UNIVERSITY OF LONDON

IMPERIAL COLLEGE OF SCIENCE AND TECHNOLOGY

DEPARTMENT OF ELECTRICAL ENGINEERING

# TELEPHONE NETWORKS FOR DEVELOPING AREAS

Presented in Partial Fulfillment of the Requirements
for the Degree Doctor of Philosophy
in Electrical Engineering

By

CARLOS AMADEO MARAZZI, DIPL. ING., M. SC.

April, 1971

## ABSTRACT

Most of the developing countries do not possess a telephone network sufficient to promote trade, industry and education at the level required for their economic development. In the present work a telephone network specifically intended for such countries has been proposed as the best solution to the problem created by the existing low traffic and the necessity of economic expansion.

A description of some existing systems has been made in order to point out characteristics relevant to their size and application. None of them, however, were designed with the needs of the developing countries in mind, nor do any fit those needs and conditions.

The system proposed here can be built with the existing technology. Moreover, most of its components are already available in the market.

The system, including the satellite exchanges, data link, central processor and controlling program have been described. Special attention has been paid to the controlling program because it largely determines the signalling between the satellites and the central processor.

A simulation language has been written to operate the system under simulation conditions in a large computer. The simulation program has been described in some detail

because it is felt that it can be used not only to write and test the controlling program, but to study traffic related problems, such as the influence of the type of of processor, number of digit receivers in the exchanges, and so on, in function of the traffic.

Besides selecting the type of network and forecasting the demand of telephones, a crucial point in the implementation of a network is to find an optimum plan of investment. A method to find such an optimum plan was designed. It has the property of optimizing the invested capital, maintaining the flexibility of admitting modifications when the circumstances so require it.

For completeness, a model case is described and an optimum investment plan for the imaginary country Teleria is found.

The consideration of social weights for the selection of different telephone projects during their period of commissioning is discussed and a satisfactory social welfare function for this purpose is given.

# PREFACE

This study has primarily intended to work out a telephone network specifically for rural, sparsely populated developing areas. The need for such a telephone system is undeniable if those lagging areas are to develop an infrastructural base that will thrust them into the process towards progress and modernization. It might be argued that those sparsely populated areas have little need for communication links. However, one feels that at some point the vicious circle of underdevelopment must be broken.

A telephone system, in the same way as for instance, a main highway, can give a new impetus to a lagging area, and can be an important element that will attract other types of badly needed projects and services, both in a planned or unplanned economy.

The first chapter of this study deals with the infrastructural aspect of telecommunications in the developing countries considering the peculiarities of the areas, the obsolescence of their networks and the need versus demand criterias for telephones. In Chapter II the problems of selecting the type of network and hardware appropriate for developing countries are considered and a survey is made of existing systems some of which could be of immediate application. Also, a type of telephone

network system is recommended and in Chapter III a description of that system is presented.

The telephone network described in this work is electronic and remote controlled, mainly because it is felt that this is more adequate to meet the needs of the areas concerned, than manual or electromechanically controlled systems. A remote controlled system requires less technically trained personnel and can be easily expanded in size, security and number of services, as demand increases. In Chapter IV, the signalling system of the network and a method for simulating the system in operation are described.

Again with the needs of these lagging areas in mind, the author has found it essential to dedicate Chapter V to the development of an optimum investment plan. The purpose has been to optimize investments in terms of economic and social profits.

Chapter VI considers an imaginary developing country for which a telephone network has been planned, and a plan of investment has to be elaborated. This chapter presents the computation of a model case, applying the knowledge from Chapter V.

When an optimum plan of investment for a telecommunication network is evaluated, special attention should be paid to the social needs since the ultimate goal will be that of improving the social conditions of life. In Chapter VII, the social weights for the selection of

telephone projects are considered and a satisfactory social welfare function for this purpose is given.

