ORDER ADAPTIVE FILE ORGANIZATIONS TO BE USED IN CONNECTION WITH SERIAL SCAN

by

THOMAS HOWARD ELROD

B. S., Kansas State University
of Agriculture and Applied Science, 1955

A THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Electrical Engineering

KANSAS STATE UNIVERSITY
OF AGRICULTURE AND APPLIED SCIENCE

1960

TABLE OF CONTENTS

INTRODUCTION 1

IDEAL ORDERING 3

SYSTEMS 4

FORMULATION OF THE FEED SYSTEM 5

Single Feed System 10

Multiple Feed System 11

TRANSPOSITION SYSTEM 13

GENERAL FEED SYSTEM PROGRAM 14

Single Feed System Program 15

Multiple Feed System Program 18

TRANSPOSITION SYSTEM PROGRAM 20

DISCUSSION OF RESULTS 21

Feed System Results 22

Transposition System Results 23

MECHANIZATION OF THE TRANSPOSITION SYSTEM 24

CONCLUSION 26

ACKNOWLEDGMENT 27

REFERENCES 28

APPENDIX 29

INTRODUCTION

In large files having in the order of 10^6 items, the central problem is one of being able to retrieve any item from the file in the minimum amount of time. When the file is a large digital file, several methods of searching for and retrieving an item are available. The investigation carried out in this thesis was concerned with only one of the possible searching patterns and methods of causing the retrieval time to approach a minimum. The ideal organization was also found and is shown as the lower bound to the other solutions.

The retrieval of items from large digital files is presently accomplished by two general procedures. In the first type of search operation a correspondence is made between each item and its location in the file. To retrieve an item the location is set into the address mechanism and the item is retrieved in one try. This type of file has a very short retrieval time but the obvious disadvantage is the requirement that the human operator must have available the location of each item in the file. This implies large tables and a filing system of locations. The second type of search operation is accomplished by giving each item a unique tag and when a request for a particular item is received, a search is started for the item by examining in succession all the item tags until the proper one is found. This type of organization has the advantage that there is no requirement for knowledge of the location of each item in the file or for the correspondence between tag and location. The item tags can be generated in a quite spontaneous fashion so that there is no need to keep a record of them. For instance, a file of checking accounts at a bank could use as a tag for each account four symbols corresponding to the two initials and the first two letters of the last name of each account owner. There is a small probability that two accounts will have the same tag, in which case one can be given a slight modification. The disadvantage of this type of file organization is the marked increase in the time required to retrieve an item from the file. The average time required will be the time necessary to examine half of the items in the file.

If the search process of the serial scan system is examined in light of the information that it takes to locate the item of interest in the file, it is apparent that the same information is used for all the items. This is equivalent to choosing to neglect all the statistical information contained in the probabilities related to the usage of the items in the file. Thus the serial scan system might be called, with some justification, a blind serial search. If the statistical information contained in the probabilities of the items is made use of to influence the ordering pattern of the file, the file becomes capable of adaptation. This implies that the only human intervention required is to enter new items in the file, remove items from the file, and to present the requests to the file. The systems studied in this investigation produce an ordering of the file that tends to approximate the ordering of the probability weights induced on the elements of the file by the external users.

IDEAL ORDERING

Define as ideal ordering the ordering that makes the distribution $\{p_i\}$ a monotone non-increasing function. The criterion for judging this to be the ideal is that with this ordering the average length of the retrieval experiment is a minimum.

The average length of the retrieval experiment is given by

$$\gamma = \sum_{i=1}^{N} x_i p_i$$

This can also be expressed

$$\gamma = \sum_{\substack{i=1 \\ i \neq j,\ell}}^{N} x_i p_i + x_j p_j + x_\ell p_\ell$$

where $$j < \ell$$

Now if the j:th and the ℓ:th elements are interchanged,

$$\gamma' = \sum_{\substack{i=1 \\ \neq j,\ell}}^{N} x_i p_i + x_\ell p_j + x_j p_\ell$$

The difference $\Delta\gamma$ is formed

$$\Delta\gamma = \gamma' - \gamma$$

$$= \sum_{\substack{i=1 \\ \neq j,\ell}}^{N} x_i p_i + x_\ell p_j + x_j p_\ell - \sum_{\substack{i=1 \\ \neq j,\ell}}^{N} x_i p_i + x_j p_j + x_\ell p_\ell$$

$$= (x_\ell p_j + x_j p_\ell) - (x_j p_j + x_\ell p_\ell)$$

$$= x_\ell (p_j - p_\ell) + x_j (p_\ell - p_j)$$

Now $$p_j > p_\ell$$

and $$x_\ell > x_j$$

therefore $$\Delta\gamma > 0$$

and $$\gamma' > \gamma$$

Thus the shortest retrieval time is obtained when the file is ordered in such a way that the distribution $\{p_i\}$ is monotone non-increasing. This then establishes the criterion for the order adaptive file organization. In order to reduce the average length of the retrieval experiment, the file should spontaneously approach the ideal ordering.

SYSTEMS

The system or mechanism used in the file organization takes on three configurations. The configurations are all conceptually of the same general type, but only two can be analyzed in a similar fashion. The three systems will be called the single feed system, the multiple feed system, and the transposition system.

The single feed system employs a mechanism that takes the item read from the file and reenters this item at the origin of the file, shifting all the items between the position of the one read and the origin one position away from the origin. That this system would require complex logic to mechanize for random access drum or tape storage is evident, but it consents to easy mechanization in a card file.

The multiple feed system employs a mechanization that takes the item read from the file and reenters this item at one of several feed points in the file determined by the position from which it is removed. This system also suffers from the same disadvantages of mechanization as the single feed system. and in addition it requires more elaborate logic to determine the feed points.

The transposition system employs a mechanism that takes the item read from the file and transposes it with the item in the position α cells toward the origin from the position of the item read. This system is more easily mechanized than either of the feed systems, requiring only two intermediate storage locations and the logic to select the position α cells toward the origin.

The speed of adaptation of any of these systems is obviously dependent upon the average distance traveled by an item when it is read. This suggests immediately that the single feed system should be the fastest and that the transposition system will have an adaptation speed that is dependent on α.

FORMULATION OF THE FEED SYSTEM

The formulation of the feed system results in a set of equations that can be specialized to describe either a multiple feed or single feed system.

The complete formulation of a single feed system is exhibited to show the process to be carried out in arriving at the formulations for both systems.

The average length of the retrieval experiment depends on the average velocity of a particular item moving through the file. If the average speed can be computed, the mean free path of an item through the file, and hence γ, can be computed.

That is,

$$\gamma = \sum_{i=1}^{N} \bar{\ell}_i \, p_i$$

where $\bar{\ell}_i$ is the mean free path of the i:th item in the file, and

$$\bar{\ell}_i = \xi \, \bar{v}_i$$

where ξ is the number of interrogations per unit time and $\bar{v}_i$ is the average velocity of the i:th item through the file. The average velocity of the i:th item through the file is dependent on the probability of reading an item in the i + 1:th position. Thus

$$\bar{v}_i = \xi \, P\left\{\text{reading an item below i}\right\}$$

and

$$P\left\{\text{reading an item below i}\right\} = \sum_{i+1}^{N} p_i$$

It is apparent that

$$P\left\{\text{reading an item below i}\right\}$$

is dependent upon the instantaneous position occupied by the i:th item in the file, and hence cannot be determined until ℓ_i is known. The exact formulation for γ thus leads to a relation solvable only by trial and error or some similar type of analysis. If an approximation can be found that will represent the system to some reasonable degree of accuracy, it would be more acceptable for solution.

To obtain an approximation. the file is viewed as a probability reservoir having internal boundaries to define subfiles and external flow channels from each subfile to the top of the file. The change in file statistics can then be represented by flow equations describing the change of probability in each subfile in terms of external flows and internal flows between neighboring subfiles.

Assume there are a finite number of levels of probability in the file and designate them by

$$p_j \qquad j = 1, 2, 3, \ldots m$$

The number of subfiles will be designated by N; then define the coefficients α_{ij} to represent the number of items in the i:th subfile having probability p_j. The external differential flow of items with probability p_j into the k:th subfile will be, for k = 1,

$$d\alpha_{kj} = \left(\sum_{\ell=2}^{N} \alpha_{\ell j}\, p_j\right) \xi\, dt$$

For k = 2, 3, ..., N

$$d\alpha_{kj} = (-\alpha_{kj}\, p_j)\, \xi\, dt$$

The internal differential flow of items with probability p_j into the k:th subfile will be, for k = 1

$$d\alpha_{kj} = 0$$

because there is no neighboring subfile above from which the flow can emanate.

For k = 2, 3, ... N

$$d\alpha_{kj} = Q_{k-1,j}\left(\sum_{i=k}^{N} \pi_i\right) \xi\, dt$$

where Q_{kj} is the probability coefficient defined by

$$Q_{k-1,j} = \frac{\eta_{k-1,j}}{\sum_{\ell=1}^{m} \eta_{k-1,\ell}}$$

and $\eta_{k=1,j}$ is defined by

$$\eta_{k-1,j} = \frac{(1 - p_j)^{\sum_{i=1}^{k-1} n_i}}{\eta_{k-1}} \alpha_{k-1,j} \quad \text{for } \alpha_{k-1,j} \geq 0$$

$$= 0 \qquad\qquad \text{for } \alpha_{k-1,j} < 0$$

Thus $\eta_{k-1,j}$ is the ratio of the number of items of probability p_j in the k-1:th subfile to the number of items in the subfile multiplied by the probability that an item with probability p_j is not read over $\sum_{i=1}^{k-1} n_i$ steps. The exponent $\sum_{i=1}^{k-1} n_i$ represents the number of interrogations required for an item to travel from one boundary of the subfile to the other. That more interrogations than this will be required is evident and some better approximation to the correct value might be an average defined to compensate for the changing speed of an item in the subfile. Neglect of this refinement will cause the incremental flows to be larger, and therefore the file will have a weaker tendency toward ordering. The term

$$\left(\sum_{i=k}^{N} \pi_i \right)$$

represents the probability of reading an item from the subfiles below the k-1:th. π_k is the total probability of the k:th subfile defined by

$$\pi_k = \sum_{j=1}^{m} \alpha_{kj}\, p_j, \quad k = 1, 2, 3, \ldots, N$$

The internal differential flow of items with probability p_j from the k:th into the k+1:th subfile will be:

for $\quad k = 1, 2, 3, \ldots, N-1$

$$d\alpha_{kj} = Q_{kj}\left(\sum_{i=k+1}^{N} \pi_i\right)\xi\, dt$$

and for $k = N$

$$d\alpha_{kj} = 0$$

because there is no neighboring file below. These differential flows are defined in the same fashion as those going from the k-1:th to the k:th subfile and the Q_{kj}'s are determined by replacing k-1 by k in the definitions for the $Q_{k-1,j}$'s given above.

Collecting all the differential flows, the total differential flow into the k:th subfile will be, for $k = 1$

$$d\alpha_{kj} = \left\{\left(\sum_{\ell=2}^{N} \alpha_{\ell j} p_j\right) - Q_{kj}\left(\sum_{i=k+1}^{N} \pi_i\right)\right\}\xi\, dt$$

for $k = 2, 3, \ldots, N - 1$

$$d\alpha_{kj} = \left\{(-\alpha_{kj} p_j) + Q_{k-1,j}\left(\sum_{i=k}^{N} \pi_i\right) - Q_{k,j}\left(\sum_{i=k+1}^{N} \pi_i\right)\right\}\xi\, dt$$

and for $k = N$

$$d\alpha_{kj} = \left\{(-\alpha_{kj} p_j) + Q_{k-1,j}\left(\sum_{i=k}^{N} \pi_i\right)\right\}\xi\, dt$$

These equations can be approximated by a set of difference equations. For $k = 1$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\left\{\left(\sum_{\ell=2}^{N} \alpha_{\ell j} p_j\right) - Q_{kj}\left(\sum_{i=k+1}^{N} \pi_i\right)\right\}$$

for $k = 2, 3, 4, \ldots, N - 1$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\left\{(-\alpha_{kj}p_j) + Q_{k-1,j}\left(\sum_{i=k}^{N} \pi_i\right) - Q_{k,j}\left(\sum_{i=k+1}^{N} \pi_i\right)\right\}$$

and for k = N

$$\alpha'_{kj} = \alpha_{kj} + \lambda\left\{(-\alpha_{kj}p_i) + Q_{k-1,j}\left(\sum_{i=k}^{N} \pi_i\right)\right\}$$

where α'_{kj} is the value of α_{kj} after Δt time has elapsed and

$$\lambda = \xi \, \Delta t$$

λ then has the dimensions of interrogations and this set of equations will represent the change in file statistics after λ interrogations each time they are applied in an iterative solution. The average length of the retrieval experiment can then be found approximately, assuming the subfiles to have equiprobable items in each, by the relation

$$\nu = \sum_{j=1}^{N}\left(\sum_{i=1}^{j-1} n_i + \frac{n_i}{2}\right)\pi_j$$

where the π_j are as defined above. This approximation becomes better as the ideal ordering is approached.

Single Feed System

The single feed system chosen for analysis was a system having four levels of probability p_j and ten subfiles N. The set of equations that describes the system is:

for k = 1

$$\alpha'_{kj} = \alpha_{kj} + \lambda\left\{\left(\sum_{i=k+1}^{10} \alpha_{ij}\right)p_j - Q_{kj}\left(\sum_{i=k+1}^{10} \pi_i\right)\right\}$$

for $k = 2, 3, \ldots, 9$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\left\{(-\alpha_{kj})p_j + Q_{k-1,j}\left(\sum_{i=k}^{10} \pi_i\right)\right.$$

$$\left. - Q_{kj}\left(\sum_{i=k+1}^{10} \pi_i\right)\right\}$$

and for $k = 10$

$$\alpha'_{k,j} = \alpha_{kj} + \lambda\left\{(-\alpha_{kj})p_j + Q_{k-1,j}\left(\sum_{i=k}^{10} \pi_i\right)\right\}$$

where $j = 1, 2, 3, 4.$

Multiple Feed System

The multiple feed system chosen for analysis was a system having four levels of probability p_j and ten subfiles N. The feed points were into subfiles one, four, and seven. The set of equations that describe the system are:

for $k = 1$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\left\{\left(\sum_{i=k+1}^{5} \alpha_{ij}\right)p_j - Q_{kj}\left(\sum_{i=k+1}^{5} \pi_i\right)\right\}$$

for $k = 2, 3$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\left\{(-\alpha_{kj})p_j + Q_{k-1,j}\left(\sum_{i=k}^{5} \pi_i\right)\right.$$

$$\left. - Q_{kj}\left(\sum_{i=k+1}^{5} \pi_i\right)\right\}$$

for $k = 4$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\Big\{\big(\sum_{i=k+2}^{8} \alpha_{ij} - \alpha_{kj}\big)p_j$$

$$+ Q_{k-1,j} \big(\sum_{i=k}^{5} \pi_i\big) - Q_{kj} \big(\sum_{i=k+1}^{5} \pi_i\big) - T_{kj} \big(\sum_{i=k+2}^{8} \pi_i\big)\Big\}$$

for $k = 5$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\Big\{(-\alpha_{kj})p_j + Q_{k-1,j} \big(\sum_{i=k}^{5} \pi_i\big)$$

$$+ T_{k-1,j} \big(\sum_{i=k+1}^{8} \pi_i\big) - T_{k,j} \big(\sum_{i=k+1}^{8} \pi_i\big)\Big\}$$

for $k = 6$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\Big\{(-\alpha_{kj})p_j + T_{k-1,j} \big(\sum_{i=k}^{8} \pi_i\big)$$

$$- T_{kj} \big(\sum_{i=k+1}^{8} \pi_i\big)\Big\}$$

for $k = 7$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\Big\{\big(\sum_{i=k+2}^{10} \alpha_{ij} - \alpha_{kj}\big)p_j + T_{k-1,j} \big(\sum_{i=k}^{8} \pi_i\big)$$

$$- T_{kj} \big(\sum_{i=k+1}^{8} \pi_i\big) - U_{kj} \big(\sum_{i=k+2}^{10} \pi_i\big)\Big\}$$

for $k = 8$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\Big\{(-\alpha_{kj})p_j + T_{k-1,j} \big(\sum_{i=k}^{8} \pi_i\big)$$

$$+ U_{k-1,j} \big(\sum_{i=k+1}^{10} \pi_i\big) - U_{kj} \big(\sum_{i=k+1}^{10} \pi_i\big)\Big\}$$

for $k = 9$

$$\alpha'_{kj} = \alpha_{kj} + \lambda\Big\{(-\alpha_{kj})p_j + U_{k-1,j} \big(\sum_{i=k}^{10} \pi_i\big)$$

$$- U_{kj} \left(\sum_{i=k+1}^{10} \pi_i \right) \Big\}$$

for k = 10

$$\alpha'_{kj} = \alpha_{kj} + \lambda \left\{ (-\alpha_{kj}) pj + U_{k-1,j} \left(\sum_{i=k}^{10} \pi_i \right) \right\}$$

This set of equations is derived from the general formulation of the feed systems by considering each region of the file having a feed loop as being a separate single feed system. The T_{kj} and U_{kj} coefficients are defined in the same manner as the $Q_{k,j}$ coefficients with the exception that the lower boundary N of the Q_{kj}'s is replaced by σ or ω. That the overall system can be considered as a superposition of several single feed systems is evident from the formulation of the feed systems.

TRANSPOSITION SYSTEM

Because the investigation to be carried out was intended to discover only the adaption characteristics of the systems, more sophisticated statistical analysis of the transposition system was discarded in favor of the more straightforward direct simulation.

A set of identification tags were placed in the file with a known weight attached to each tag. An interrogation set was prepared in such a fashion that

$$m\left[a_i \in I\right] = w(a_i)$$

where $m\left[a_i \in I\right]$ is the measure of a_i in the interrogation set and $w(a_i)$ is the weight of a_i in the identification tag set, or $w(a_i)$ is the measure induced on the identification tag set by the interrogation set. These weights must satisfy

$$\sum_{i=1}^{N} w(a_i) = 1$$

in order that the measure be a probability measure. The average length of the retrieval experiment is then the mean of the identification tag set and is given by

$$\sum_{i=1}^{N} x_i \; w(a_i)$$

where x_i is the range of the identification tag set.

This type of analysis led to a very simple simulation program with the disadvantage that the number of members in the identification tag set was limited to a relatively small number.

GENERAL FEED SYSTEM PROGRAM

The two systems, multiple feed and single feed, have many things in common from the point of view of the type of analysis used, and therefore should lend to a general program conceived in such a way that it would accommodate both types of systems.

The general feed system program was arranged to analyze a system that could be described by a system of 100 equations. The number of subfiles, N, and the number of levels of probability, m, could range from 1 to 33 and 1 to 10, respectively, with the condition that $mN \leq 100$. The systems described were restricted to those with the flow from the last subfile to a feed point coming from the subfile below the next lower feed point. The flow diagrams were labeled in terms of m, N, and the feed point subfile numbers σ, ϵ, ω, and ρ.

The calculation of the next instant α_{kj}'s could be accomplished by a transformation matrix applied to the present instant α_{kj}'s. This would require a 100 x 100 matrix and storage locations in excess of 10^4 in the computer. This was an impossibility and a pseudo-matrix transformation was used instead.

It should be noted that this program is general enough to describe the solution of any set of flow equations that can be arranged in the form

$$Q' = Q + \lambda \Big\{ (\text{external flow}) + (\text{incoming internal flow}) - (\text{outgoing internal flow}) \Big\}$$

where the Q's represent quantity and λ is a proportionality-time constant. The internal and external flows may be changing and may be recalculated after each application of the equations.

Single Feed System Program

Plate I shows the combination of subroutines used to accomplish the iterative solution of the single feed system equations. This program was derived from the general feed system program and the memory locations are those for the general feed system program.

Plate II is the block diagram for the subroutine that calculates the π_k's. The π_k's are given by

$$\pi_k = \sum_{j=1}^{m} \alpha_{kj}\, pj \qquad k = 1, 2, 3, \ldots, N$$

and this routine takes the α_{kj}'s from locations 0001 to 0040, forms the sum of products with the pj in locations 301 to 304, and stores the resultant π_k's in locations 311 to 320.

Plate III is the block diagram for the subroutine that calculates the Q_{kj}'s. The Q_{kj}'s are given by

$$Q_{kj} = \frac{\eta_{k,j}}{\sum_{\ell=1}^{m} \eta_{k,\ell}}$$

and

$$\eta_{kj} = \frac{(1 - pj)^{\sum_{i=1}^{k-1} n_i}}{\eta_k} \alpha_{kj} \qquad \text{for } \alpha_{kj} \geq 0$$

$$= 0 \qquad \text{for } \alpha_{kj} < 0$$

$$k = 1, 2, 3, \ldots, (N - 1); \; j = 1, 2, 3, 4$$

This routine tests the α_{kj}'s for sign and stores zero or the product in an intermediate location depending on the outcome of the sign test, and forms the sum of the η_{kj} at the same time. After m terms η_{kj} and their sum have been formed, the intermediate locations are divided by their sum and stored in locations 0681 to 0716.

Plate IV is the block diagram for the subroutine that calculates the sums $\sum_{i=k+1}^{N} \pi_i$, the sums $\sum_{\ell=2}^{N} \alpha_{\ell j}$, and the terms $-\alpha_{kj}$, and stores them in the locations for the transformation operation. The first part of the routine forms the sums

$$\sum_{i=k+1}^{N} \pi_i$$

with k increasing through k = 1, 2, 3, ..., (N - 1), taking the

π_i from locations 0311 to 0320, and stores these coefficients in locations 0346 to 0354. The second part of the routine forms the sums

$$\sum_{\ell=2}^{N} \alpha_{\ell j}$$

with ℓ increasing through $\ell = 1, 2, 3, 4$ taking the $\alpha_{\ell j}$ from locations 0005 to 0040, and stores these coefficients in locations 0381 to 0383. The third part of the routine changes the sign on the α_{kj}'s with k increasing through k = 2, 3, 4, ..., N, and j increasing through j = 1, 2, 3, 4, taking the α_{kj}'s from locations 0005 to 0040 and storing $-\alpha_{kj}$ in locations 0385 to 0420.

Plate V is the block diagram for the subroutine that applies the transformation to the α_{kj} to form the α'_{kj}. The routine forms the sum of α_{kj}, taken from locations 0001 to 0040, the product λ(external flow coefficient)pj, taken from locations 0000, 0381 to 0420, and 0301 to 0304, the product λ(internal flow coefficient) (incremental flow coefficient), taken from 0000, 0345 to 354, and 0681 to 0716, and stores these sums in locations 0101 to 0140. The last part of the routine replaces the α_{kj} in locations 0001 to 0040 by the α'_{kj} in locations 0101 to 0140.

Plates VI through IX show the block diagram for the subroutine that controls the punch-out and calculates the output data. All the α_{kj}'s in locations 0001 to 0040 are punched out, four words to a card. The sums

$$\sum_{j=1}^{m} \alpha_{kj} \qquad\qquad k = 1, 2, 3, \ldots, N$$

are punched out. These sums are the number of items in the k:th subfile and will be constant. The sums

$$\sum_{k=1}^{N} \alpha_{kj} \qquad\qquad j = 1, 2, 3, 4$$

are punched out. These sums are the number of items in the file of probability pj and will also be constant. These two sets of row and column sums are used as checks to indicate whether the program is performing properly. The average length of the retrieval experiment and the probabilities π_k are also punched out.

Plate VI is the block diagram for the punch-out of the α_{kj}'s. Plate VII is the block diagram for the calculation and punch-out of the row sums. Plate VIII is the block diagram for the calculation and punch-out of the column sums and γ. Plate IX is the block diagram for the punch-out of the π_k's. These routines also count the number of passes through the program and number the output cards as they are punched.

Multiple Feed System Program

Plate I shows the combination of the subroutines used to accomplish the iterative solution of the multiple feed system equations. This program was derived from the general feed system program.

Plate II is the block diagram for the subroutine that calculates the π_k's. This routine takes the α_{kj}'s in locations

0001 to 0040, forms the sum of products with the pj in locations 0301 to 0304, and stores the resultant π_k's in locations 0311 to 0320.

Plates X and XI are the block diagrams for the subroutine that calculates the incremental flow coefficients. The α_{kj} in locations 0001 to 0040 are tested for sign, then zero or the product $\frac{(1 - pj)^{\sum_{i=1}^{k} \eta_i}}{\eta_k} \alpha_{kj}$ is stored in an intermediate location and added to the sum. After the m = 4, η_{kj}'s have been formed and stored in the intermediate locations and added to the sum, each intermediate location is divided by the sum to form the incremental flow coefficients which are then stored in locations 0681 to 0724. After a fixed number of passes through the routine, controlled by auxiliary location 0678, the data addresses of the instructions that pick up the α_{kj}'s are modified to allow a skip of m = 4 in picking up the α_{kj}'s for the next set of passes through the program. This method was used instead of changing the index registers because not all the addresses modified by the index registers needed to be changed. Auxiliary locations 674 and 680 were used to control the number of passes until the next modification of the data addresses was required.

Plates XII through XV are the block diagrams that calculate and store the elements of the transformation applied to the α_{kj} to produce the α'_{kj}. The first part of Plate XII forms the terms $\sum_{i=k}^{5} \pi_i$, k + 2, 3, 4, 5, and stores them in locations 346 to 349. The second part of Plate XII forms the terms $\sum_{i=k}^{8} \pi_i$, $k = 6, 6, 7, 8$, and stores them in locations 350 to 353. The first part of Plate XIII forms the terms $\sum_{i=k}^{10} \pi_i$, $k = 9, 9, 10$, and stores them in locations 354 to 356. The second part of Plate XIII forms the terms $\sum_{k=2}^{5} \alpha_{kj}$, $j = 1, 2, 3, 4$, and stores them in locations 381 to 383. The first part of Plate XIV forms the terms $-\alpha_{kj}$, $k = 2, 3$, $j = 1, 2, 3, 4$, and stores them in locations 385 to 392. The second part of Plate XIV forms the terms $-\alpha_{4j} + \sum_{k=6}^{8} \alpha_{kj}$, $j = 1, 2, 3, 4$, and stores them in locations 393 to 396. The third part of Plate XIV forms the terms $-\alpha_{kj}$, $k = 5, 6$, $j = 1, 2, 3, 4$, and stores them in locations 397 to 404. The first part of Plate XV forms the terms $-\alpha_{7j} + \sum_{k=9}^{10} \alpha_{kj}$, $j = 1, 2, 3, 4$, and stores them in locations 405 to 408. The second part of Plate XV forms the terms $-\alpha_{kj}$, $k = 8, 9, 10$, $j = 1, 2, 3, 4$, and stores them in locations 409 to 420.

Plates XVI and XVII are the block diagrams that calculate the α_{kj} transformation and replace the α_{kj}'s by the α'_{kj}'s. Data address modification and auxiliary multipliers are used in the transformation calculation to control the picking up of terms and to allow sign changes on certain of the terms.

