE IS
OF POOR QUALITY

		PROJ 1	PROJ 2	PROJ 3	PROJ 5
y = a + b*x	a	45.3	59.3	31.3	68.9
	b	-.145	-.200	-.036	-.340
	r	.57	.70	.081	.65
y=a*exp(b*x)	a	78.9	103	24.1	89.2
	b	-1.28	-.012	-.001	-.013
	r	.82	.80	.044	.65
y=a*(x**b)	a	423	286	51.2	139
	b	-.085	-.064	-.193	-.431
	r	.47	.40	.14	.38
y=a+(b/x)	a	15.9	23.9	20.8	30.4
	b	8.57	7.75	7.92	7.11
	r	.27	.21	.49	.22
y=1/(a+b*x)	a	-.502	-.220	-.054	-7.58
	b	.005	.003	4.22e-5	.006
	r	.41	.37	.015	.22
y=x/(a+b*x)	a	-.058	-2.33	-5.5e-3	-9.32
	b	.57	.233	.065	.099
	r	.006	.006	.003	.006

Table 12. Components Worked On/ Existing Components

ORIGINAL PAGE IS
OF POOR QUALITY

		PROJ 1	PROJ 2	PROJ 3	PROJ 5
y = a + b*x	a	3.58	6.96	25.0	12.4
	b	.025	.006	-4.75	4.33
	r	.006	.002	.31	.002
y=a*exp(b*x)	a	3.40	6.15	24.3	13.0
	b	.002	5.29	-.0036	-.0003
	r	.004	.007	.30	.001
y=a*(x**b)	a	5.41	3.17	17.4	21.5
	b	-.030	.159	-.11	-.030
	r	.002	.11	.026	.070
y=a+(b/x)	a	8.31	8.04	13.2	16.6
	b	.339	-.567	-.88	1.06
	r	.002	.045	.013	.011
y=1/(a+b*x)	a	.292	.203	.033	.086
	b	.0013	-5.62	.0004	.0001
	r	.002	.0008	.13	.014
y=x/(a+b*x)	a	-.024	2.22	.008	-6.34
	b	.323	.189	.141	9.92
	r	.011	.026	.065	.026

Table 13. Effort/ Components Worked On

The best fit was given by the straight line to existing components. Existing components was also fit well by the exponential curve. The ratio of components worked on to existing components was best fit by the exponential and secondly by the line. The ratio between effort and the number of components was not fit well by any of the curve types. Visual inspection of the plots of this ratio for each of the subcycles suggested the possibility that the ratio was constant during the coding phase. This observation was not substantiated by curve fitting however.

If what seems to be the best curve types are multiplied as was illustrated above, the following is obtained.

effort/week = (a1+b1+t)*(a2*exp(b2*t))*a3

a1, a2 and a3 are constants resulting from the fits. This equation can be rewritten in the form,

effort/week = (C1+C2*t)*(exp(C3*t))

As can be seen this equation differs from the Rayleigh equation. The expression in the exponential is a function of time whereas in the Rayleigh equation it is a function of time squared. If the resulting equation had had the same form as the Rayleigh equation, it would have lent some support to the model. However, nothing can be said about the rayleigh model from these results.

The fact that the plot of total number of existing components, was best fit by the straight line suggests that for this

environment, the system being developed grows by a constant number of components. The relation that is observed can be summarized as follows: the limit of the difference between the total number of components in week i and the total number of components in week i-1 as i approaches td, the time to the end of development, is constant.

Other relations studied were total cumulative effort, the number of components worked on for a given week and the ratio of effort to the number of components in existence in the system. None of these relations proved very useful. The distribution of the number of components worked on in a given week does suggest a Rayleigh shape but there is too much noise in this data to be certain even after smoothing. The plots for all the ratios for one of the projects studied are given in Appendix B.

There do not seem to be any relations in the data which can be used in smoothing. Other than the total number of components in existence none of the other relations could be fit very well by any of the curve types. As far as gaining any further insight into the behavior of man-power not much can be said. There does not seem to be any obvious relation between the number of components worked on and the effort expended.

8. CONCLUSION

It is clear that the Rayleigh curve is not the best fitting curve for the effort data in the SEL environment. Because of the contractor's familiarity with the problem area, a unique development environment exists which varies significantly from the environments studied by Putnam. Therefore it is natural that the Rayleigh model would not be adequate for this environment. The SEL environment differs principally in the design and testing phases. It is clear that much of the effort which would normally be required during the design phase is eliminated because of the contractor's knowledge of the problem area. The testing effort curve is different only in how testing effort is accounted for and how the time schedule for testing differs. Testing and acceptance testing are done as two distinct phases. If the effort data was collected as two different phases, it is very possible that each phase would exhibit a Rayleigh man-power distribution. The addition of the two curves would not necessarily result in a Rayleigh curve as was illustrated in figure 17.

The coding phase for the SEL environment seems to follow the Rayleigh curve closely. It may be that the coding phase is less affected by the contractor's familiarity with the problem area. Added experience may aid programmers in finding more efficient ways of implementing a particular design and in reducing the total amount of time spent on developing the code, but still not change the basic shape of the curve because the problem of coding is not really changed. Even with the added knowledge there is an

upper limit to how fast code can be produced. In many cases the programmer may depend on another module to be completely coded. Other problems which are unique from project to project may still need to be solved. Whereas many design problems were eliminated because the design already existed, coding specific solutions may be different enough from previous solutions that the code has to be redone or significantly modified.

Since the Rayleigh curve fits this data well and since it is possible that the environmental differences did not cause significant deviations from a typical man-power distribution, this makes the Rayleigh model at least a possible candidate model for the man-power distribution of medium scale projects. However there are at least four factors which have been studied that affect the overall weekly effort distribution for medium scale projects which must be taken into account.

First, the underlying subcycles of design, code and test may not all be Rayleigh in shape due to differences in the development environment. This was definitely the case for the SEL environment. Differences in management strategies can also cause significant deviations. This was observed in the testing phase and it also has been pointed out by Putnam. Second, the effort data gathered may not be that which the Rayleigh model is intended to model. The data should match as closely as possible the type of data used to formulate the model. Before attempting to apply the model one must carefully consider what the model is intended to model. Thirdly, the effect of holidays on medium

scale system's total effort distribution is much more pronounced than on large scale systems. This may be as a result of the differences in granularity of the data collected for small to medium scale systems versus large scale systems (weekly vs. monthly or yearly). These predictable disturbances must be taken into account in small to medium scale projects while they might be ignored for large projects. Finally, the difference in the relative dates for the start of various phases between projects may vary significantly. The ideal project phasing has not been thoroughly worked out. One solution for all projects most probably does not exist. Nevertheless, the timing of these milestones is under the control of management.

All these factors must be taken into account in any future studies of this model's applicability to medium scale projects. The environment studied is rather unusual and somewhat unrepresentative of a large section of the industry. The data used contains a lot of noise. Also, the smoothing techniques used may have caused some facts to be overlooked.

The optimum man-power distribution curve for a typical development situation is not necessarily the optimal man-power solution for all situations. This is strongly suggested by the environment studied in the SEL. Furthermore, there are not really that many "typical" environments. The notion of what is "typical" is very hard to define. Putnam has tried to define the average behavior of man-power curves. Individual deviations will always exist. However, it is felt that the model does show promise not

only for large scale projects but also for the smaller sized projects studied here.

9. BIBLIOGRAPHY

[Putnam1] Lawrence H. Putnam, "A General Empirical Solution to the Macro Software Sizing and Estimating Problem", IEEE Transactions on Software Engineering, Vol. se-4,No. 4, July 1978, p.345-361.

[Putnam2] Lawrence H. Putnam,"Lecture 1. The Software Life: Evidence and Foundation, Quantitative Management, Software Cost Estimating", COMPSAC 77, Nov. 1977, p. 1-37.

[Putnam3] Lawrence H. Putnam, "A Macro Estimating Methodology for Software Development", IEEE Computer Society COMPCON, Wash., D.C., Sept. 1976, p.138-143.

[Putnam4] Lawrence H. Putnam, "The Influence of the Time-Difficulty Factor in Large Scale Software Development", Software Phenomenology, Working Papers of the Software Life Cycle Management Workshop, August 1977, p.307-312.

[B-Z] Victor R. Basili and Marvin V. Zelkowitz, "Analyzing Medium Scale Development", Proceedings of the 3rd International Conference on Software Engineering, May 1978, p.116-123.

[M] Thomas E. Mapp, "Applicability of the Rayleigh Curve to the SEL Environment", University of Maryland Computer Science Technical Report.

[Basili,et. al.] Victor R. Basili, et. al., "The Software Engineering Laboratory", University of Maryland Computer Science Technical Report.

[P] F. N. Parr, "An Alternative to the Rayleigh Curve Model for Software Development Effort", IEEE Transactions on Software Engineering, May 1980.

[BB1] Victor R. Basili and John Beane, "Can the Parr Curve Help With the Manpower Distribution and Resource Estimation Problem?", Journal of Systems and Software, Vol 2, No. 1.

[BB2] John W. Bailey and Victor R. Basili, "A Meta Model for Software Development for Resource Expenditures", University of Maryland Computer Science Technical Report 935, August 1980.

CONTENTS

1. INTRODUCTION 1

2. DESCRIPTION OF THE DATA EMPLOYED 4

3. EARLY WORK 7

4. ANALYSIS OF TOTAL WEEKLY EFFORT 13

4.1 Predictions of ta, yd, and K 13

4.2 Curve Fitting and Smoothing of Total Weekly Effort 16

4.3 General Trends in the Total Weekly Effort Distribution 30

5. SUBCYCLE DATA ANALYSIS 39

5.1 Curve Fitting and Smoothing of Subcycle Effort 40

5.2 Comparison of Putnam and SEL data 49

5.3 General Trends in the Total Weekly and Development Distributions 52

5.4 General Trends in the Subcycle Effort Distribution 58

6. USE OF SOFTWARE PARAMETERS TO CLASSIFY SYSTEMS 62

7. ANALYSIS OF COMPONENT DATA 73

8. CONCLUSION 80

9. BIBLIOGRAPHY 84

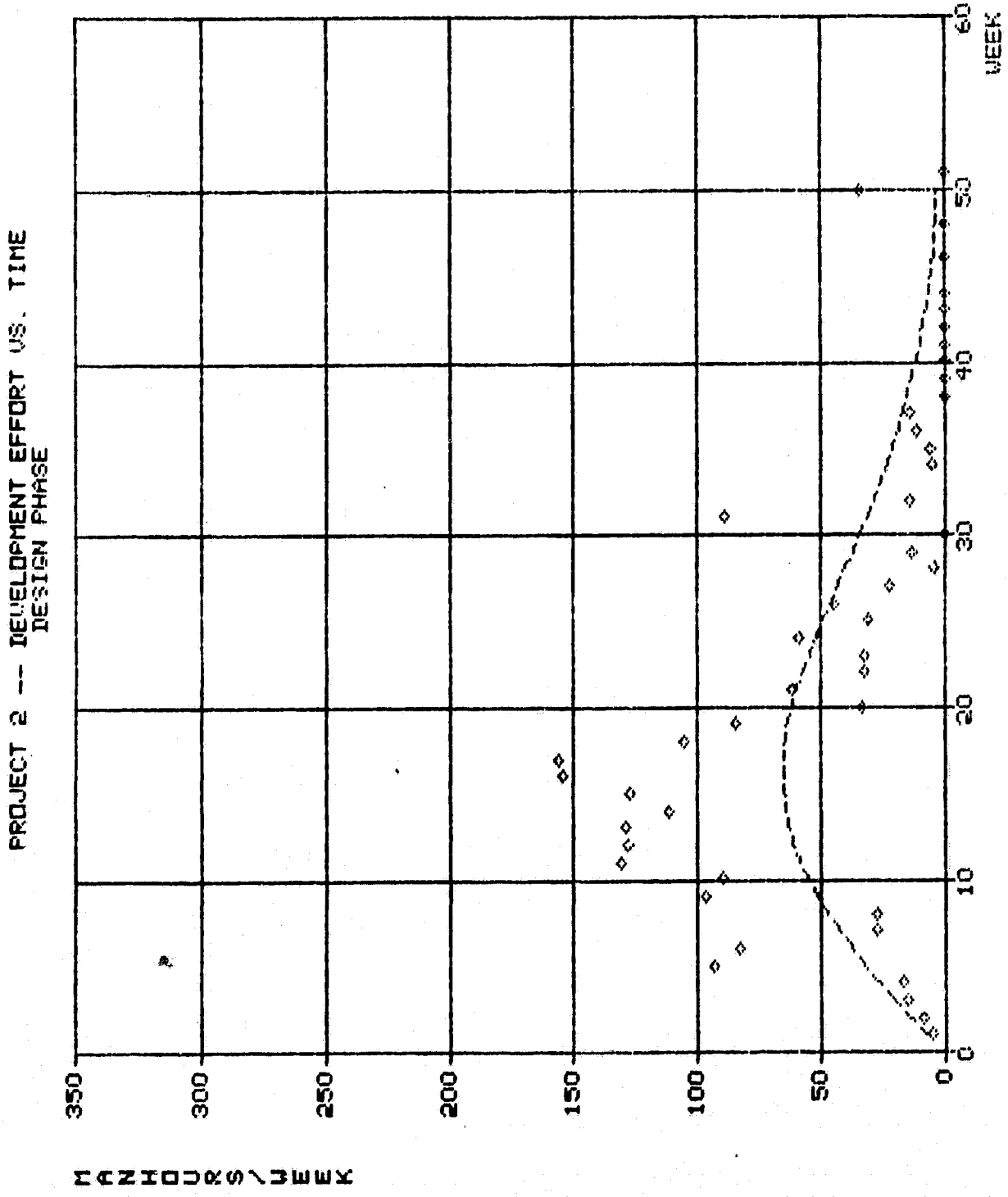
APPENDIX A 85

APPENDIX B 106

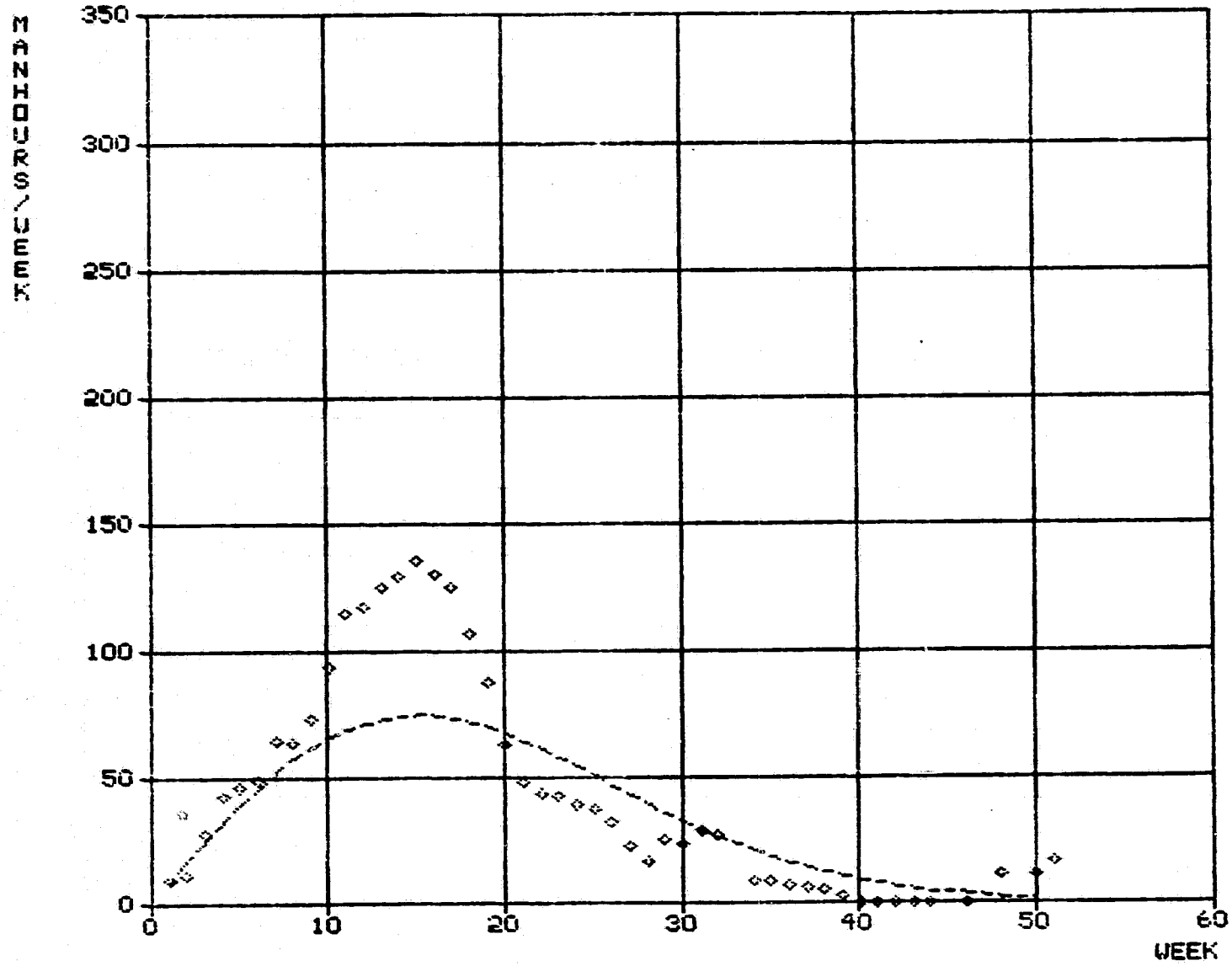
84A

APPENDIX A

ORIGINAL PAGE IS
OF POOR QUALITY

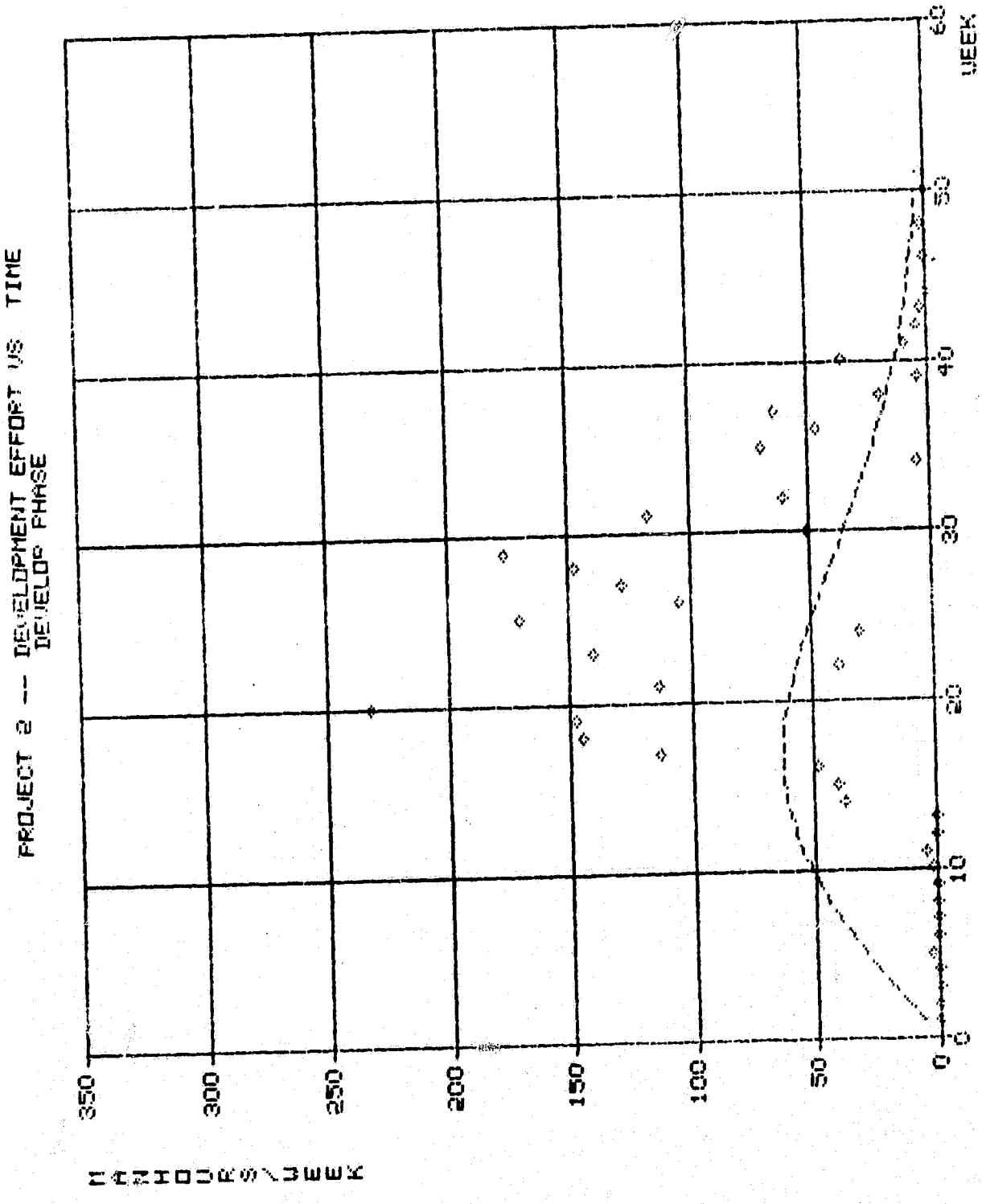


PROJECT 2 -- DEVELOPMENT EFFORT VS. TIME
DESIGN PHASE



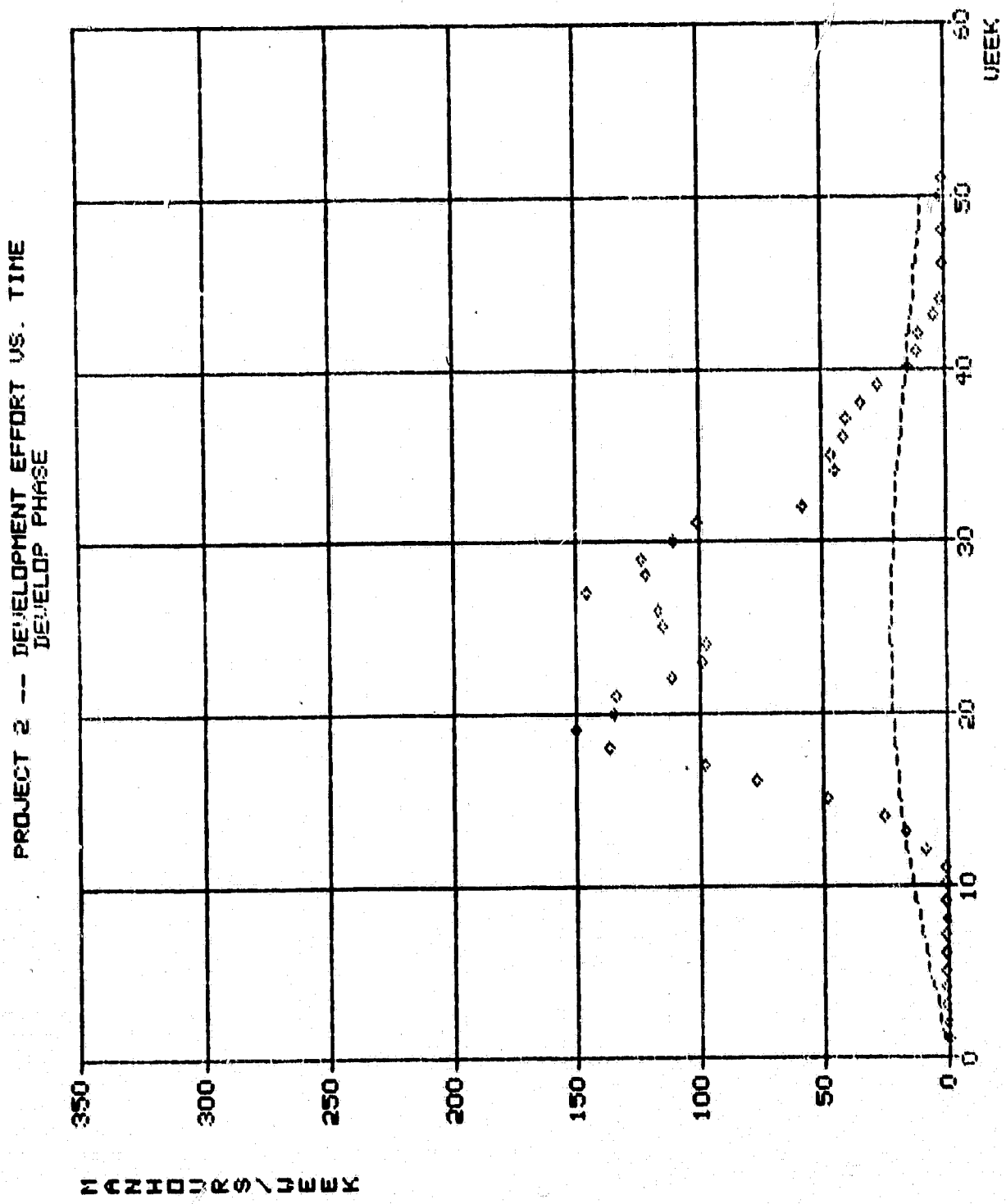
ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