The author is grateful to Professor Colin Cherry for his help and encouragement throughout the time spent at the Imperial College, and also to Mr. Robert Bogner for his helpful advice in the initial period. He is particularly indebted to Mr. Charles J. Hughes of the Post Office Research Department, and his colleagues, for the many valuable suggestions and orientation, and to Dr. Nikolaus Christofides of the Imperial College for his most valuable advice and stimulus. Dr. Christofides read and gave constructive criticisms to the chapter dealing with operations research.

The author also wishes to convey his thanks to Messrs. Robert King, Max Casley, Denis Dicks and Robert Andrews for participating in the proof-reading.

Finally, the author is indebted to the College of Engineering of the University of Puerto Rico, without whose financial aid this study could not have been carried out.

TABLE OF CONTENTS


ABSTRACT . . . . . ii

PREFACE . . . . . iv

LIST OF ABBREVIATIONS . . . . . x

INTRODUCTION . . . . . 1

CHAPTER I. INFRASTRUCTURAL ASPECTS . . . . . 5

I.1 Telecommunications as Part of the Infrastructure in the Developing Countries. . 5

I.2 Environmental Conditions . . . . . 10

I.3 Need and Demand of Telephones. . . . . 11

CHAPTER II. SELECTION OF A NETWORK. . . . . 14

II.1 Problems in the Selection of the Type of Network. . . . . 14

II.2 Proposition for a Developing Area. . . . 16

II.3 The Control of Switching Networks. . . . 19

Background . . . . . 19

Common control principles. . . . . 22

Stored-program control . . . . . 22

II.4 Some Existing Systems. . . . . 27

CHAPTER III. A NETWORK WITH STORED-PROGRAM CONTROL EXCHANGES . . . . . 37

III.1 System Description . . . . . 37

The central processor. . . . . 42

The data link. . . . . 44

The satellite exchanges. . . . . 48

TABLE OF CONTENTS--Continued


| | | |
|---|---|---|
| III.2 | The Stored Program. . . . . | 50 |
| III.3 | The Program's Blocks. . . . . | 58 |
| | The off-hook block (OFH). . . . . | 58 |
| | The dialled digits block (DD) . . . . | 62 |
| | The data unit block (DAUNIT). . . . . | 63 |
| | The trunk calls' blocks . . . . . | 64 |
| | The on-hook block (ONH) . . . . . | 67 |
| CHAPTER IV. | COMPUTER SIGNALLING SIMULATION. . . | 68 |
| IV.1 | Signalling System . . . . . | 68 |
| IV.2 | A Language for Signalling Systems Simulation. . . . . | 69 |
| | The subroutines . . . . . | 71 |
| | The generation of random arrival time . | 75 |
| | The time structure of the S.S.S.. . . . | 77 |
| CHAPTER V. | OPTIMUM INVESTMENT STRATEGY . . . . | 78 |
| V.1 | Plan of Investment. . . . . | 78 |
| V.2 | Stages of Optimization. . . . . | 79 |
| V.3 | The Investment Model. . . . . | 82 |
| V.4 | Subdivision of the Network. . . . . | 84 |
| V.5 | Variables of the Model. . . . . | 86 |
| V.6 | Macro-Program: an Algorithm for the Selection of the Best Set of Projects . . | 93 |
| V.7 | Economic Profit . . . . . | 97 |
| V.8 | Program within One Period . . . . . | 108 |
| V.9 | One Stage Optimization. . . . . | 117 |