TRANSPOSITION SYSTEM PROGRAM

Plate XVIII shows the combination of subroutines to accomplish the simulation of the transposition system. The program was conceived to read the identification tag from a deck of interrogation cards representing the interrogation set, compare the tag read with the tags in the serial locations of the file until the same tag is found. The program then interchanges this item and the item $\propto$ units toward the origin and returns to read another interrogation card. The initial position of the items in the file was determined by reading a deck of cards, thus allowing random distribution of the items. The identification tag cards also carried the weight of the tag and this weight was used to calculate the mean value.

Plate XIX shows the routine for reading the random distribution of the identification cards into serial drum locations and the interrogation and interchange of the items in the file.

Plate XX shows the routine for the punch-out and calculation of the mean of the identification tag set.

DISCUSSION OF RESULTS

The results of the studies of all of the systems show a distinct trend toward ordering of the file. The degree of ordering in all cases is greater for more peaked usage probability density functions. The number of interrogations required to order the file is determined to be dependent upon the average number of locations moved by an interrogated item. This distance is greatest for the single feed system and is about the same for the multiple feed and transposition systems.

That the results of the investigation should show the same type of results for all the systems is apparent if the multiple feed system is viewed as the superposition of several single feed systems and the transposition system as a multiple feed system with randomly defined boundaries and feed points.

Feed System Results

Plate XX shows γ for the feed systems for the two densities presented to the systems. Density No. 1 was a relatively flat function, while density No. 2 was a relatively peaked function.

When presented density No. 2, γ for both systems approaches an asymptote of about .13 N after 3 N interrogations. The only significant difference in their response is that the initial speed of the single feed system is greater.

When presented with density No. 1, both systems show some ordering with the same characteristics noted for density No. 2, but the multiple feed system appears to have a stronger ordering tendency.

Plates XXI through XXXII show the total weight contained in each subfile of the systems versus interrogations for the two density presented to the systems.

By comparing Plates XXIV, XXV, and XXVI for the single feed system with Plates XXX, XXXI, and XXXII for the multiple feed system, it becomes apparent why the single feed system has a greater initial ordering speed. The multiple feed system orders each region defined by the feed points and this reduces the speed of ordering the whole file.

Transposition System Results

Plates XXXIII through XLII show γ as a function of the number of interrogations, with α as a parameter, for two densities. Density No. 1 was a relatively flat function and density No. 2 was a relatively peaked function.

The results show this system to have an obvious ordering trend dependent in degree upon the density function of the usage probabilities and the parameter α. The ordering for any α is stronger with a more peaked density function and becomes stronger for increasing α until some optimum α is reached, and then trades overall ordering for initial speed. This characteristic is shown very clearly by Plate XLIII, which shows γ for α ranging from 1 to 6 when the system was presented density No. 2.

Plates XLIV and XLV show the distribution of weights of the identification tags in the interrogation set for density No. 2 under initial conditions, after 3 N interrogations of the file and in the ideally ordered file. These results show the actual ordering of the weights of the items and the tendency toward ideal ordering is clearly evidenced.

Plate XLVI shows γ as a function of the number of interrogations for five arrangements of the interrogation sequence for density No. 2 and $\alpha = 6$. These curves show that the detailed ordering is determined by the sequence of the interrogations presented but that the adaptation characteristic is changed very little with different sequences.

MECHANIZATION OF THE TRANSPOSITION SYSTEM

It was pointed out in passing that of the three systems considered, the one consenting to the simplest mechanization was the transposition system. The only hardware required, in addition to the searching logic necessary in any system of serial scan type, is three registers with logic circuits to connect them to any location in the file.

To explain the system operation, assume the item requested is in the i:th location in the file. The i:th item is transferred from the i:th location to an intermediate register, then the $i - \alpha$:th item is transferred to the i:th position, and the i:th item is transferred from the intermediate register to the $i - \alpha$:th location.

Plate XLVII shows a block diagram of the system. The control block determines the sequence of events and provides the timing for the system.

When an item is requested from the file, the identification tag is set into the identification tag input and the control started. The control then produces the following sequence of events:

1. Increments the main address register by one.
2. Reads the identification tag of the location in the main address register in the comparator.
3. Senses the comparator output and branches; to 4 if the tags are unequal, to 5 if the tags are equal.

4. Increments the main address register by one and returns to 2.
5. Reads item of the location in the main address register into a display unit and into the first intermediate storage register.
6. Reads item of the location in the lag address register into the second intermediate storage register.
7. Reads the item in the first intermediate storage register into the location in the lag address register.
8. Reads the item in the second intermediate storage register into the location in the main address register.
9. Resets the main address and intermediate storage registers to zero, the lag address register to one, and returns to ready state to be started by the next request for an item.

The lag address register should be designed so that incrementing the main address register has no effect on the lag address register until α increments have gone by. The lag register then should be incremented along with the main register. This has the effect of keeping the address in the lag register α units toward the origin from the main register after at least α units have been added to the main register.

CONCLUSION

Three systems were investigated to determine the existence of an order adaptive characteristic and to determine the parameters affecting the speed and degree of adaptation. Because exact explicit mathematical representations of the systems proved unwieldy for solution, approximate models were used and their solutions programmed and run on the digital computer.

All of the systems investigated exhibit adaptation characteristics with well determined parameters. More extensive investigation will lead to closer determination of the relation between the adaptation and the parameters.

ACKNOWLEDGMENT

The author wishes to express his gratitude to Dr. Ing. Ugo Gagliardi for his posing of the problem and suggestions concerning the formulation of the solutions, and also to Dr. S. T. Parker for his assistance in the author's attempts to program the problems for the IBM 650 computer.

REFERENCES

Feller, W.
An introduction to probability theory and its applications. New York: John Wiley and Sons, 1950.

Joint committee report on large scale memory addressing. IBM San Jose Product Planning Technical Report No. 3, San Jose, California. 1958.

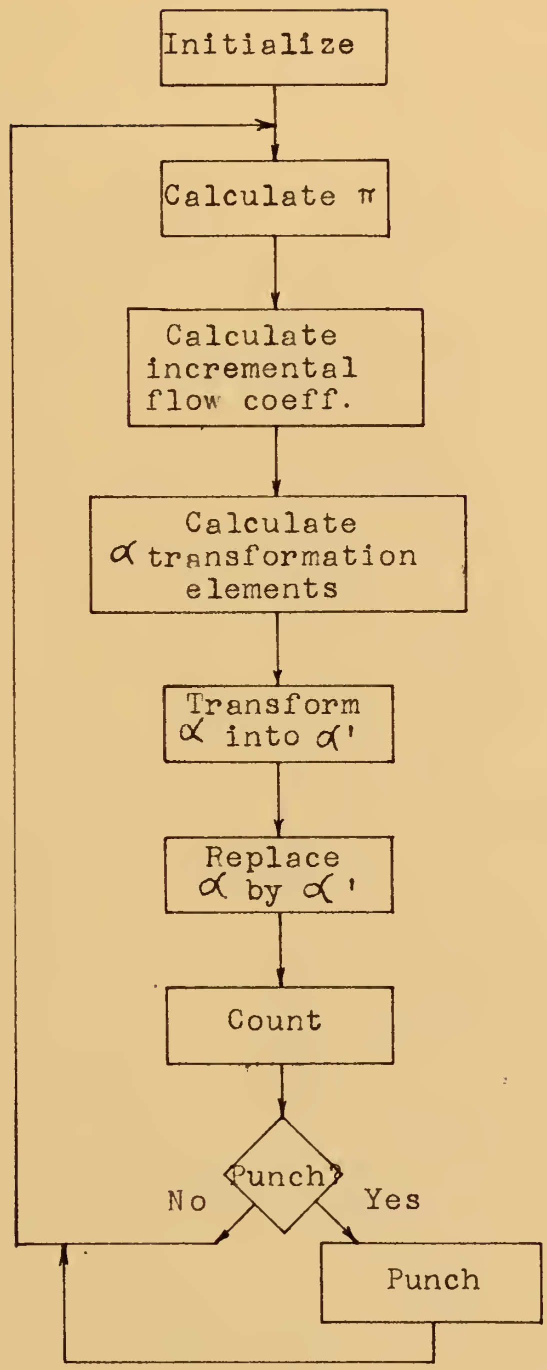
Shannon, C. E., and W. Weaver.
The mathematical theory of communication. Urbana: University of Illinois Press, 1949.

APPENDIX

EXPLANATION OF PLATE I

Symbolic routine for single feed and multiple feed systems.

PLATE I

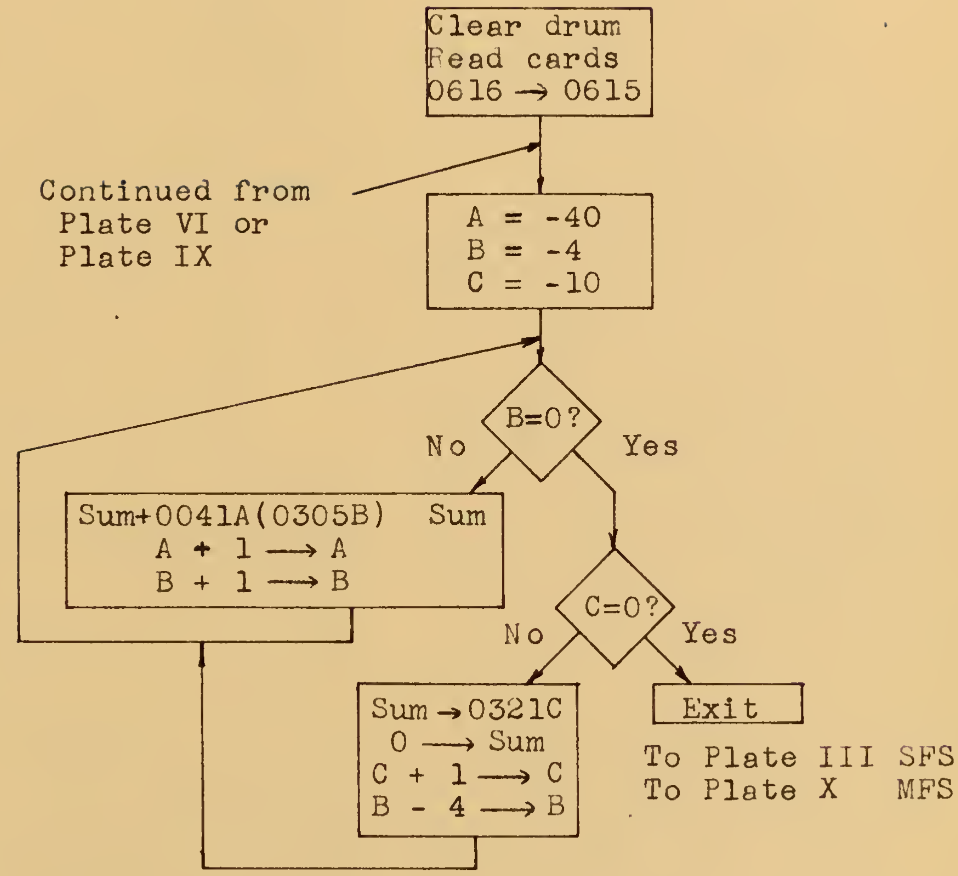


Initialize
Calculate π
Calculate incremental flow coeff.
Calculate α transformation elements
Transform α into α'
Replace α by α'
Count
Punch?
No
Yes
Punch

EXPLANATION OF PLATE II

π calculation subroutine for single feed and multiple feed systems.

PLATE II

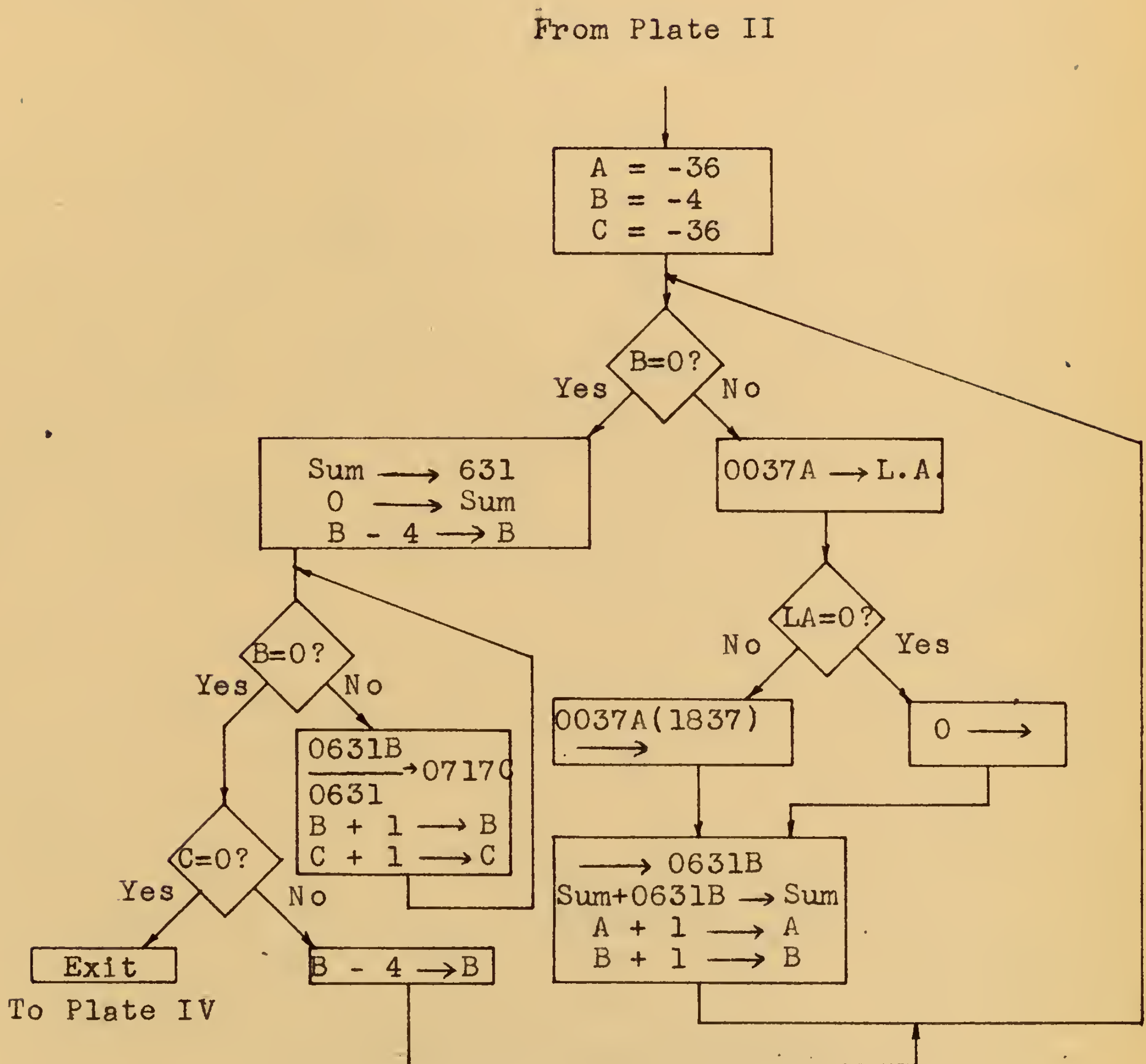


Clear drum
Read cards
0616 → 0615
Continued from
Plate VI or
Plate IX
A = -40
B = -4
C = -10
B=0?
No
Yes
Sum+0041A(0305B) Sum
A + 1 → A
B + 1 → B
C=0?
No
Yes
Sum → 0321C
0 → Sum
C + 1 → C
B - 4 → B
Exit
To Plate III SFS
To Plate X MFS

EXPLANATION OF PLATE III

Incremental flow calculation subroutine for single feed system.

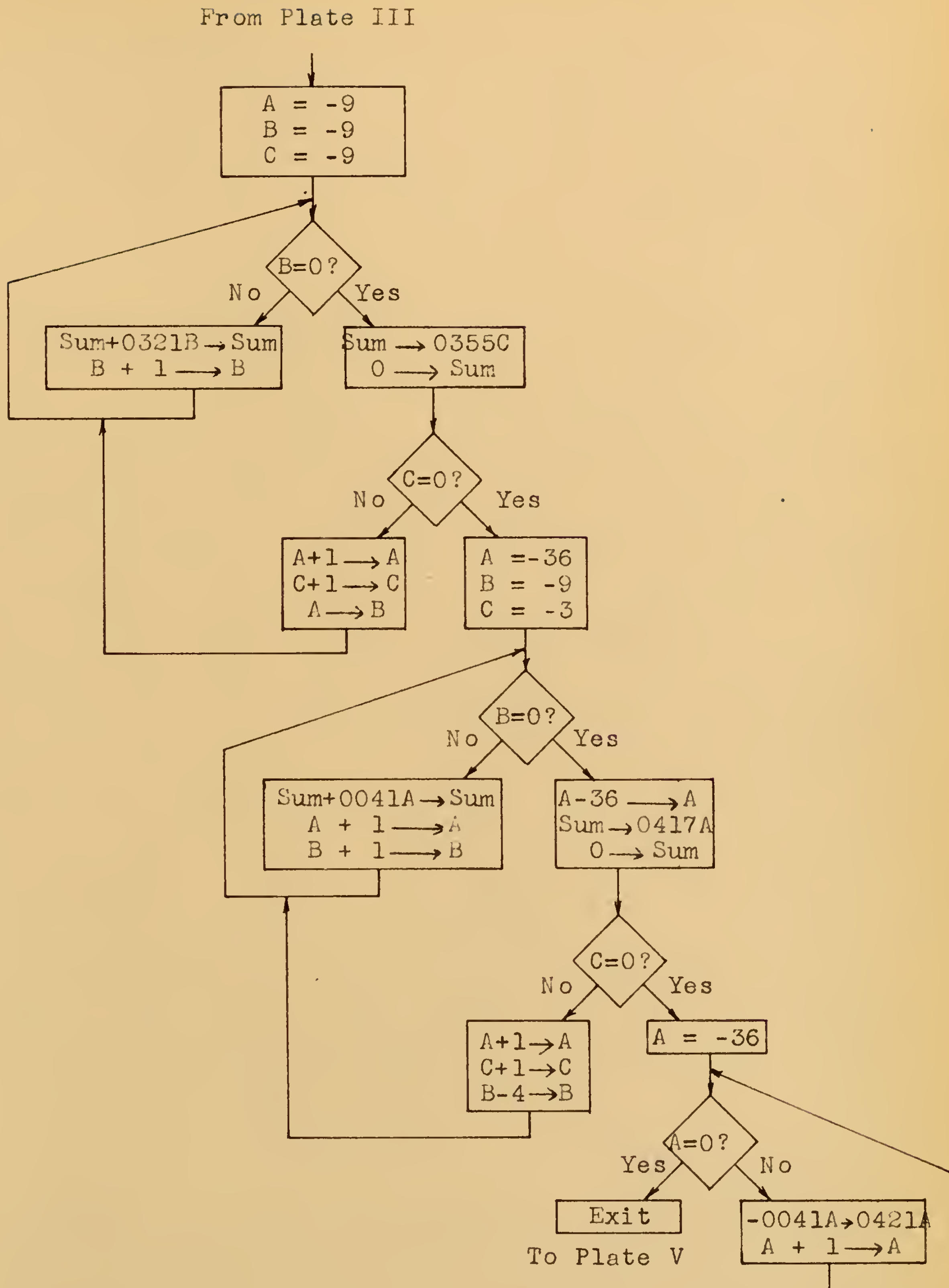
PLATE III


From Plate II
A = -36
B = -4
C = -36
B=0?
Yes
No
Sum ⟶ 631
0 ⟶ Sum
B - 4 ⟶ B
0037A ⟶ L.A.
LA=0?
No
Yes
B=0?
Yes
No
0631B ⟶ 0717C
0631
B + 1 ⟶ B
C + 1 ⟶ C
0037A(1837) ⟶
0 ⟶
⟶ 0631B
Sum+0631B ⟶ Sum
A + 1 ⟶ A
B + 1 ⟶ B
C=0?
Yes
No
Exit
B - 4 ⟶ B
To Plate IV

EXPLANATION OF PLATE IV

T elements calculation subroutine for single feed system.

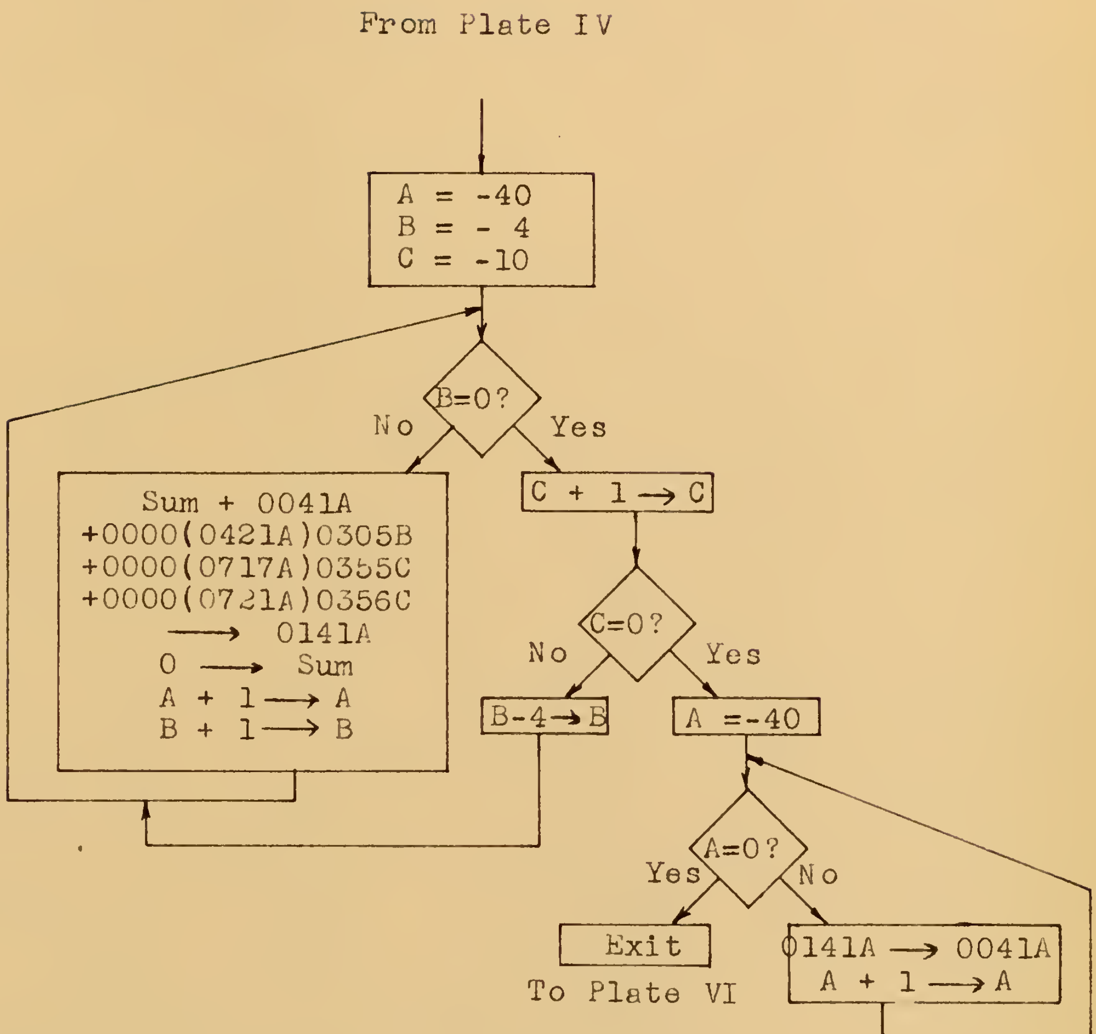
PLATE IV


From Plate III
A = -9
B = -9
C = -9
B=0?
No
Yes
Sum+0321B → Sum
B + 1 → B
Sum → 0355C
0 → Sum
C=0?
No
Yes
A+1 → A
C+1 → C
A → B
A =-36
B = -9
C = -3
B=0?
No
Yes
Sum+0041A → Sum
A + 1 → A
B + 1 → B
A-36 → A
Sum → 0417A
0 → Sum
C=0?
No
Yes
A+1 → A
C+1 → C
B-4 → B
A = -36
A=0?
Yes
No
Exit
-0041A → 0421A
A + 1 → A
To Plate V

EXPLANATION OF PLATE V

α' calculation subroutine for single feed system.

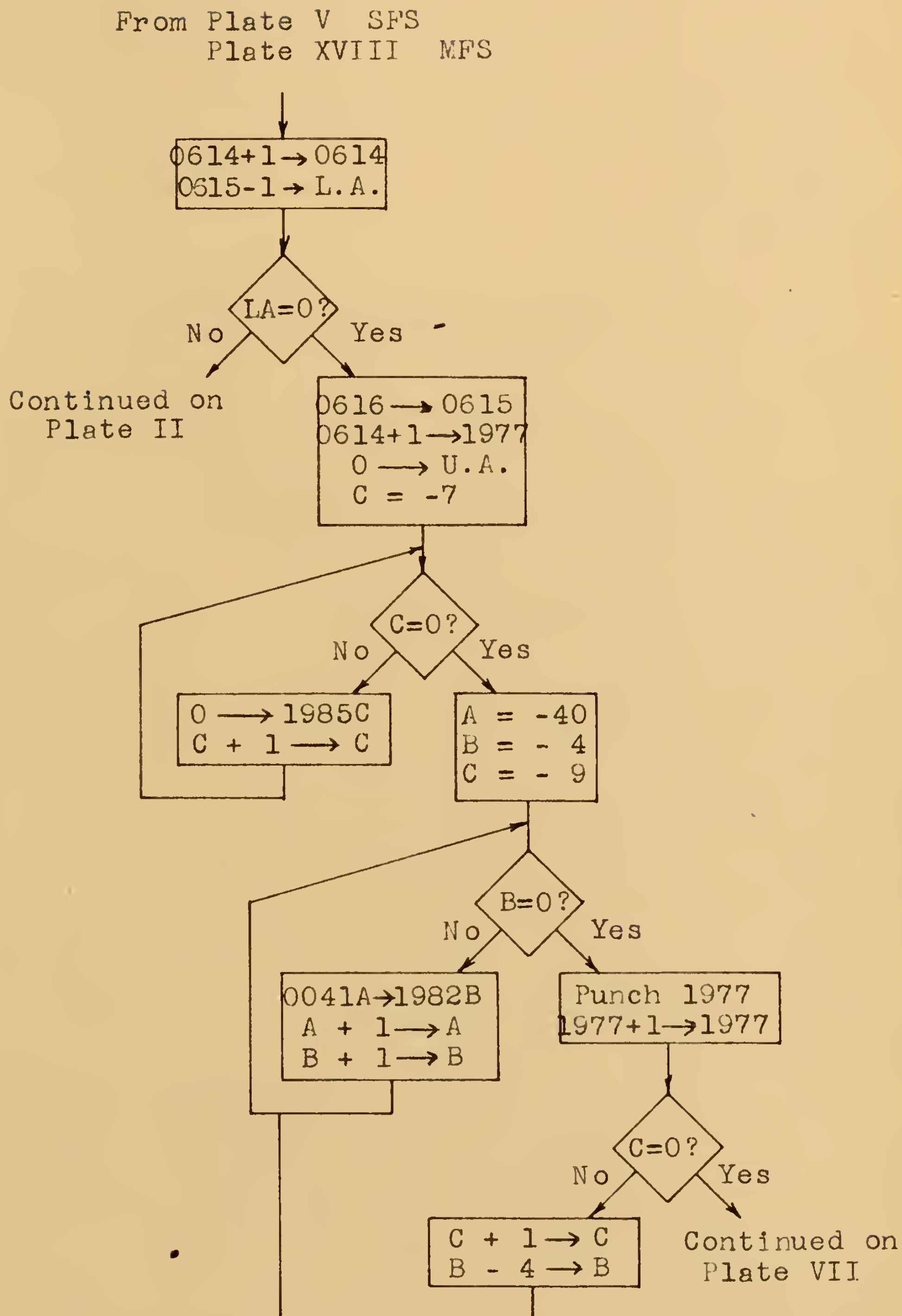
PLATE V


From Plate IV
A = -40
B = - 4
C = -10
B=0?
No
Yes
Sum + 0041A
+0000(0421A)0305B
+0000(0717A)0355C
+0000(0721A)0356C
⟶ 0141A
0 ⟶ Sum
A + 1 ⟶ A
B + 1 ⟶ B
C + 1 ⟶ C
C=0?
No
Yes
B-4 ⟶ B
A =-40
A=0?
Yes
No
Exit
To Plate VI
0141A ⟶ 0041A
A + 1 ⟶ A

EXPLANATION OF PLATE VI

Count-punch subroutine for single feed and multiple feed systems.