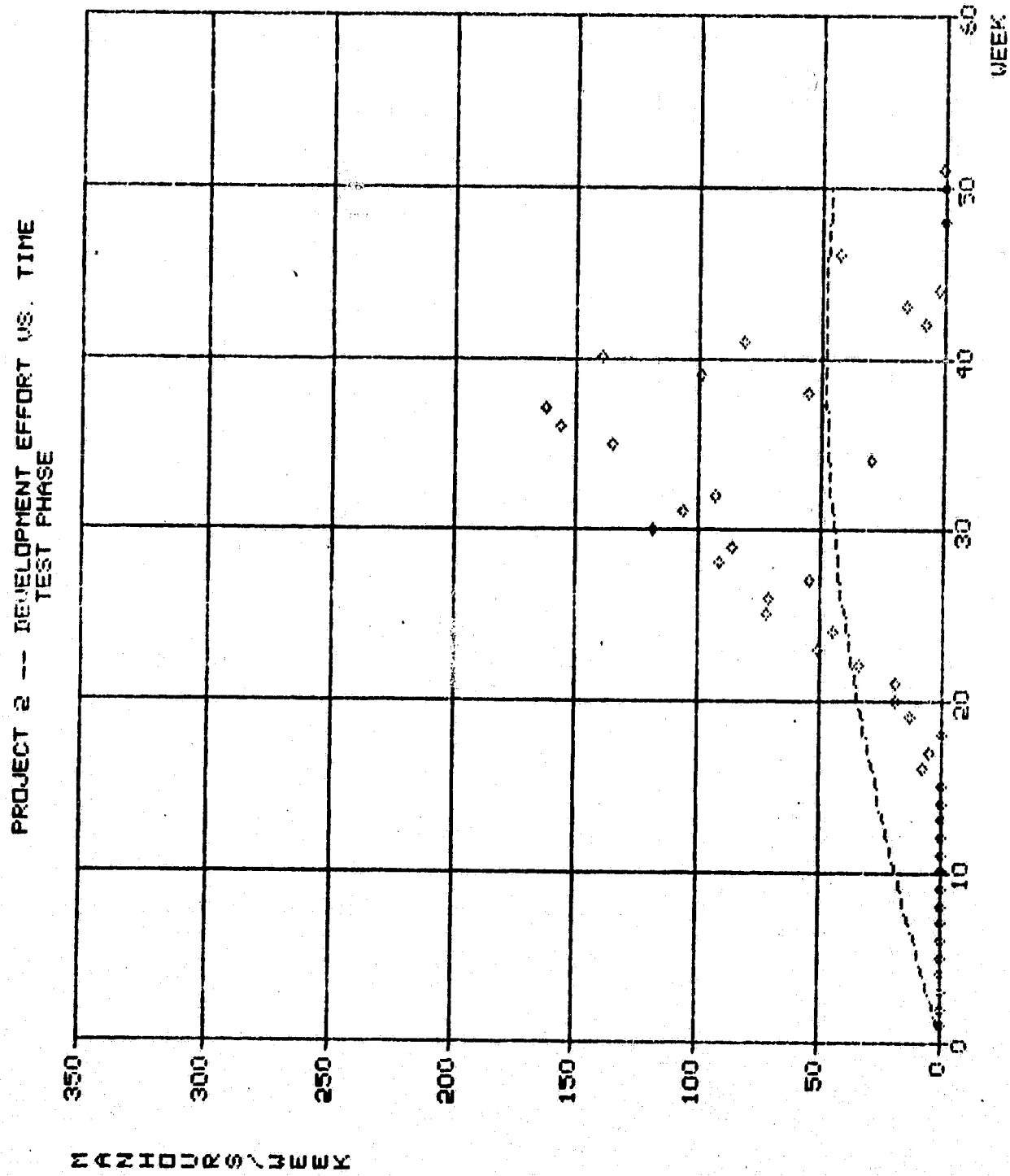


C-2

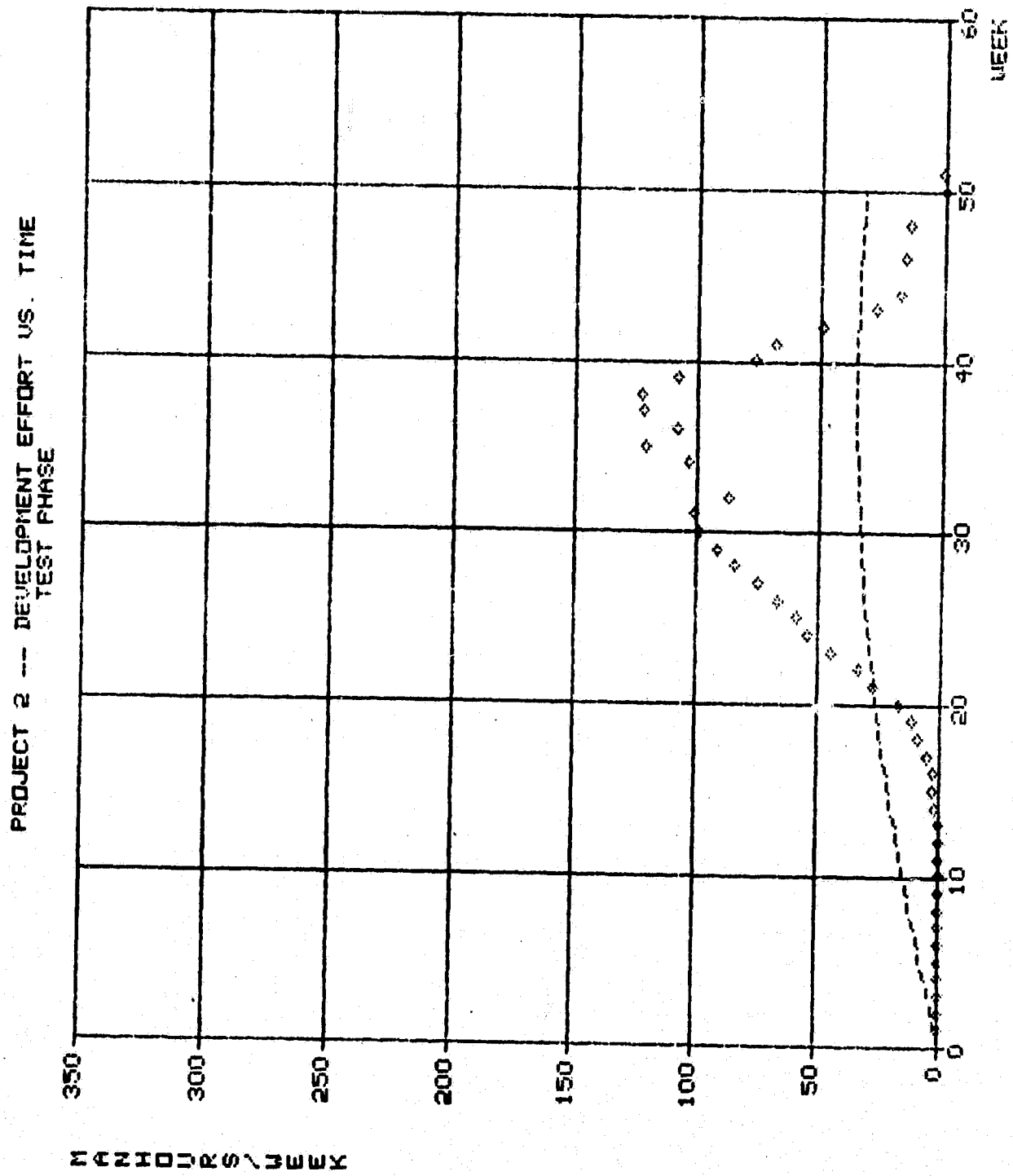
ORIGINAL PAGE IS
OF POOR QUALITY



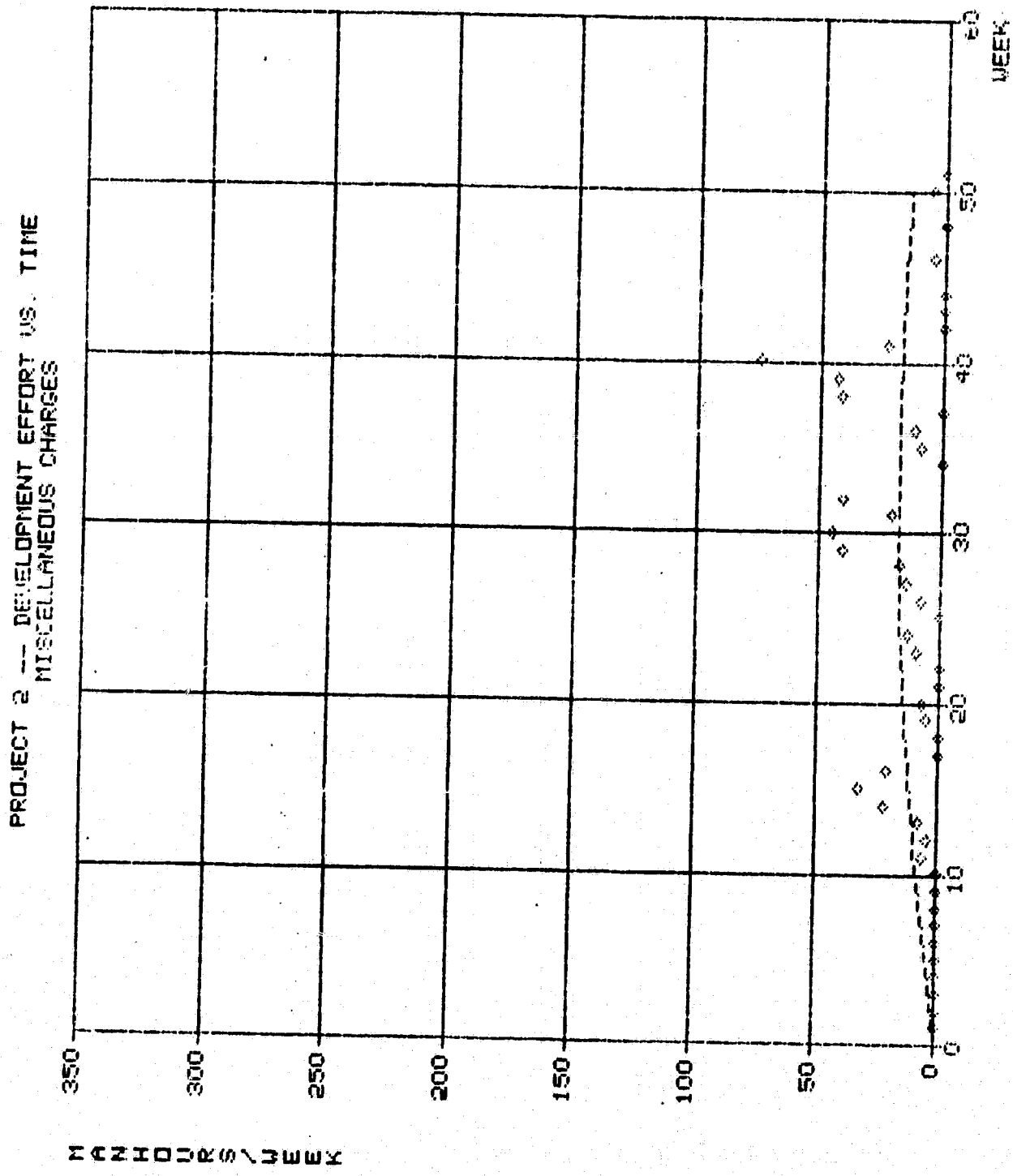
ORIGINAL PAGE IS
OF POOR QUALITY



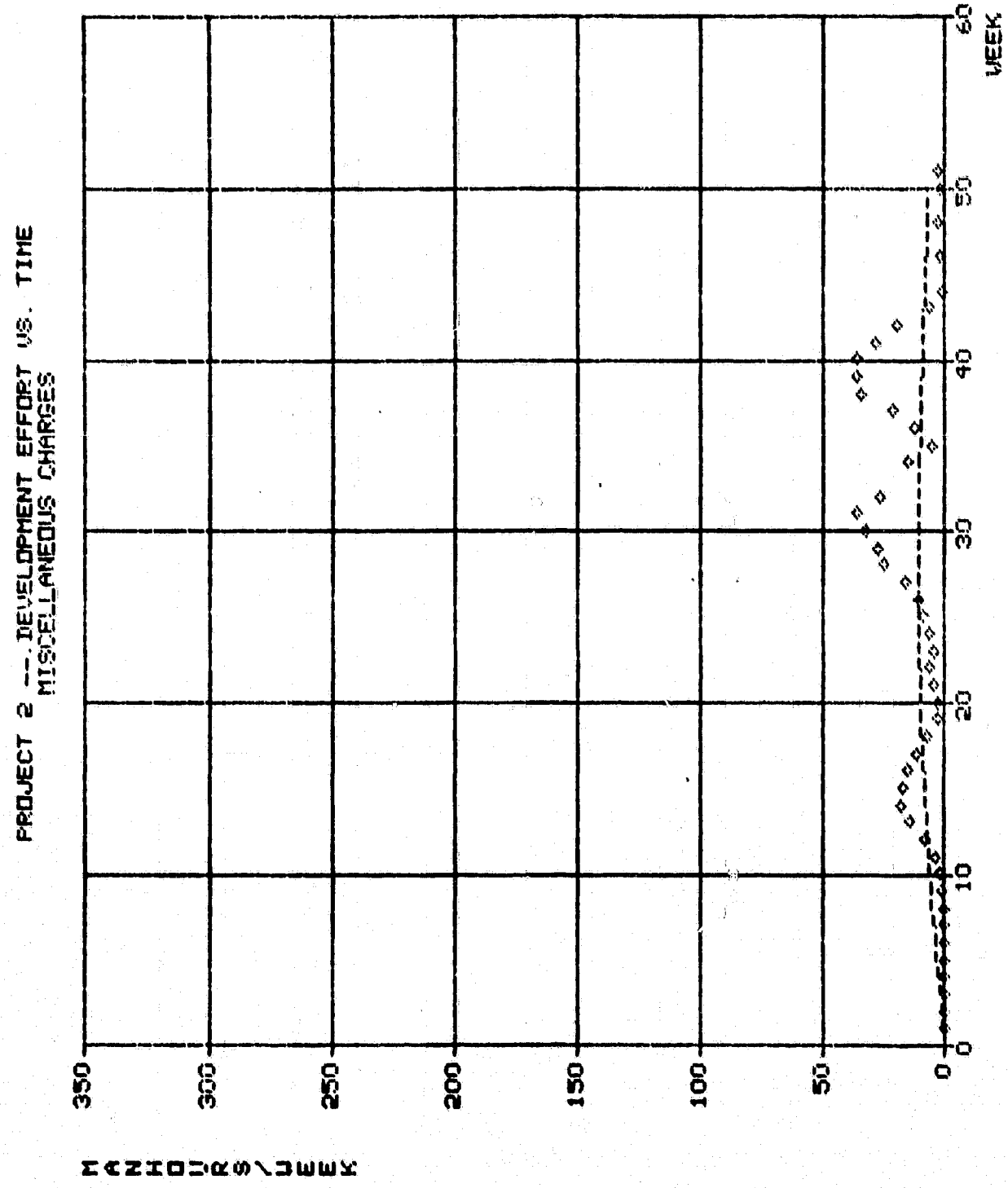
ORIGINAL PAGE IS
OF POOR QUALITY



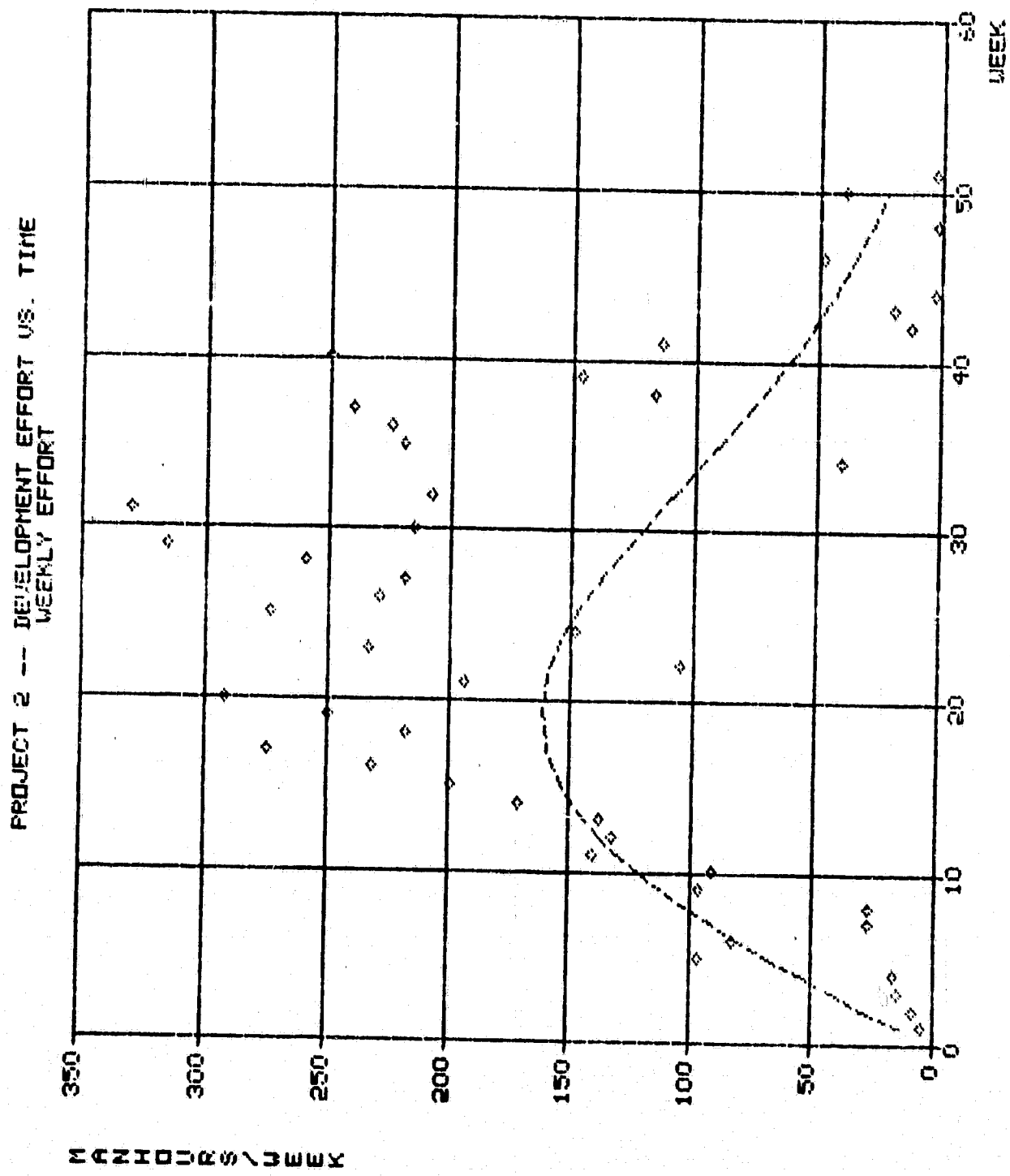
ORIGINAL PAGE IS
OF POOR QUALITY

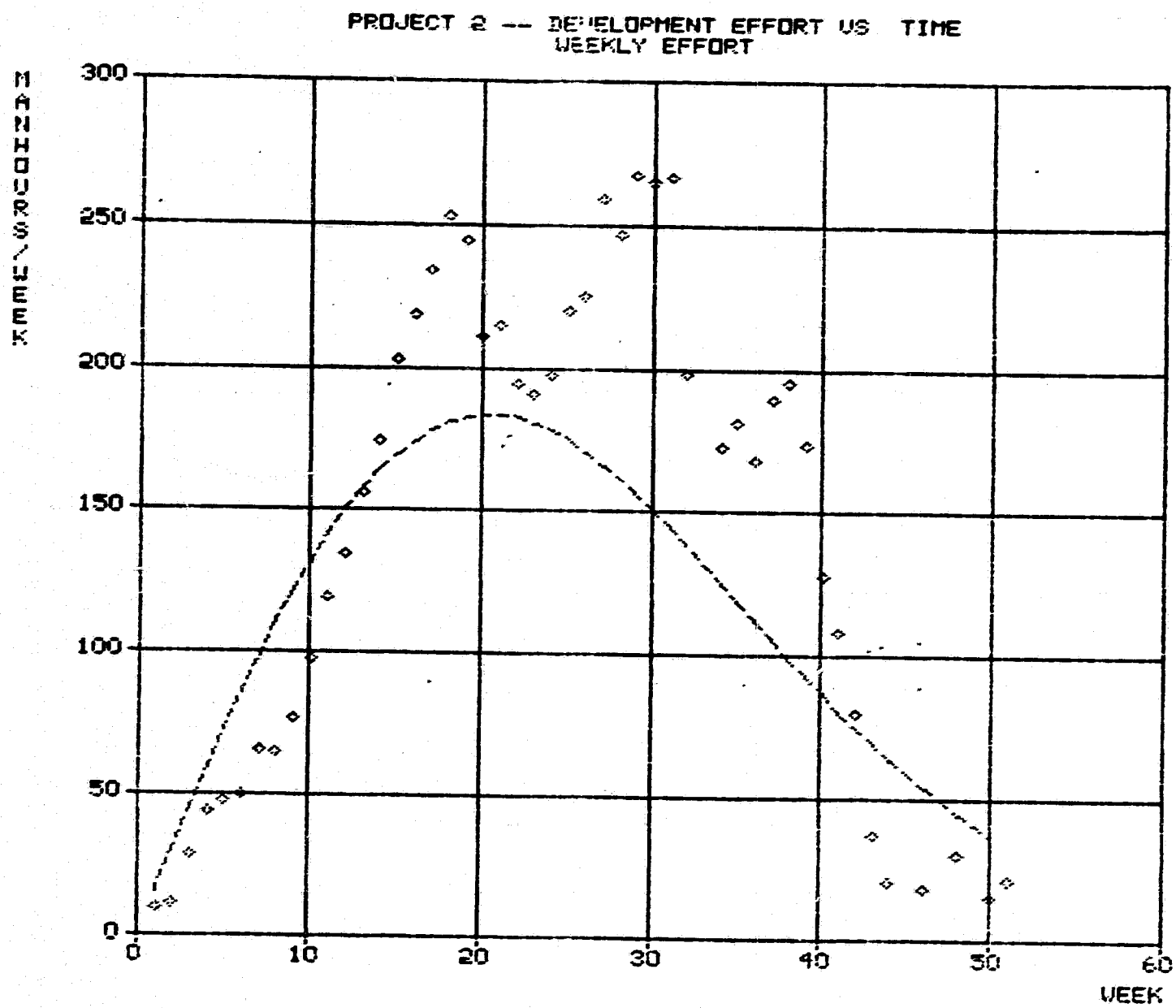


ORIGINAL PAGE IS
OF POOR QUALITY



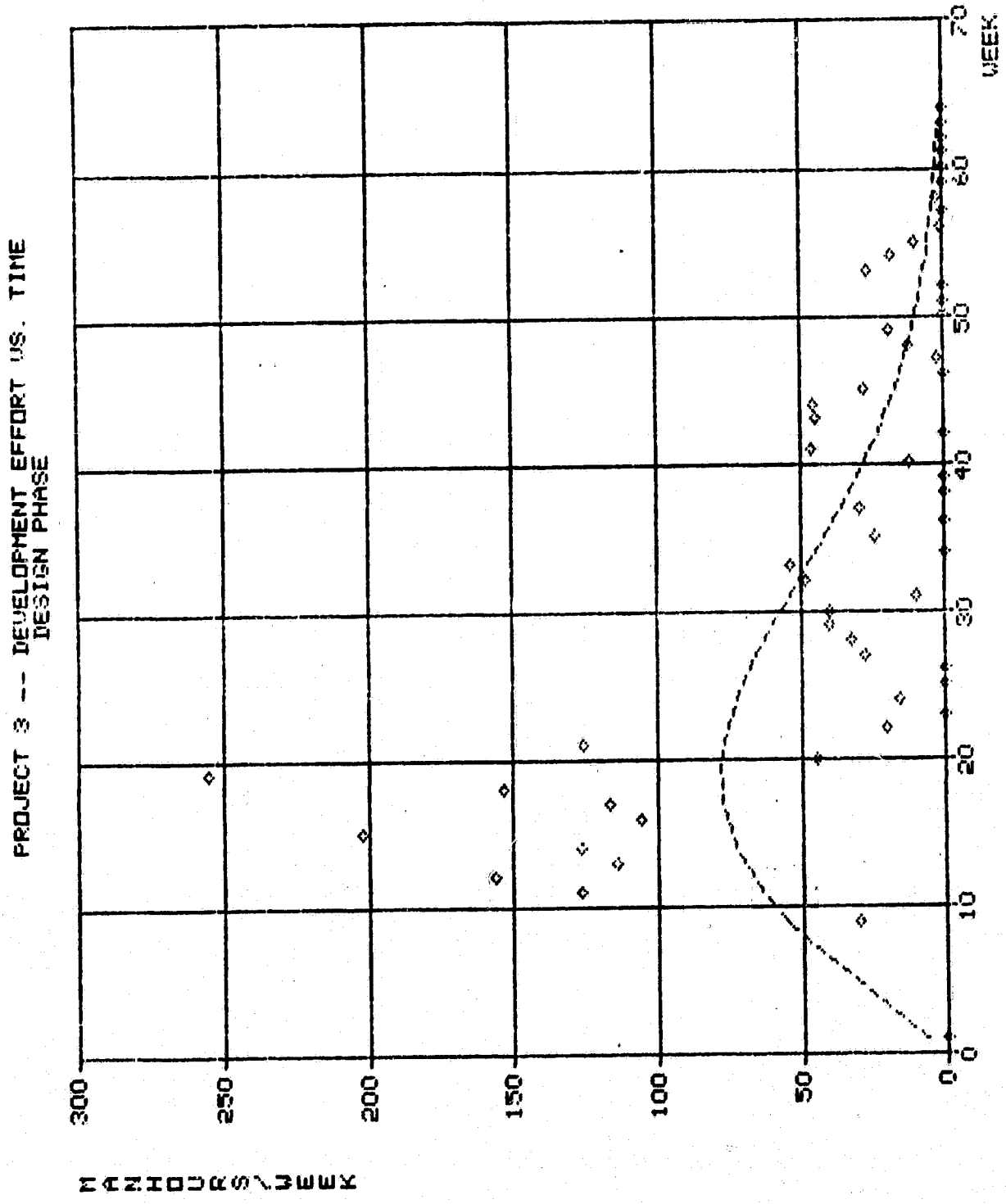
ORIGINAL PAGE IS
OF POOR QUALITY