TABLE OF CONTENTS--Continued

CHAPTER VI. A MODEL CASE PLANNING . . . . . 121

VI.1 The Country. . . . . 122

VI.2 The Telephone Network and Forecasted Needs. . . . . 124

VI.3 The Plan of Investment . . . . . 131

CHAPTER VII. SOCIAL PROFITS AND OTHER PRIORITIES 135

VII.1 The Ranking of Projects. . . . . 135

VII.2 The Social Welfare Function. . . . . 141

CONCLUDING REMARKS . . . . . 149

APPENDIX I. AN ALGORITHM FOR THE SELECTION OF INVESTMENT PROJECTS. . . . . 151

APPENDIX II. AN ALGORITHM FOR THE KNAPSACK PROBLEM WITH CONSTRAINTS IN EACH VARIABLE . . . . . 164

APPENDIX III. AXIOMATIC ANALYSIS OF THE SOCIAL WELFARE FUNCTION'S THEOREM . . . . 170

APPENDIX IV. LOCAL NETWORK OPTIMIZATION . . . . 174

APPENDIX V. PROGRAM LISTINGS AND FLOW GRAPHS . 176

SELECTED REFERENCES. . . . . 200

## LIST OF ABBREVIATIONS

## FREQUENTLY USED IN TELEPHONY LITERATURE AND IN THIS WORK

| | |
|---|---|
| AT&T | American Telephone and Telegraph |
| CC | Common Control |
| CCITT | Comité Consultatif International de Téléphonie et Télégraphie |
| CDS | Called Subscriber |
| CGS | Calling Subscriber |
| COS | Class of Service |
| DC | Distributed Control (step-by-step control) |
| DD | Dialled Digit |
| DDR | Dialled Digit Receiver |
| DPI | Dial Pulse Interpreter |
| DSR | Discriminator Selector Repeater |
| DSX | Discriminating Satellite Exchange |
| EN | Equipment Number |
| FDM | Frequent Division Multiplex |
| GOS | Grade of Service |
| GSC | Group Switching Centre |
| I/O | Input/Output |
| ISD | International Subscriber Dialling |
| ITT | International Telephone & Telegraph |
| ITU | International Telecommunication Union |
| LSI | Large Scale Integration |
| LU | Line Unit |
| MDF | Main Distribution Frame |
| NS | Normal Subscriber |
| OFH | Off-Hook |
| ONH | On-Hook |
| PABX | Private Automatic Branch Exchange |

| | |
|---|---|
| PBX | Private Branch Exchange |
| RPC | Remote Processor Control |
| SDM | Space Division Multiplex |
| SN | Subscribers Number |
| SPC | Stored-Program Control |
| STD | Subscriber Trunk Dialling |
| TDM | Time Division Multiplex |
| TO | Trunk Offering |
| TOS | Temporarily Out of Service |
| TS | Tone Sender |
| TSC | Transit Switching Centre |

# INTRODUCTION

Economic growth generally refers to the raising of the income level in the rich areas of the world, whereas, for the poor areas, the same process is called development.[J1,M1] One of the main reasons for this differentiation lies in the urgency of the process. In the rich areas economic growth is less urgent since the main concern is usually to bring about an improvement in the quality of life. In the poor areas, however, the urgency is for breaking with the obstacles hindering the exploitation of resources for the welfare of the area, and bringing about a total change in society, both in quantitative and qualitative terms.

In recent literature on economic development, one finds a wide range of opinions on the geographical, sociological and historical reasons for the present state of underdevelopment of the poor areas* and on how to go ahead with development. Even though these arguments and facts are numerous and sometimes differ widely, those refering to the infrastructural conditions find common ground in most authors. It is interesting to point out that at the end of the nineteenth century and the beginning

---

*The expression underdeveloped or developing areas is applied in this work to any group of countries whose economic and social status is lagging, as would be the case of East Africa, or a lagging area within a developed country.

of the twentieth, the modern development experienced by Great Britain, the United States and France was not merely due to the spontaneous desire for a better life. It was greatly determined by the introduction of new technology, and as a consequence, by the rapid development of cheap transportation with facilities for large cargoes, steam shipping, transportation for refrigerated foodstuffs, and the introduction of fast and efficient means of communication and of industrial machinery. The introduction of cable communications, for instance, was a necessary complement to the railways before it was possible to exploit fully the benefits of international trade. Of course, the development of modern technology and its effects on economic development is closely associated to the social organization and patterns of the area concerned. But the fact remains that infrastructural elements are essential.