PLATE VI

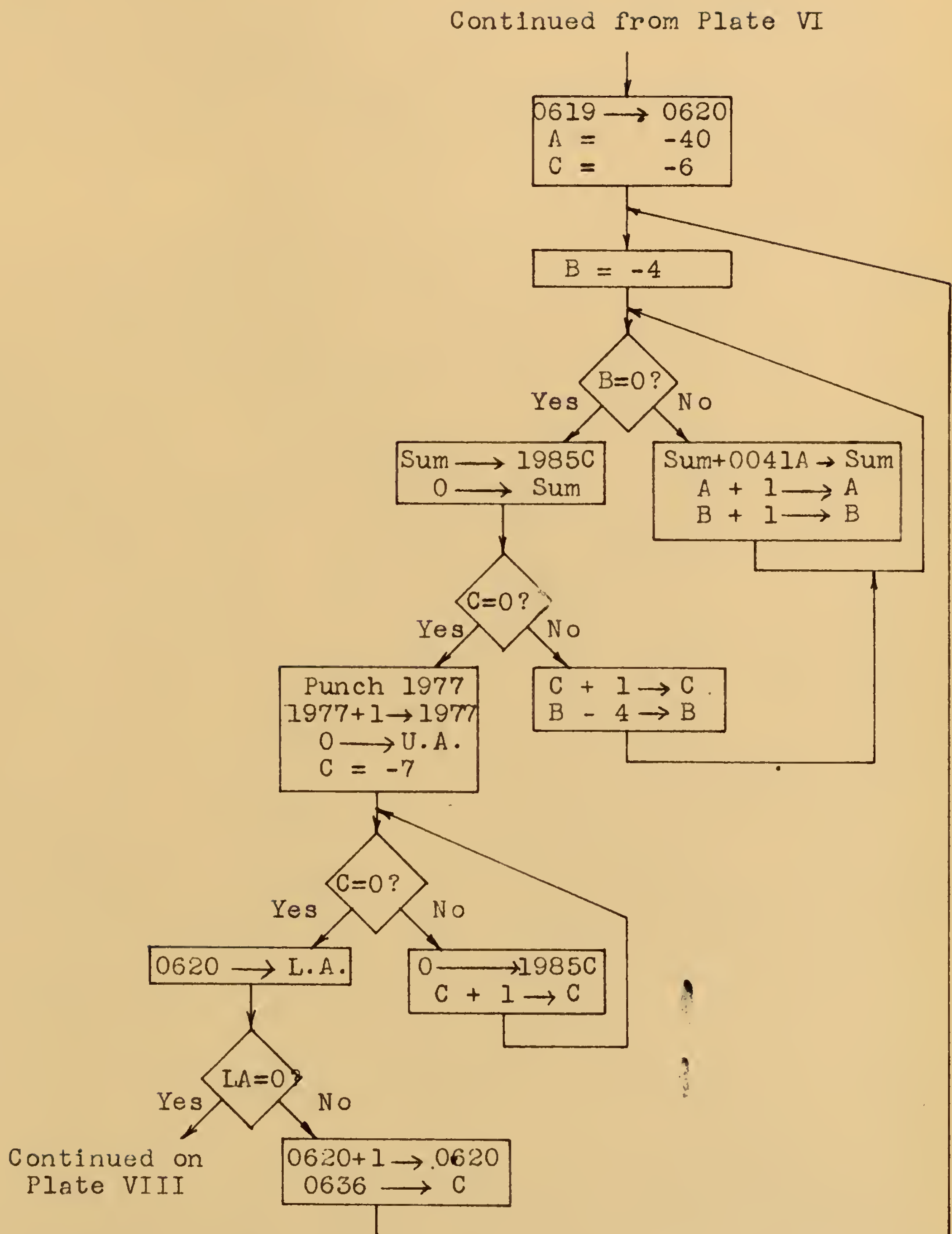


From Plate V SFS
Plate XVIII MFS
0614+1→0614
0615-1→L.A.
LA=0?
No
Yes
Continued on
Plate II
0616→0615
0614+1→1977
0→U.A.
C = -7
C=0?
No
Yes
0→1985C
C + 1→C
A = -40
B = - 4
C = - 9
B=0?
No
Yes
0041A→1982B
A + 1→A
B + 1→B
Punch 1977
1977+1→1977
C=0?
No
Yes
C + 1→C
B - 4→B
Continued on
Plate VII

EXPLANATION OF PLATE VII

Punch subroutine single feed and multiple feed systems.

PLATE VII

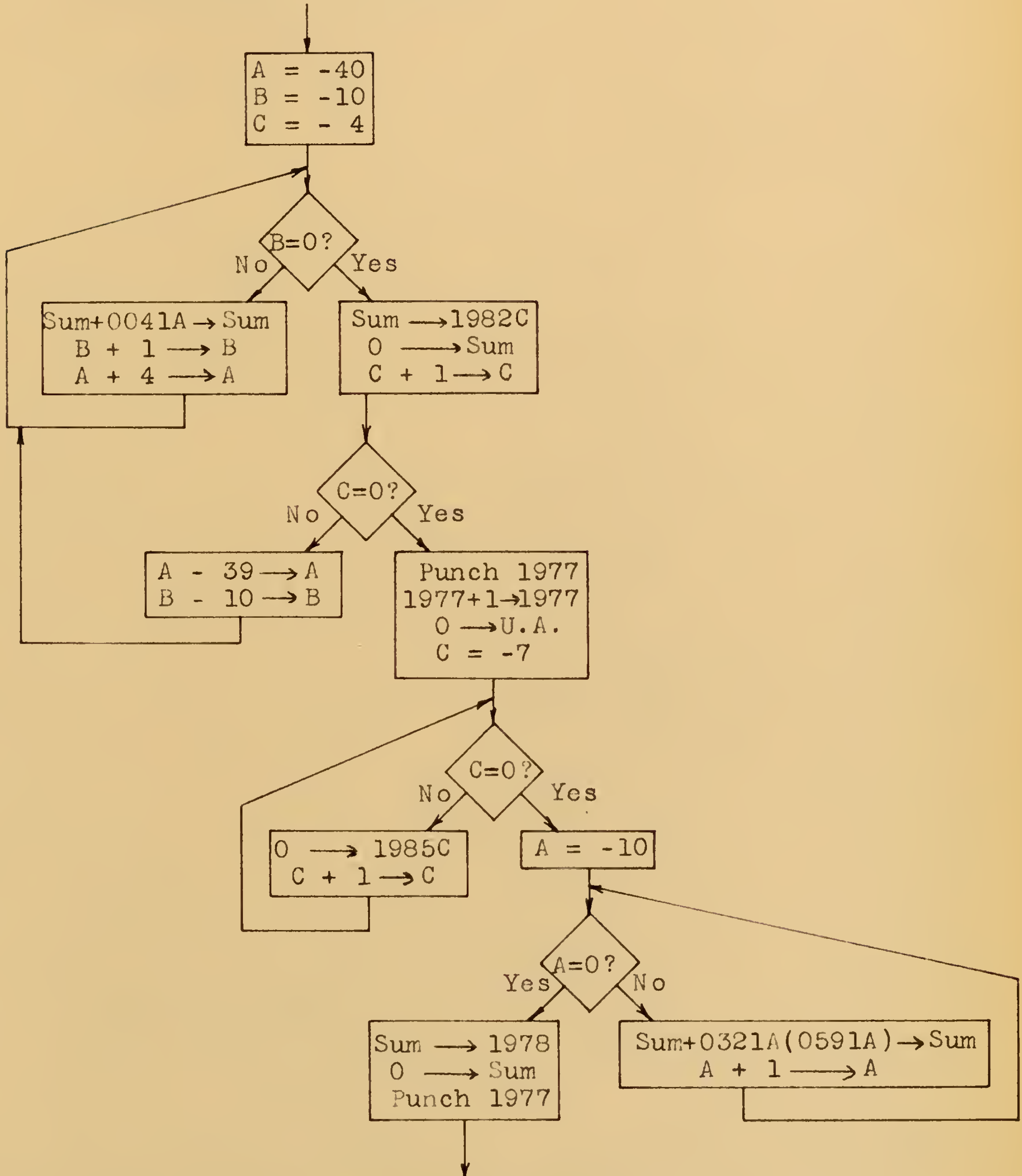

Continued from Plate VI
0619 → 0620
A = -40
C = -6
B = -4
B=0?
Yes
No
Sum → 1985C
0 → Sum
Sum+0041A → Sum
A + 1 → A
B + 1 → B
C=0?
Yes
No
Punch 1977
1977+1→1977
0 → U.A.
C = -7
C + 1 → C
B - 4 → B
C=0?
Yes
No
0620 → L.A.
0 → 1985C
C + 1 → C
LA=0?
Yes
No
Continued on
Plate VIII
0620+1 → 0620
0636 → C

EXPLANATION OF PLATE VIII

Punch subroutine for single feed and multiple feed system.

PLATE VIII

Continued from Plate VII

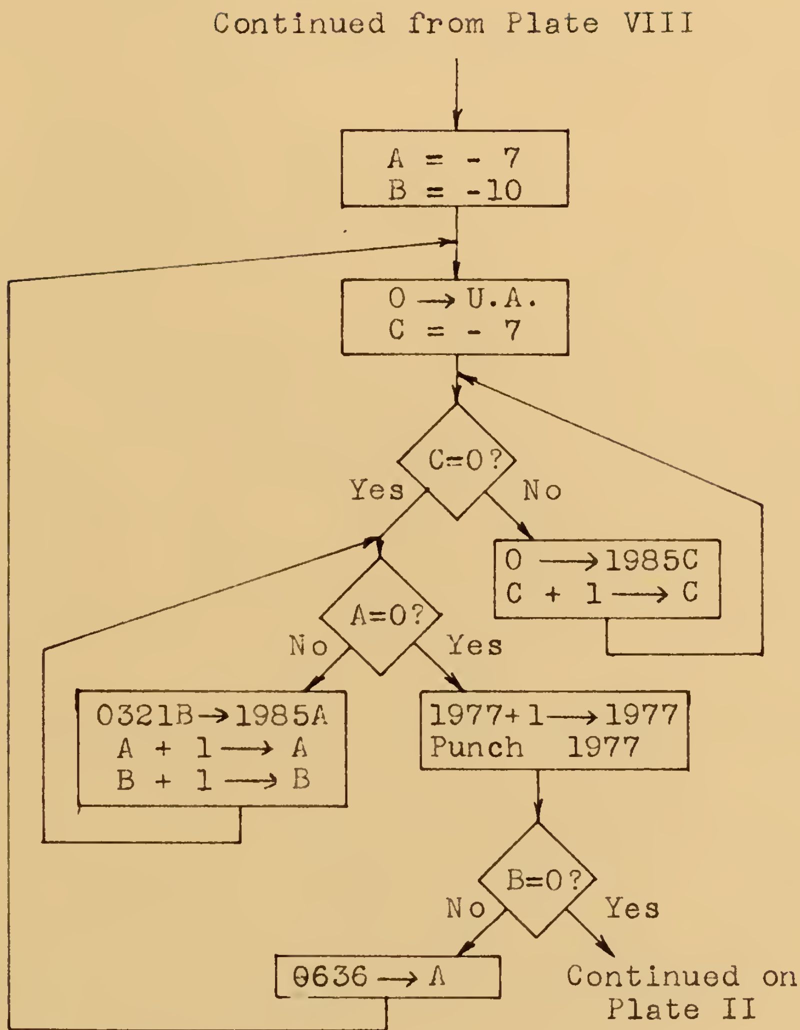


Continued on Plate IX

EXPLANATION OF PLATE IX

Punch subroutine for single feed and multiple feed system.

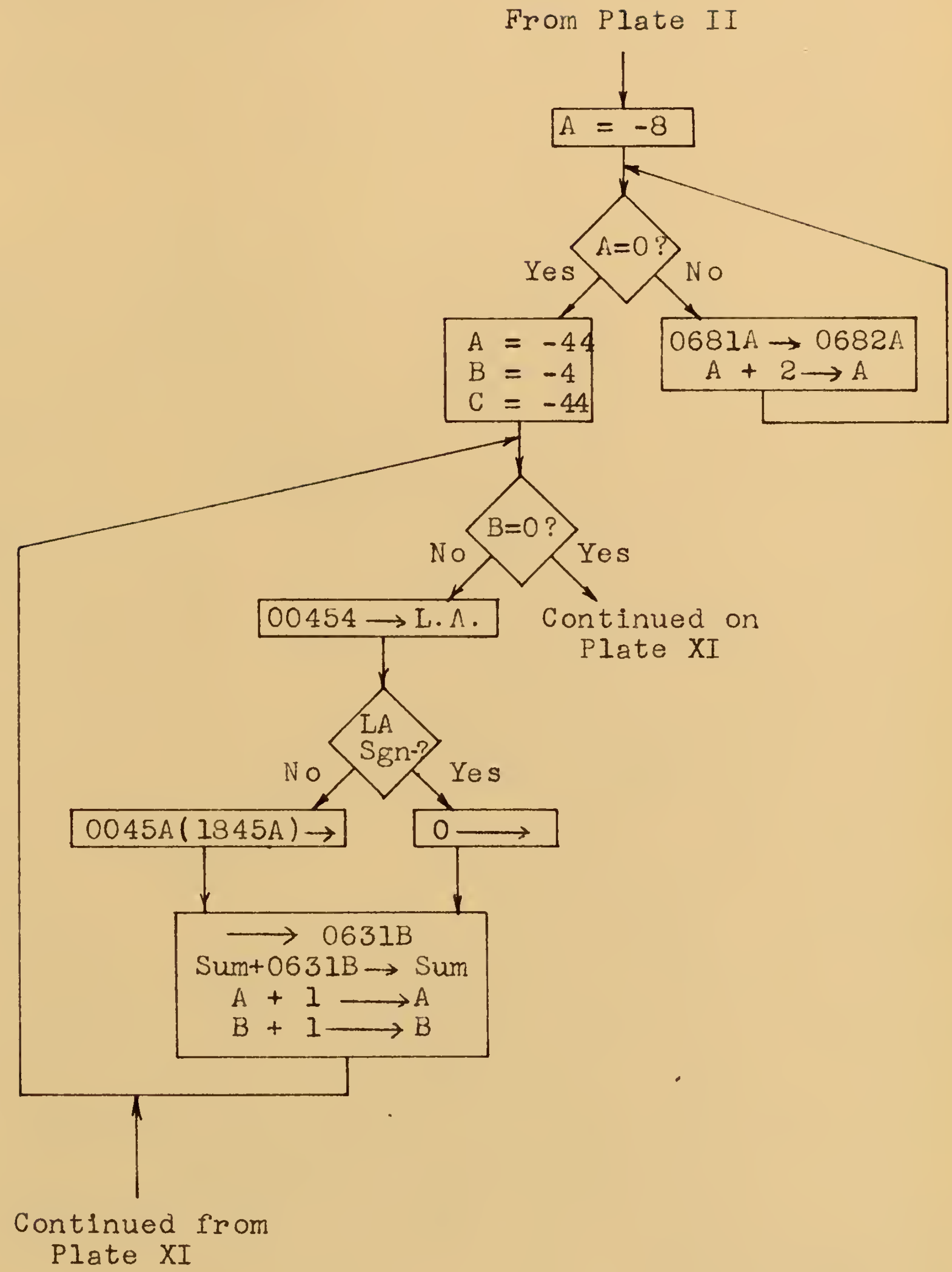
PLATE IX


Continued from Plate VIII
A = - 7
B = -10
0 → U.A.
C = - 7
C=0 ?
Yes
No
0 → 1985C
C + 1 → C
A=0 ?
No
Yes
0321B → 1985A
A + 1 → A
B + 1 → B
1977+1 → 1977
Punch 1977
B=0 ?
No
Yes
0636 → A
Continued on
Plate II

EXPLANATION OF PLATE X

Incremental flow calculation subroutine for multiple feed system.

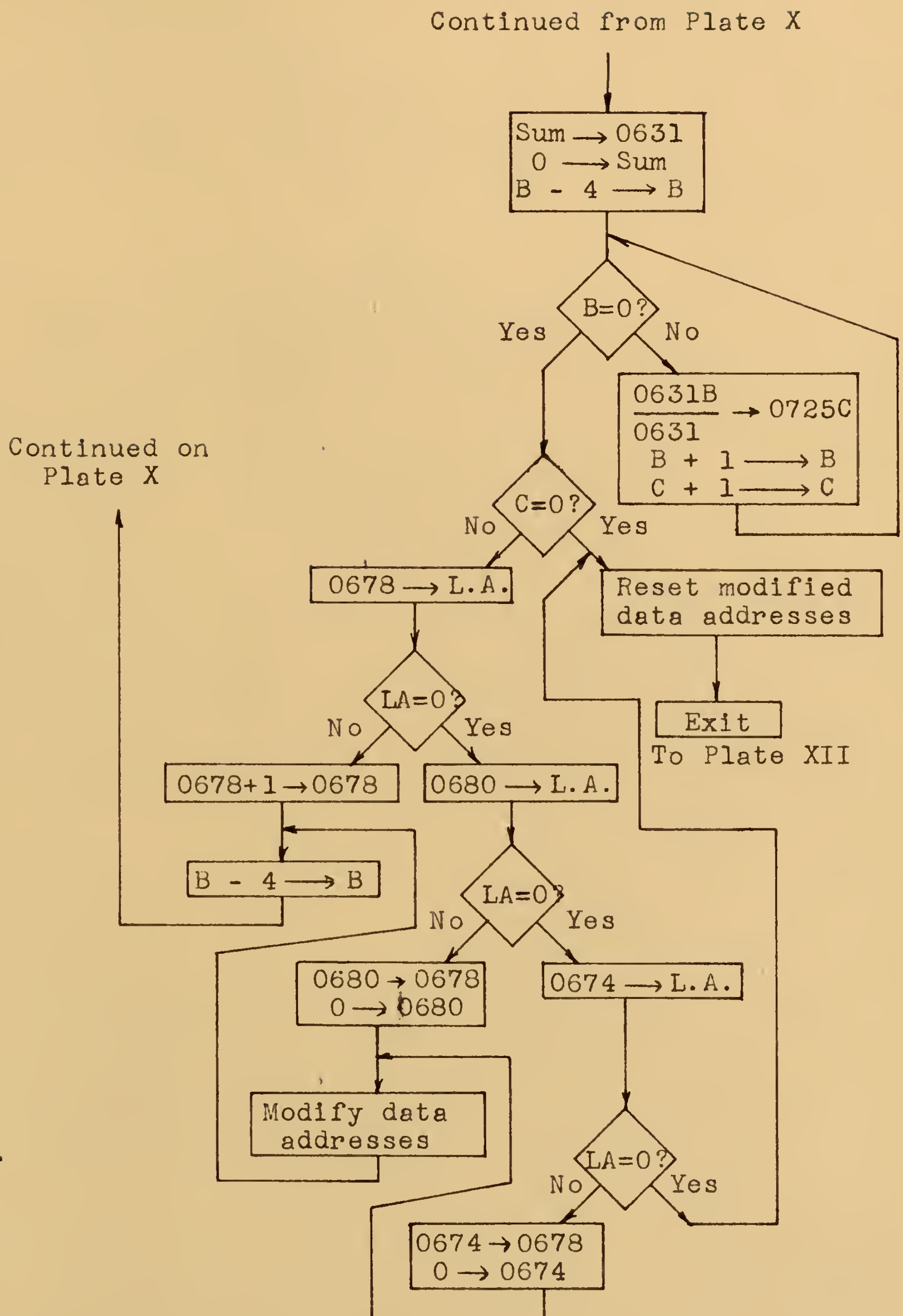
PLATE X


From Plate II
A = -8
A=0 ?
Yes
No
A = -44
B = -4
C = -44
0681A → 0682A
A + 2 → A
B=0 ?
No
Yes
00454 → L. A.
Continued on
Plate XI
LA
Sgn-?
No
Yes
0045A(1845A) →
0 →
→ 0631B
Sum+0631B → Sum
A + 1 → A
B + 1 → B
Continued from
Plate XI

EXPLANATION OF PLATE XI

Incremental flow calculation subroutine for multiple feed system.

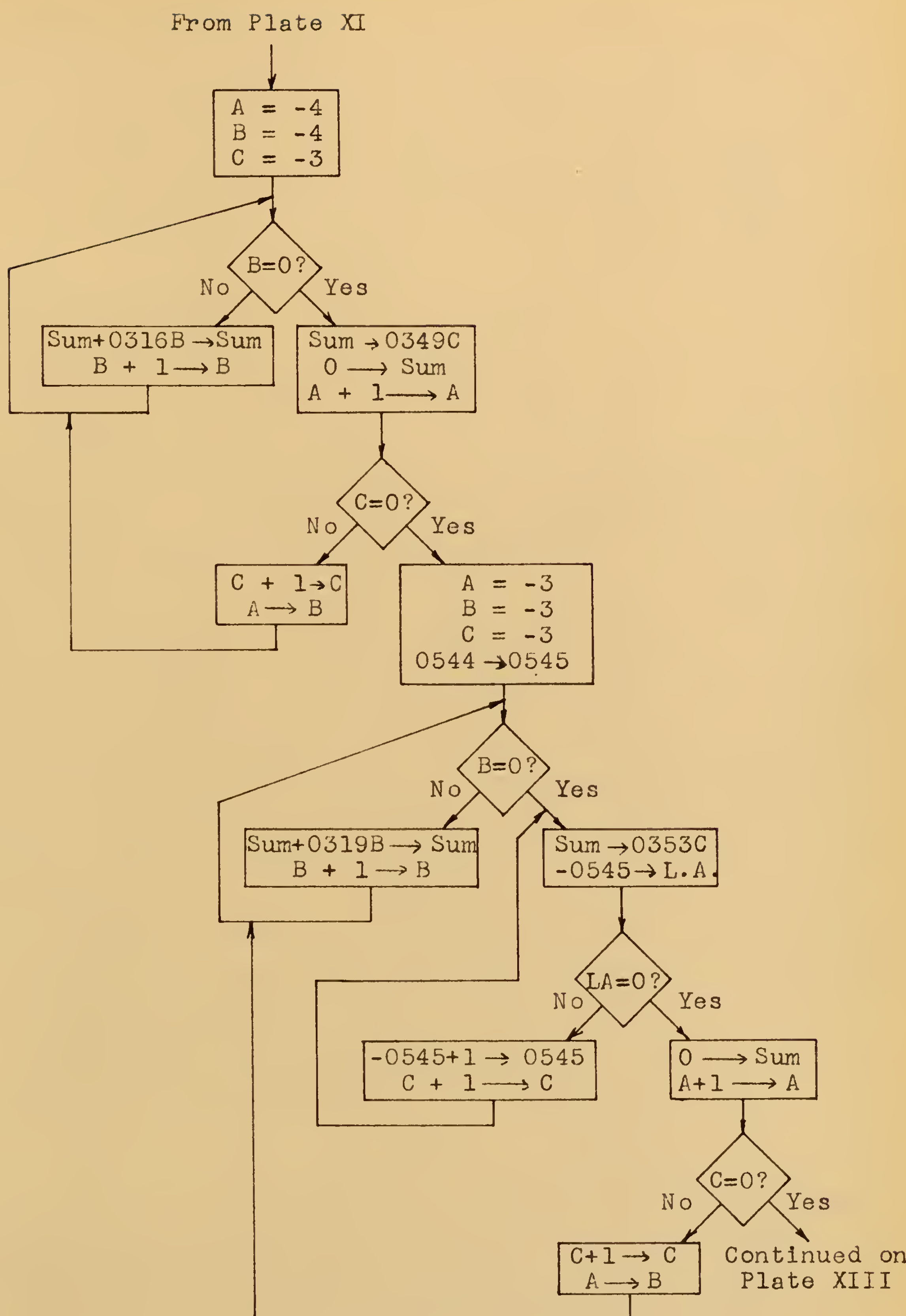
PLATE XI


Continued from Plate X
Sum → 0631
0 → Sum
B - 4 → B
B=0?
Yes
No
0631B / 0631 → 0725C
B + 1 → B
C + 1 → C
Continued on Plate X
C=0?
No
Yes
0678 → L.A.
Reset modified data addresses
LA=0?
No
Yes
Exit
To Plate XII
0678+1→0678
0680 → L.A.
B - 4 → B
LA=0?
No
Yes
0680 → 0678
0 → 0680
0674 → L.A.
Modify data addresses
LA=0?
No
Yes
0674 → 0678
0 → 0674

EXPLANATION OF PLATE XII

T elements calculation subroutine for multiple feed system.