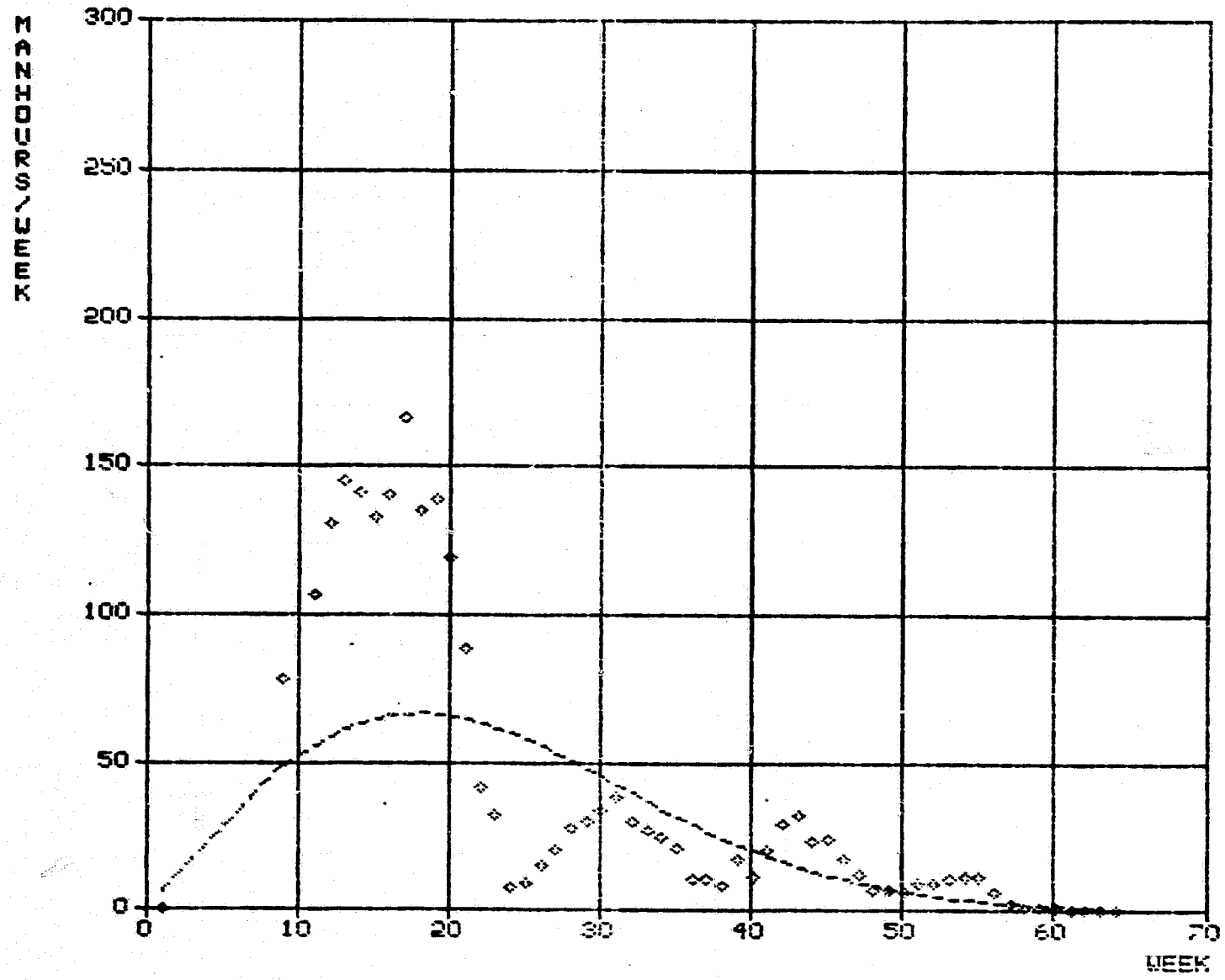


ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

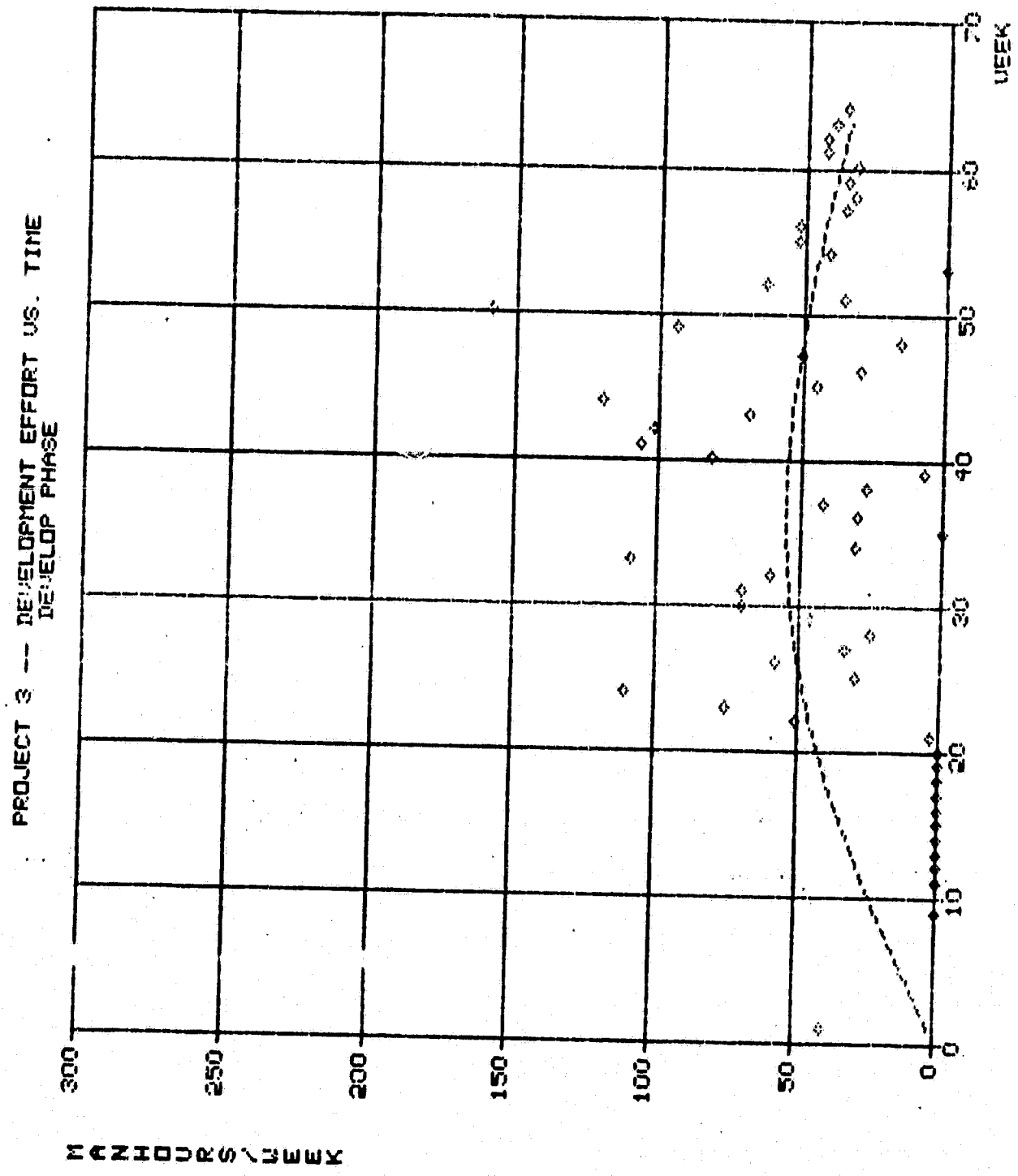


PROJECT 3 -- DEVELOPMENT EFFORT VS. TIME
DESIGN PHASE

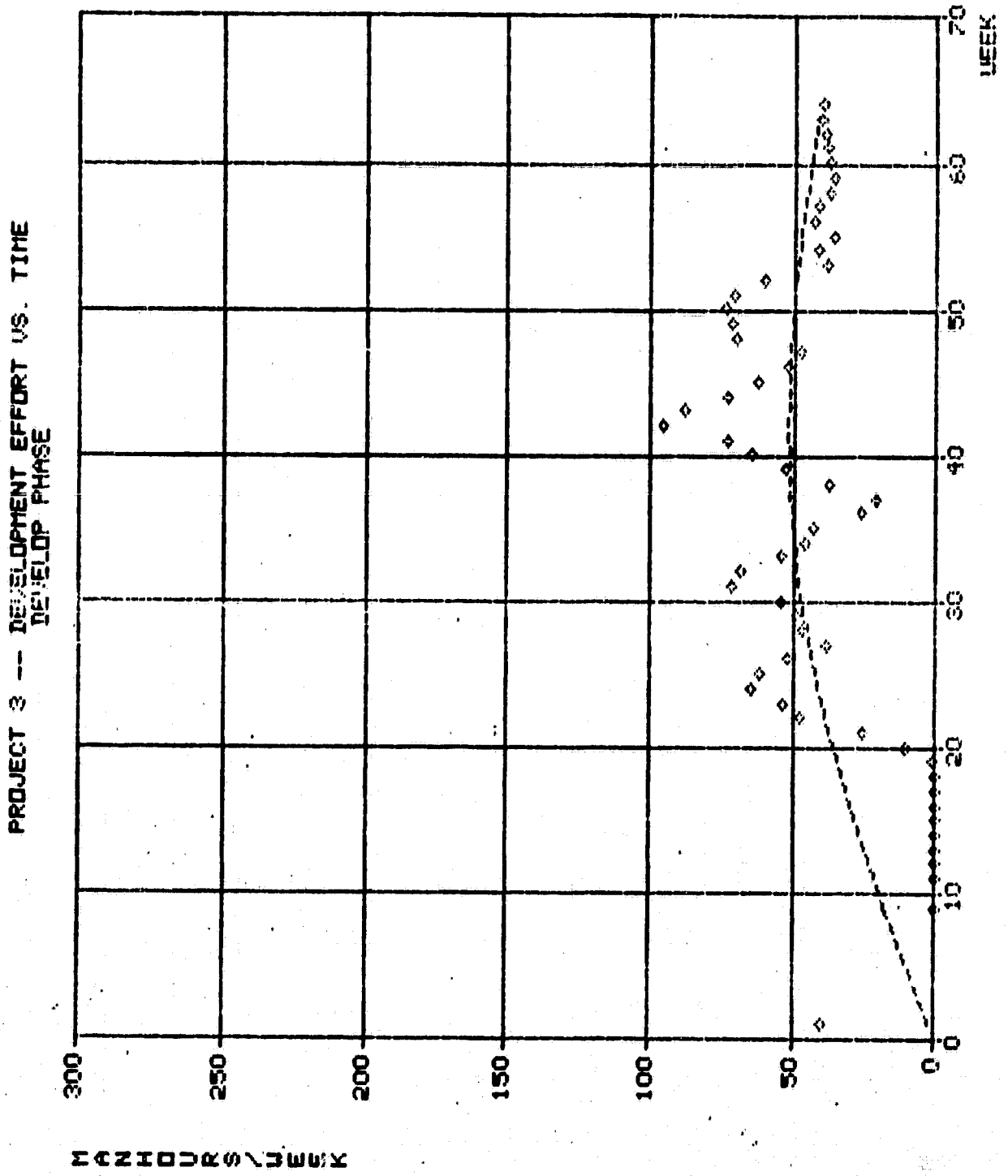


ORIGINAL PAGE IS
OF POOR QUALITY

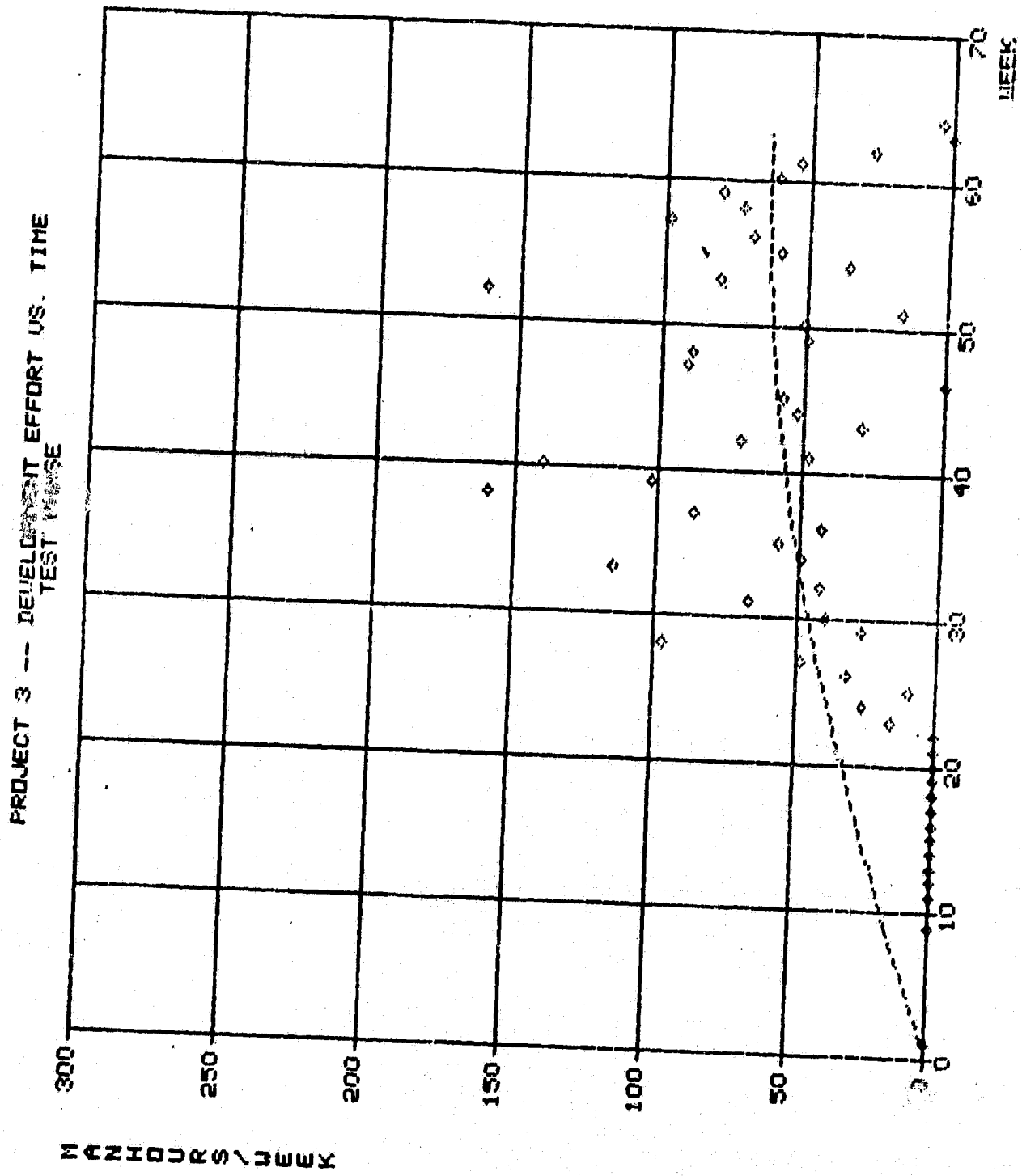
ORIGINAL PAGE IS
OF POOR QUALITY



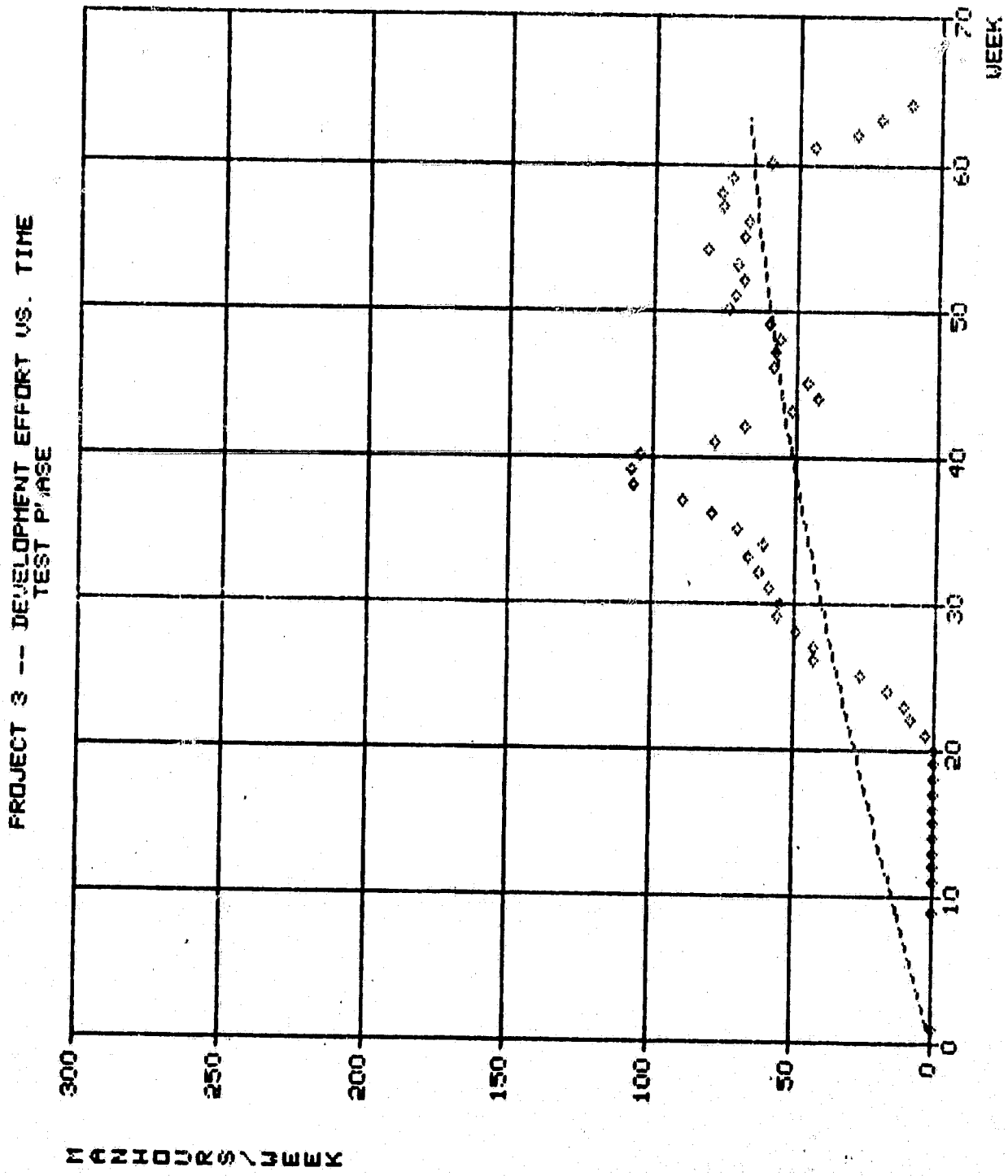
ORIGINAL PAGE IS
OF POOR QUALITY



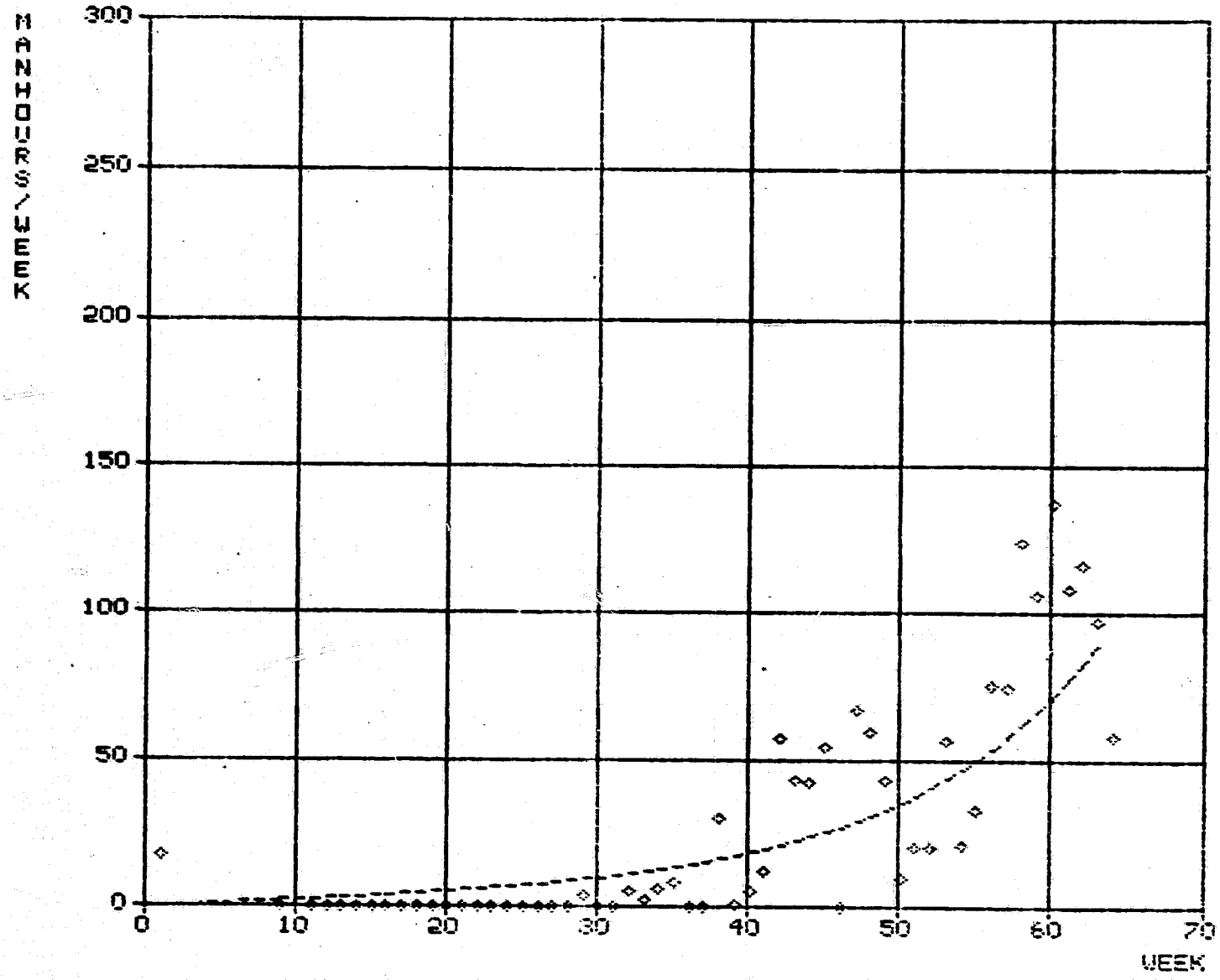
ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY

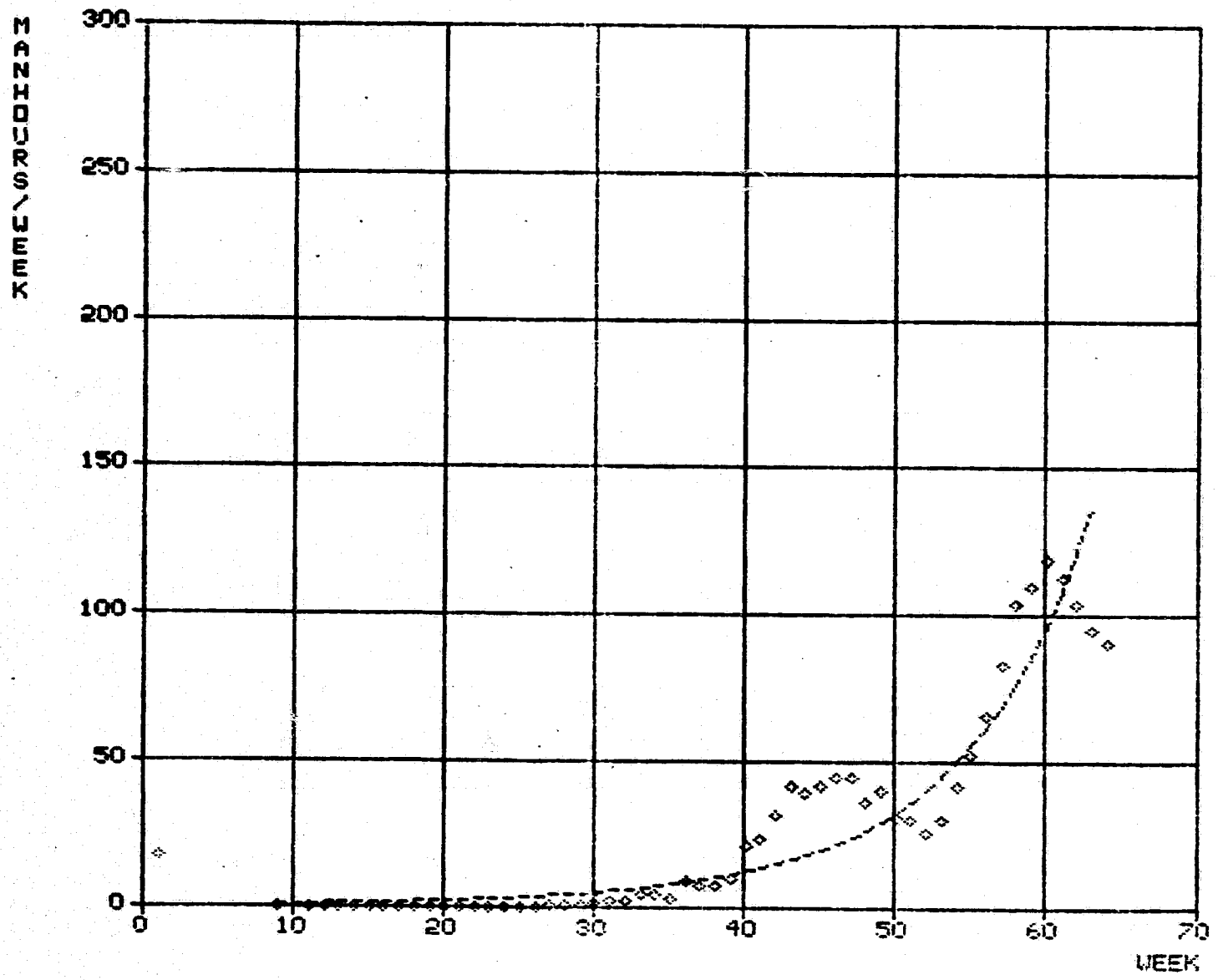


PROJECT 3 -- DEVELOPMENT EFFORT VS. TIME
MISCELLANEOUS CHARGES



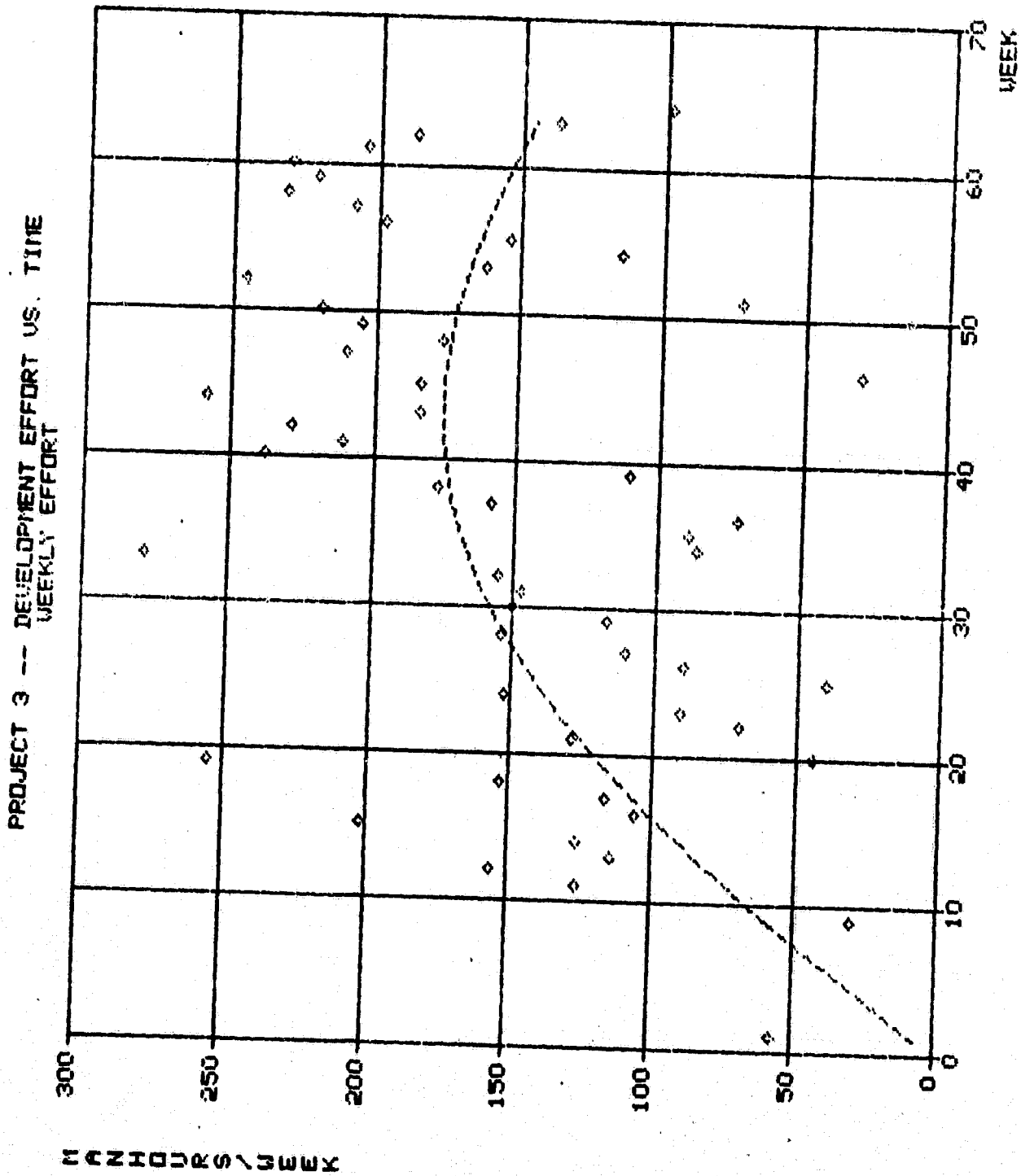
ORIGINAL PAGE IS
OF POOR QUALITY

PROJECT 3 -- DEVELOPMENT EFFORT VS. TIME
MISCELLANEOUS CHARGES

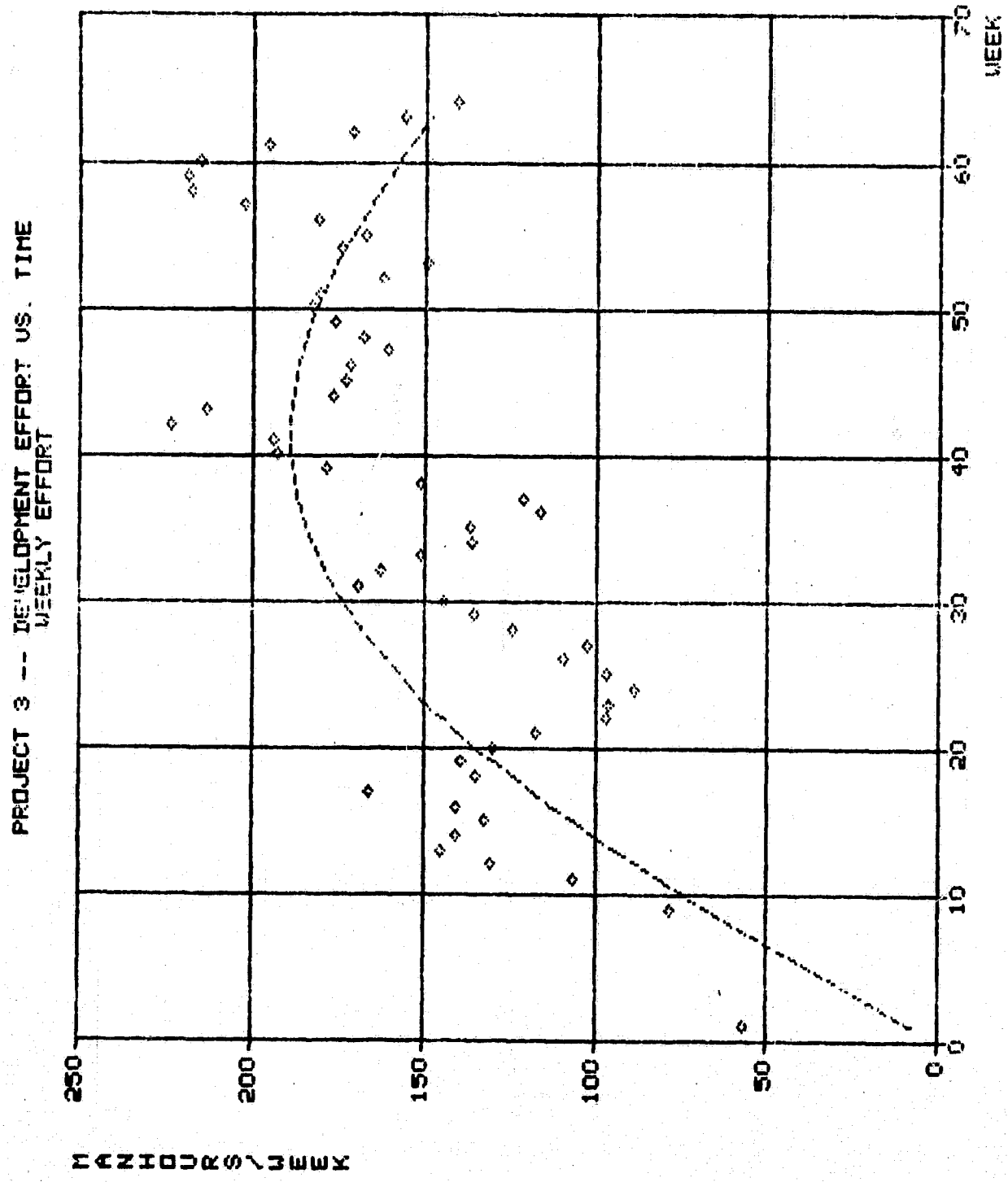


ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

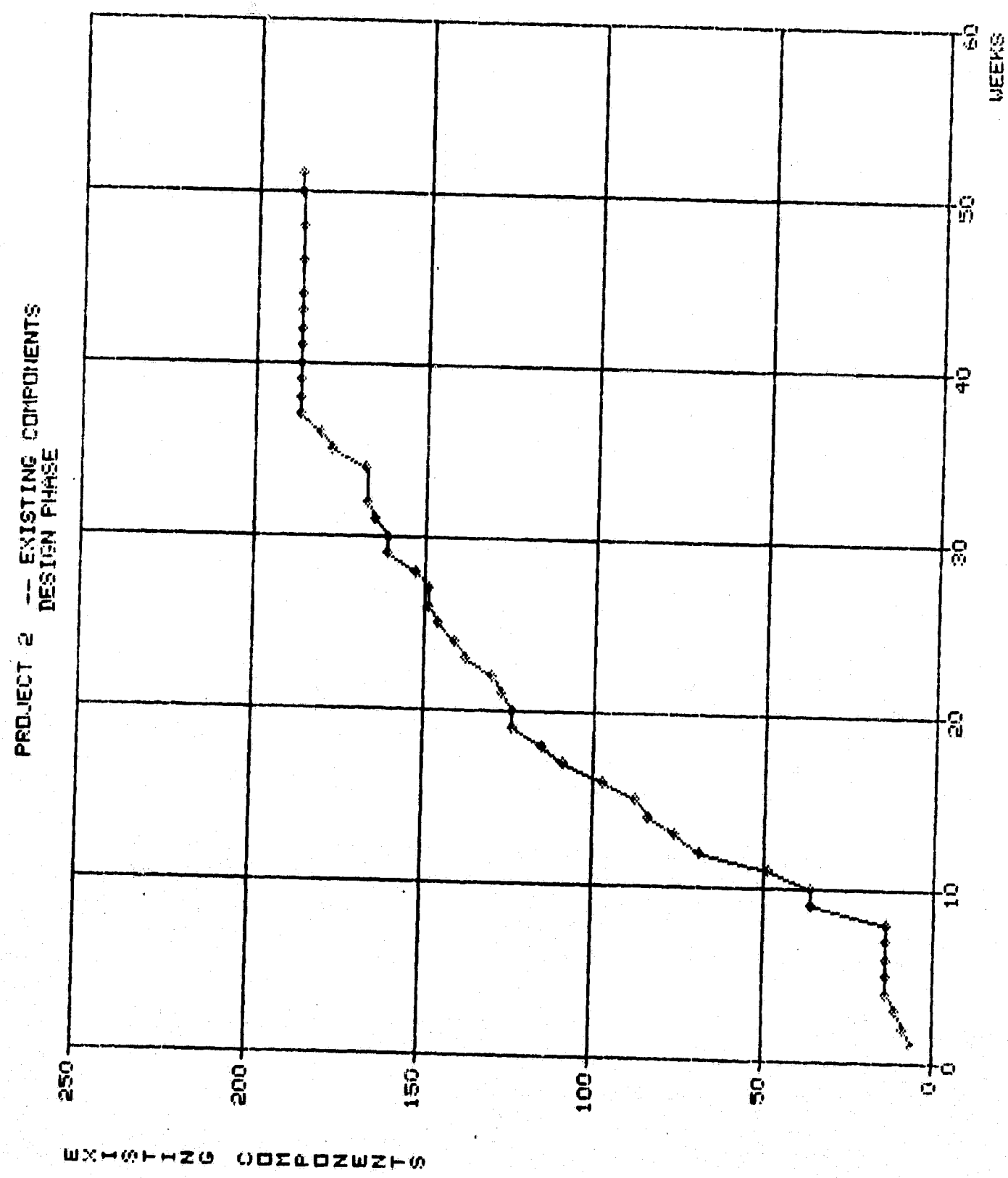


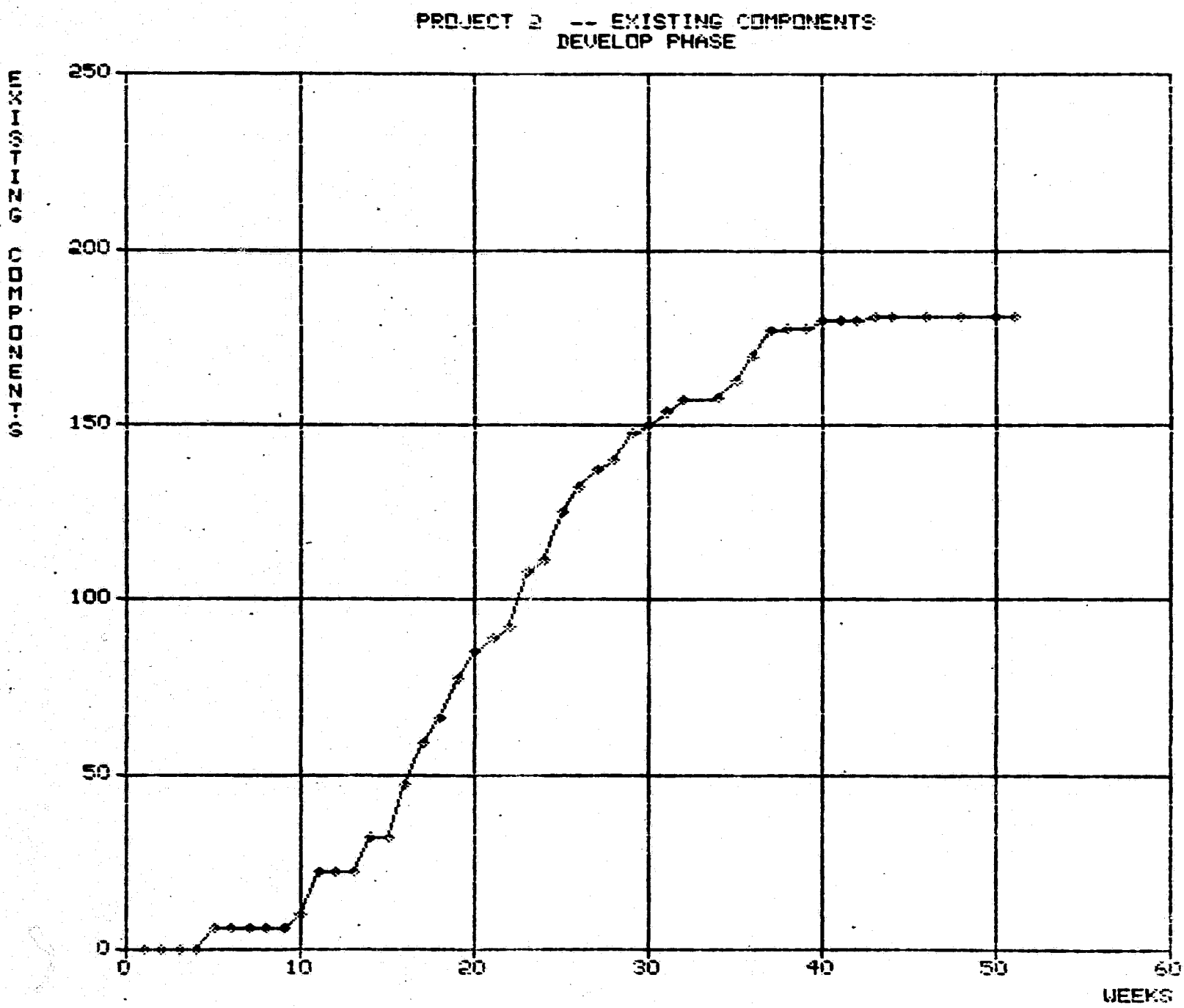
ORIGINAL PAGE IS
OF POOR QUALITY



APPENDIX B

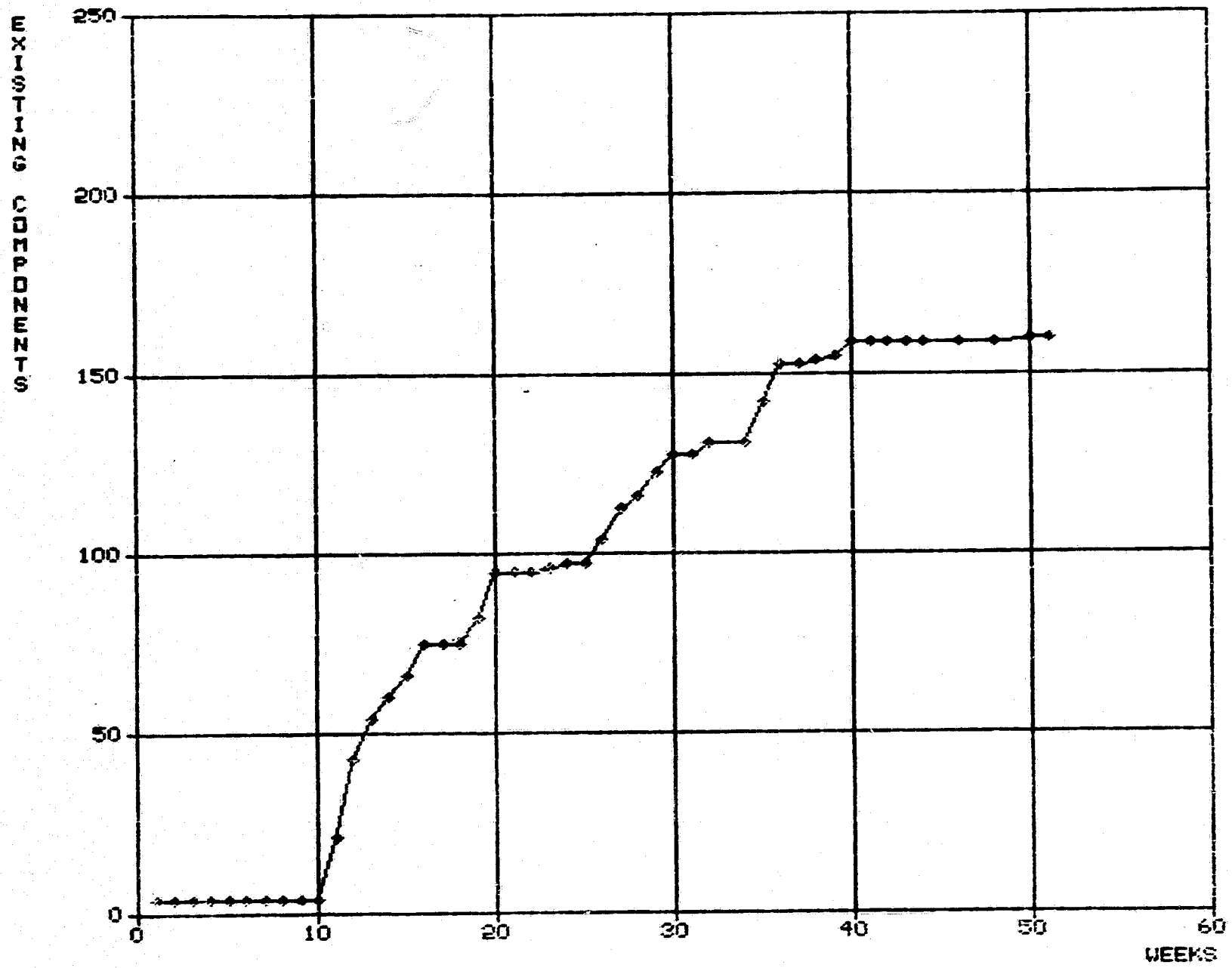
ORIGINAL PAGE IS
OF POOR QUALITY





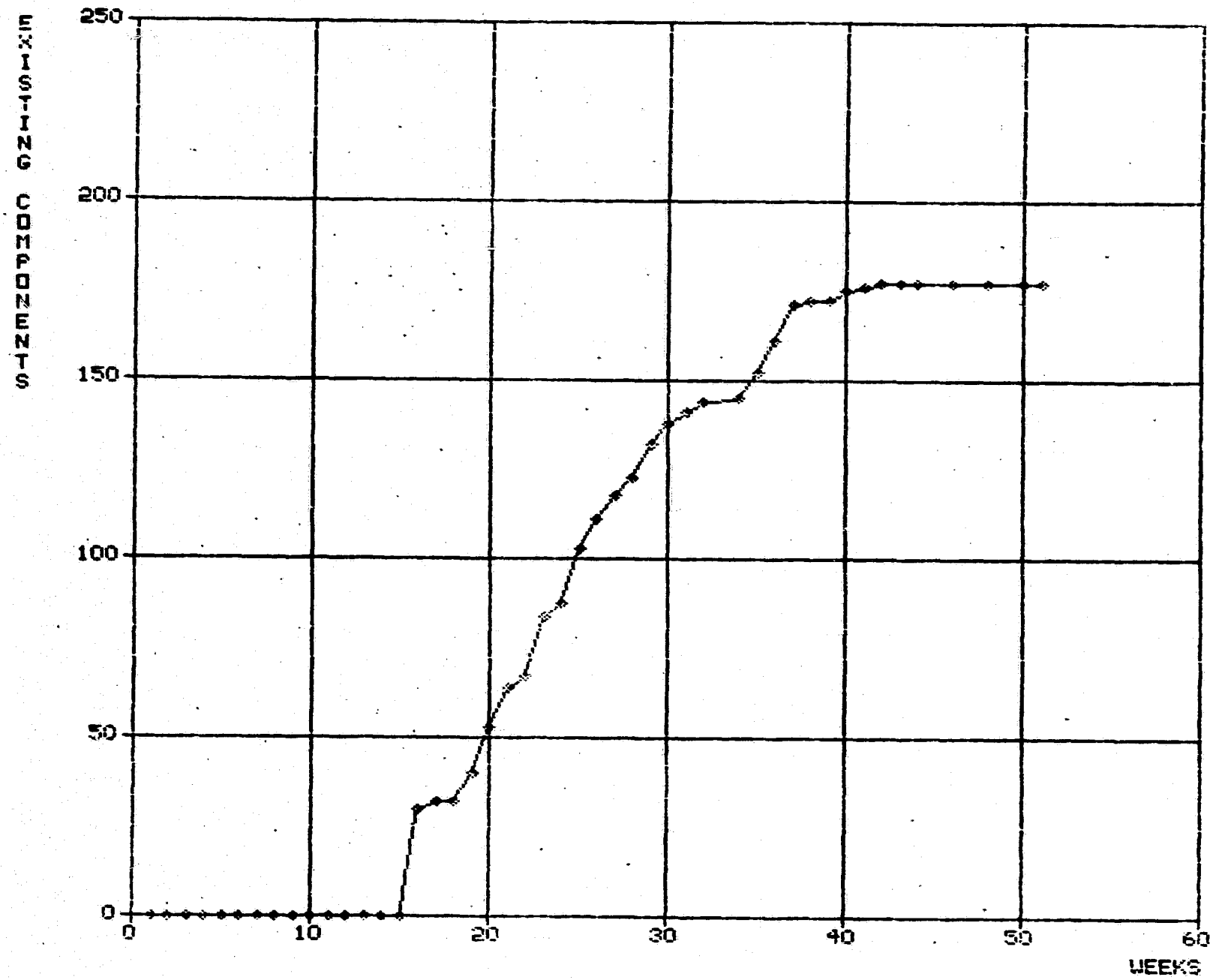
ORIGINAL PAGE IS
OF POOR QUALITY

PROJECT 2 -- EXISTING COMPONENTS
OTHER CHARGES



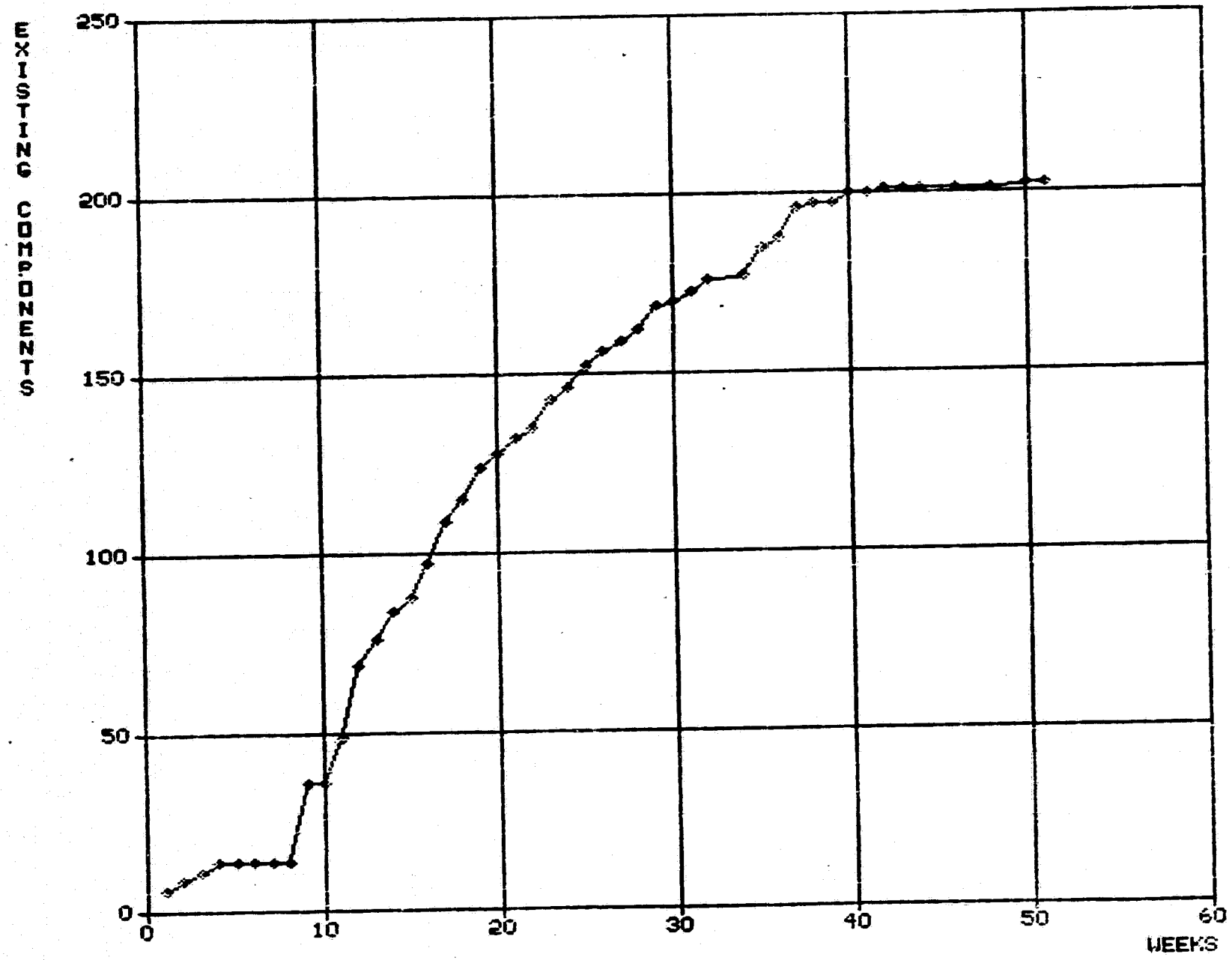
ORIGINAL PAGE IS
OF POOR QUALITY