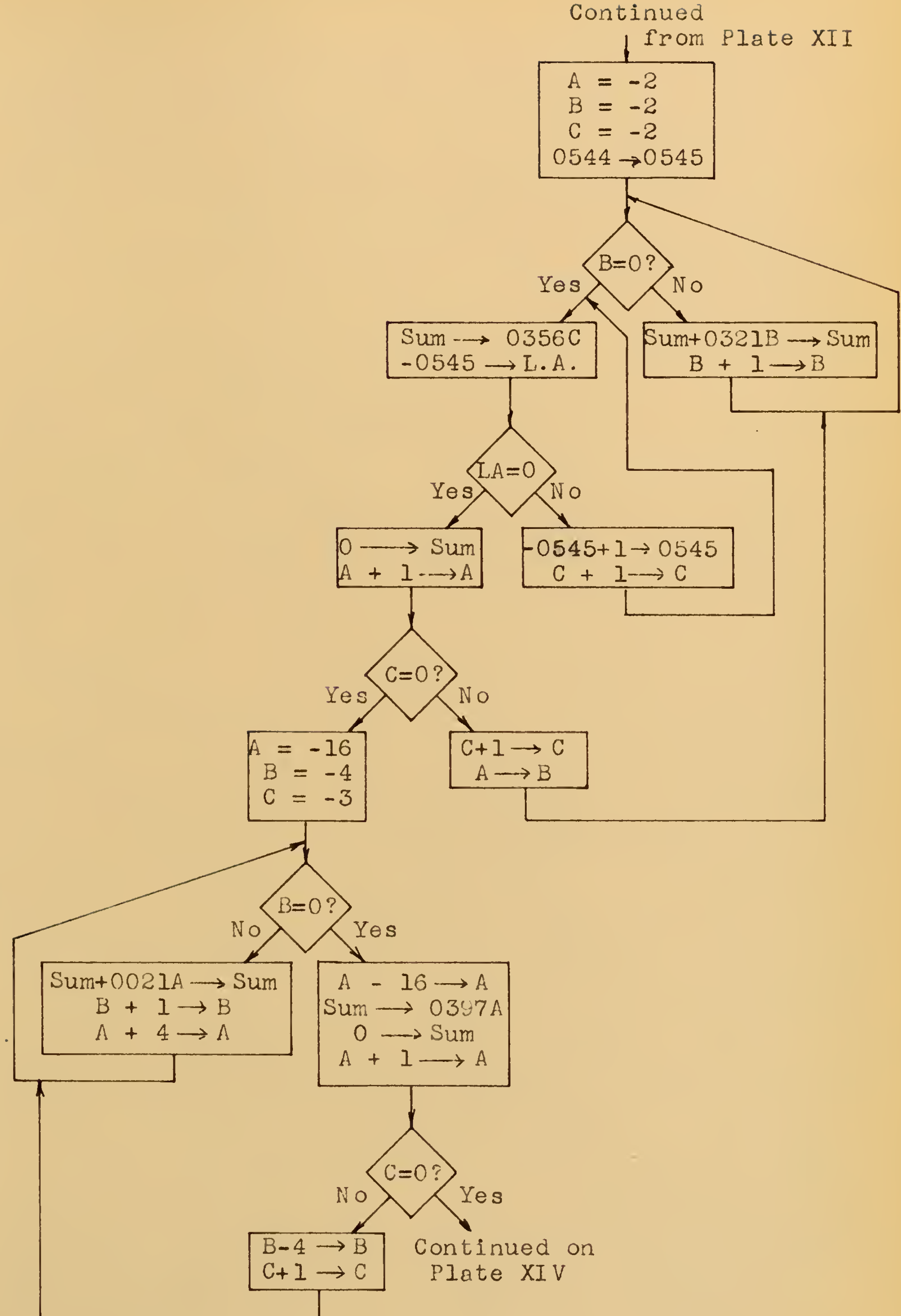
PLATE XII


From Plate XI
A = -4
B = -4
C = -3
B=0?
No
Yes
Sum+0316B → Sum
B + 1 → B
Sum → 0349C
0 → Sum
A + 1 → A
C=0?
No
Yes
C + 1 → C
A → B
A = -3
B = -3
C = -3
0544 → 0545
B=0?
No
Yes
Sum+0319B → Sum
B + 1 → B
Sum → 0353C
-0545 → L.A.
LA=0?
No
Yes
-0545+1 → 0545
C + 1 → C
0 → Sum
A+1 → A
C=0?
No
Yes
C+1 → C
A → B
Continued on Plate XIII

EXPLANATION OF PLATE XIII

T elements calculation subroutine for multiple feed system.

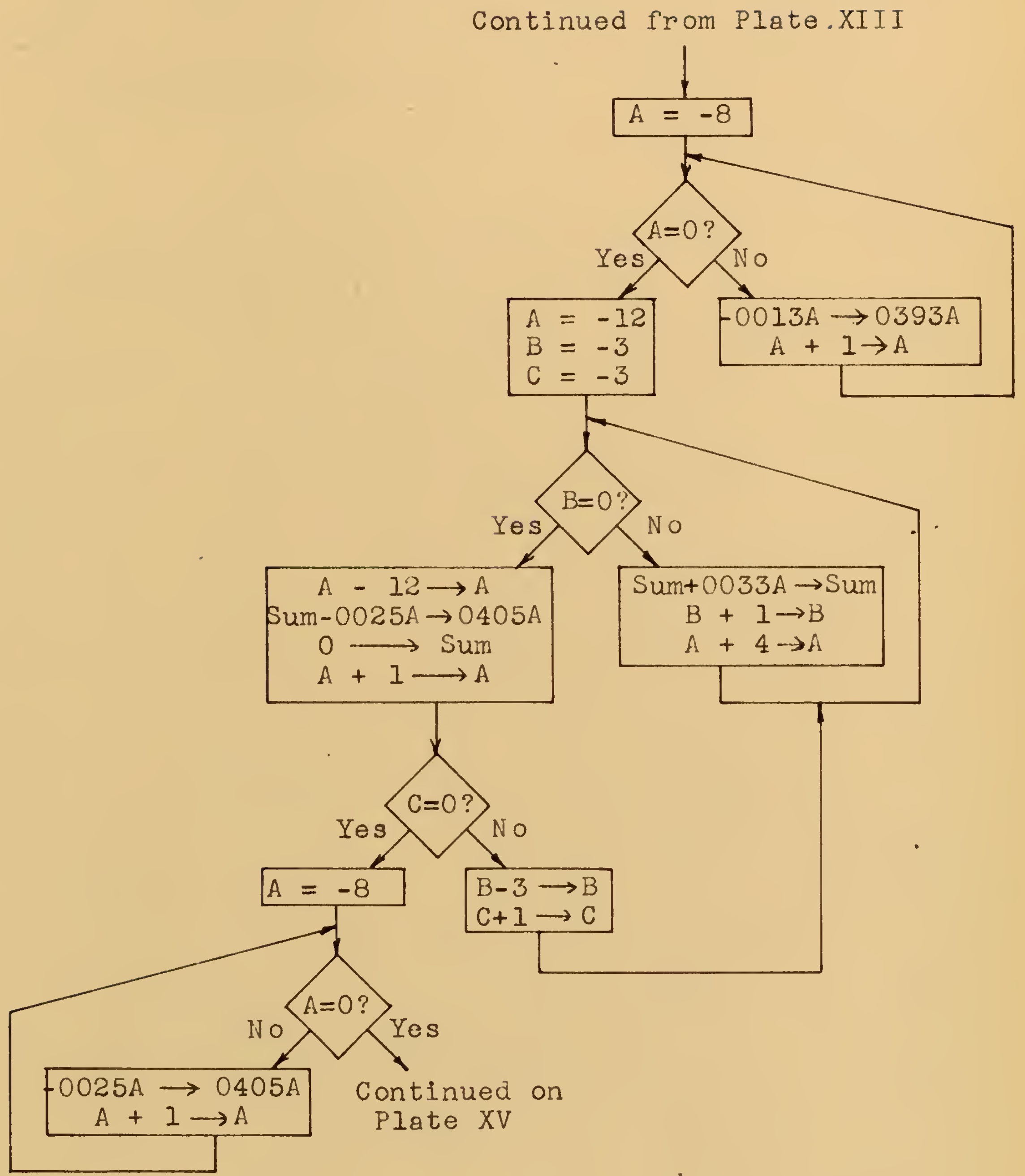
PLATE XIII


Continued from Plate XII
A = -2
B = -2
C = -2
0544 → 0545
B=0?
Yes
No
Sum → 0356C
-0545 → L.A.
Sum+0321B → Sum
B + 1 → B
LA=0
Yes
No
0 → Sum
A + 1 → A
0545+1 → 0545
C + 1 → C
C=0?
Yes
No
A = -16
B = -4
C = -3
C+1 → C
A → B
B=0?
No
Yes
Sum+0021A → Sum
B + 1 → B
A + 4 → A
A - 16 → A
Sum → 0397A
0 → Sum
A + 1 → A
C=0?
No
Yes
B-4 → B
C+1 → C
Continued on Plate XIV

EXPLANATION OF PLATE XIV

T elements calculation subroutine for multiple feed system.

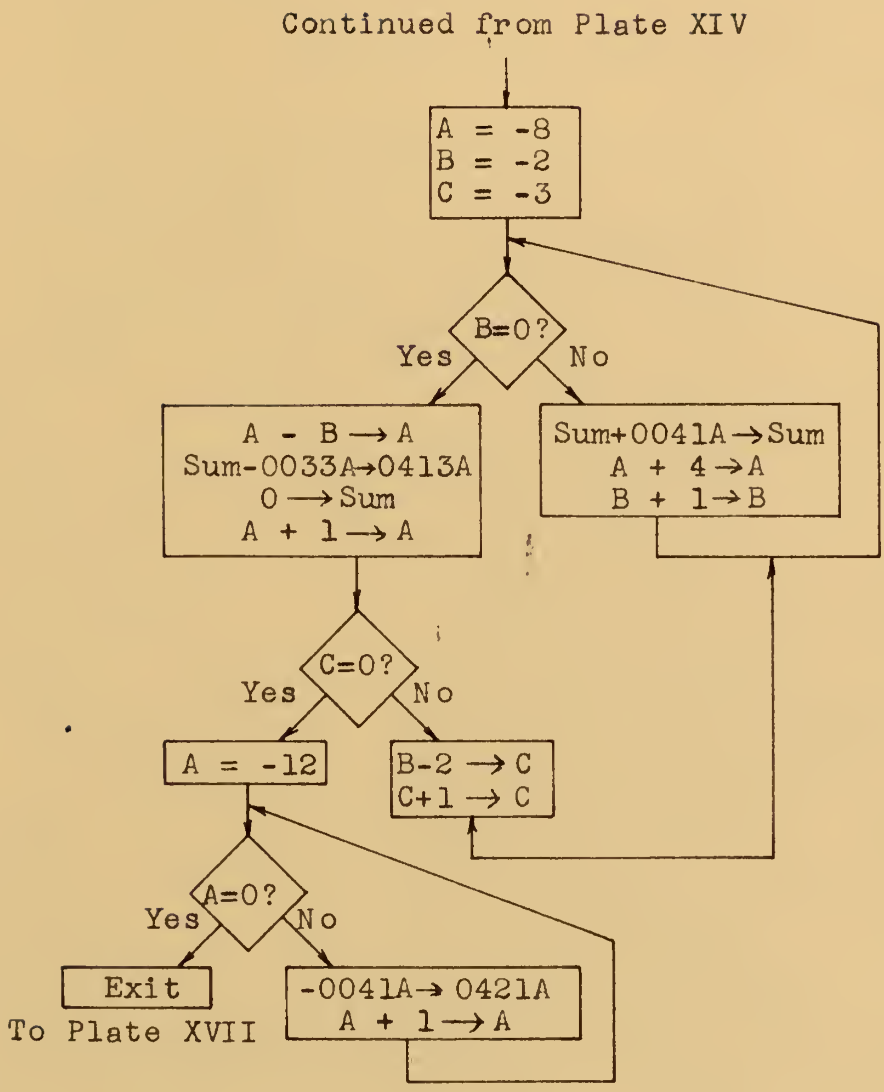
PLATE XIV


Continued from Plate.XIII
A = -8
A=0?
Yes
No
A = -12
B = -3
C = -3
-0013A → 0393A
A + 1→A
B=0?
Yes
No
A - 12 → A
Sum-0025A → 0405A
0 → Sum
A + 1 → A
Sum+0033A → Sum
B + 1→B
A + 4 → A
C=0?
Yes
No
A = -8
B-3 → B
C+1 → C
A=0?
No
Yes
-0025A → 0405A
A + 1 → A
Continued on
Plate XV

EXPLANATION OF PLATE XV

T elements calculation subroutine for multiple feed system.

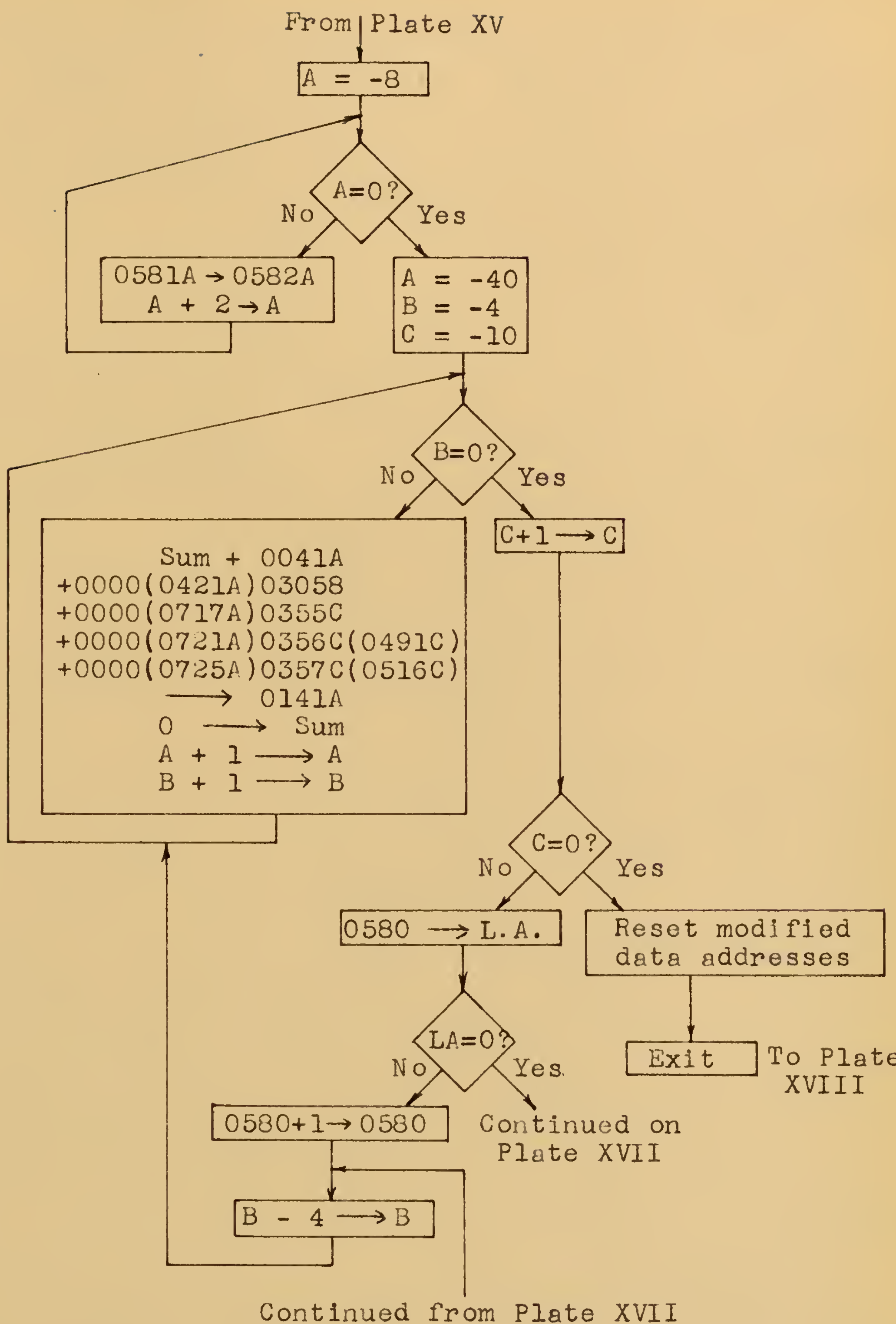
PLATE XV


Continued from Plate XIV
A = -8
B = -2
C = -3
B=0?
Yes
No
A - B → A
Sum-0033A→0413A
0 → Sum
A + 1 → A
Sum+0041A→Sum
A + 4→A
B + 1→B
C=0?
Yes
No
A = -12
B-2 → C
C+1 → C
A=0?
Yes
No
Exit
To Plate XVII
-0041A→ 0421A
A + 1 → A

EXPLANATION OF PLATE XVI

α' calculation subroutine for multiple feed system.

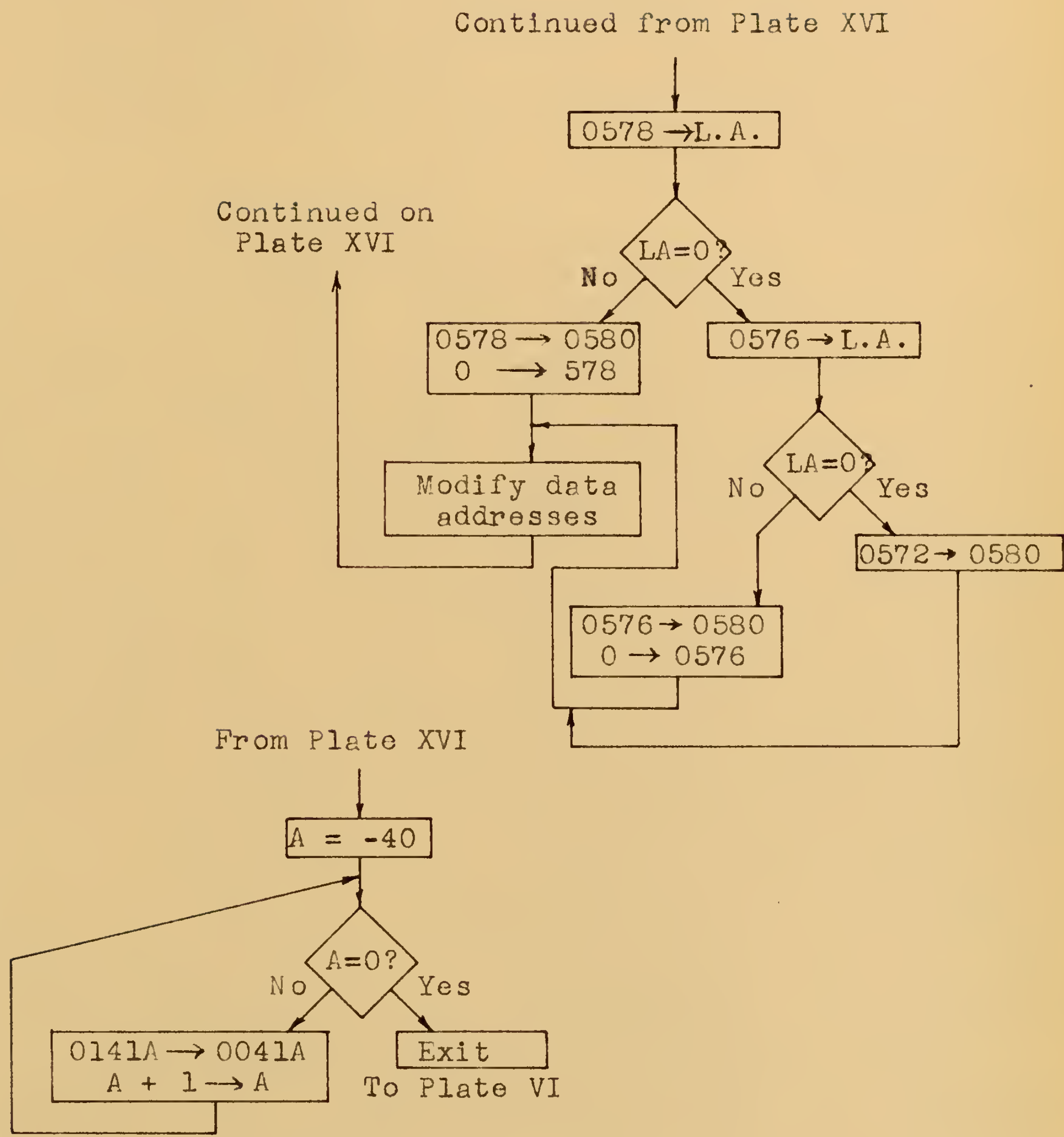
PLATE XVI


From Plate XV
A = -8
A=0 ?
No
Yes
0581A → 0582A
A + 2 → A
A = -40
B = -4
C = -10
B=0 ?
No
Yes
Sum + 0041A
+0000(0421A)03058
+0000(0717A)0355C
+0000(0721A)0356C(0491C)
+0000(0725A)0357C(0516C)
⟶ 0141A
0 ⟶ Sum
A + 1 ⟶ A
B + 1 ⟶ B
C+1 ⟶ C
C=0 ?
No
Yes
0580 ⟶ L.A.
Reset modified data addresses
LA=0 ?
No
Yes
Exit
To Plate XVIII
0580+1 → 0580
Continued on Plate XVII
B - 4 ⟶ B
Continued from Plate XVII

EXPLANATION OF PLATE XVII

α' calculation and transfer subroutines for multiple feed system.

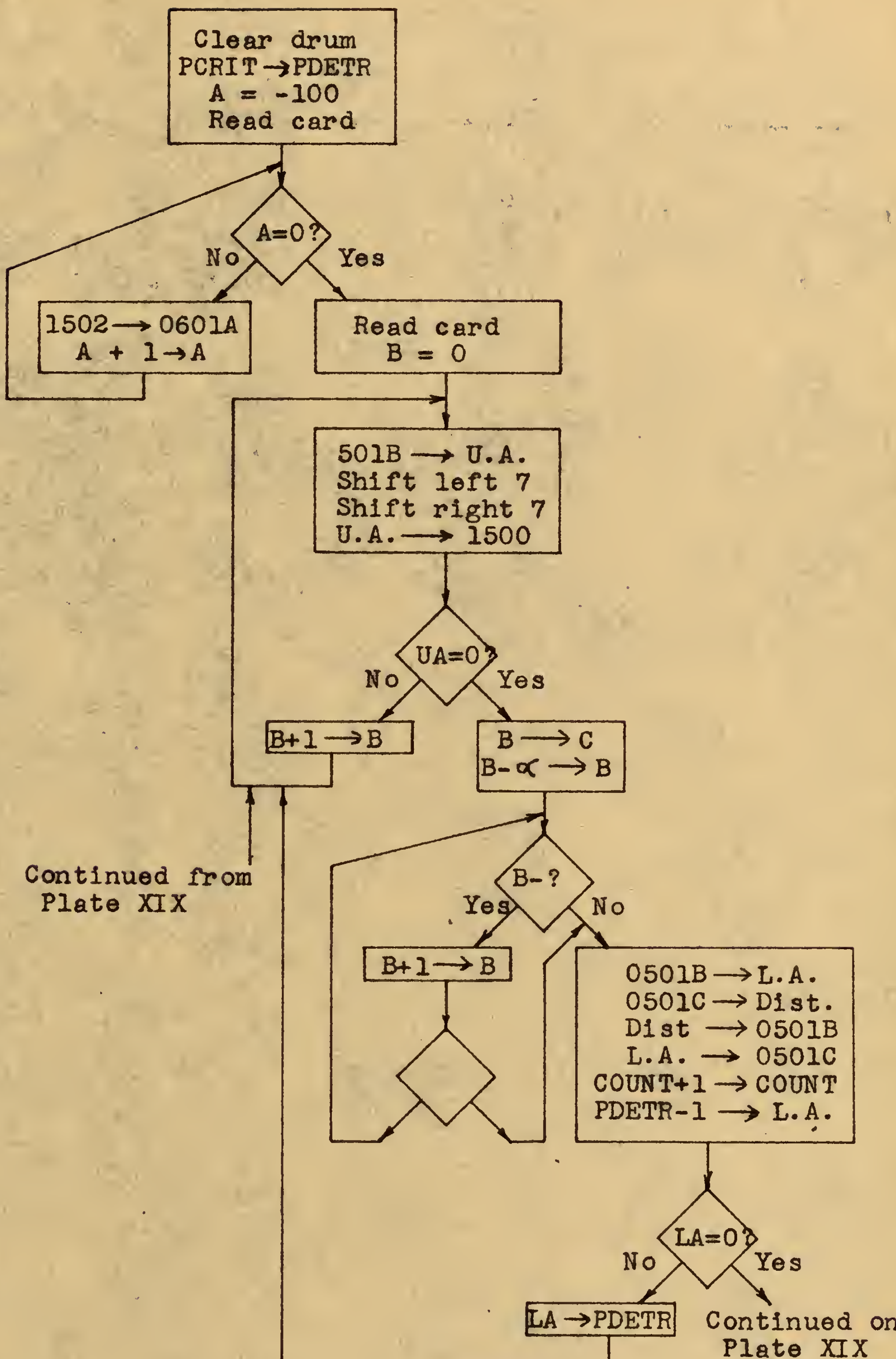
PLATE XVII


Continued from Plate XVI
0578 → L. A.
LA=0?
No
Yes
Continued on
Plate XVI
0578 → 0580
0 → 578
0576 → L. A.
Modify data
addresses
LA=0?
No
Yes
0572 → 0580
0576 → 0580
0 → 0576
From Plate XVI
A = -40
A=0?
No
Yes
0141A → 0041A
A + 1 → A
Exit
To Plate VI

EXPLANATION OF PLATE XVIII

Identification tag storage and interrogation subroutines for transposition system.