PROJECT 2 -- EXISTING COMPONENTS
TEST PHASE



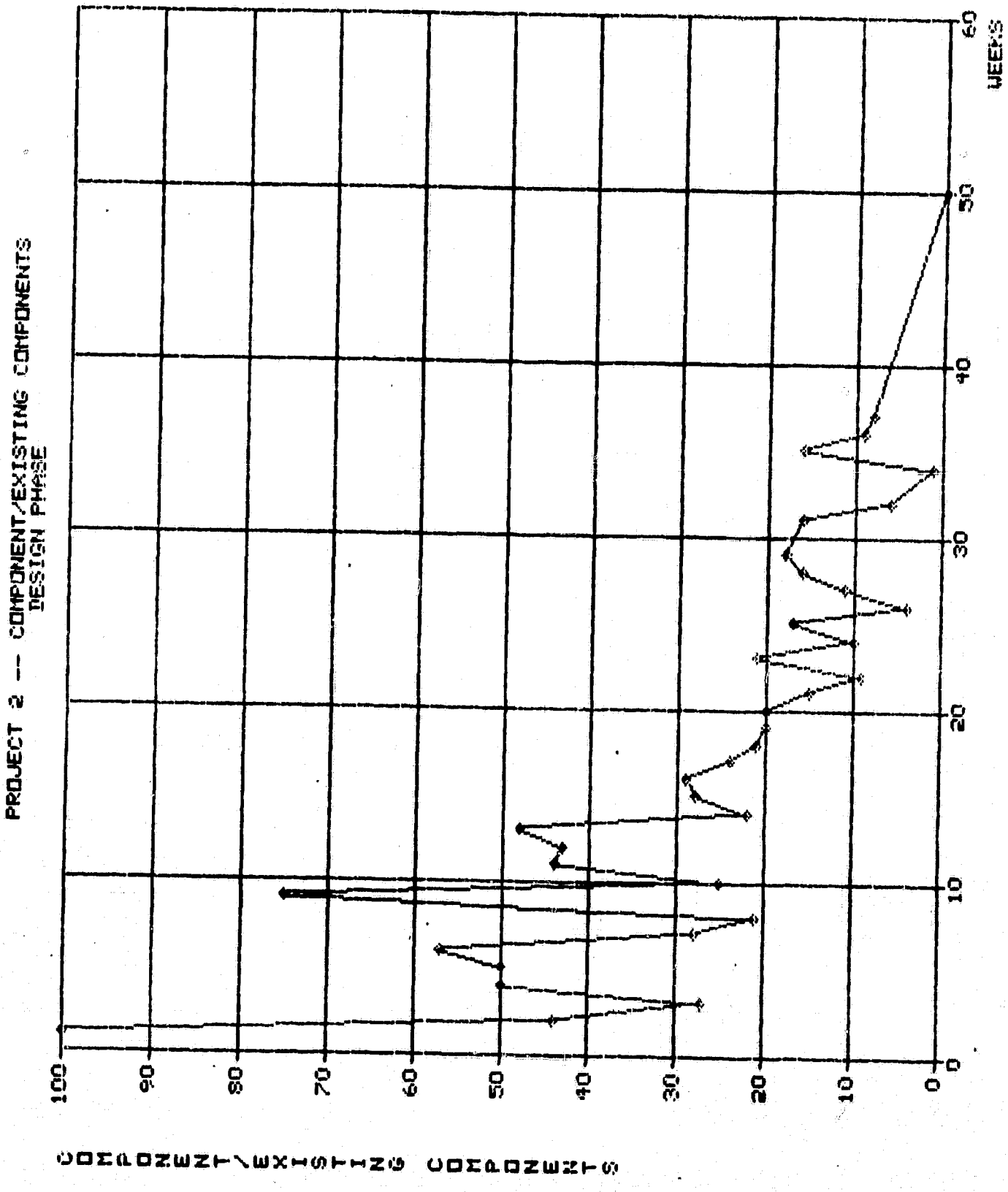
ORIGINAL PAGE IS
OF POOR QUALITY

PROJECT 2 -- EXISTING COMPONENTS
WEEKLY TOTALS

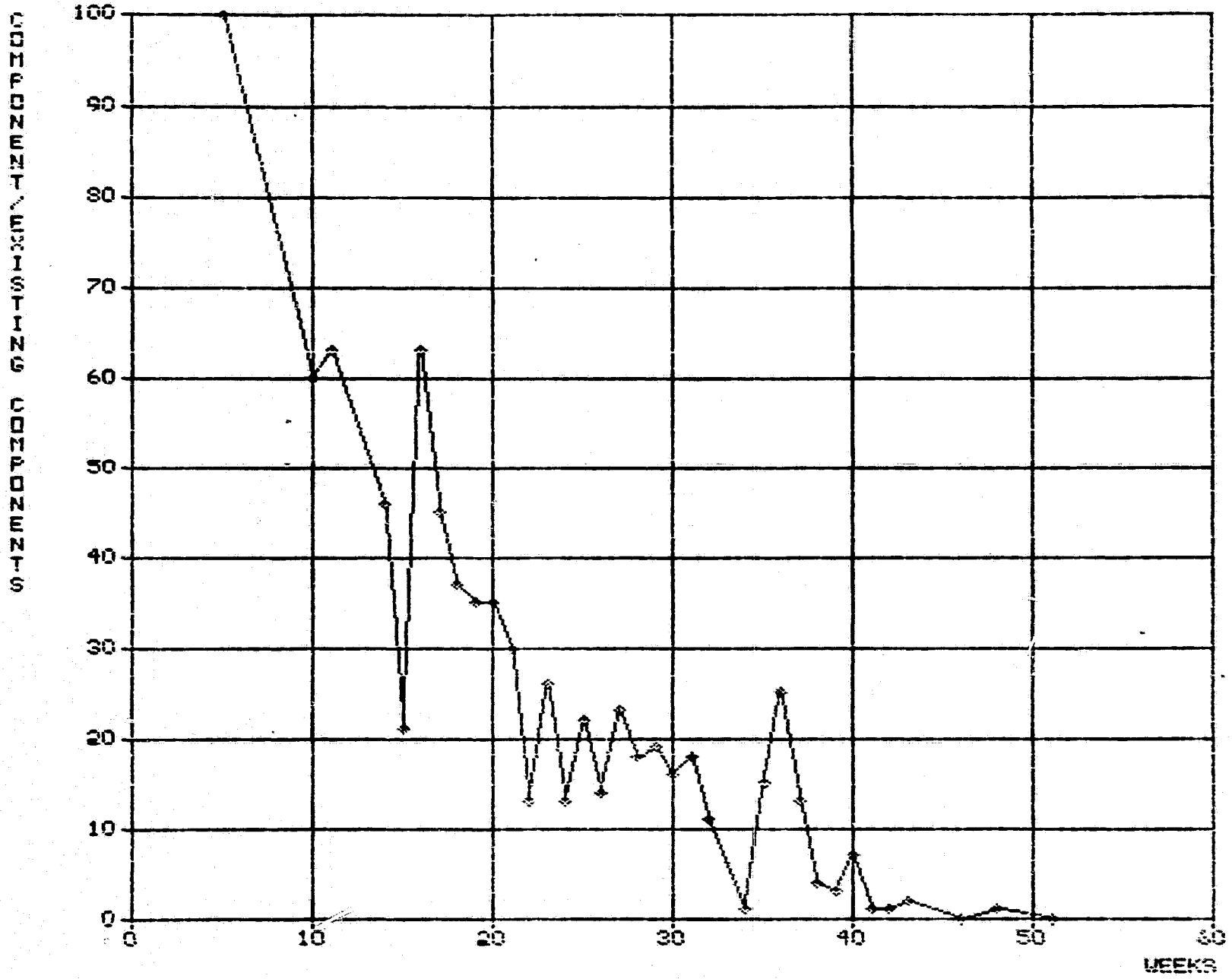


ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

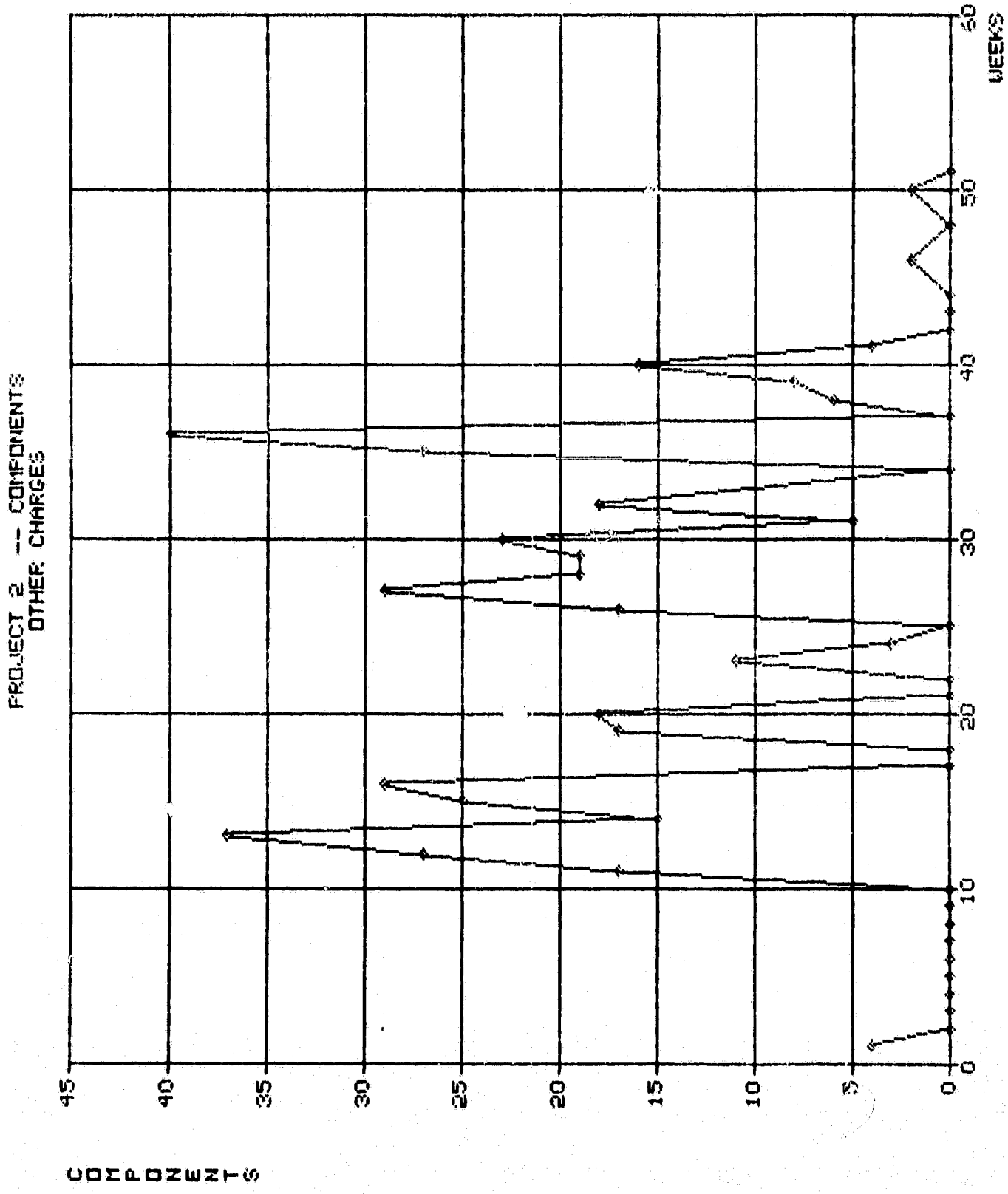


PROJECT 2 -- COMPONENT/EXISTING COMPONENTS
DEVELOP PHASE

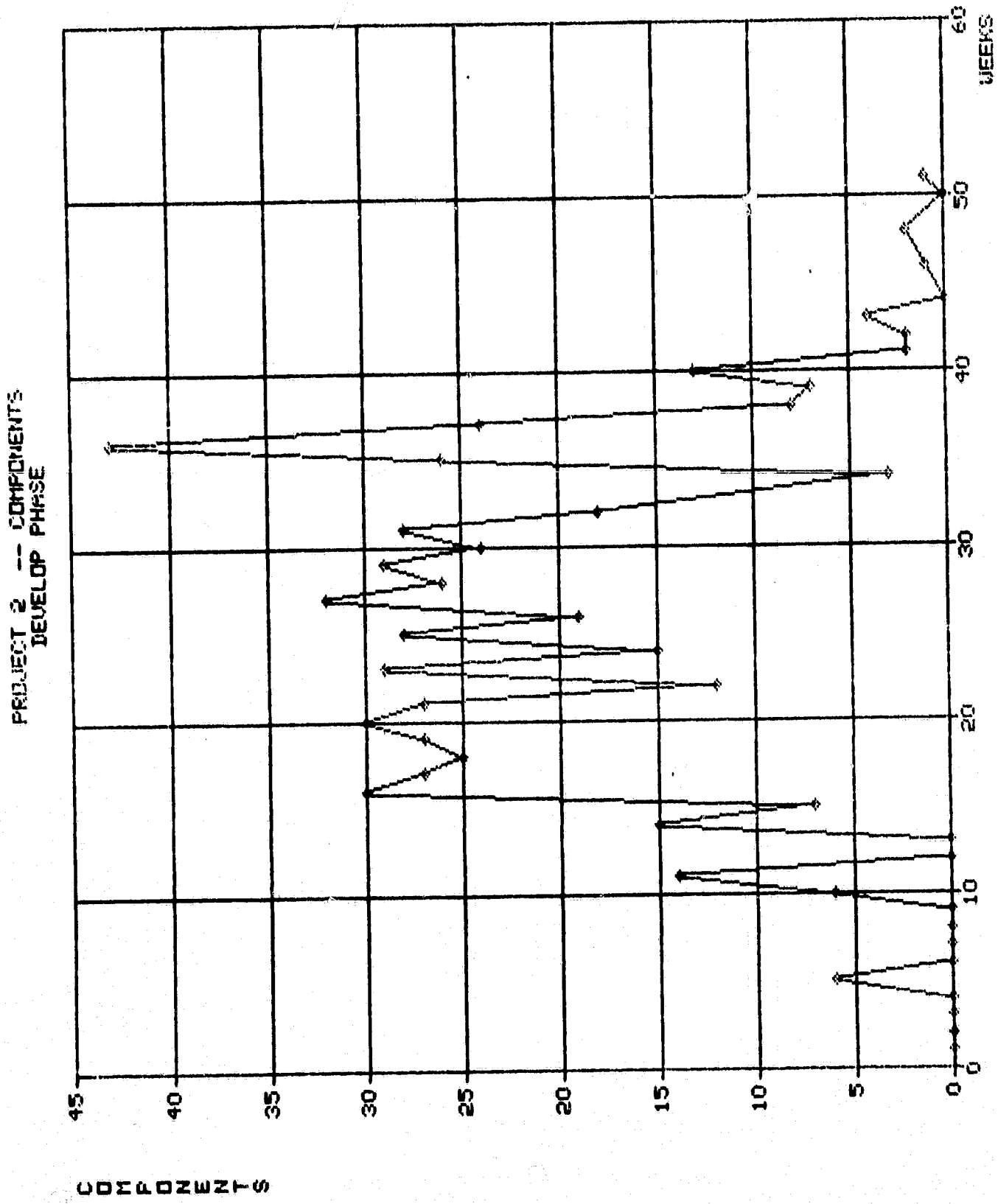


ORIGINAL PAGE IS
OF POOR QUALITY

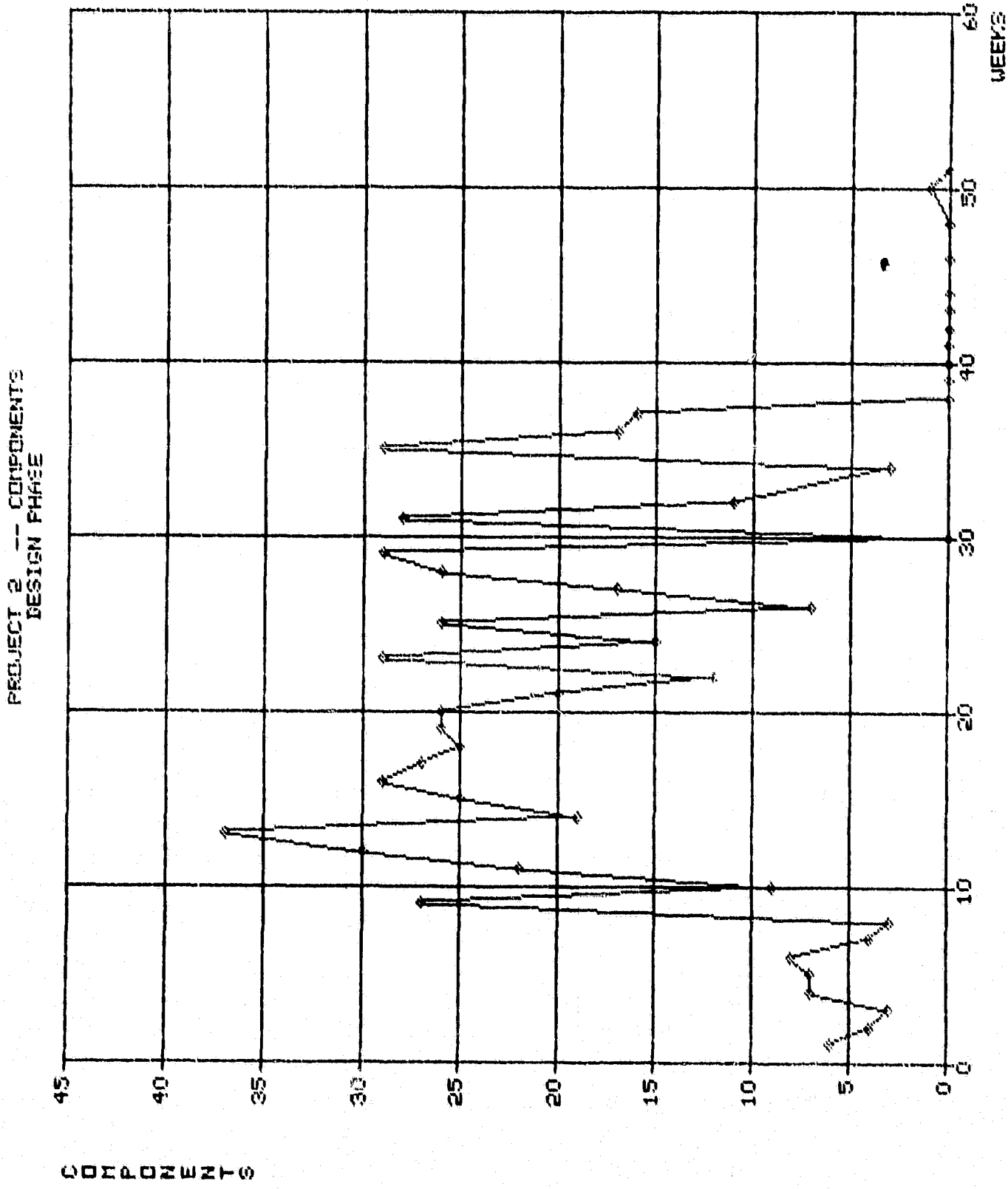
ORIGINAL PAGE IS
OF POOR QUALITY



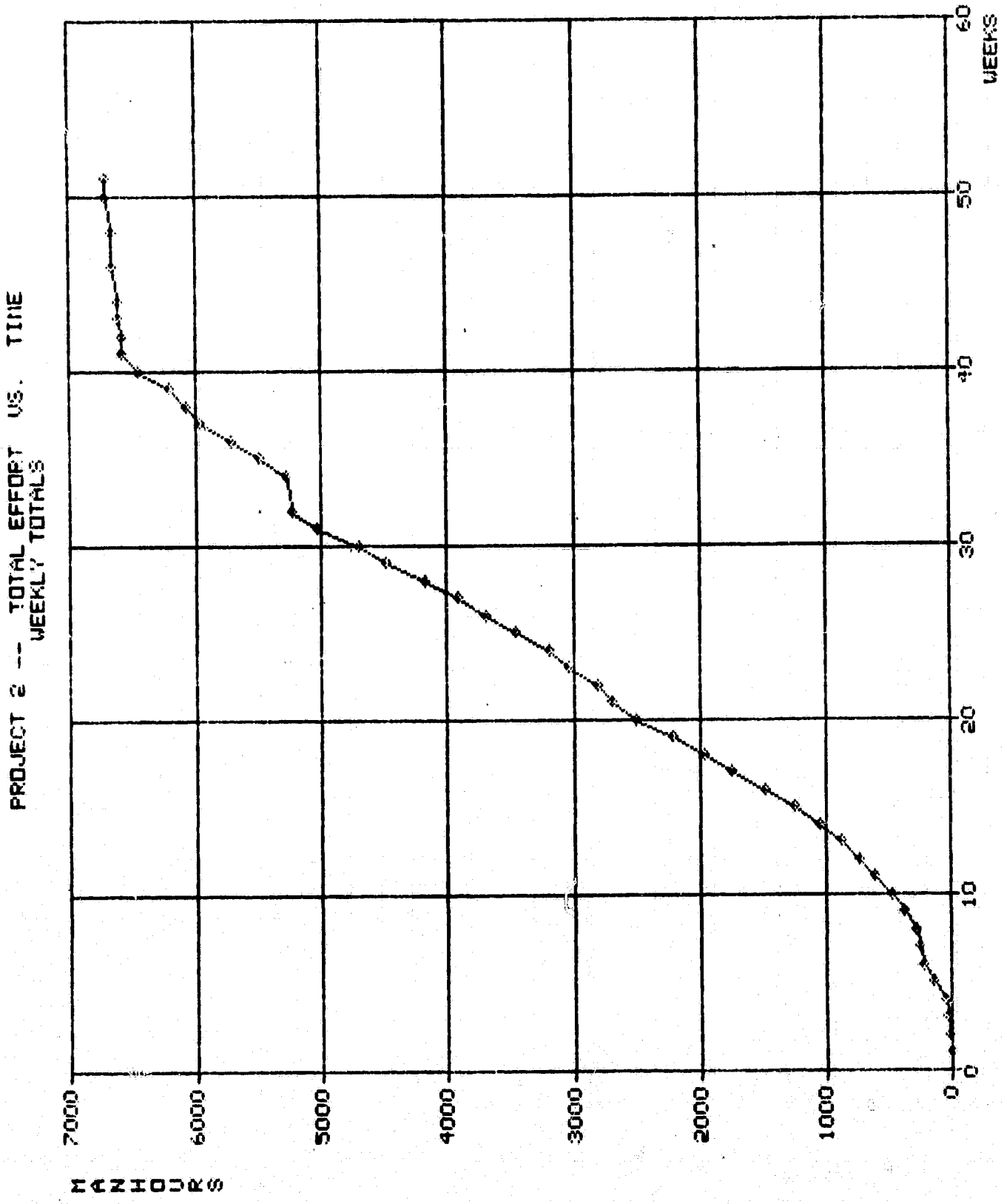
ORIGINAL PAGE IS
OF POOR QUALITY



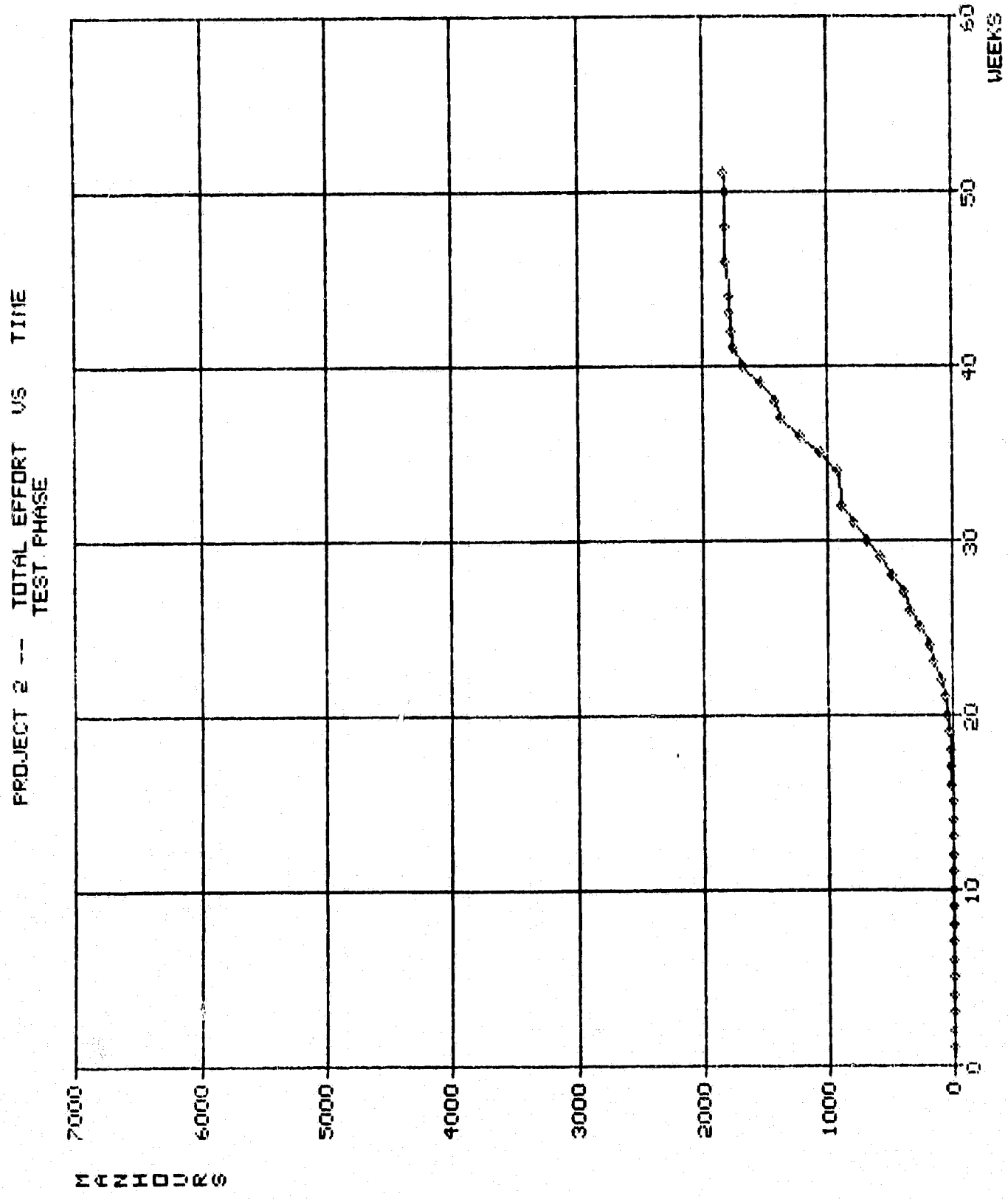
ORIGINAL PAGE IS
OF POOR QUALITY



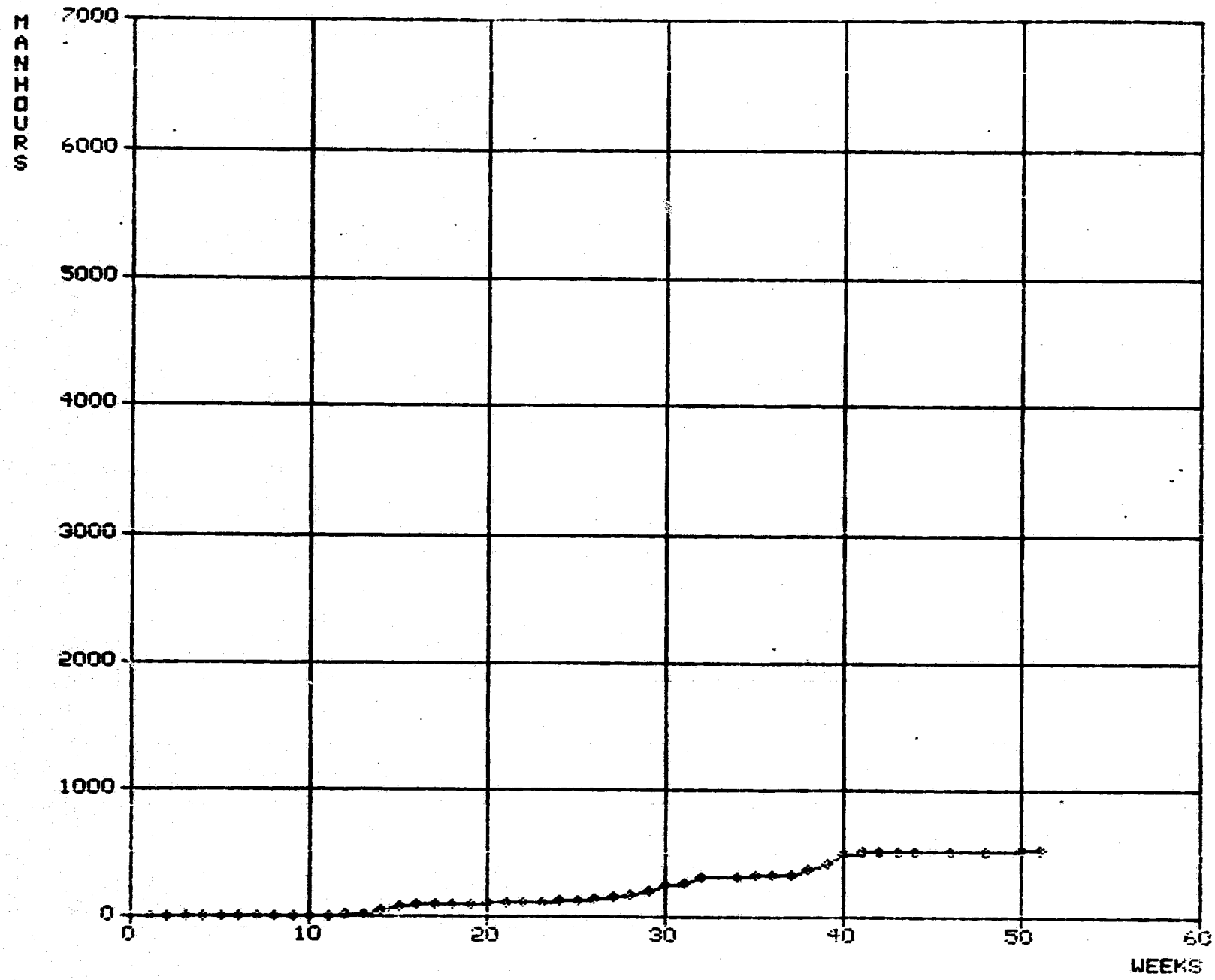
ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY

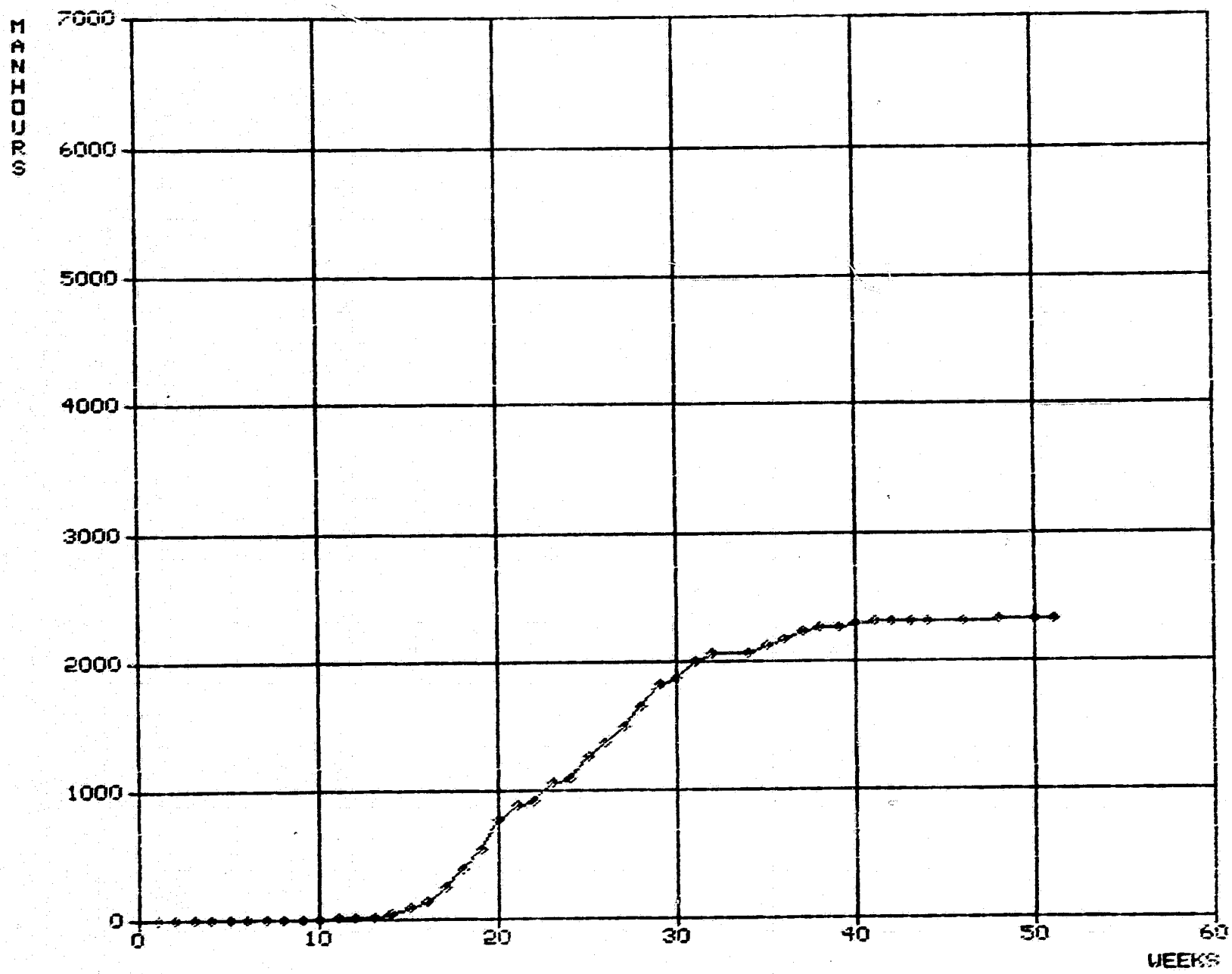


PROJECT 2 -- TOTAL EFFORT VS. TIME
OTHER CHARGES



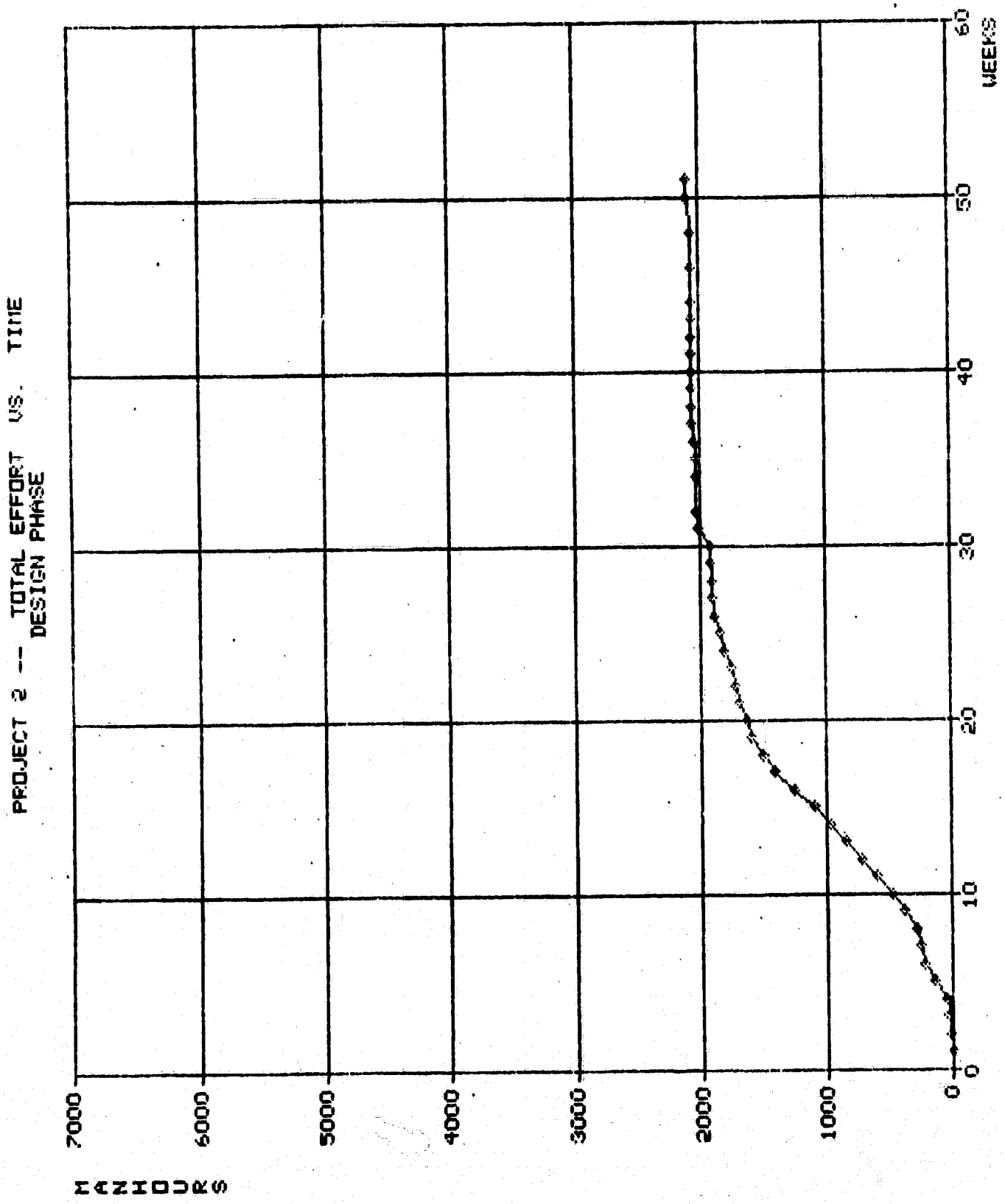
ORIGINAL PAGE IS
OF POOR QUALITY

PROJECT 2 -- TOTAL EFFORT VS TIME
DEVELOP PHASE

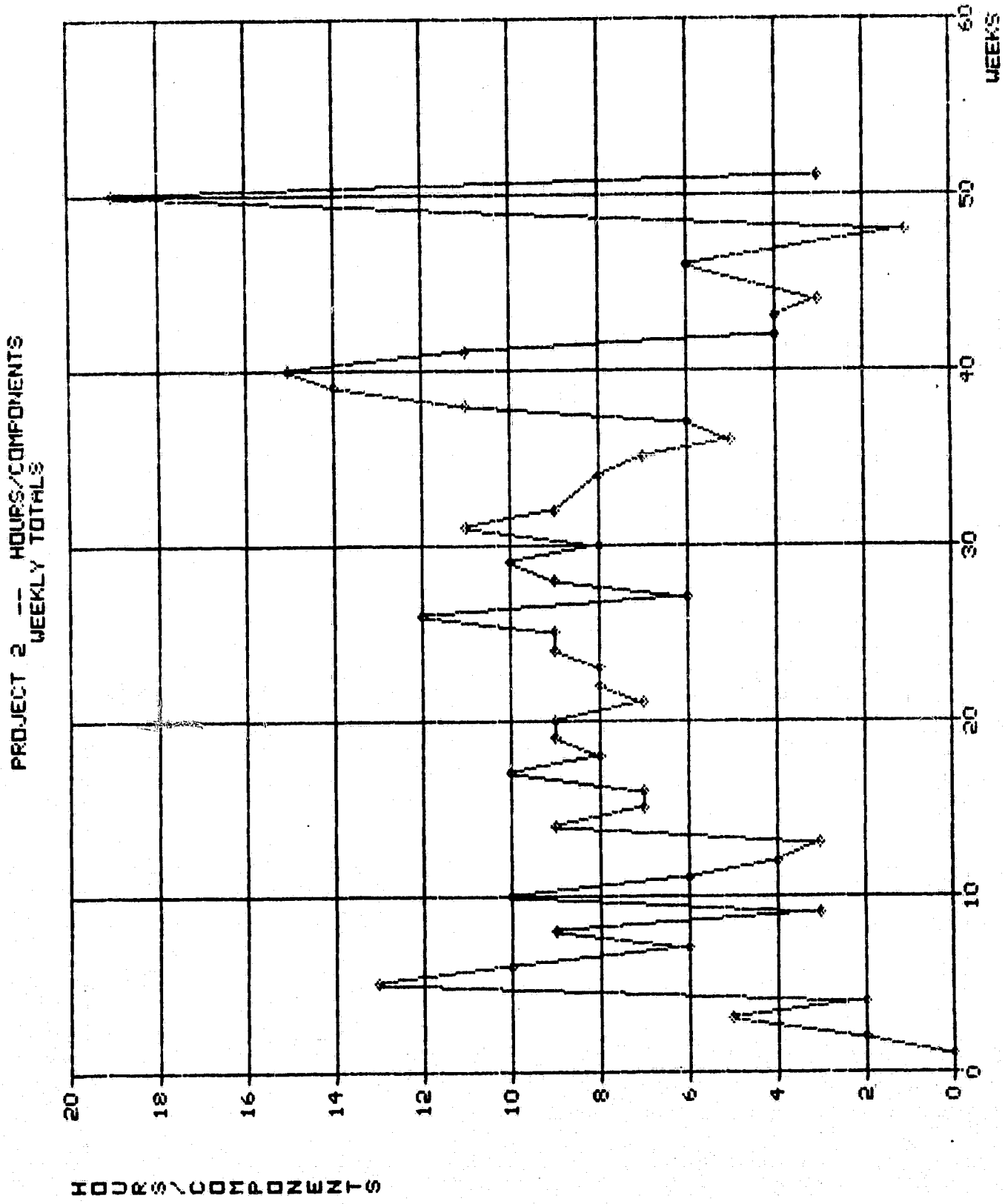


ORIGINAL PAGE IS
OF POOR QUALITY

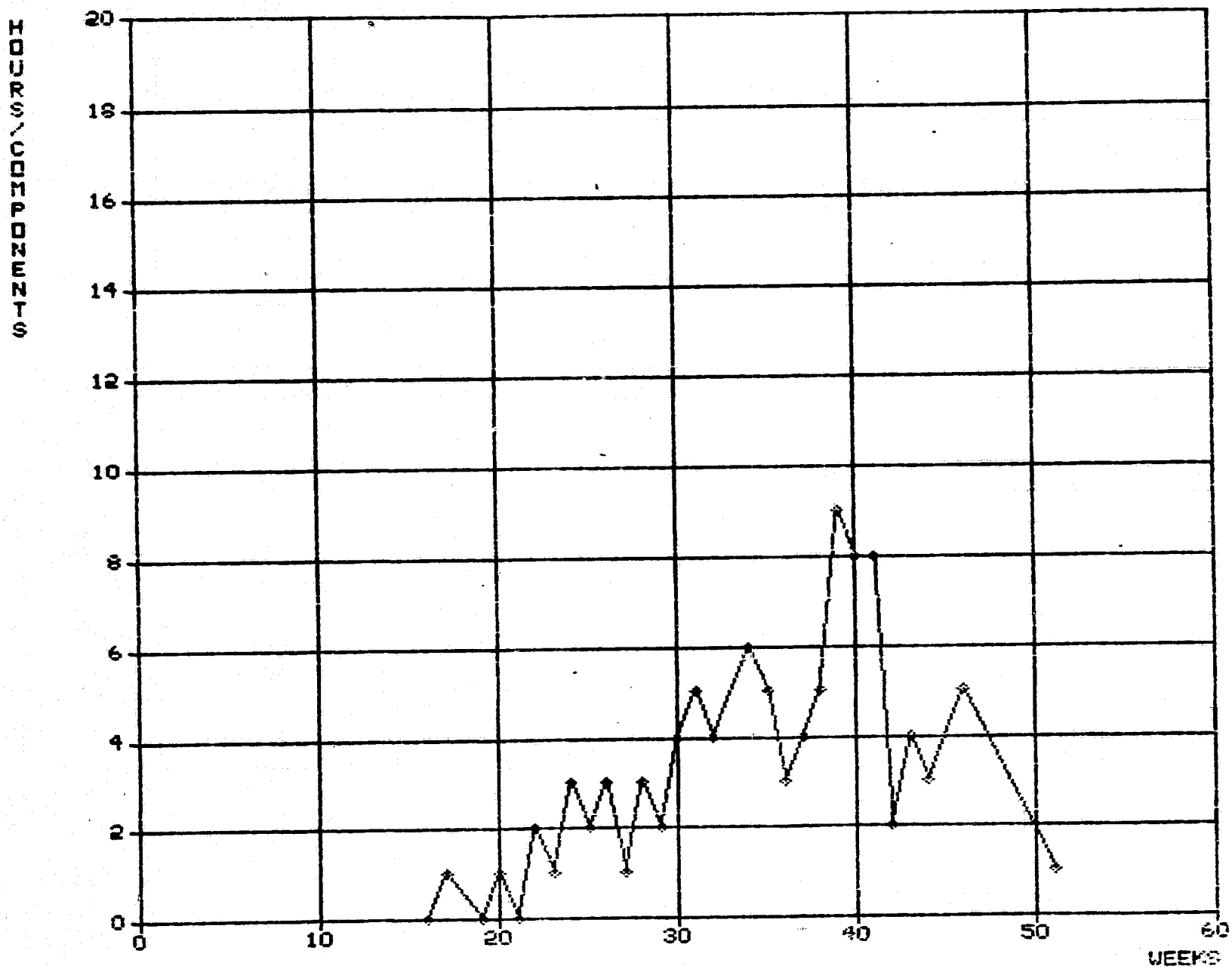
ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY

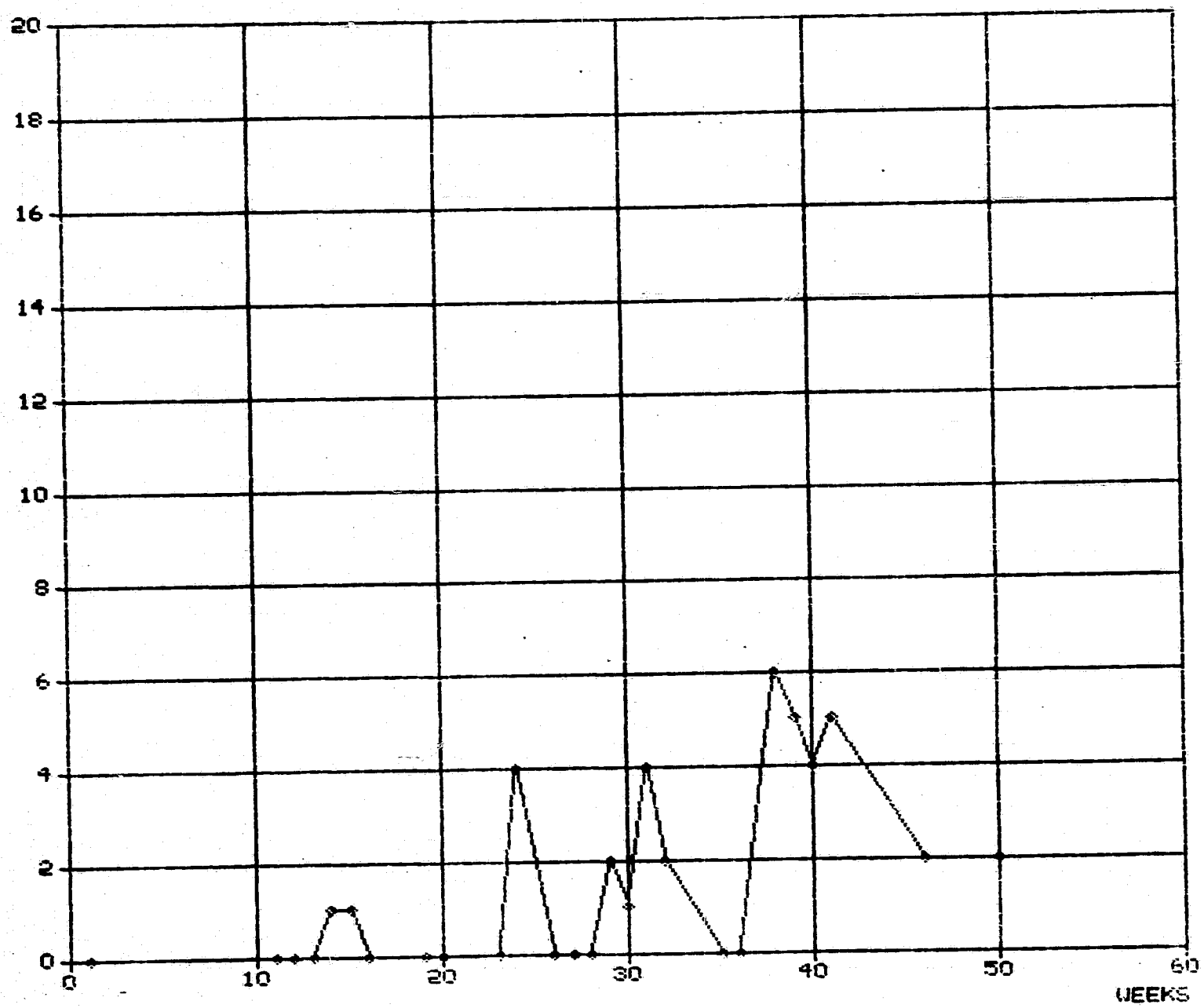


PROJECT 2 -- HOURS/COMPONENTS
TEST PHASE



ORIGINAL PAGE IS
OF POOR QUALITY

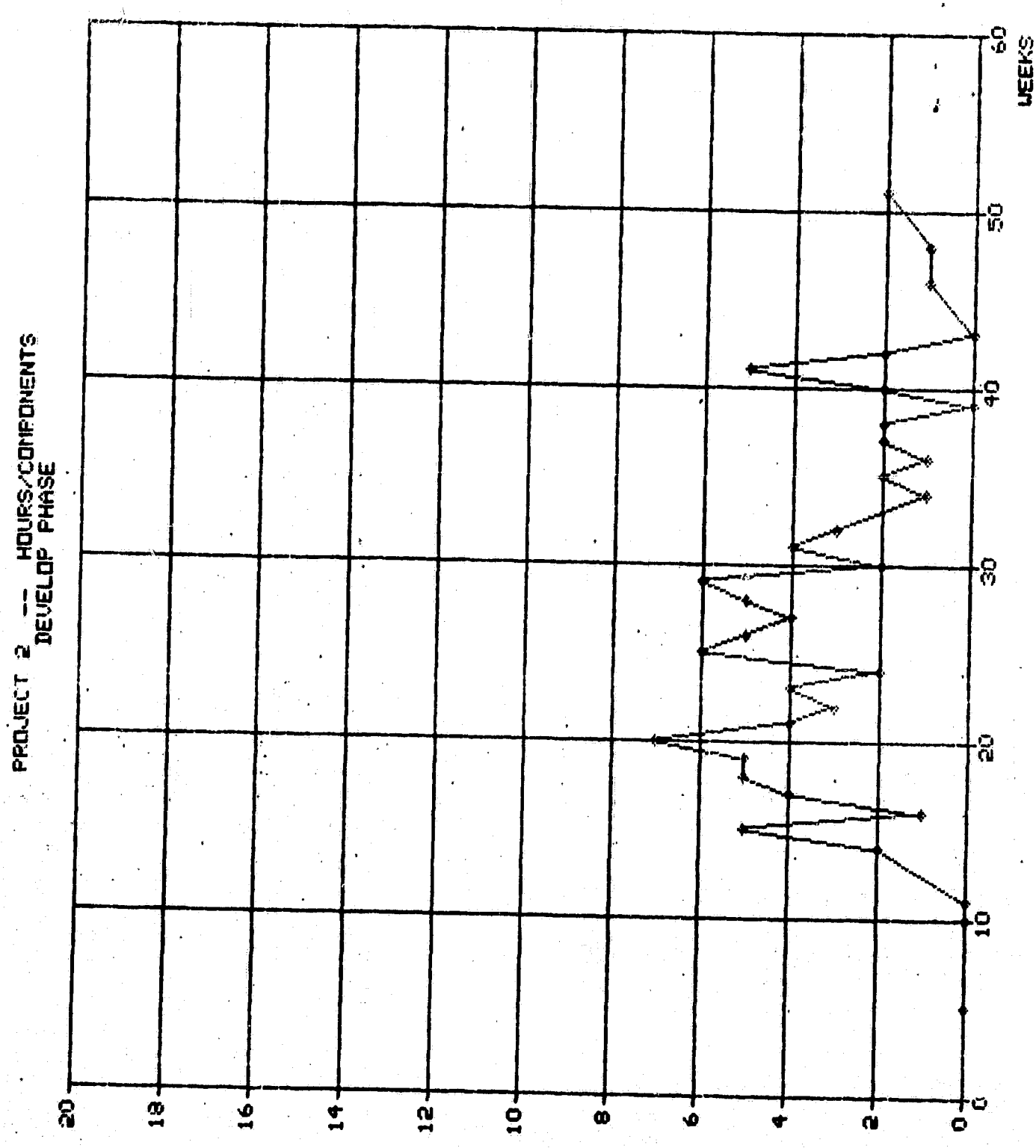
PROJECT 2 -- HOURS/COMPONENTS
OTHER CHARGES