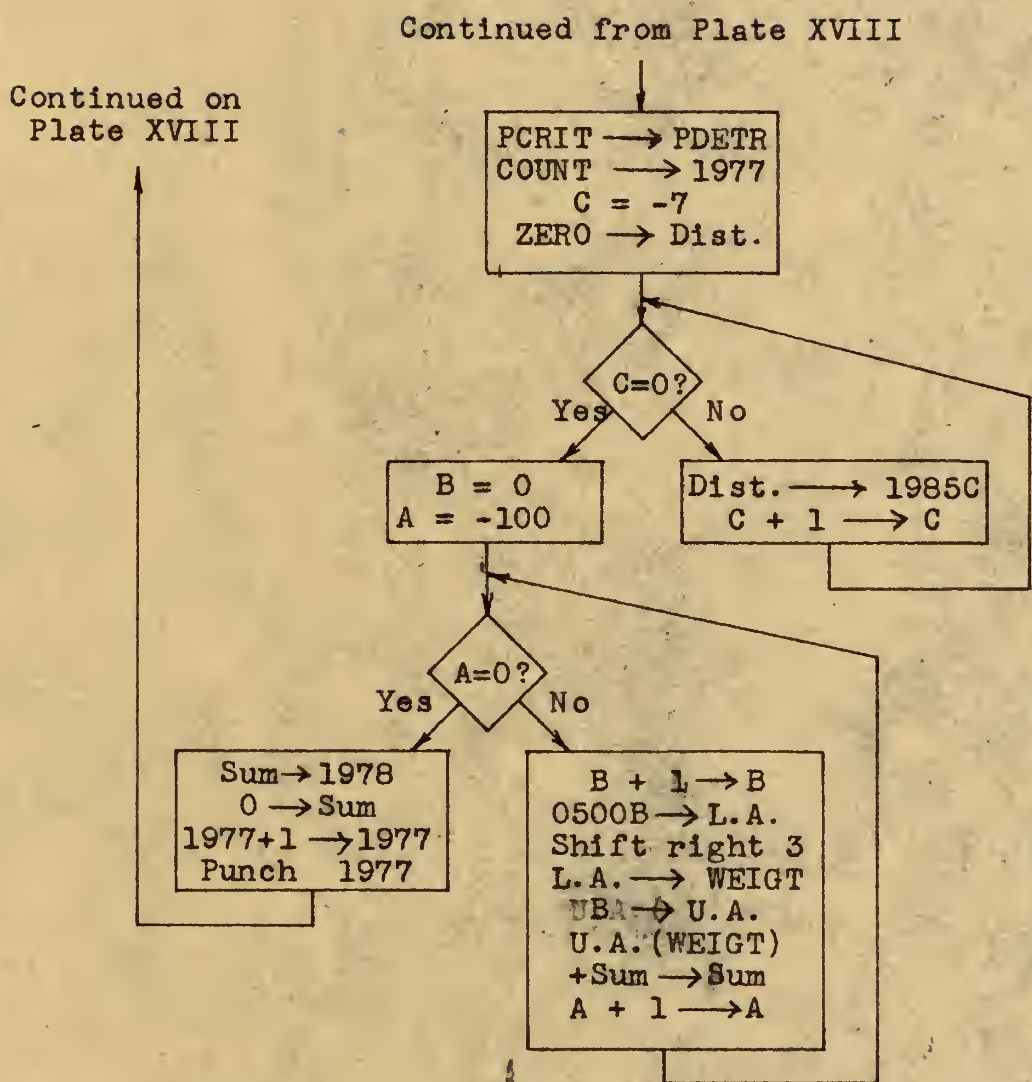
PLATE XVIII


Clear drum
PCRIT → PDETR
A = -100
Read card
A=0?
No
Yes
1502 → 0601A
A + 1 → A
Read card
B = 0
501B → U.A.
Shift left 7
Shift right 7
U.A. → 1500
UA=0?
No
Yes
B+1 → B
B → C
B-α → B
Continued from
Plate XIX
B-?
Yes
No
B+1 → B
0501B → L.A.
0501C → Dist.
Dist → 0501B
L.A. → 0501C
COUNT+1 → COUNT
PDETR-1 → L.A.
LA=0?
No
Yes
LA → PDETR
Continued on
Plate XIX

EXPLANATION OF PLATE XIX

Count-punch subroutine for transposition system.

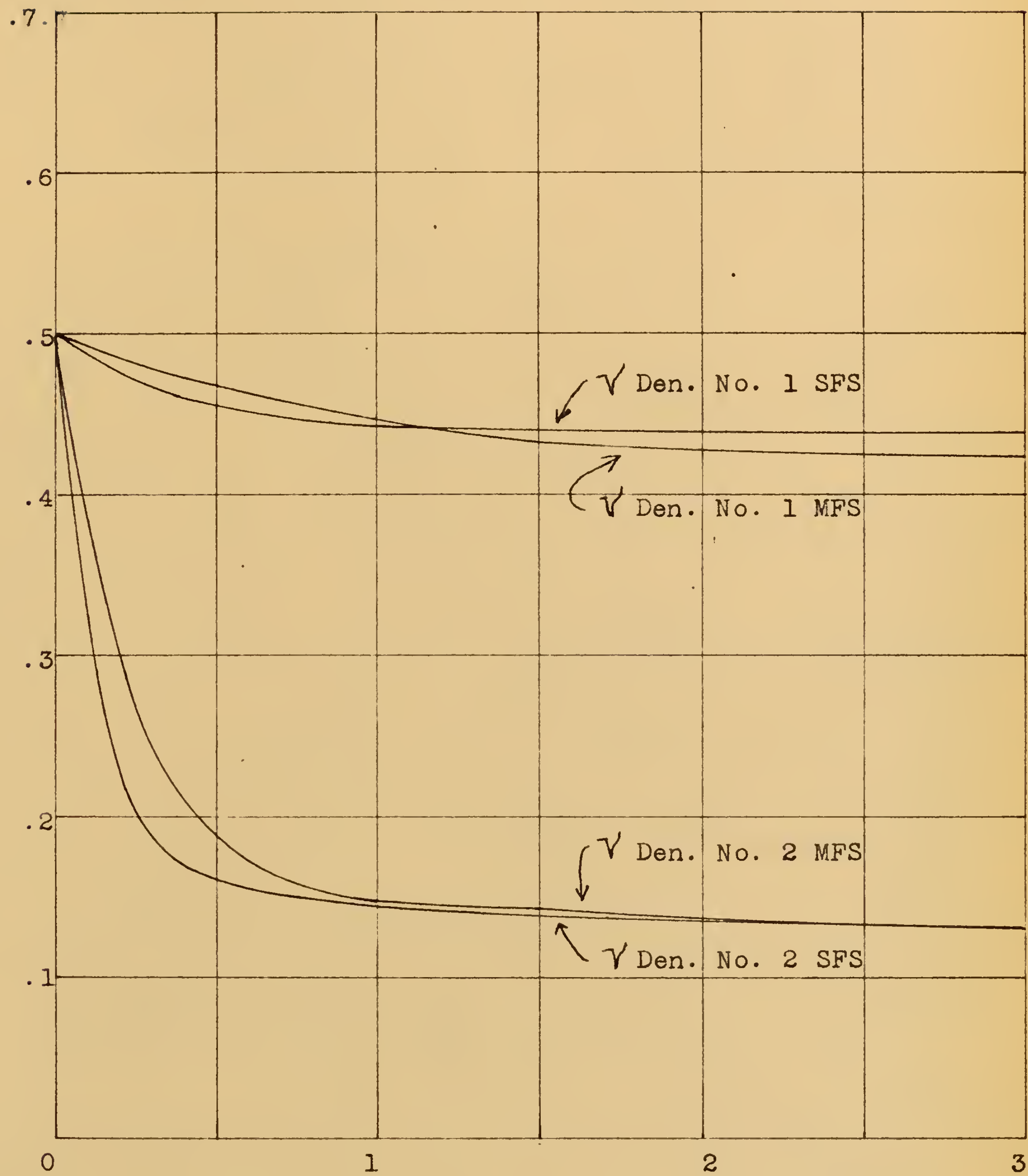
PLATE XIX


Continued from Plate XVIII
Continued on Plate XVIII
PCRIT → PDETR
COUNT → 1977
C = -7
ZERO → Dist.
C=0?
Yes
No
B = 0
A = -100
Dist. → 1985C
C + 1 → C
A=0?
Yes
No
Sum → 1978
0 → Sum
1977+1 → 1977
Punch 1977
B + 1 → B
0500B → L.A.
Shift right 3
L.A. → WEIGT
UBA → U.A.
U.A. (WEIGT)
+Sum → Sum
A + 1 → A

EXPLANATION OF PLATE XX

√ (in file population) versus interrogations (in file populations) for the single feed and multiple feed systems confronted with density No. 1 and No. 2.

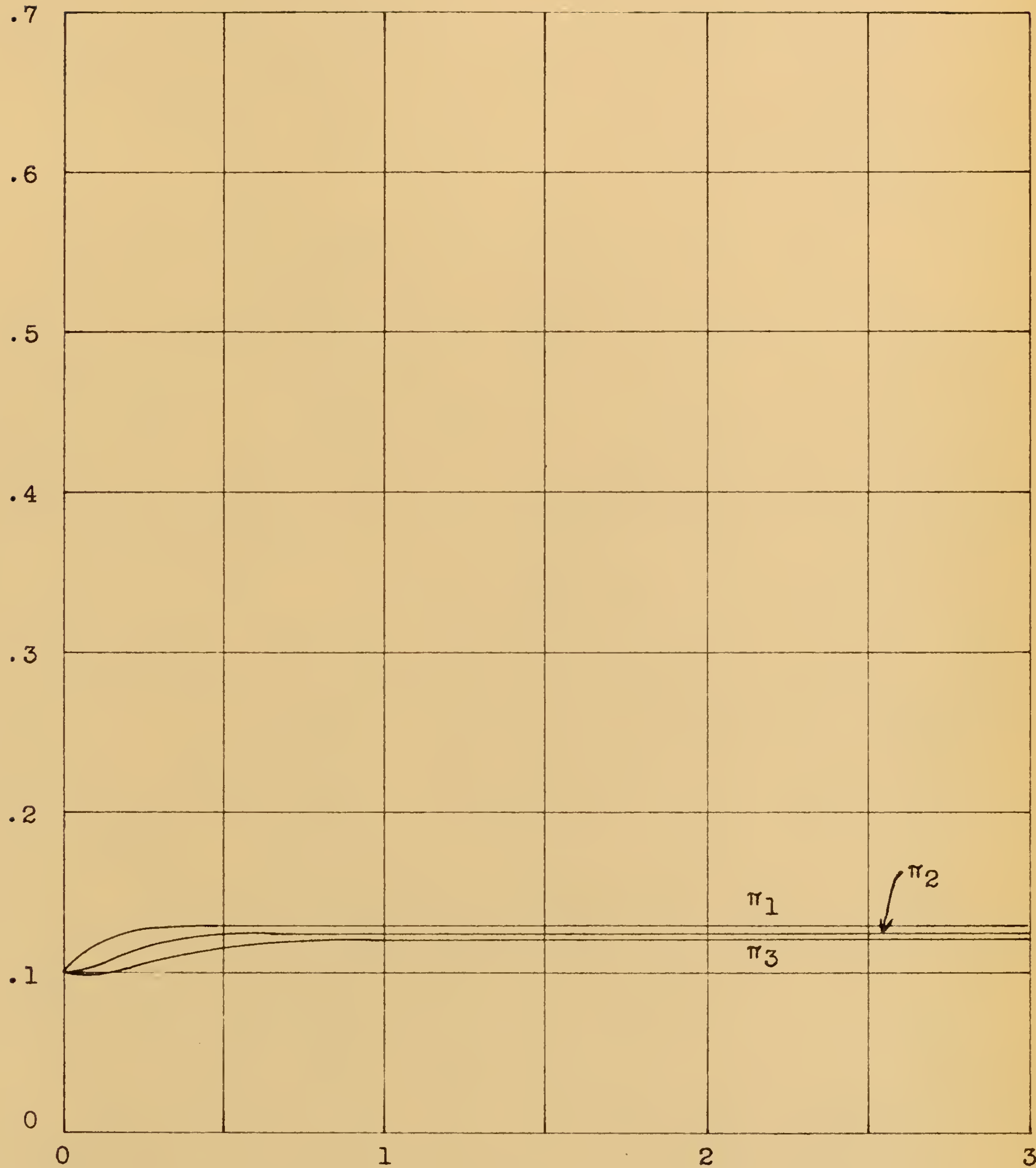
PLATE XX



EXPLANATION OF PLATE XXI

π_1, π_2, and π_3 (in file population) versus interrogations (in file populations) for the single feed system confronted with density No. 1.

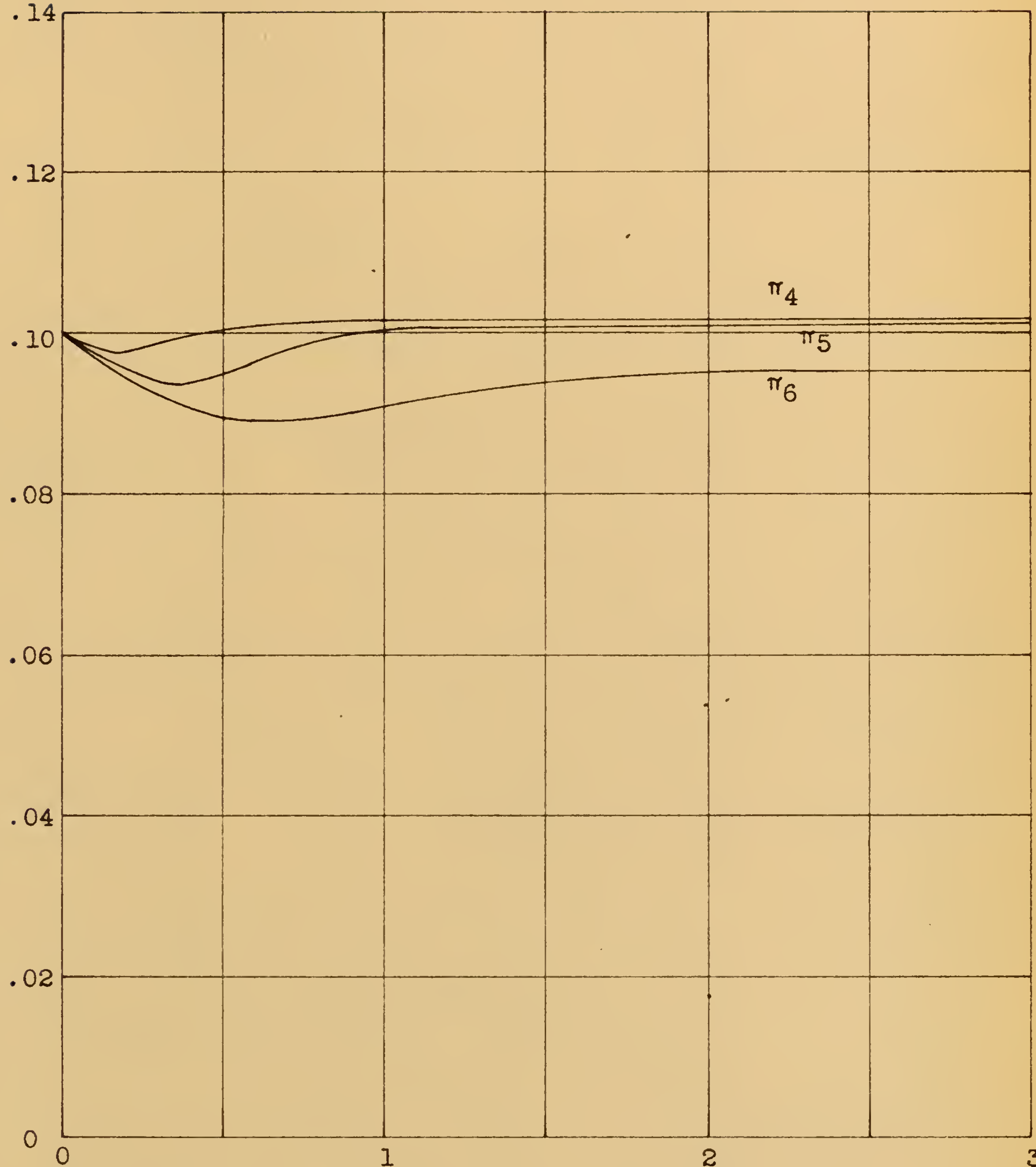
PLATE XXI



EXPLANATION OF PLATE XXII

π_4, π_5, and π_6 (in file population) versus interrogations (in file populations) for the single feed system confronted with density No. 1.

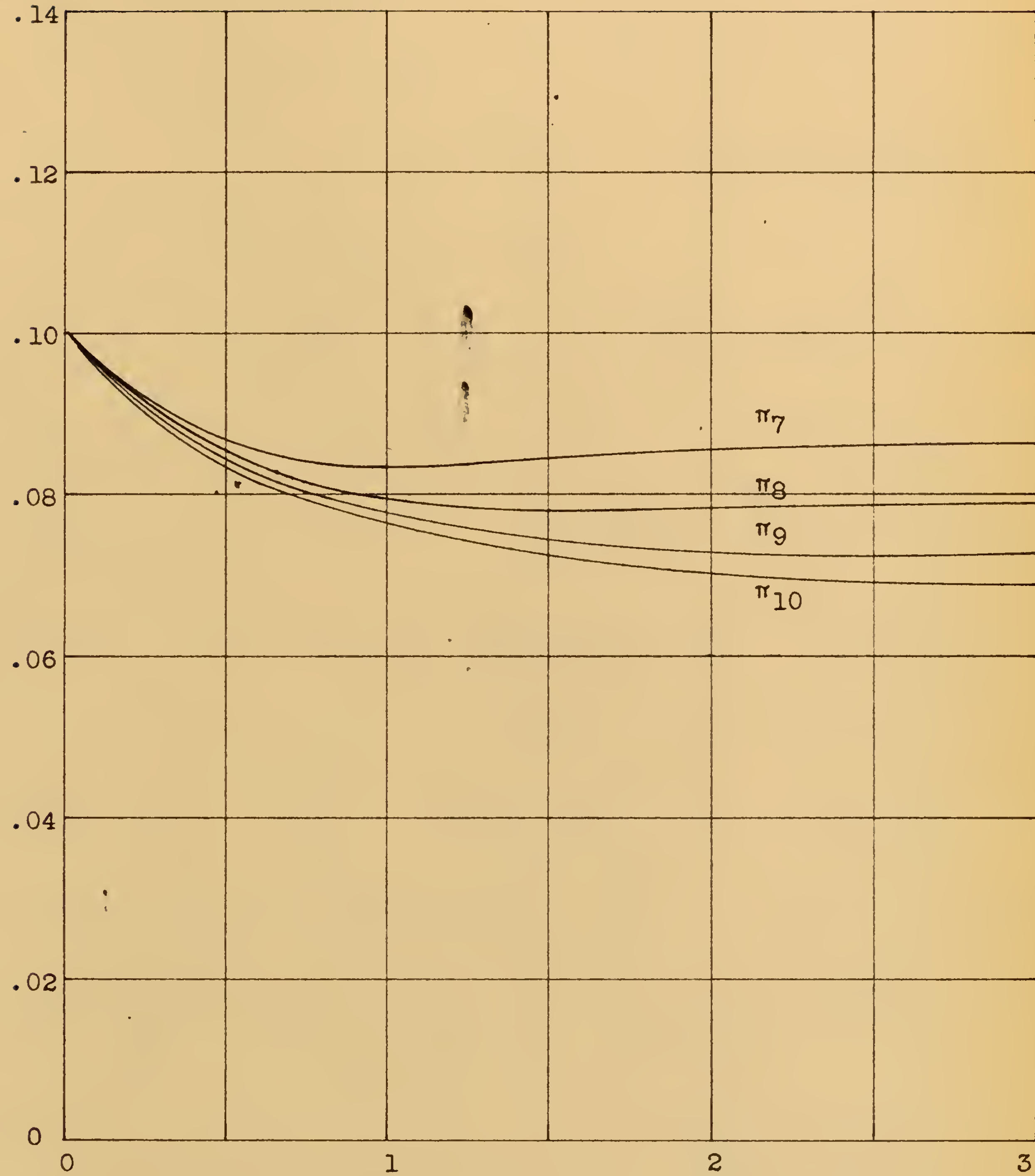
PLATE XXII



EXPLANATION OF PLATE XXIII

π_7, π_8, π_9, and π_{10} (in file population) versus interrogations (in file populations) for the single feed system confronted with density No. 1.

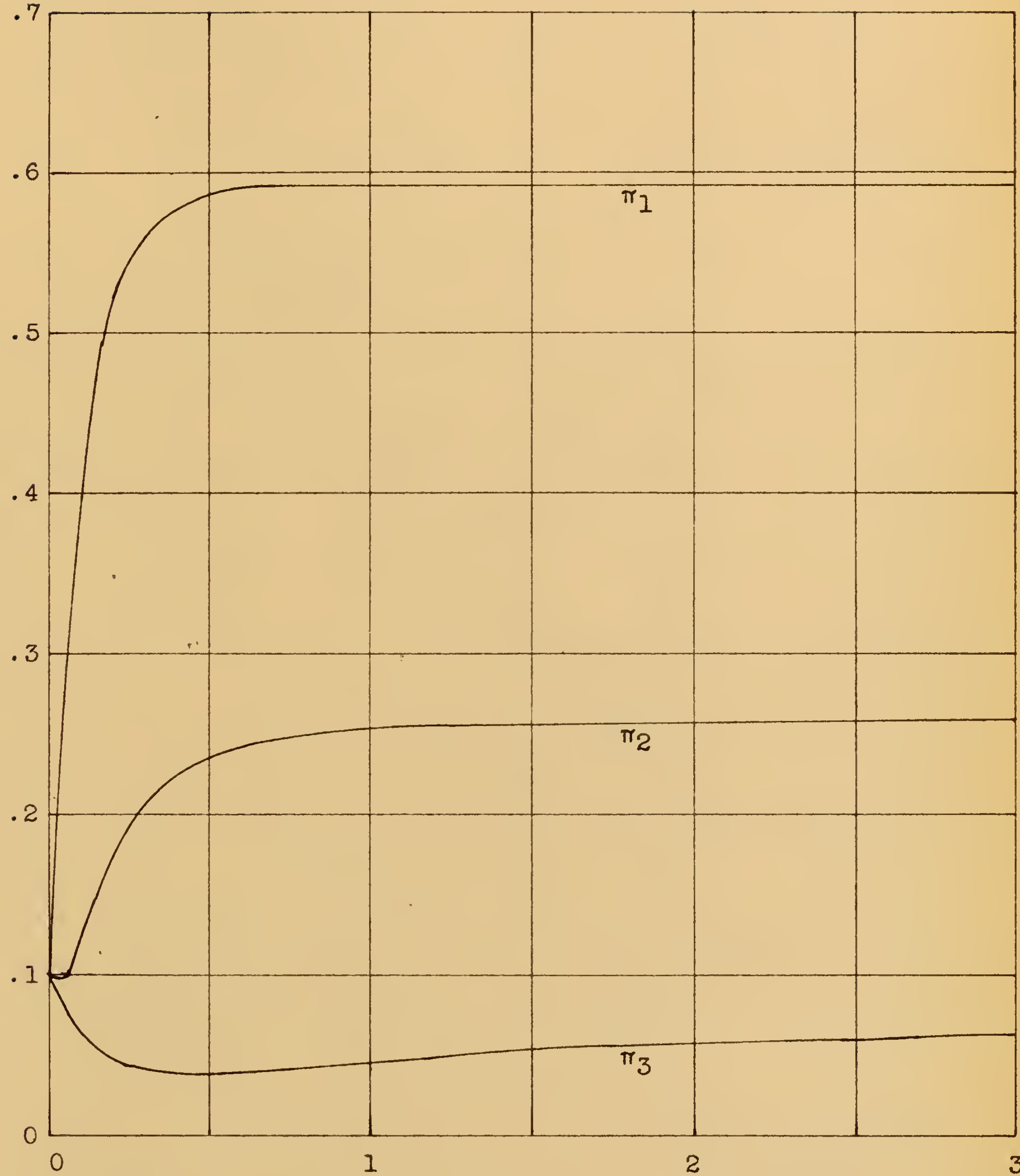
PLATE XXIII



EXPLANATION OF PLATE XXIV

π_1, π_2, and π_3 (in file population) versus interrogations (in file populations) for the single feed system confronted with density No. 2.

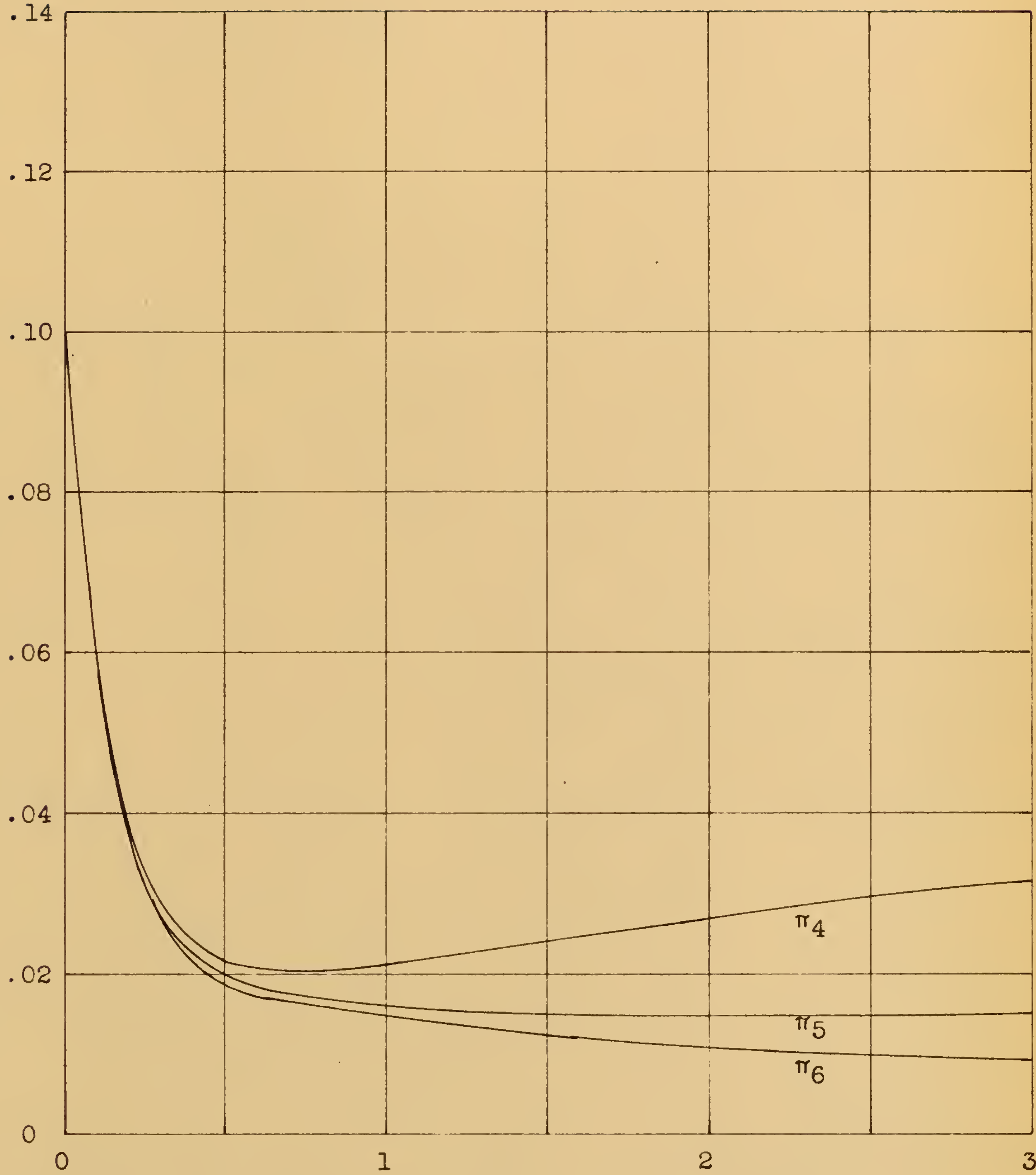
PLATE XXIV



EXPLANATION OF PLATE XXV

π_4, π_5, and π_6 (in file population) versus interrogations (in file populations) for the single feed system confronted with density No. 2.

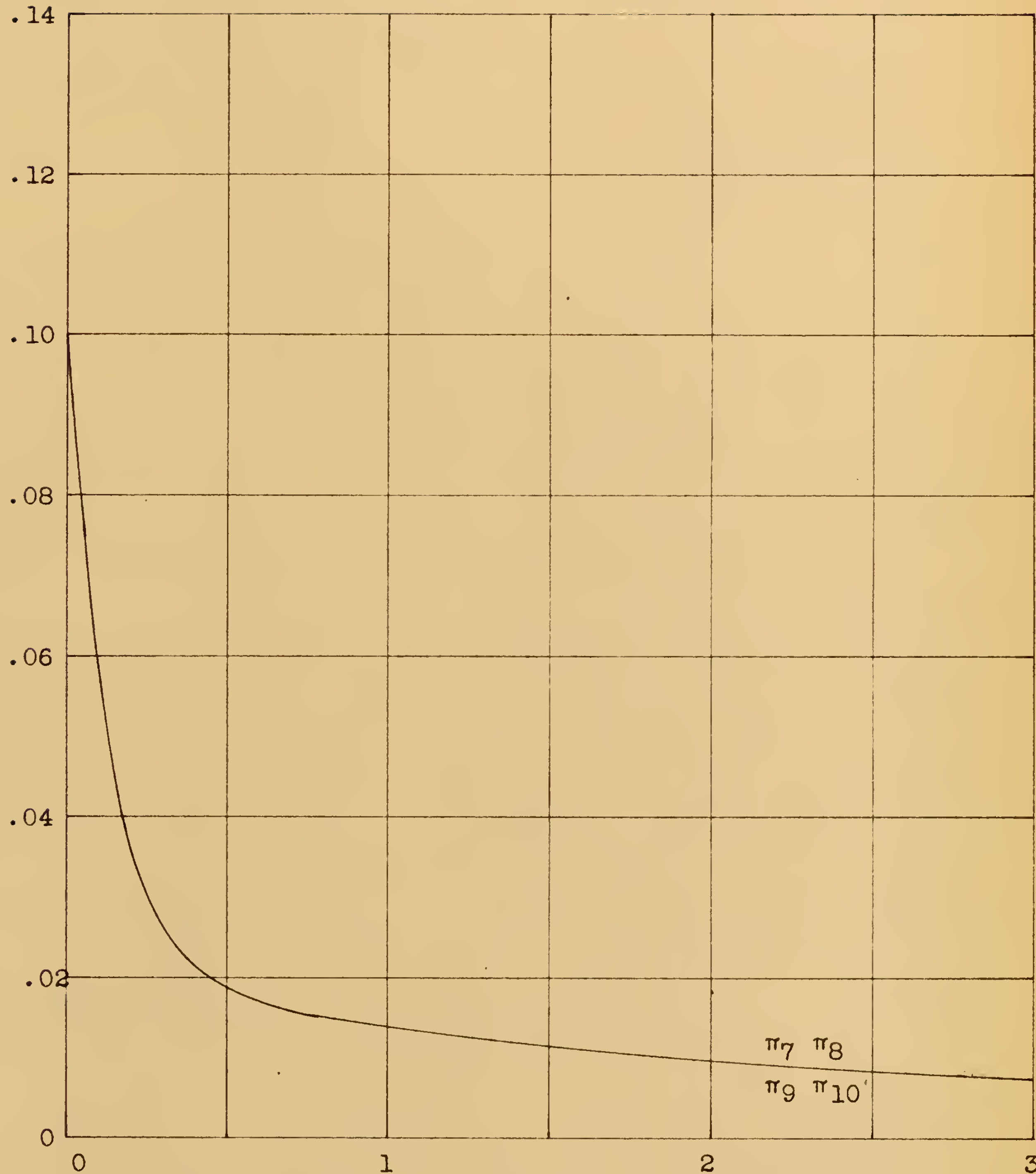
PLATE XXV



EXPLANATION OF PLATE XXVI

π_7, π_8, π_9, and π_{10} (in file population) versus interrogations (in file populations) for the single feed system confronted with density No. 2.

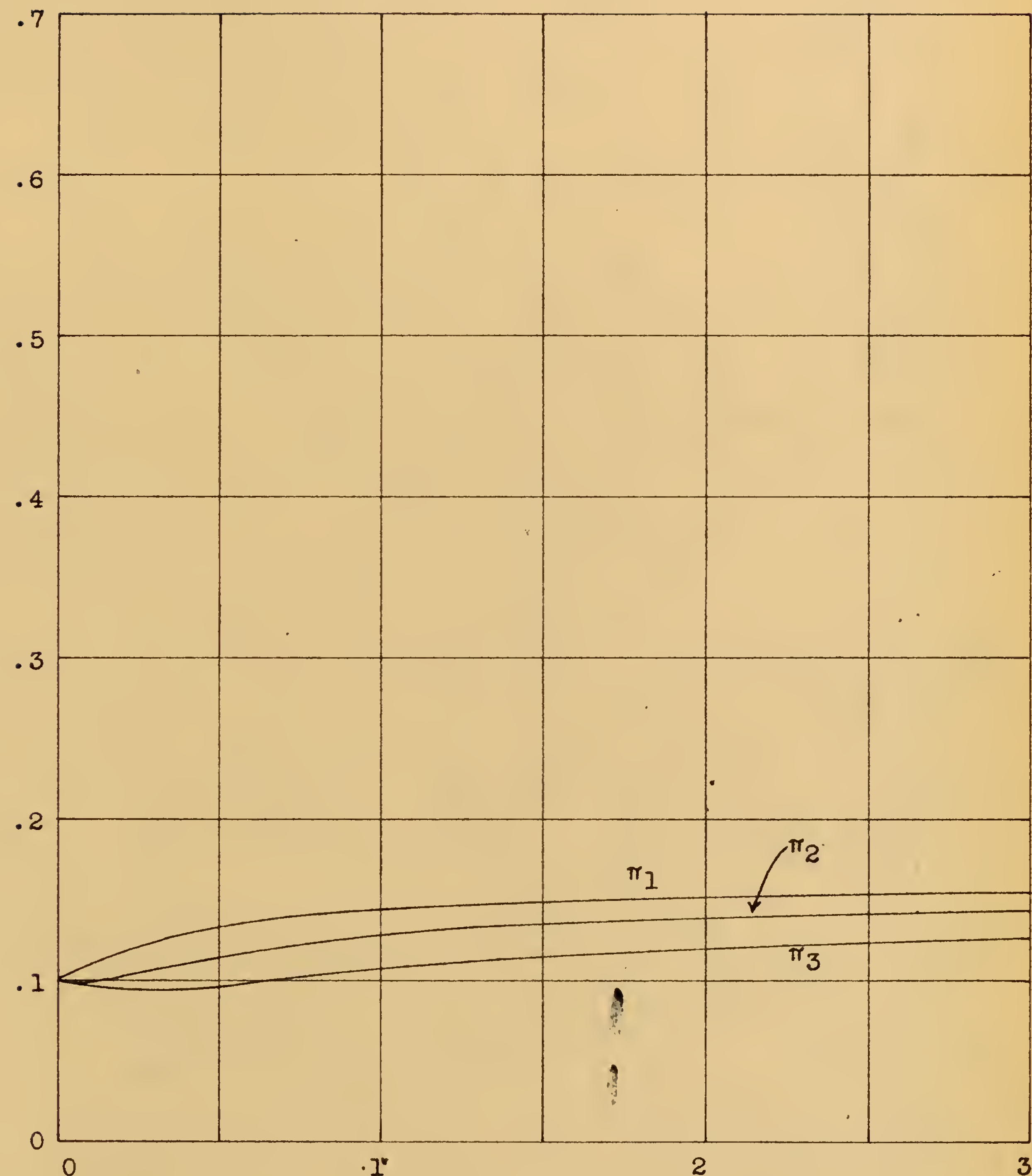
PLATE XXVI



EXPLANATION OF PLATE XXVII

π_1, π_2, and π_3 (in file population) versus interrogations (in file populations) for the multiple feed system confronted with density No. 1.

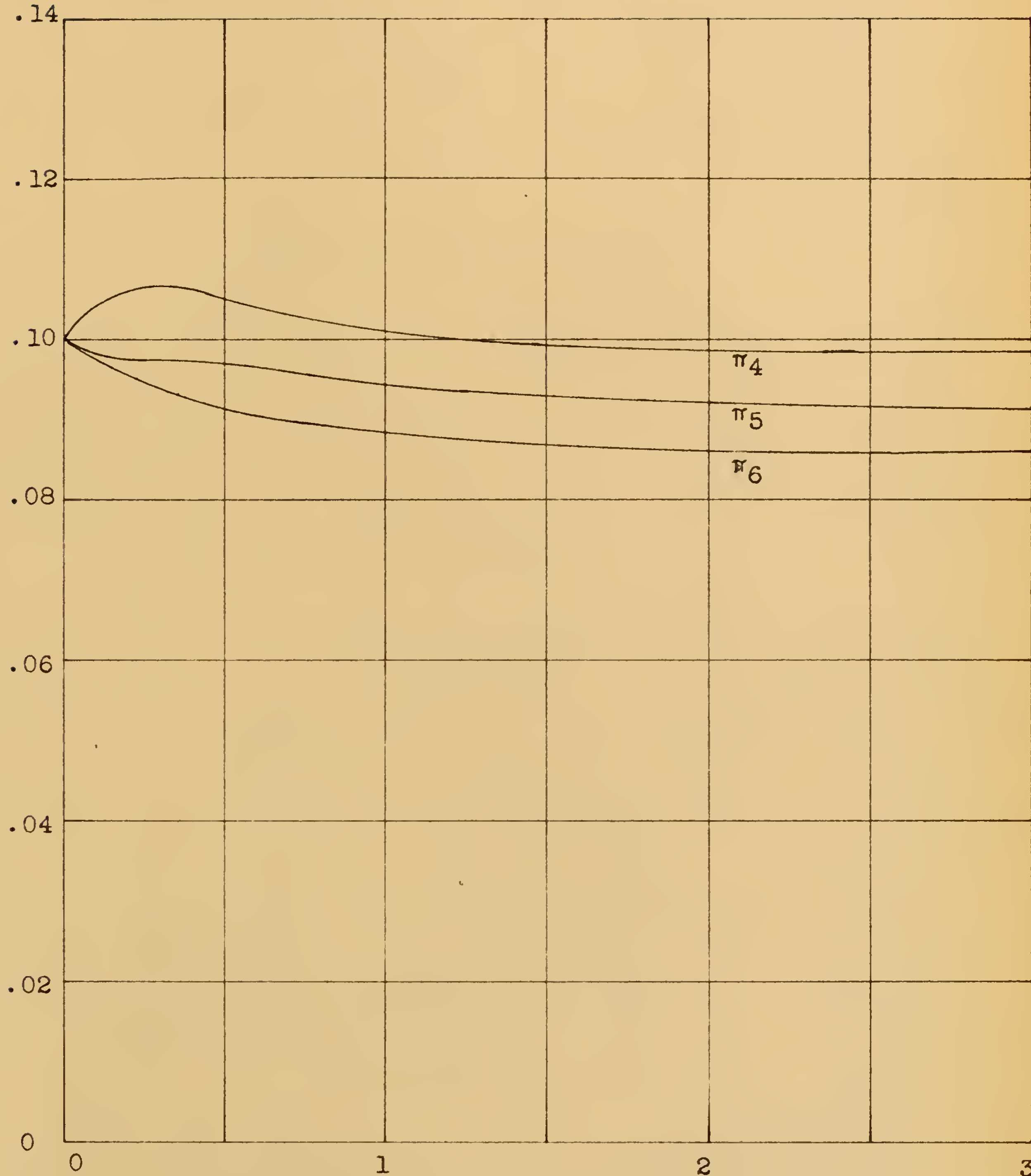
PLATE XXVII



EXPLANATION OF PLATE XXVIII

π_4, π_5, and π_6 (in file population) versus interrogations (in file populations) for the multiple feed system confronted with density No. 1.

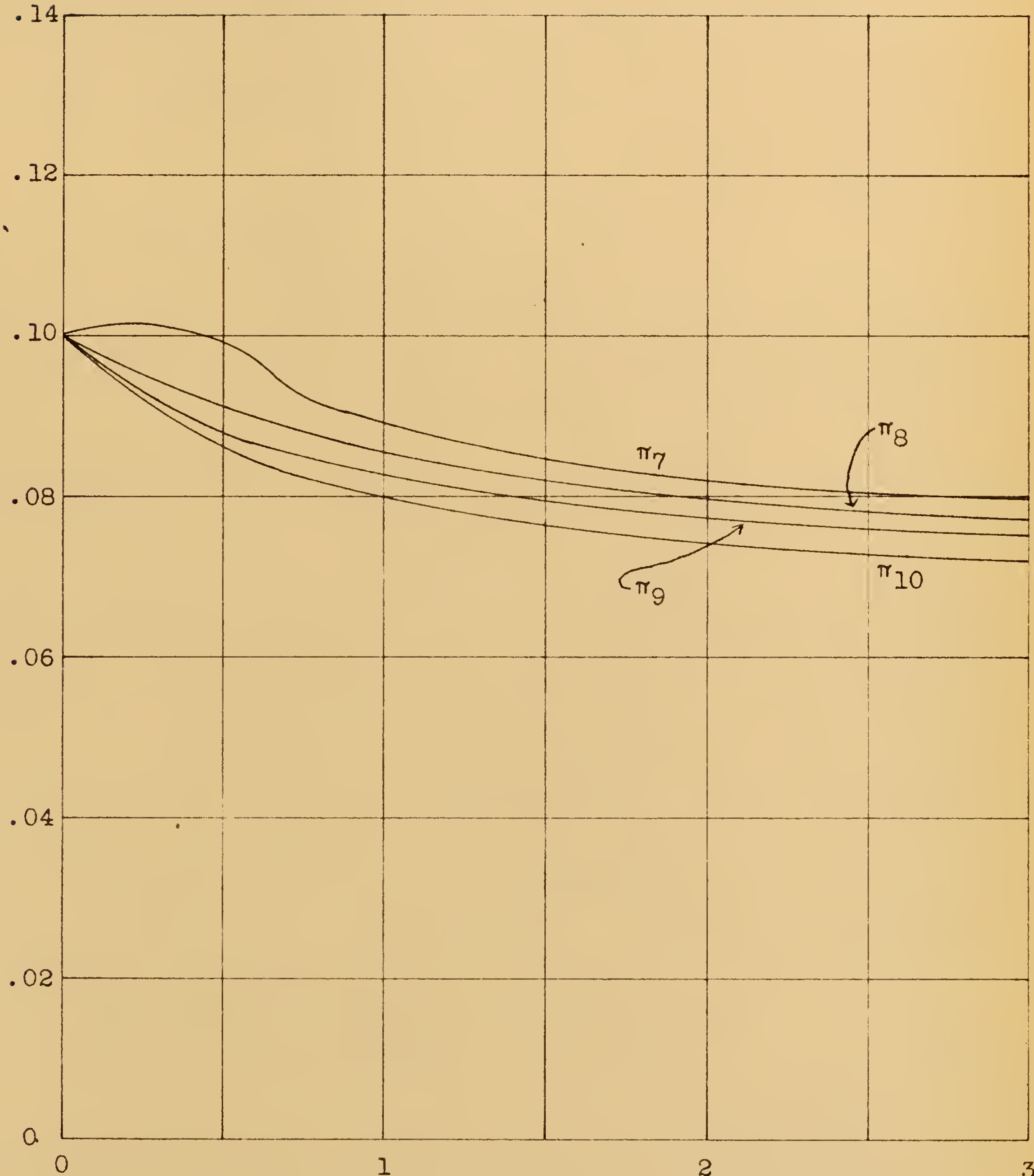
PLATE XXVIII



EXPLANATION OF PLATE XXIX

π_7, π_8, π_9, and π_{10} (in file population) versus interrogations (in file populations) for the multiple feed system confronted with density No. 1.

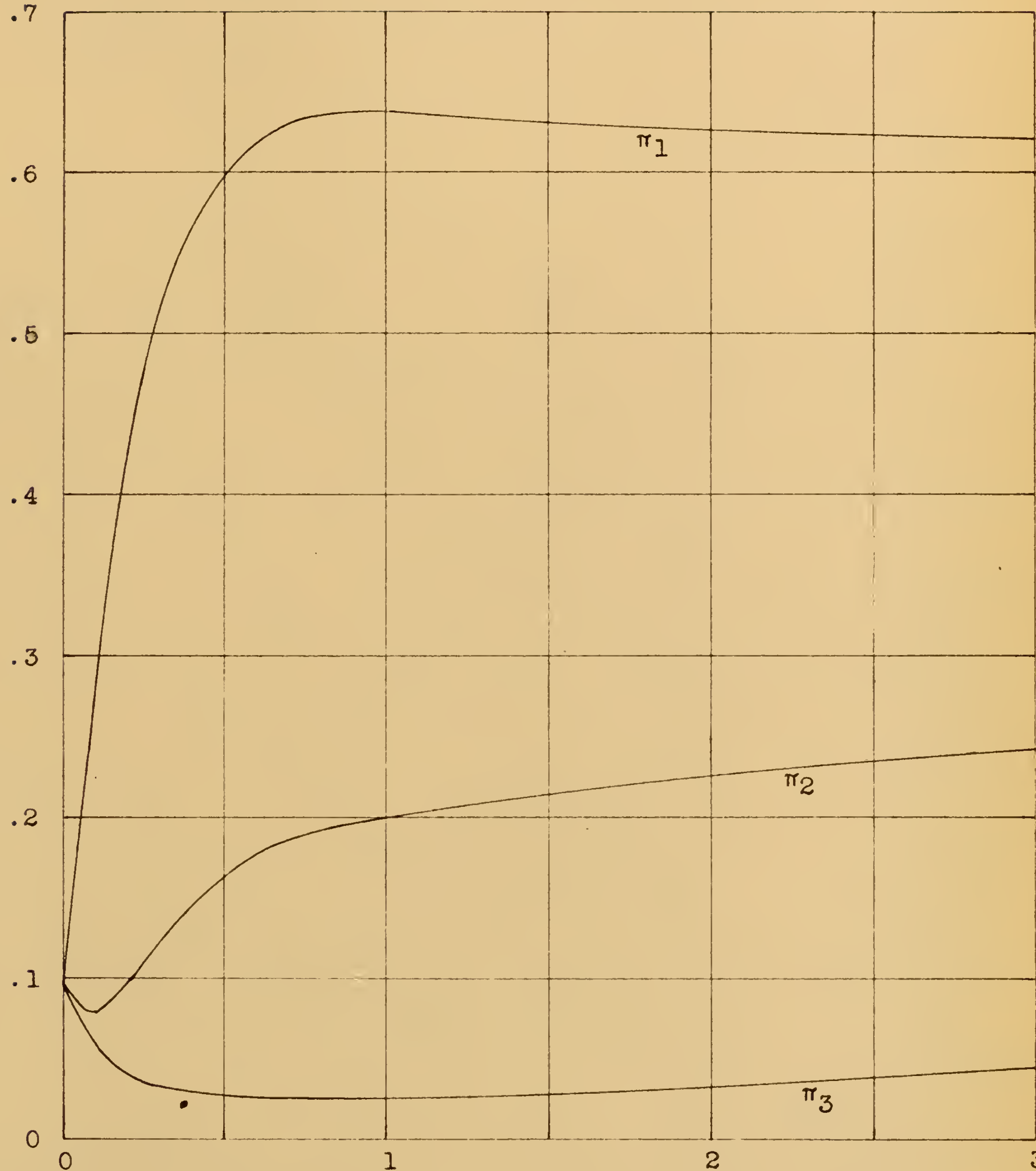
PLATE XXIX



EXPLANATION OF PLATE XXX

π_1, π_2, and π_3 (in file population) versus interrogations (in file populations) for the multiple feed system confronted with density No. 2.

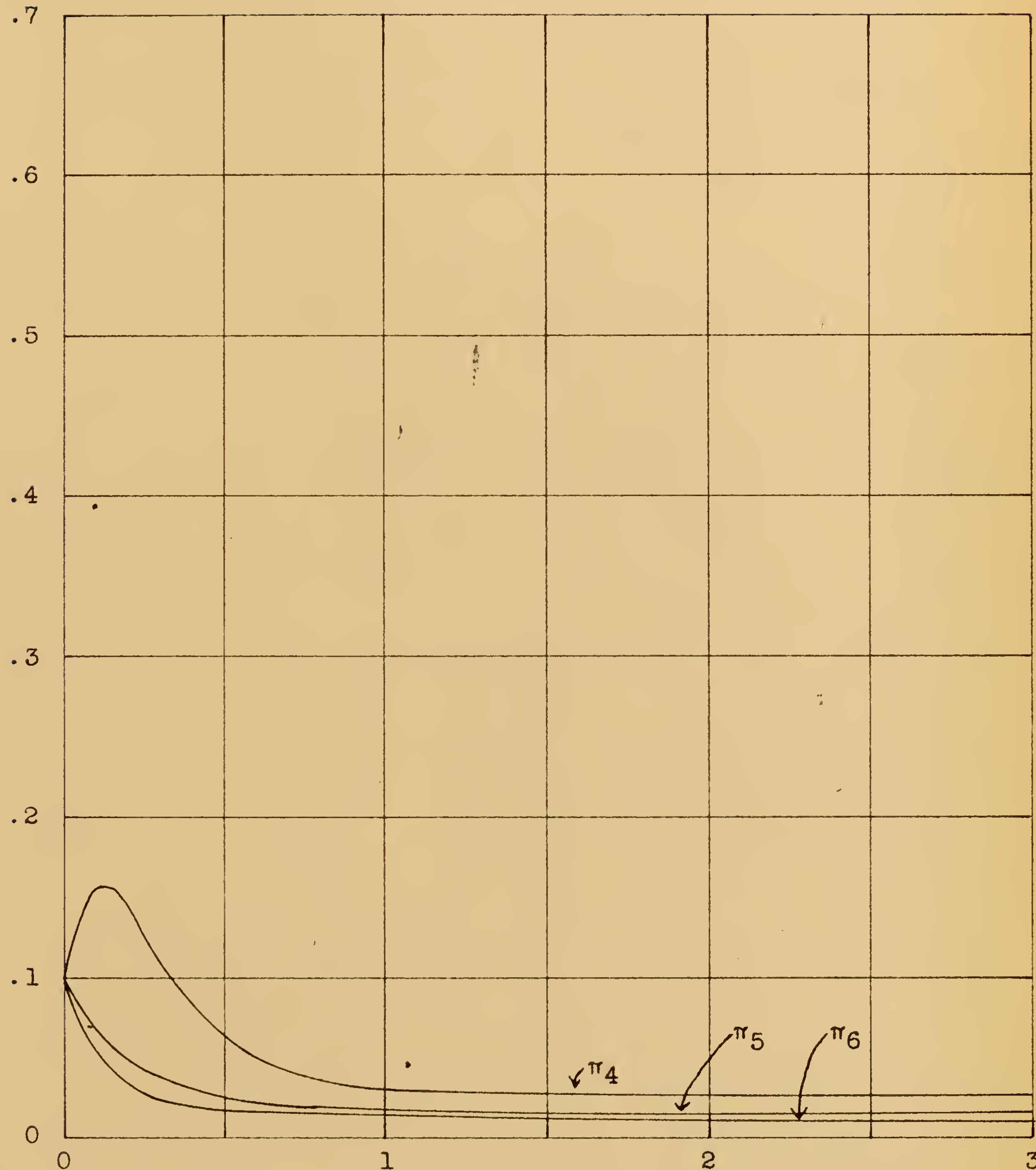
PLATE XXX



EXPLANATION OF PLATE XXXI

π_4, π_5, and π_6 (in file population) versus interrogations (in file populations) for the multiple feed system confronted with density No. 2.

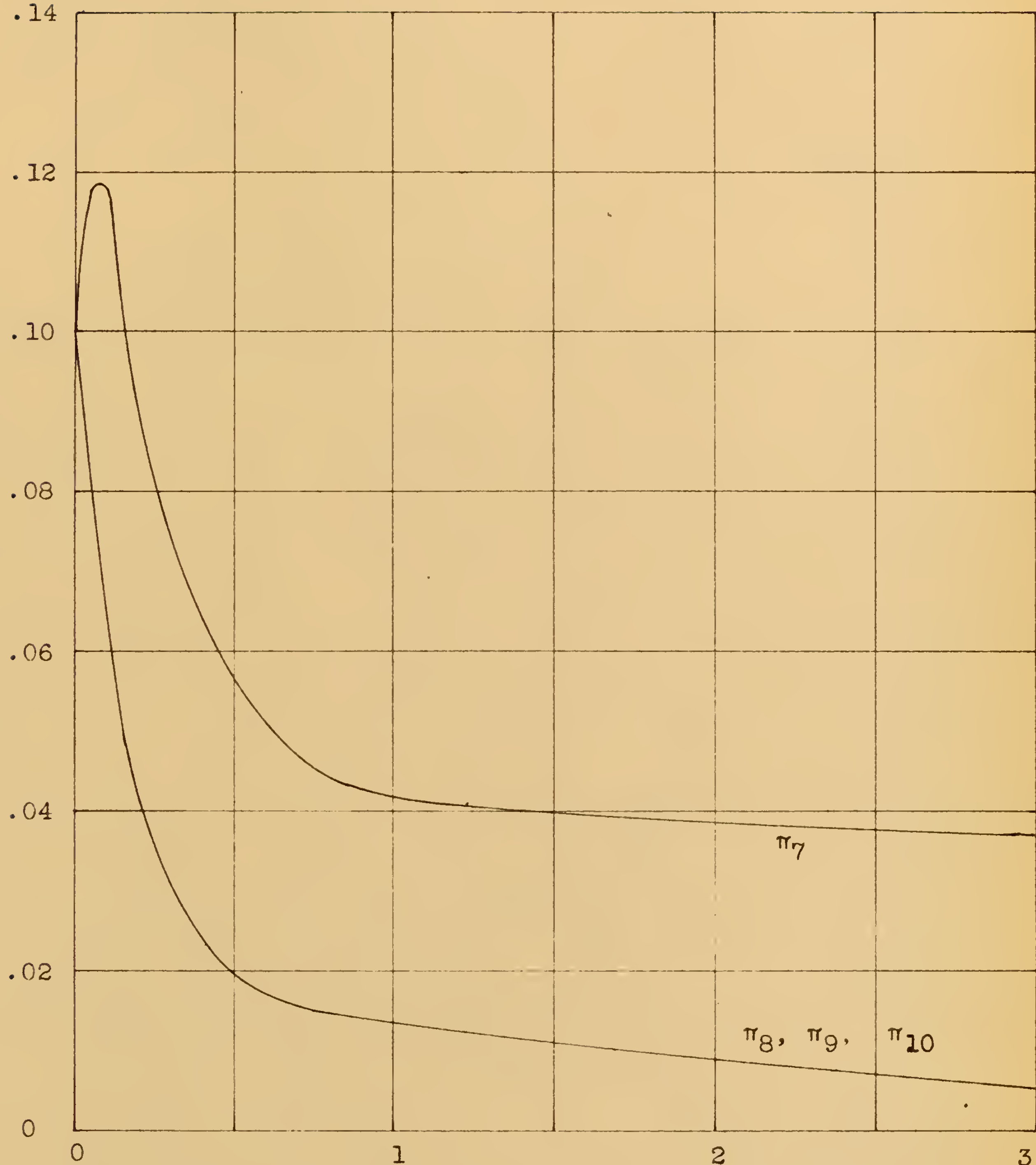
PLATE XXXI



EXPLATION OF PLATE XXXII

π_7, π_8, π_9, and π_{10} (in file population) versus interrogations (in file populations) for the multiple feed system confronted with density No. 2.