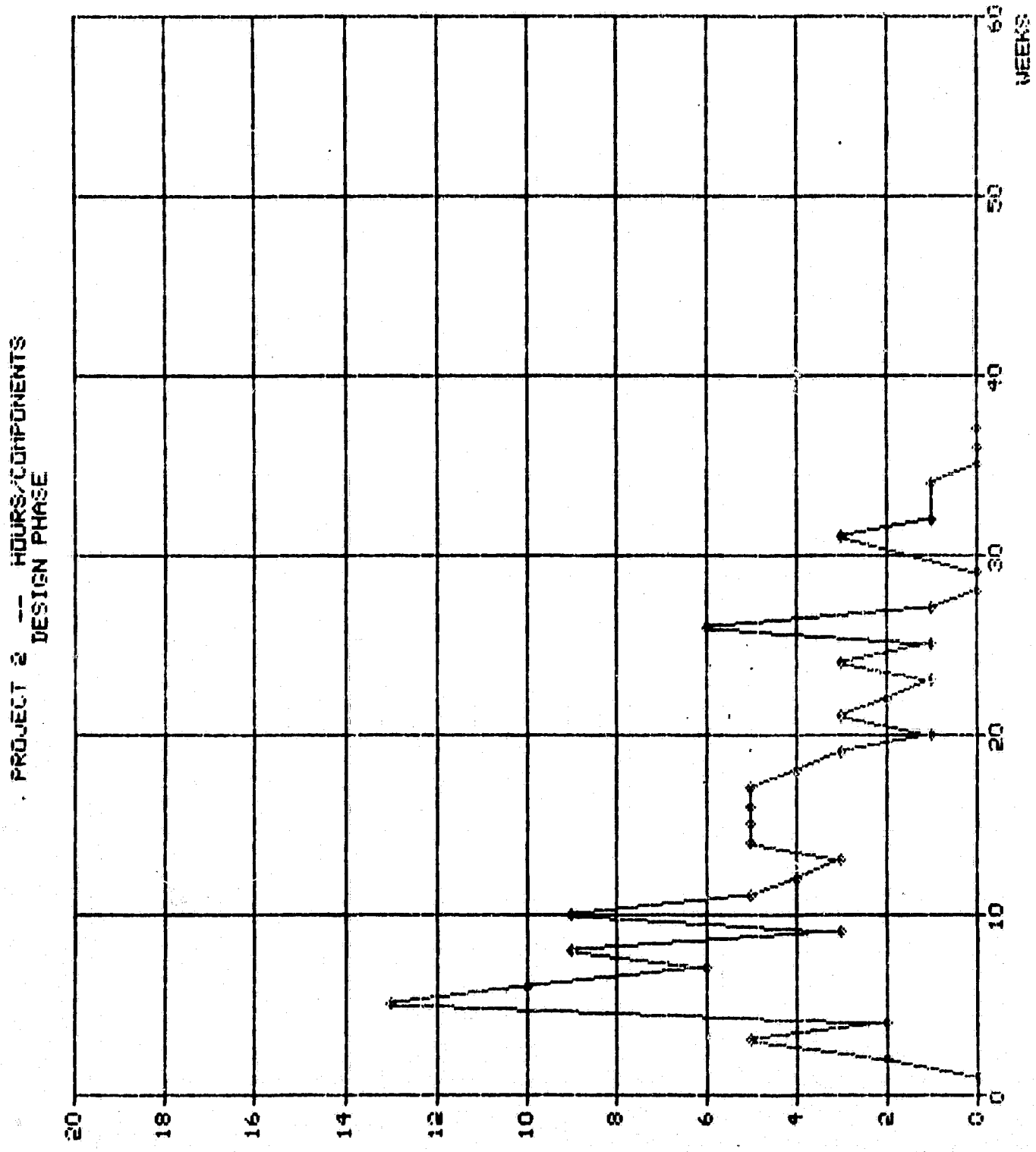
ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

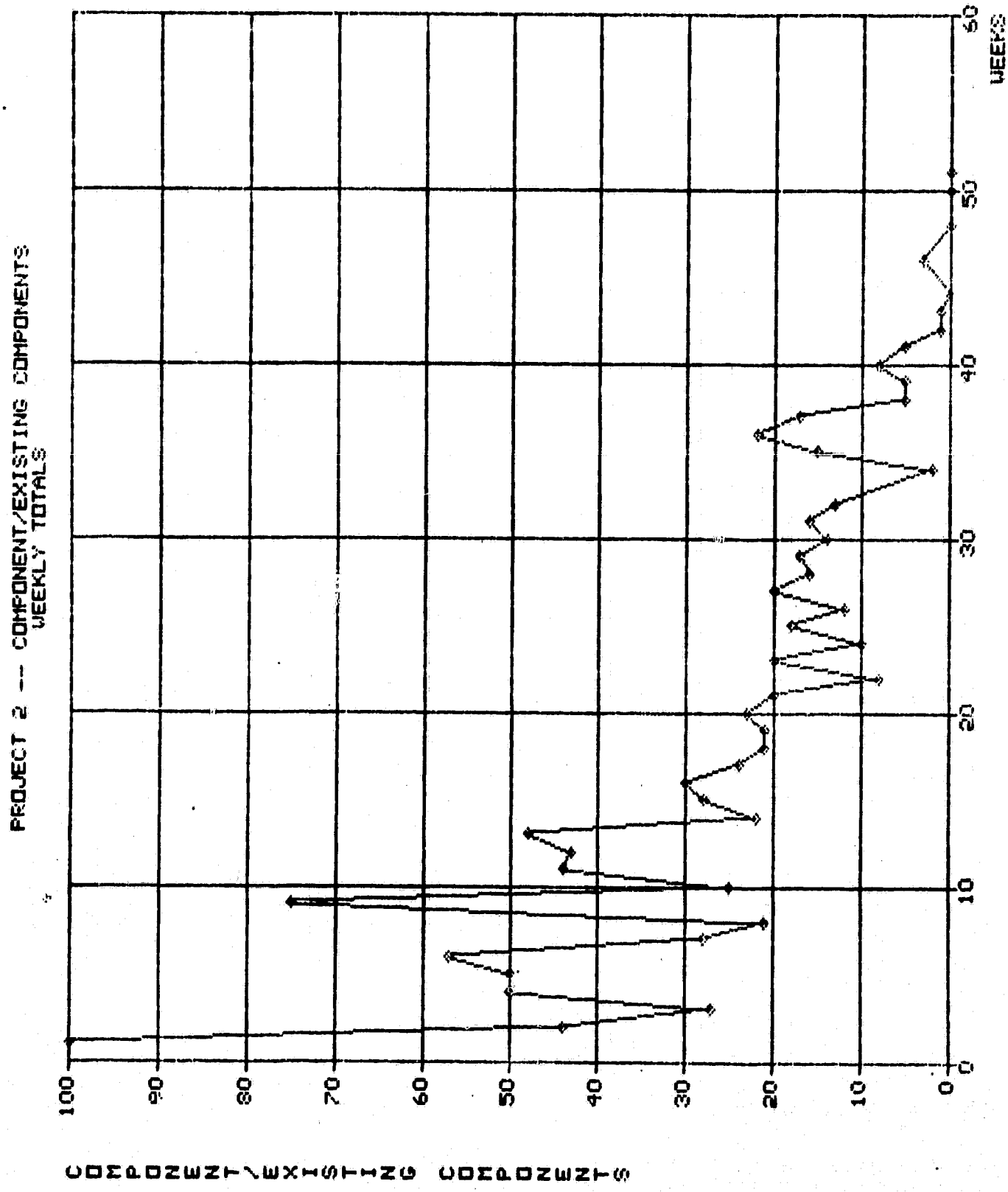
ORIGINAL PAGE IS
OF POOR QUALITY



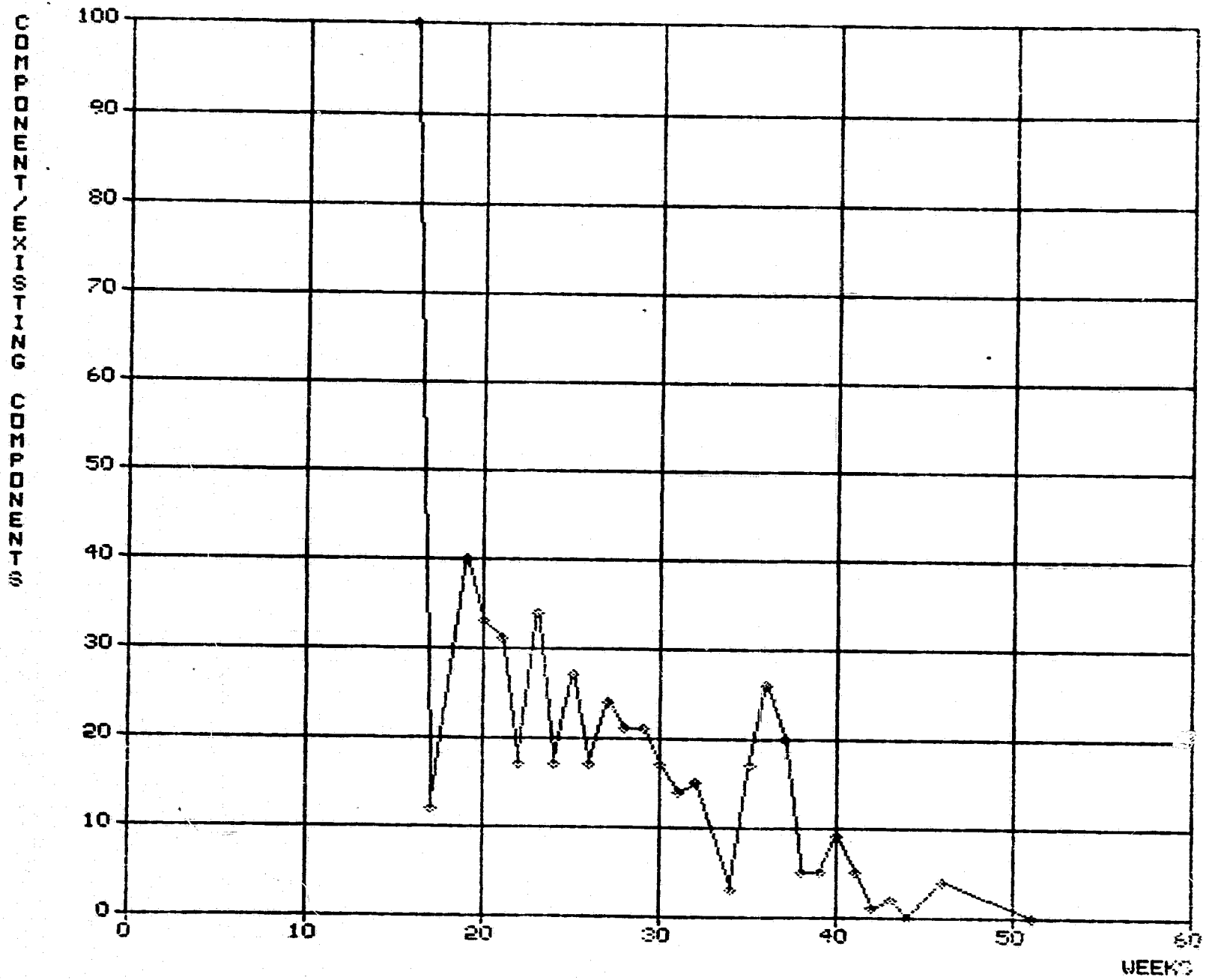
ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY

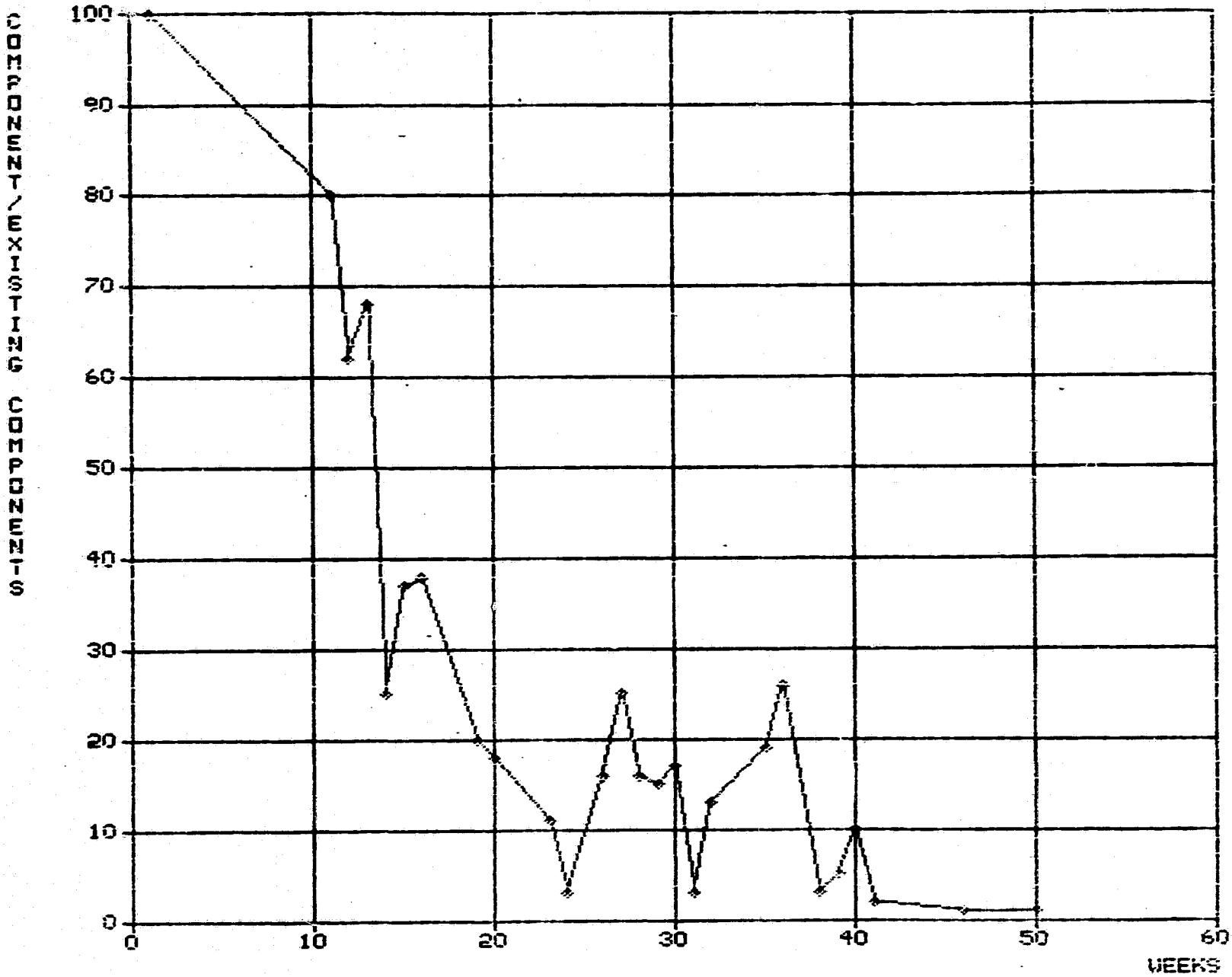


PROJECT 2 -- COMPONENT/EXISTING COMPONENTS
TEST PHASE



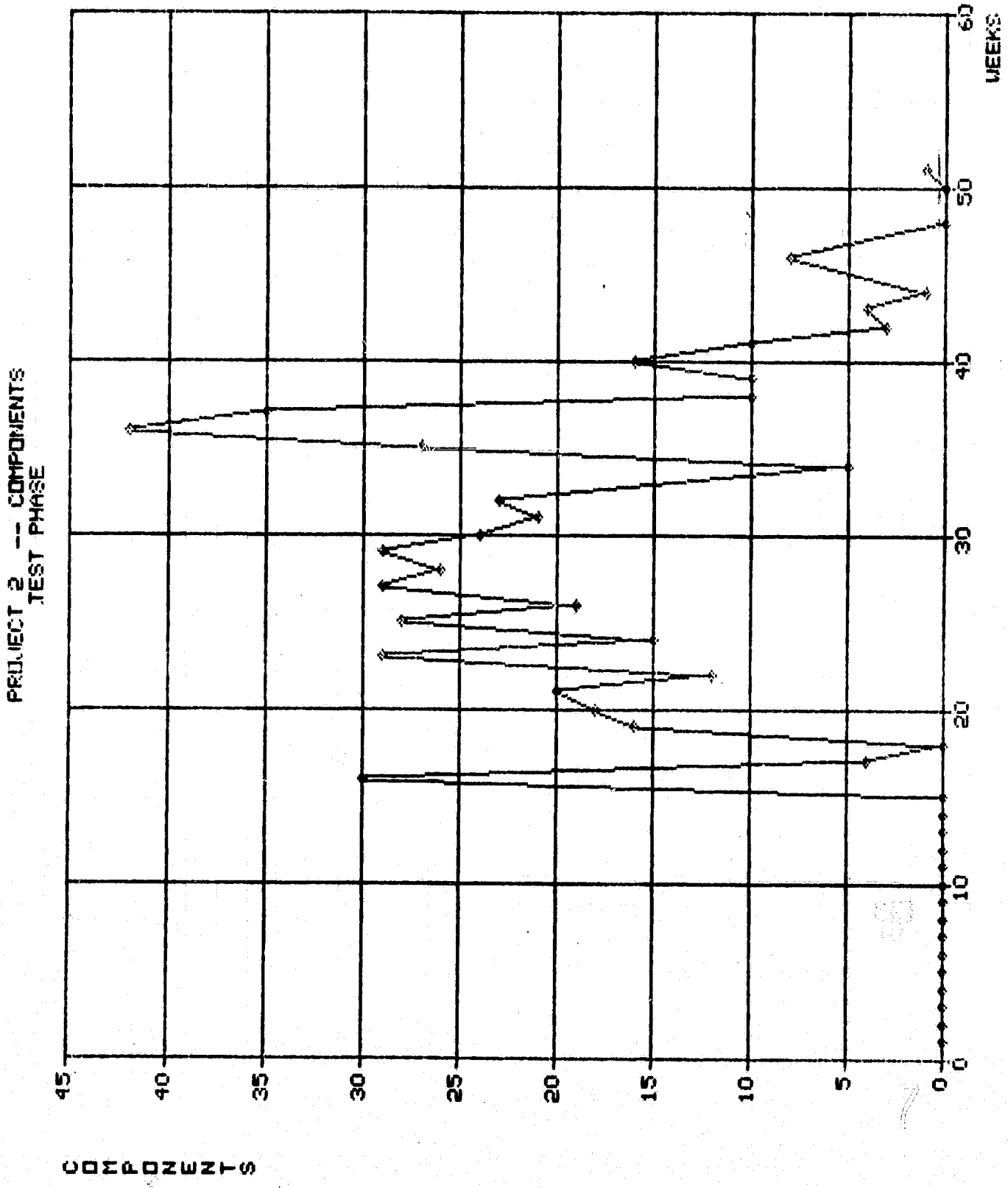
ORIGINAL PAGE IS
OF POOR QUALITY

PROJECT 2 -- COMPONENT/EXISTING COMPONENTS
OTHER CHARGES

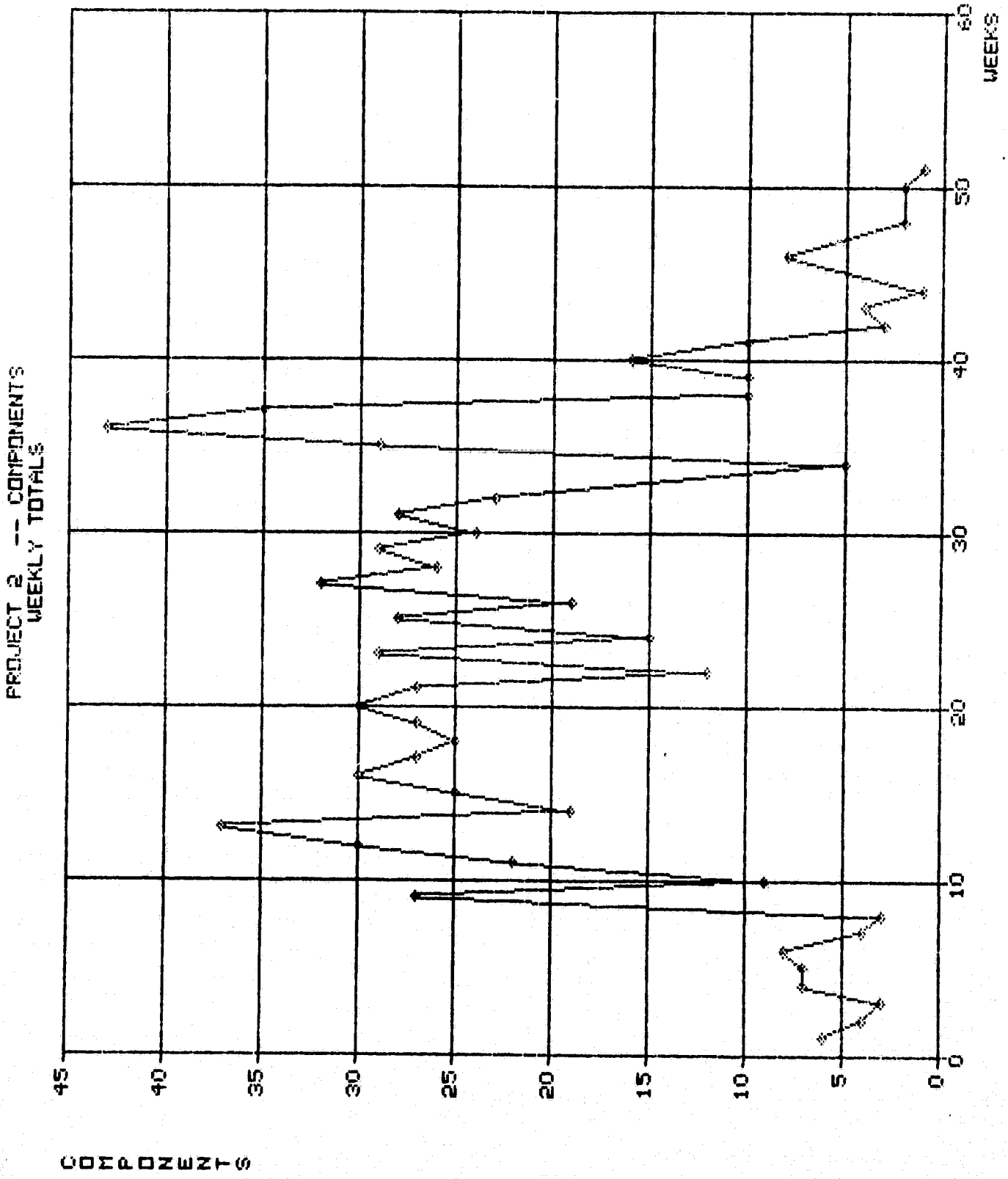


ORIGINAL PAGE IS
OF POOR QUALITY

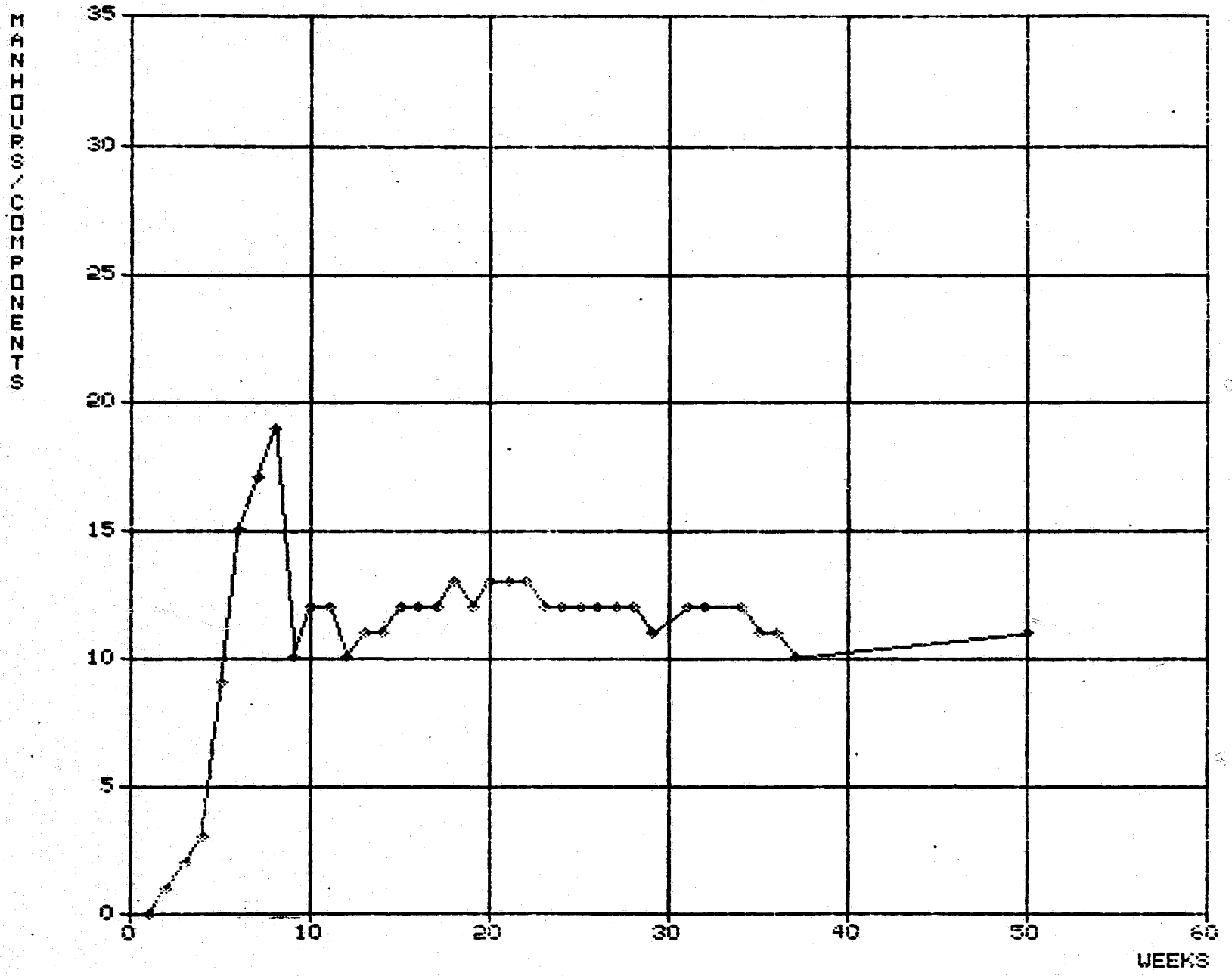
ORIGINAL PAGE IS
OF POOR QUALITY



ORIGINAL PAGE IS
OF POOR QUALITY

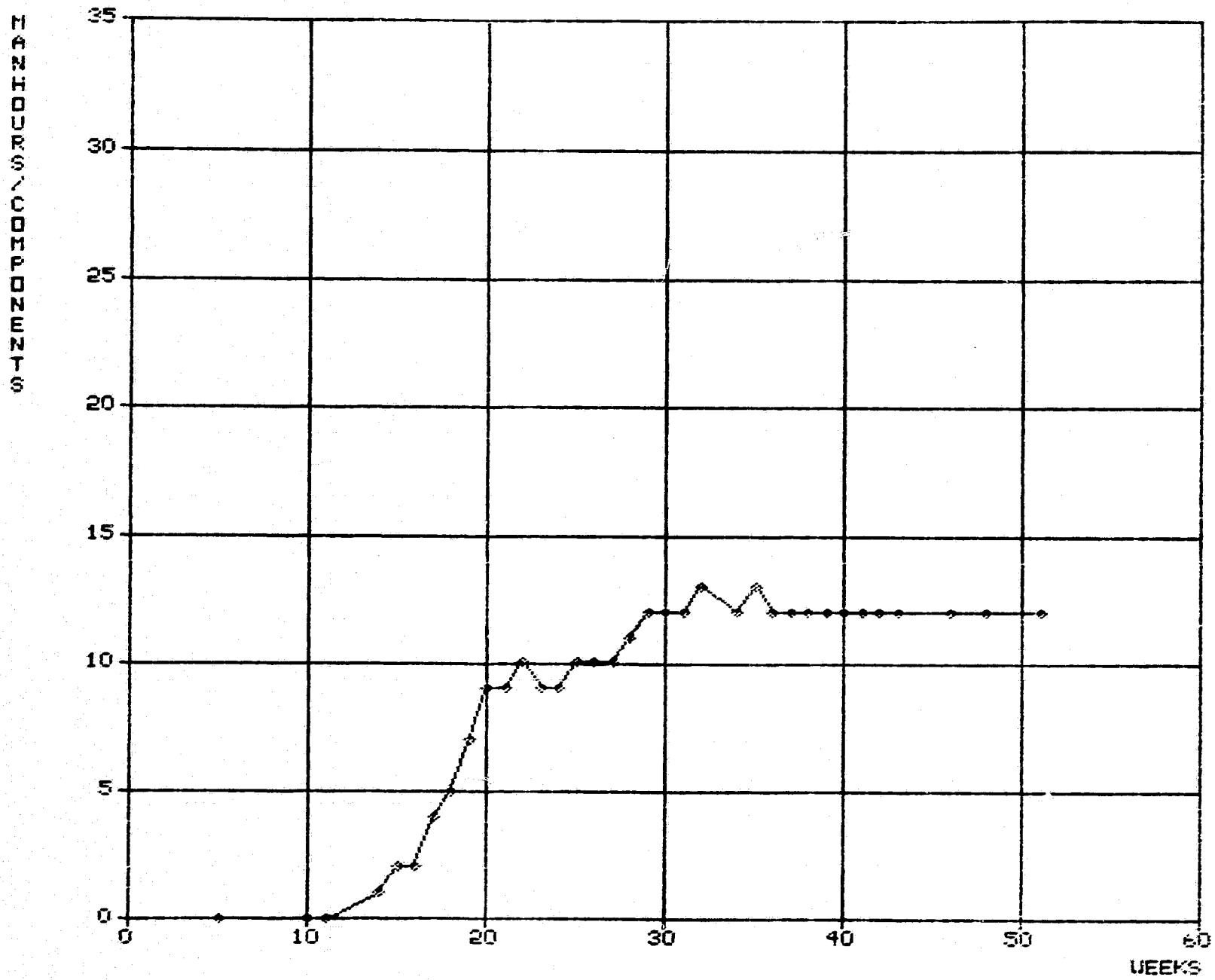


PROJECT 2 -- TOTAL HOURS/EXISTING COMPONENTS
DESIGN PHASE



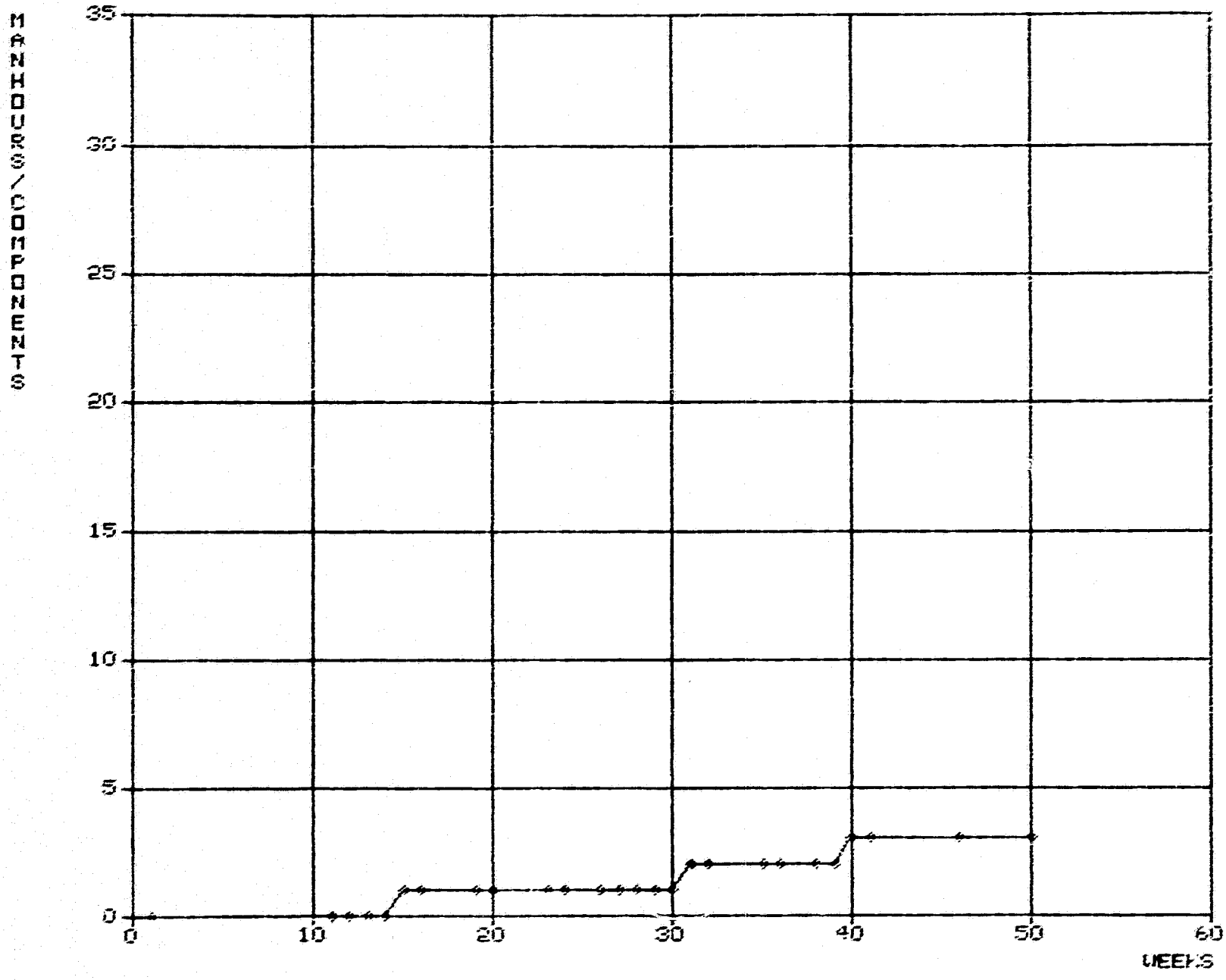
ORIGINAL PAGE IS
OF POOR QUALITY

PROJECT 2 -- TOTAL HOURS/EXISTING COMPONENTS
DEVELOP PHASE



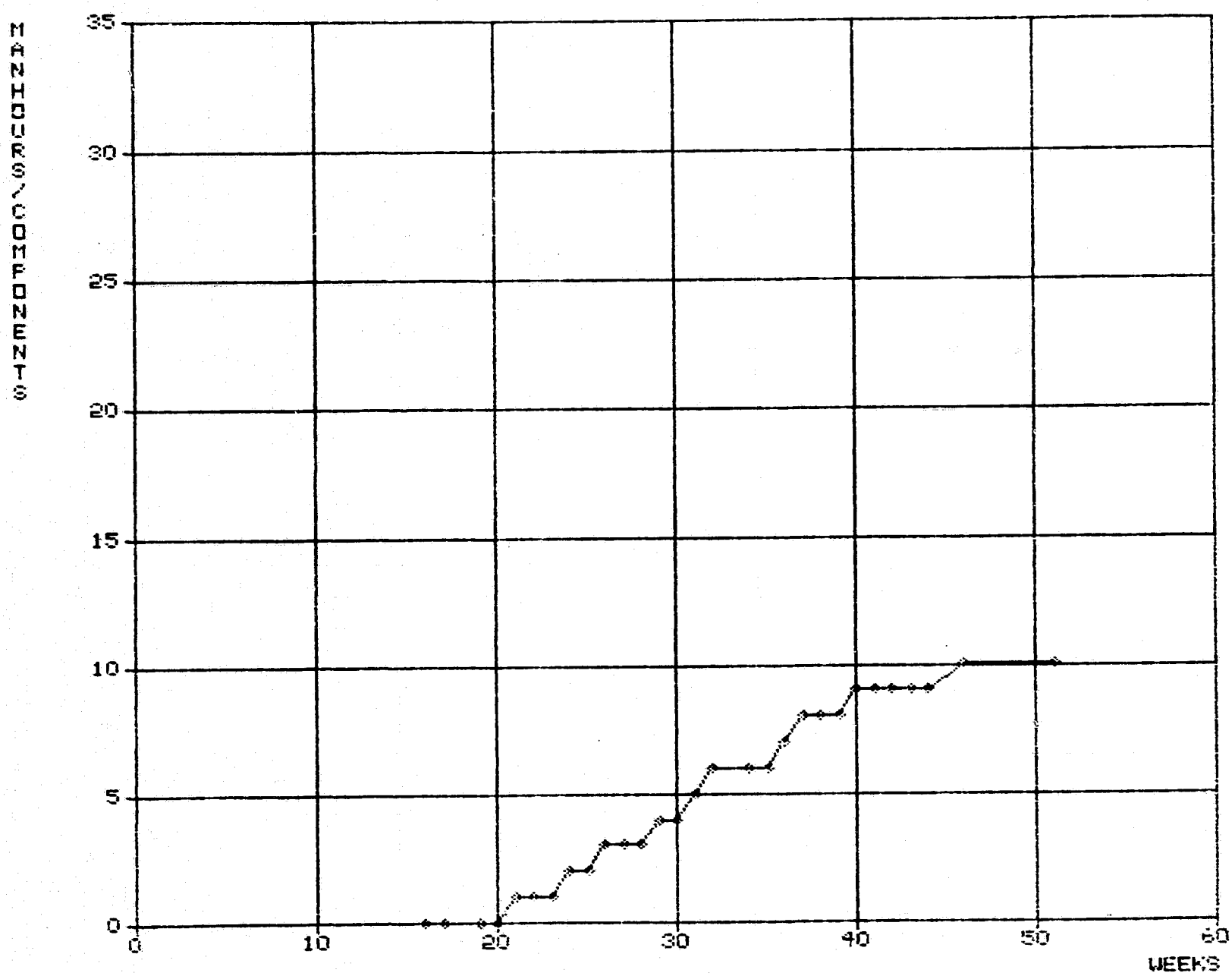
ORIGINAL PAGE IS
OF POOR QUALITY

PROJECT 2 -- TOTAL HOURS/EXISTING COMPONENTS
OTHER CHARGES



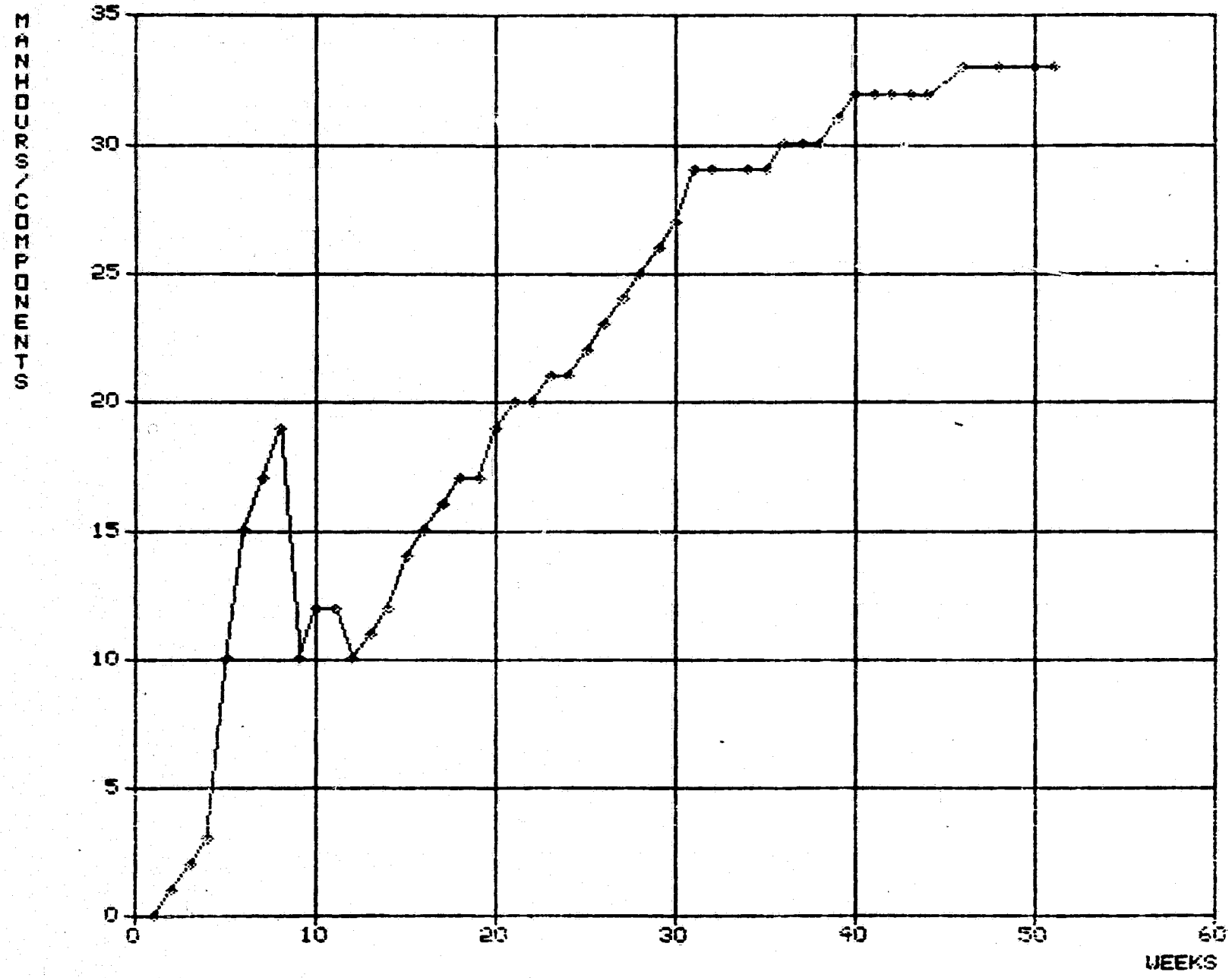
ORIGINAL PAGE IS
OF POOR QUALITY

PROJECT 2 -- TOTAL HOURS/EXISTING COMPONENTS
TEST PHASE



ORIGINAL PAGE IS
OF POOR QUALITY

PROJECT 2 -- TOTAL HOURS/EXISTING COMPONENTS
WEEKLY CHARGES



ORIGINAL PAGE IS
OF POOR QUALITY

BIBLIOGRAPHY OF SEL LITERATURE

The technical papers, memorandums, and documents listed in this bibliography are organized into two groups. The first group is composed of documents issued by the Software Engineering Laboratory (SEL) during its research and development activities. The second group includes materials that were published elsewhere but pertain to SEL activities.

SEL-Originated Documents

Software Engineering Laboratory, SEL-76-001, Proceedings From the First Summer Software Engineering Workshop, August 1976

SEL-77-001, The Software Engineering Laboratory, V. R. Basili, M. V. Zelkowitz, F. E. McGarry, et al., May 1977

SEL-77-002, Proceedings From the Second Summer Software Engineering Workshop, September 1977

SEL-77-003, Structured FORTRAN Preprocessor (SFORT), B. Chu, D. S. Wilson, and R. Beard, September 1977

SEL-77-004, GSFC NAVPAK Design Specifications Languages Study, P. A. Scheffer and C. E. Velez, October 1977

SEL-78-001, FORTRAN Static Source Code Analyzer (SAP) Design and Module Descriptions, E. M. O'Neill, S. R. Waligora, and C. E. Goorevich, January 1978

†SEL-78-002, FORTRAN Static Source Code Analyzer (SAP) User's Guide, E. M. O'Neill, S. R. Waligora, and C. E. Goorevich, February 1978

SEL-78-102, FORTRAN Static Source Code Analyzer Program (SAP) User's Guide (Revision 1), W. J. Decker and W. A. Taylor, May 1982 (preliminary)

SEL-78-003, Evaluation of Draper NAVPAK Software Design, K. Tasaki and F. E. McGarry, June 1978

†This document superseded by revised document.

SEL-78-004, Structured FORTRAN Preprocessor (SFORT) PDP-11/70 User's Guide, D. S. Wilson, B. Chu, and G. Page, September 1978

SEL-78-005, Proceedings From the Third Summer Software Engineering Workshop, September 1978

SEL-78-006, GSFC Software Engineering Research Requirements Analysis Study, P. A. Scheffer, November 1978

SEL-78-007, Applicability of the Rayleigh Curve to the SEL Environment, T. E. Mapp, December 1978

SEL-79-001, SIMPL-D Data Base Reference Manual, M. V. Zelkowitz, July 1979

SEL-79-002, The Software Engineering Laboratory: Relationship Equations, K. Freburger and V. R. Basili, May 1979

SEL-79-003, Common Software Module Repository (CSMR) System Description and User's Guide, C. E. Goorevich, S. R. Waligora, and A. L. Green, August 1979

SEL-79-004, Evaluation of the Caine, Farber, and Gordon Program Design Language (PDL) in the Goddard Space Flight Center (GSFC) Code 580 Software Design Environment, C. E. Goorevich, A. L. Green, and F. E. McGarry, September 1979

SEL-79-005, Proceedings From the Fourth Summer Software Engineering Workshop, November 1979

SEL-80-001, Functional Requirements/Specifications for Code 580 Configuration Analysis Tool (CAT), F. K. Banks, C. E. Goorevich, and A. L. Green, February 1980

SEL-80-002, Multi-Level Expression Design Language-Requirement Level (MEDL-R) System Evaluation, W. J. Decker, C. E. Goorevich, and A. L. Green, May 1980

SEL-80-003, Multimission Modular Spacecraft Ground Support Software System (MMS/GSSS) State-of-the-Art Computer Systems/Compatibility Study, T. Welden, M. McClellan, P. Liebertz, et al., May 1980

SEL-80-004, System Description and User's Guide for Code 580 Configuration Analysis Tool (CAT), F. K. Banks, W. J. Decker, J. G. Garrahan, et al., October 1980

SEL-80-005, A Study of the Musa Reliability Model, A. M. Miller, November 1980

SEL-80-006, Proceedings From the Fifth Annual Software Engineering Workshop, November 1980

SEL-80-007, An Appraisal of Selected Cost/Resource Estimation Models for Software Systems, J. F. Cook and F. E. McGarry, December 1980

SEL-81-001, Guide to Data Collection, V. E. Church, D. N. Card, F. E. McGarry, et al., September 1981

SEL-81-002, Software Engineering Laboratory (SEL) Data Base Organization and User's Guide, D. C. Wyckoff, G. Page, F. E. McGarry, et al., September 1981

SEL-81-003, Software Engineering Laboratory (SEL) Data Base Maintenance System (DBAM) User's Guide and System Description, D. N. Card, D. C. Wyckoff, G. Page, et al., September 1981

†SEL-81-004, The Software Engineering Laboratory, D. N. Card, F. E. McGarry, G. Page, et al., September 1981

SEL-81-104, The Software Engineering Laboratory, D. N. Card, F. E. McGarry, G. Page, et al., February 1982

†SEL-81-005, Standard Approach to Software Development, V. E. Church, F. E. McGarry, G. Page, et al., September 1981

SEL-81-105, Recommended Approach to Software Development, S. Eslinger, F. E. McGarry, V. E. Church, et al., May 1982

SEL-81-006, Software Engineering Laboratory (SEL) Document Library (DOCLIB) System Description and User's Guide, W. Taylor and W. J. Decker, December 1981

†SEL-81-007, Software Engineering Laboratory (SEL) Compendium of Tools, W. J. Decker, E. J. Smith, A. L. Green, et al., February 1981

SEL-81-107, Software Engineering Laboratory (SEL) Compendium of Tools, W. J. Decker, E. J. Smith, W. A. Taylor, et al., February 1982

SEL-81-008, Cost and Reliability Estimation Models (CAREM) User's Guide, J. F. Cook and E. Edwards, February 1981

---

†This document superseded by revised document.

SEL-81-009, Software Engineering Laboratory Programmer Workbench Phase 1 Evaluation, W. J. Decker, A. L. Green, and F. E. McGarry, March 1981

SEL-81-010, Performance and Evaluation of an Independent Software Verification and Integration Process, G. Page and F. E. McGarry, May 1981

SEL-81-011, Evaluating Software Development by Analysis of Change Data, D. M. Weiss, November 1981

SEL-81-012, The Rayleigh Curve As a Model for Effort Distribution Over the Life of Medium Scale Software Systems, G. O. Picasso, December 1981 (also published as University of Maryland Technical Report TR-1186, July 1982)

SEL-81-013, Proceedings From the Sixth Annual Software Engineering Workshop, December 1981

SEL-81-014, Automated Collection of Software Engineering Data in the Software Engineering Laboratory (SEL), A. L. Green, W. J. Decker, and F. E. McGarry, September 1981

SEL-82-001, Evaluation and Application of Software Development Measures, D. N. Card, G. Page, and F. E. McGarry, July 1982

SEL-82-002, FORTRAN Static Source Code Analyzer Program (SAP) System Description, W. Taylor and W. Decker, August 1982

SEL-82-003, Software Engineering Laboratory (SEL) Data Base Reporting Software User's Guide and System Description, P. Lo and S. Eslinger, September 1982

SEL-82-004, Collected Software Engineering Papers: Volume 1, July 1982

SEL-Related Literature

Anderson, L., "SEL Library Software User's Guide," Computer Sciences-Technicolor Associates, Technical Memorandum, June 1980

††Bailey, J. W., and V. R. Basili, "A Meta-Model for Software Development Resource Expenditures," Proceedings of the Fifth International Conference on Software Engineering. New York: Computer Societies Press, 1981

Banks, F. K., "Configuration Analysis Tool (CAT) Design," Computer Sciences Corporation, Technical Memorandum, March 1980

††Basili, V. R., "The Software Engineering Laboratory: Objectives," Proceedings of the Fifteenth Annual Conference on Computer Personnel Research, August 1977

††Basili, V. R., "Models and Metrics for Software Management and Engineering," ASME Advances in Computer Technology, January 1980, vol. 1

Basili, V. R., "SEL Relationships for Programming Measurement and Estimation," University of Maryland, Technical Memorandum, October 1980

Basili, V. R., Tutorial on Models and Metrics for Software Management and Engineering. New York: Computer Societies Press, 1980 (also designated SEL-80-008)

††Basili, V. R., and J. Beane, "Can the Parr Curve Help with Manpower Distribution and Resource Estimation Problems?", Journal of Systems and Software, February 1981, vol. 2, no. 1

††Basili, V. R., and K. Freburger, "Programming Measurement and Estimation in the Software Engineering Laboratory," Journal of Systems and Software, February 1981, vol. 2, no. 1

††Basili, V. R., and T. Phillips, "Evaluating and Comparing Software Metrics in the Software Engineering Laboratory," Proceedings of the ACM SIGMETRICS Symposium/Workshop: Quality Metrics, March 1981

Basili, V. R., and T. Phillips, "Validating Metrics on Project Data," University of Maryland, Technical Memorandum, December 1981

Basili, V. R., and R. Reiter, "Evaluating Automatable Measures for Software Development," Proceedings of the Workshop on Quantitative Software Models for Reliability, Complexity and Cost, October 1979

---

††This article also appears in SEL-82-004, Collected Software Engineering Papers: Volume 1, July 1982

Basili, V. R., and M. V. Zelkowitz, "Designing a Software Measurement Experiment," Proceedings of the Software Life Cycle Management Workshop, September 1977

††Basili, V. R., and M. V. Zelkowitz, "Operation of the Software Engineering Laboratory," Proceedings of the Second Software Life Cycle Management Workshop, August 1978

††Basili, V. R., and M. V. Zelkowitz, "Measuring Software Development Characteristics in the Local Environment," Computers and Structures, August 1978, vol. 10

Basili, V. R., and M. V. Zelkowitz, "Analyzing Medium Scale Software Development," Proceedings of the Third International Conference on Software Engineering. New York: Computer Societies Press, 1978

Card, D. N., "Early Estimation of Resource Expenditures and Program Size," Computer Sciences Corporation, Technical Memorandum, June 1982

††Chen, E., and M. V. Zelkowitz, "Use of Cluster Analysis to Evaluate Software Engineering Methodologies," Proceedings of the Fifth International Conference on Software Engineering. New York: Computer Societies Press, 1981

Church, V. E., "User's Guides for SEL PDP-11/70 Programs," Computer Sciences Corporation, Technical Memorandum, March 1980

Freburger, K., "A Model of the Software Life Cycle" (paper prepared for the University of Maryland, December 1978)

Higher Order Software, Inc., TR-9, A Demonstration of AXES for NAVPAK, M. Hamilton and S. Zeldin, September 1977 (also designated SEL-77-005)

Hislop, G., "Some Tests of Halstead Measures" (paper prepared for the University of Maryland, December 1978)

---

††This article also appears in SEL-82-004, Collected Software Engineering Papers: Volume 1, July 1982

Lange, S. F., "A Child's Garden of Complexity Measures" (paper prepared for the University of Maryland, December 1978)

Miller, A. M., "A Survey of Several Reliability Models" (paper prepared for the University of Maryland, December 1978)

National Aeronautics and Space Administration (NASA), NASA Software Research Technology Workshop (proceedings), March 1980

Page, G., "Software Engineering Course Evaluation," Computer Sciences Corporation, Technical Memorandum, December 1977

Parr, F., and D. Weiss, "Concepts Used in the Change Report Form," NASA, Goddard Space Flight Center, Technical Memorandum, May 1978

Perricone, B. T., "Relationships Between Computer Software and Associated Errors: Empirical Investigation" (paper prepared for the University of Maryland, December 1981)

Reiter, R. W., "The Nature, Organization, Measurement, and Management of Software Complexity" (paper prepared for the University of Maryland, December 1976)

Scheffer, P. A., and C. E. Velez, "GSFC NAVPAK Design Higher Order Languages Study: Addendum," Martin Marietta Corporation, Technical Memorandum, September 1977

Turner, C., G. Caron, and G. Brement, "NASA/SEL Data Compendium," Data and Analysis Center for Software, Special Publication, April 1981

Turner, C., and G. Caron, "A Comparison of RADC and NASA/SEL Software Development Data," Data and Analysis Center for Software, Special Publication, May 1981

Weiss, D. M., "Error and Change Analysis," Naval Research Laboratory, Technical Memorandum, December 1977

Williamson, I. M., "Resource Model Testing and Information," Naval Research Laboratory, Technical Memorandum, July 1979

††Zelkowitz, M. V., "Resource Estimation for Medium Scale Software Projects," Proceedings of the Twelfth Conference on the Interface of Statistics and Computer Science. New York: Computer Societies Press, 1979

††This article also appears in SEL-82-004, Collected Software Engineering Papers: Volume 1, July 1982

Selkowitz, M. V., and V. R. Basili, "Operational Aspects of a Software Measurement Facility," Proceedings of the Software Life Cycle Management Workshop, September 1977