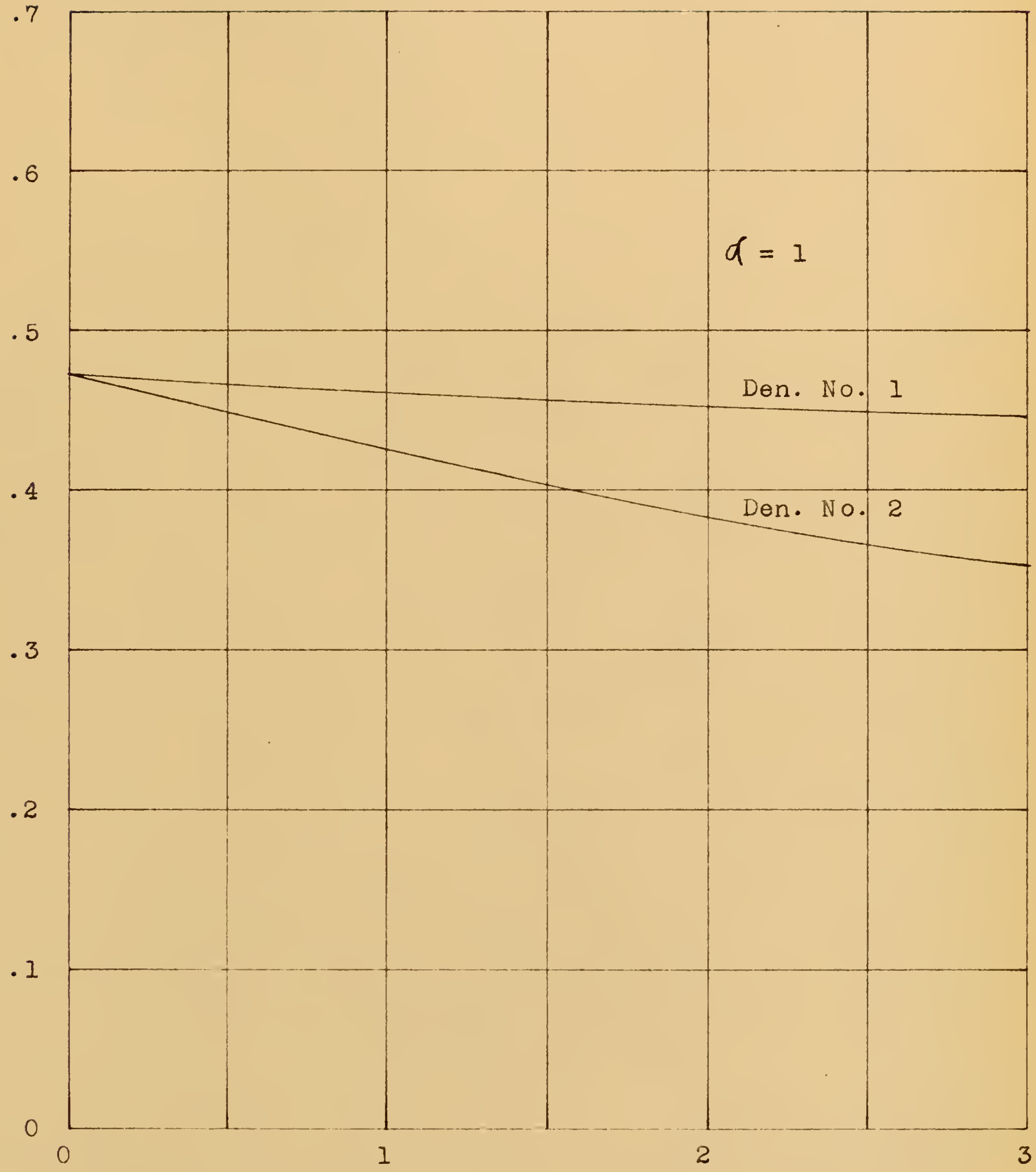
PLATE XXXII



EXPLANATION OF PLATE XXXIII

γ (in file population) versus interrogations (in file populations) for the transposition system, $\alpha = 1$, confronted with densities No. 1 and No. 2.

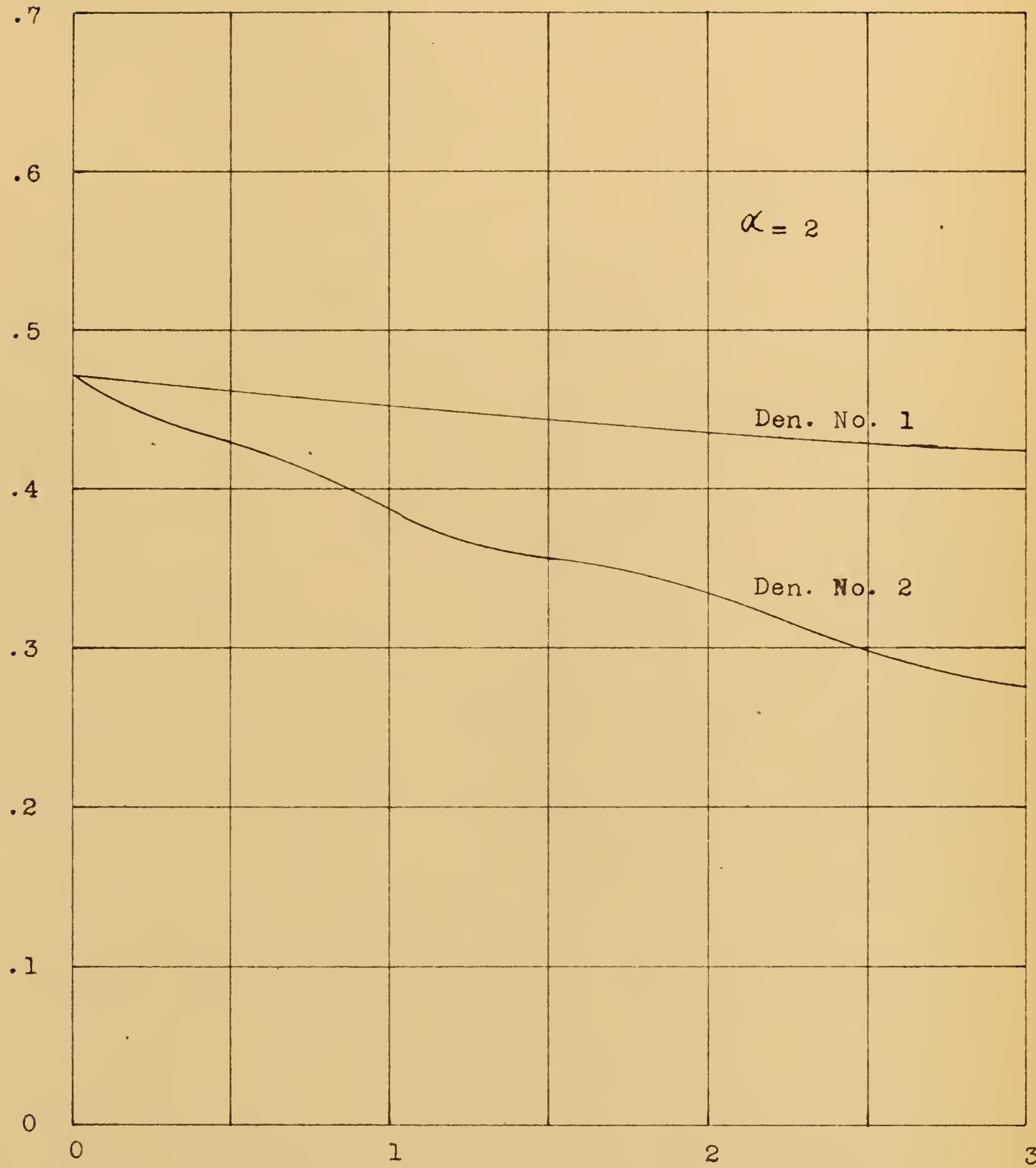
PLATE XXXIII



EXPLANATION OF PLATE XXXIV

γ (in file population) versus interrogations (in file populations) for the transposition system, $\alpha = 2$, confronted with densities No. 1 and No. 2.

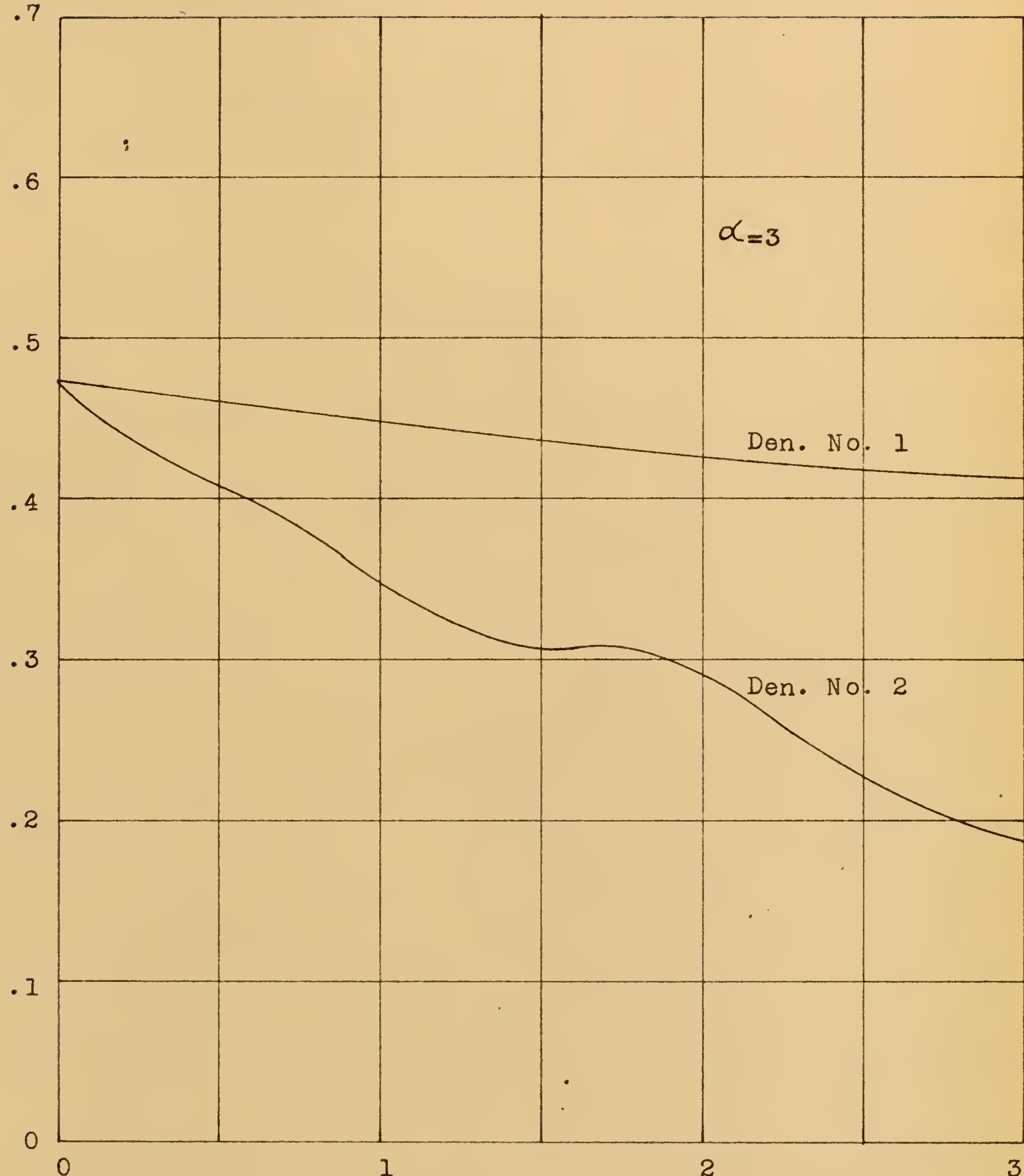
PLATE XXXIV



EXPLANATION OF PLATE XXXV

γ (in file population) versus interrogations (in file populations) for the transposition system, $\alpha = 3$, confronted with densities No. 1 and No. 2.

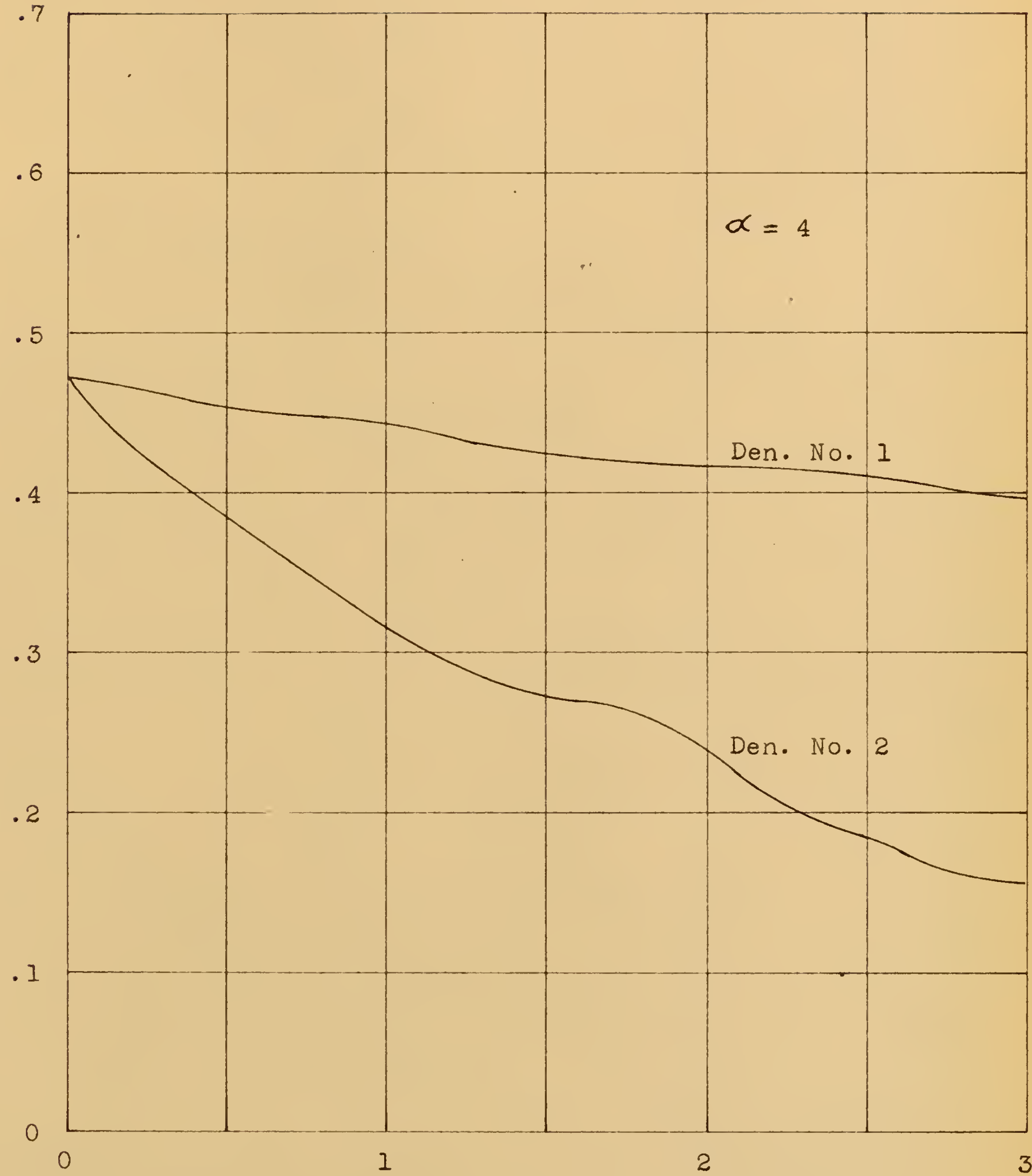
PLATE XXXV



EXPLANATION OF PLATE XXXVI

γ (in file population) versus interrogations (in file populations) for the transposition system, $\alpha = 4$, confronted with densities No. 1 and No. 2.

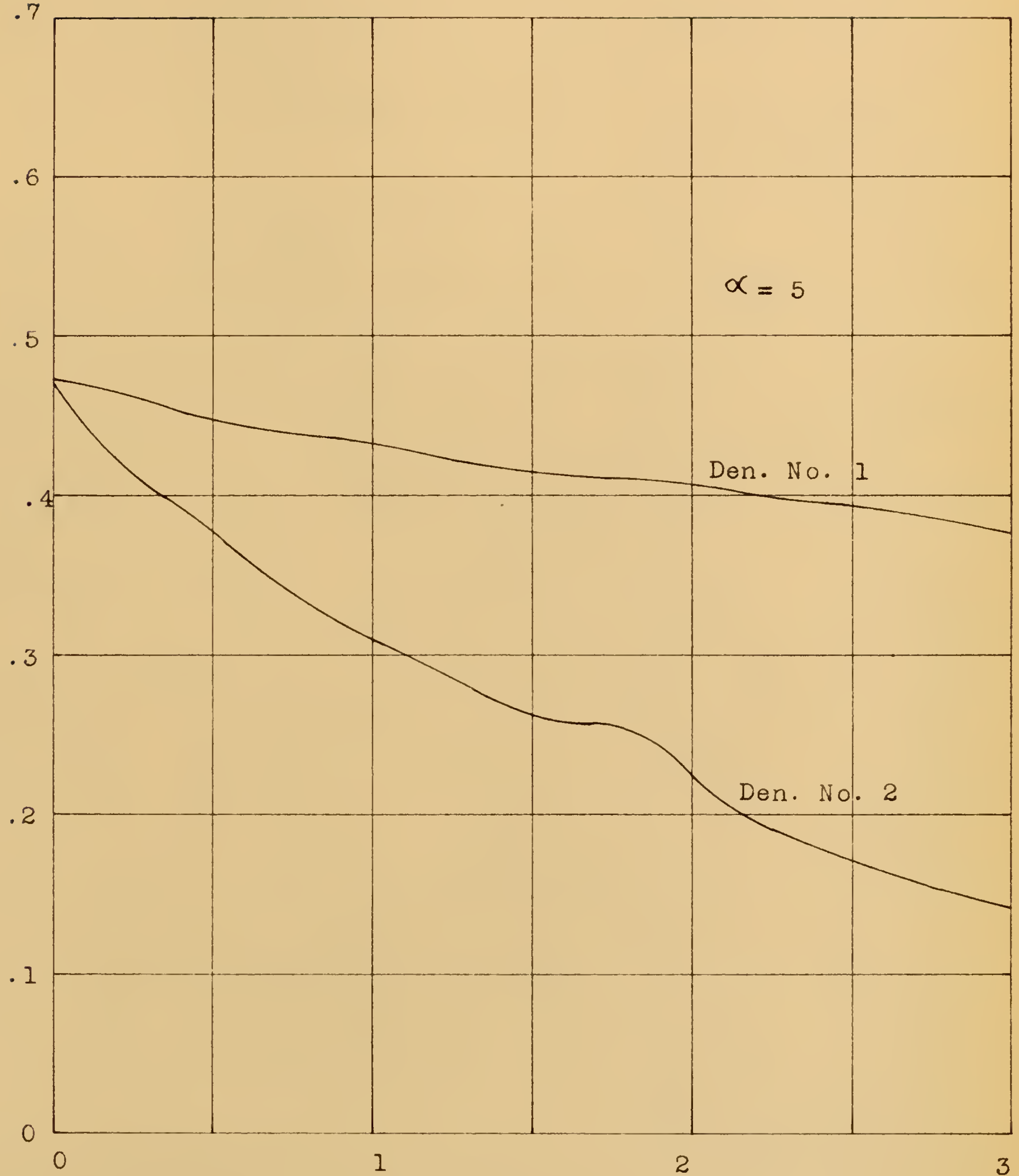
PLATE XXXVI



EXPLANATION OF PLATE XXXVII

γ (in file population) versus interrogations (in file populations) for the transposition system, $\alpha = 5$, confronted with densities No. 1 and No. 2.

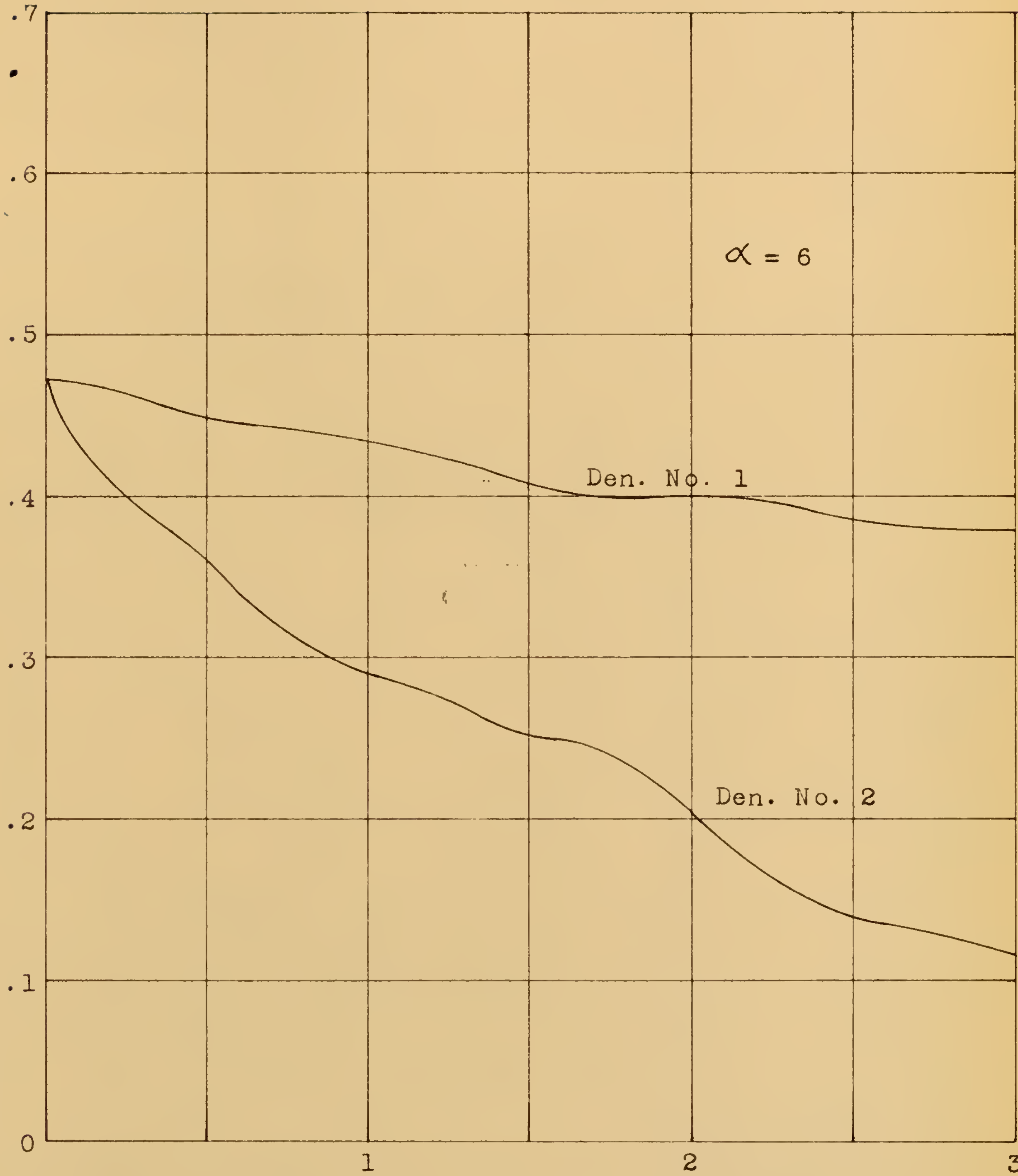
PLATE XXXVII



EXPLANATION OF PLATE XXXVIII

γ (in file population) versus interrogations (in file populations) for the transposition system, $\alpha = 6$, confronted with densities No. 1 and No. 2.

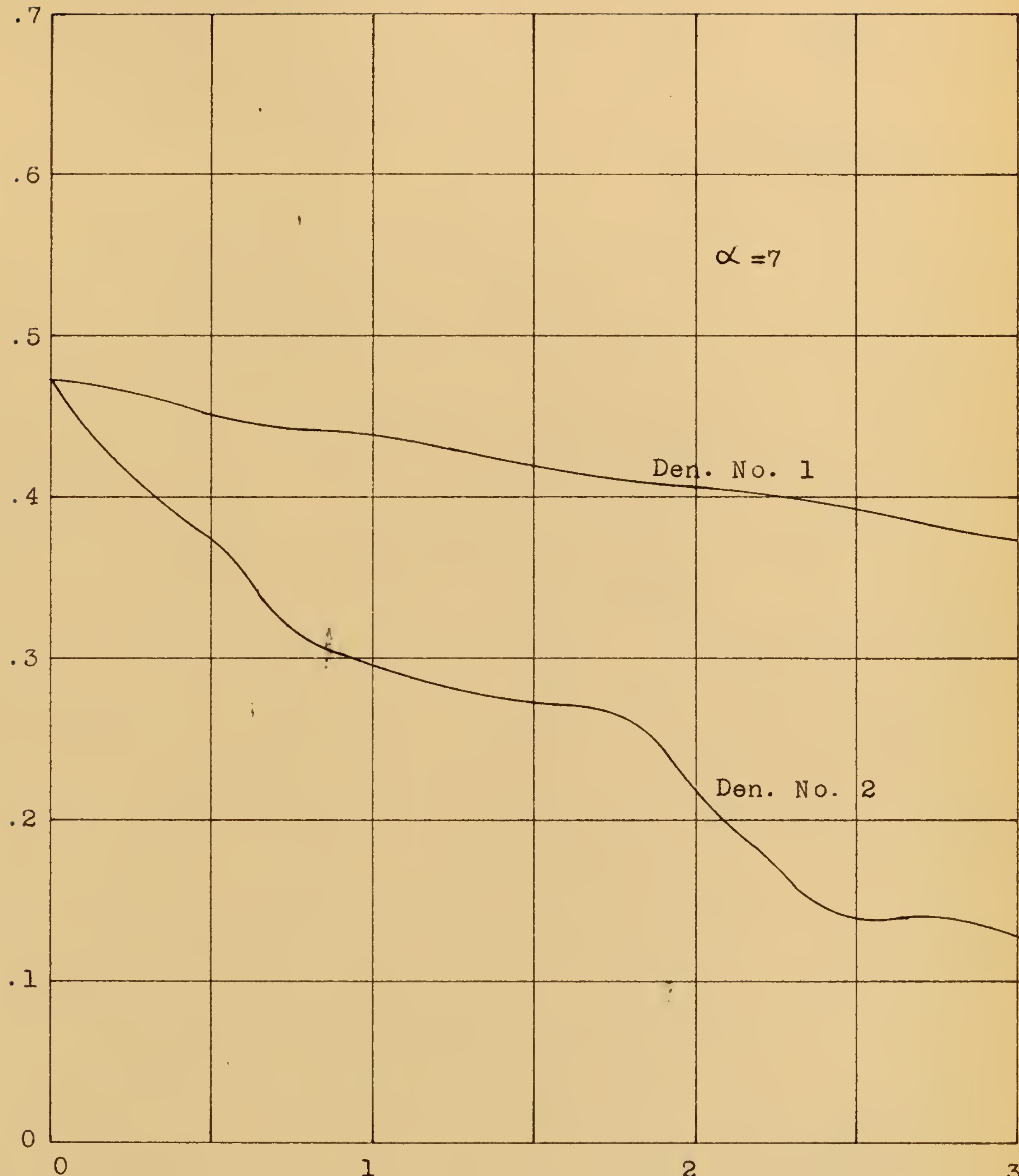
PLATE XXXVIII



EXPLANATION OF PLATE XXXIX

γ (in file population) versus interrogations (in file populations) for the transposition system, $\alpha = 7$, confronted with densities No. 1 and No. 2.

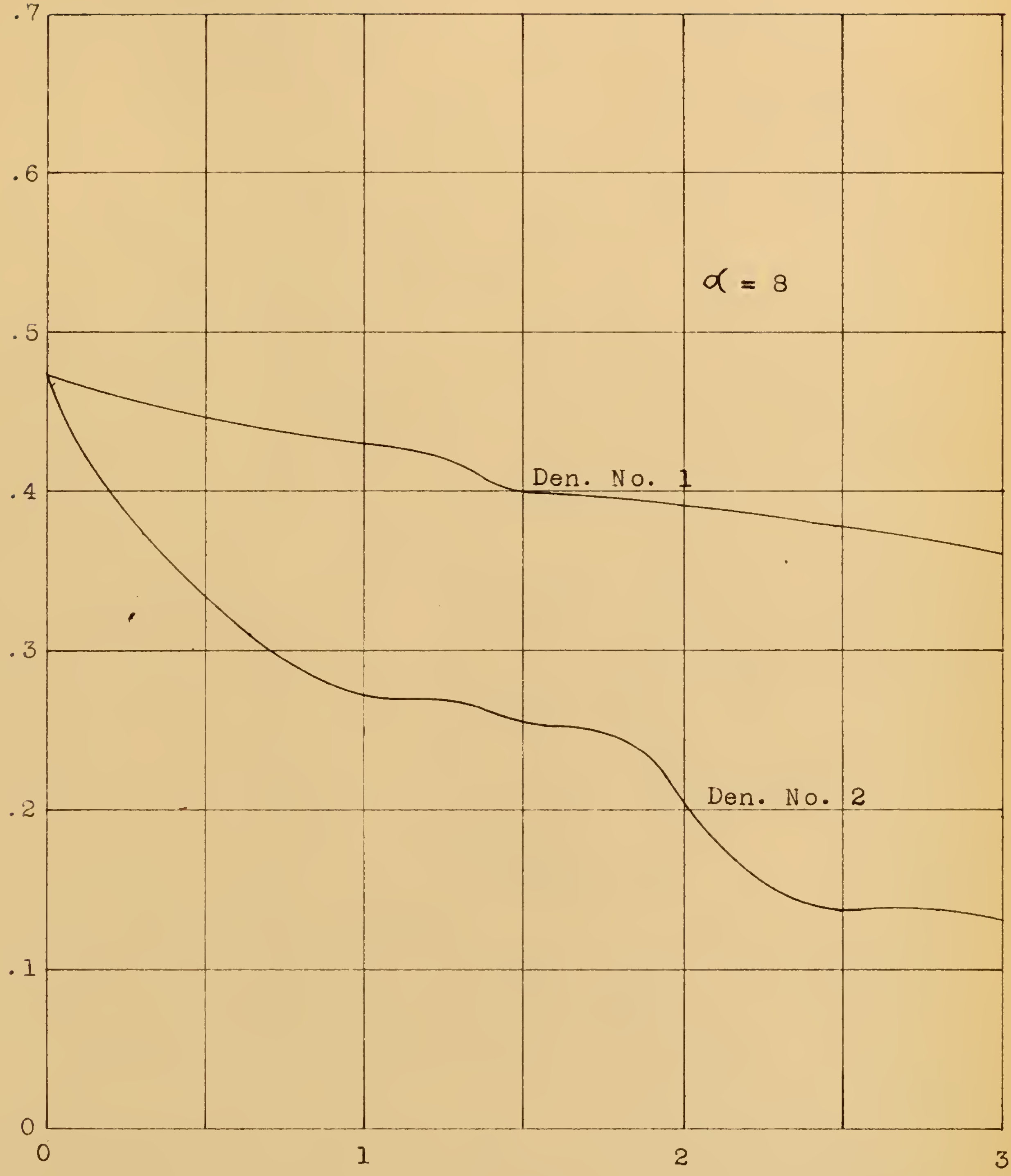
PLATE XXXIX



EXPLANATION OF PLATE XL

γ (in file population) versus interrogations (in file populations) for the transposition system, $\alpha = 8$, confronted with densities No. 1 and No. 2.

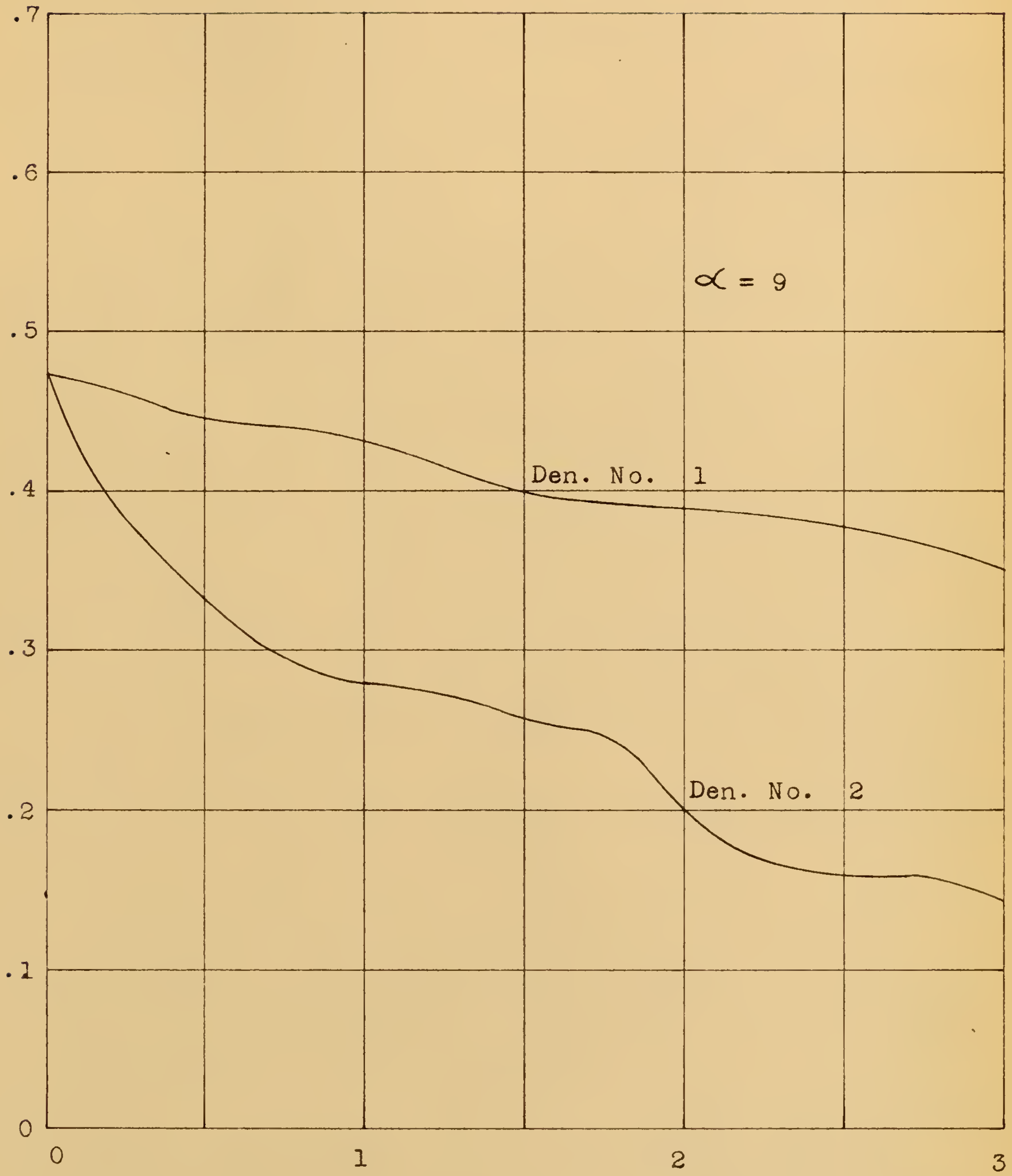
PLATE XL



EXPLANATION OF PLATE XLI

γ (in file population) versus interrogations (in file populations) for the transposition system, $\alpha = 9$, confronted with densities No. 1 and No. 2.

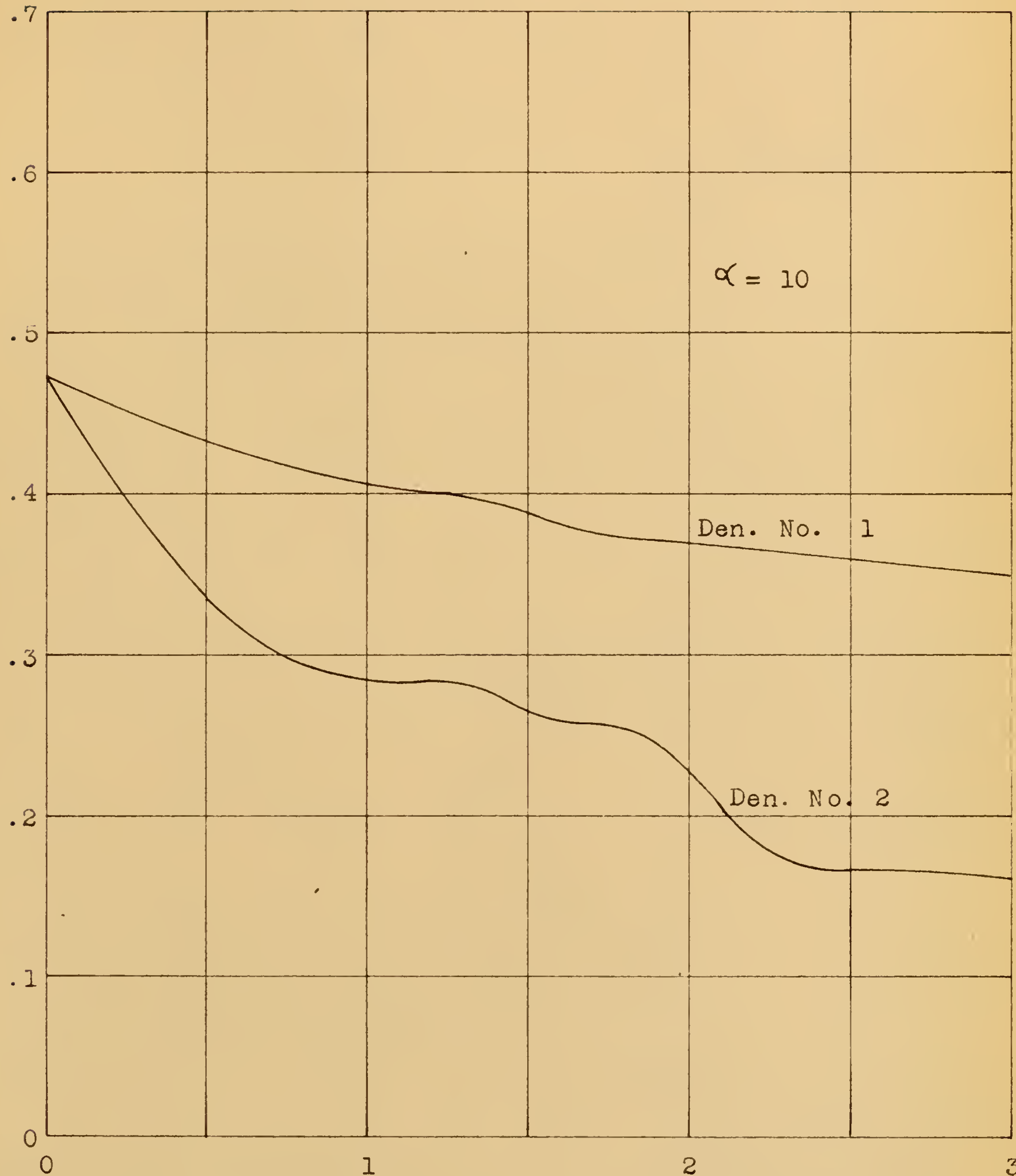
PLATE XLI



EXPLANATION OF PLATE XLII

γ (in file population) versus interrogations (in file populations) for the transposition system, $\alpha = 10$, confronted with densities No. 1 and No. 2.

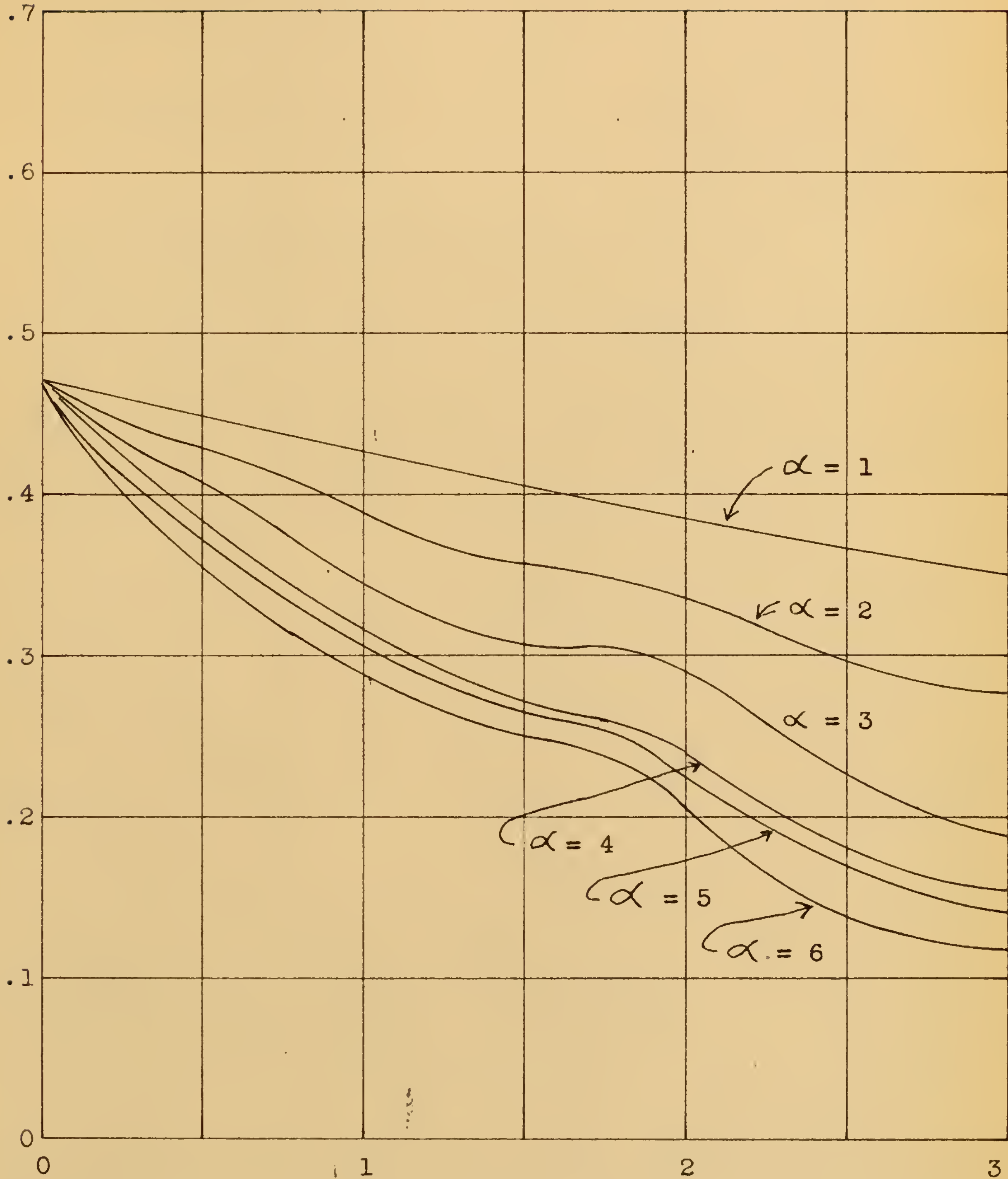
PLATE XLII



EXPLANATION OF PLATE XLIII

γ (in file population) versus interrogations (in file populations) for the transposition system confronted with density No. 2 and α ranging from 1 to 6.

PLATE XLIII



EXPLANATION OF PLATE XLIV

Fig. 1. Initial distribution of induced weights of identification tags for the transposition system, $\alpha = 6$, density No. 2.

Fig. 2. Final distribution of induced weights of identification tags after 3 N interrogations for the transposition system, $\alpha = 6$, density No. 2.

PLATE XLIV

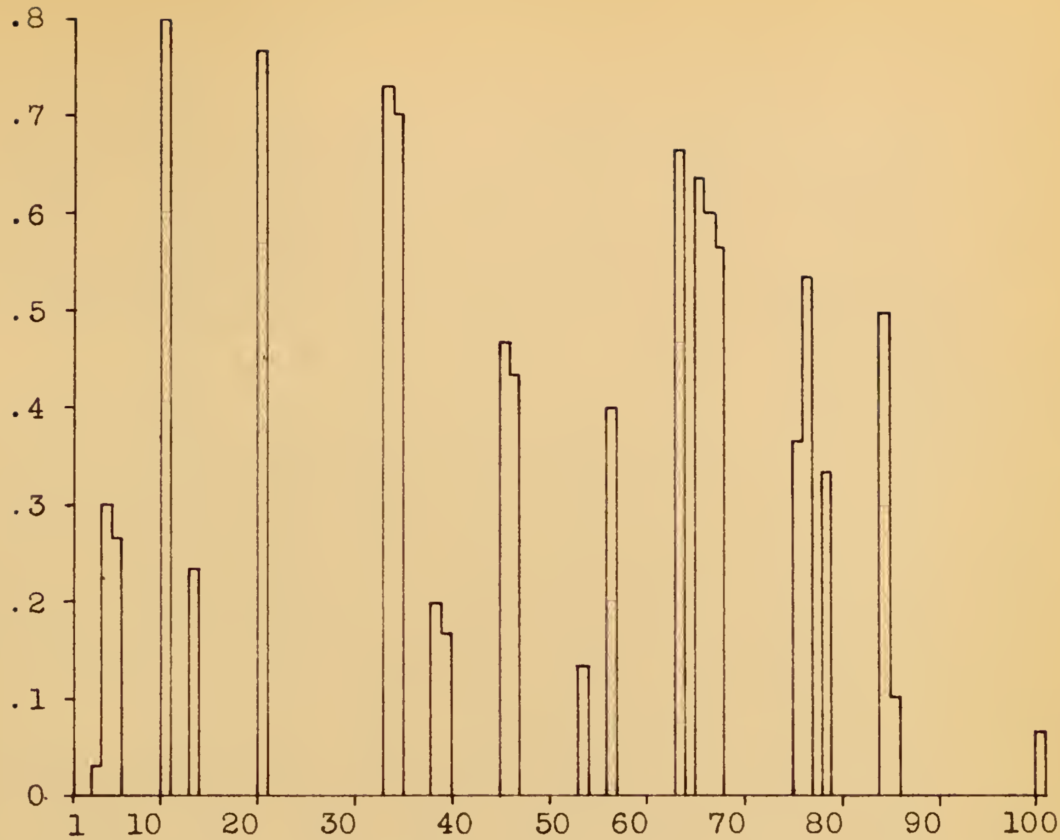


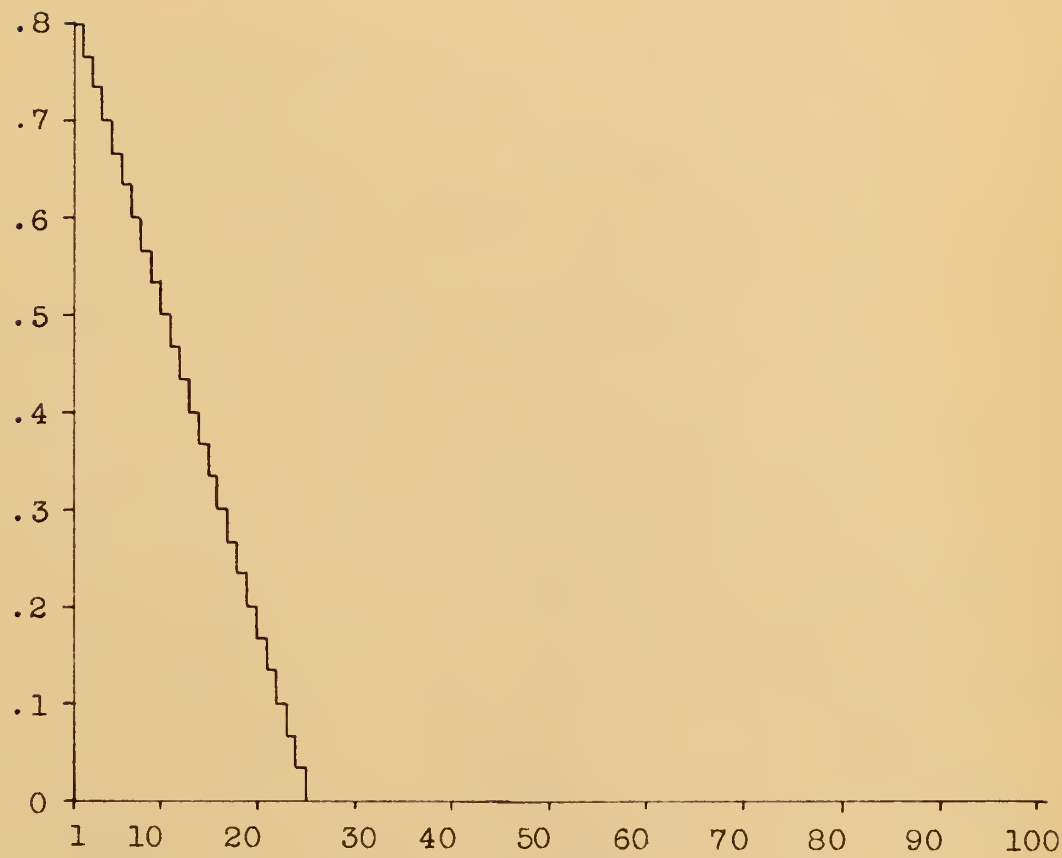
Fig. 1.

Fig. 2.

EXPLANATION OF PLATE XLV

Ideal distribution of identification tags

PLATE XLV



EXPLANATION OF PLATE XLVI

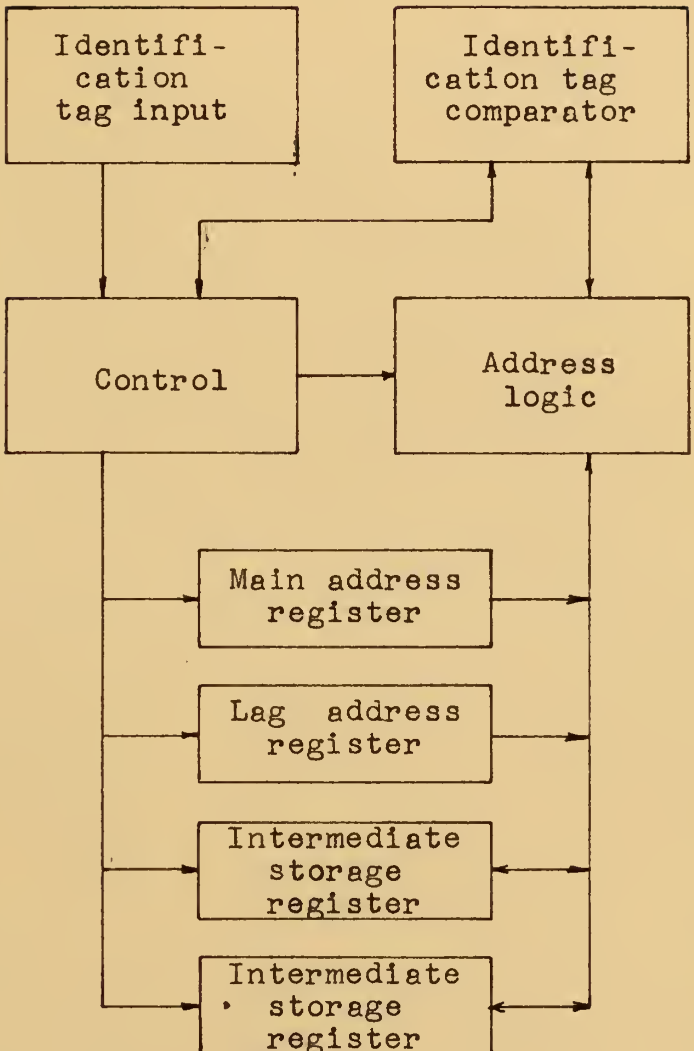
γ (in file population) versus interrogations (in file populations) for the transposition system, α = 6, confronted with density No. 2 when five different arrangements of random interrogations were used.

PLATE XLVI

EXPLANATION OF PLATE XLVII

Block diagram of mechanization components for the transposition system.

PLATE XLVII


Identification tag input
Identification tag comparator
Control
Address logic
Main address register
Lag address register
Intermediate storage register
Intermediate storage register

ORDER ADAPTIVE FILE ORGANIZATIONS TO BE USED IN CONNECTION WITH SERIAL SCAN

by

THOMAS HOWARD ELROD

B. S., Kansas State University
of Agriculture and Applied Science, 1955

AN ABSTRACT OF
A MASTER'S THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Electrical Engineering

KANSAS STATE UNIVERSITY
OF AGRICULTURE AND APPLIED SCIENCE

1960

In large files having in the order of 10^6 items, the central problem is one of being able to retrieve any item from the file in the minimum amount of time. When the file is a large digital file, several methods of searching for and retrieving an item are available. The investigation carried out in this thesis was concerned with only one of the possible searching patterns and methods of causing the retrieval time to approach a minimum. The ideal organization was also found and is shown as the lower bound to the other solutions.

Three systems were investigated to determine the existence of an order adaptive characteristic and to determine the parameters affecting the speed and degree of adaptation. Because exact explicit mathematical formulations of the systems proved unwieldy for solution, approximate models were used and their solutions programmed and run on the digital computer.

All of the systems investigated exhibit adaptation characteristics with well determined parameters. Of the three systems investigated, the transposition system shows the most promise for easy mechanization. The investigation carried out on these systems was not exhaustive but displayed the adaptation characteristics nicely.