Keeping in mind the urgency of economic development in the lagging areas and the need to create infrastructural conditions for its take-off, the present work is concerned with developing a type of telephone network and the optimal investment strategy, regarding socio-economic factors, both suitable for application in developing areas. The selection of the type of system has followed the lines expressed by J.H.H. Merriman that: "... a good system is one which has versatility and openendedness built in to permit it becoming a better

system in future".[M2]

In all problems related to the development of the infrastructural services such as roads, railways and communications, one has to face the fundamental difficulties of, on the one hand, finding the most adequate hardware for an expanding society, and, on the other hand, finding the optimum plan for the investment of scarce resources in order to obtain the maximum social and economic profit.

A further complication ensues from the terms social and economic profit. Social scientists have not found them easy to define. But even if a tight definition could be worked out, there still remain the problems of measuring social utilities and developing a methodology for making social choices, what in socio-economic terms is called the social welfare function.

When an administration responsible for telephone communications is faced with the problem of assigning priorities for the provision of service, it can either weigh the recommendations of other agencies of government according to its own value judgment, or accept an imposed scale of weights given by the policymakers or a board of overall planners of the economy.

There is no doubt that the possibility of achieving an overall optimum arrangement lies in the overall planning of the economy, where the investment of scarce resources for every public service and commodity

is considered simultaneously and subjected to the environmental constrictions of the particular area. This latter approach undoubtedly simplifies the problem of finding the real place of the telephone and in general the telecommunication network within the social state chosen.

In this thesis a network is described that meets the general requirements of most developing areas and an optimum economic plan of investment is found, by making use of the tools of operations research. The different strategies studied in the social sciences to make socio-economic decisions will be considered when evaluating the social profit of the network.

# CHAPTER I

# INFRASTRUCTURAL ASPECTS

## I.1 TELECOMMUNICATIONS AS PART OF THE INFRASTRUCTURE IN DEVELOPING COUNTRIES

An essential element for the economic development of the lagging areas of the world is a solid infrastructural base, consisting of power, ports, airports, transportation, communication networks, and so on. To provide for these, substantial investments are needed. In developing countries, these investments are not usually very attractive to private investors, since there is no prospect for a short-term recovery of the capital and profits. Hence, this type of investment has been usually made by the governments of the countries concerned, which have, of course, limited resources available.

Thus, the governments must make decisions on how to allocate the scarce resources, and care has to be taken in assessing the needs of each sector of the infrastructure, especially in the case of the telecommunications network, where the commercial or demand criteria has normally been used. In the developing countries, where telecommunications networks are lagging or lacking, the commercial or demand criteria as a measure of the amount to be invested should not be used, since other factors

that are strongly dependent on telephone services, such as access to health, security and other government services are also backbones of the infrastructure. It is a well known fact that in all developed countries the ratio of residential telephones to business telephones is larger in the cities than in the rural areas. In the rural areas business telephones predominate, including those of local government officials, police, fire department, post office, hospitals and many other essential services for rural areas like forestry, irrigation, weather, etc.[J1]

Even for the efficient use and organization of other infrastructural elements as for instance the mass media communication systems, the telephone and the telegraph network play a vital role. "Before news can be received locally, without delay, telephone and telegraph must be working efficiently...". "One of the great handicaps to any developing country's progress is its lack of internal communication facilities, in both the organizational and informational spheres".[C2] Certainly, the provision of a good telecommunication network, especially in the rural areas, where the population is dispersed, is well justified.

Table 1 shows the telephone situation in four developing countries.[P3,P4,P5] Column A shows the percentage of population with no access to any kind of telephone. Column B shows the percentage of population with access to only one public telephone (a rural coinbox

telephone). Column C shows the percentage of population either rural or urban, with access to public and private telephone, and that do not depend on a single coinbox.

TABLE 1

| Country | Column A | Column B | Column C |
|---|---|---|---|
| El Salvador | 65 | 5 | 30 |
| Peru | 20 | 1 | 79 |
| Colombia | 25 | 25 | 50 |
| Ecuador | 50 | 30 | 20 |

From this table, it could be said that, in some Latin American countries, more than half of the population either do not have any access to any kind of telephone communication, or only to a public telephone in the community. This does not necessarily imply that this people are completely isolated, since in many cases a telegraph wire or military outpost with radio communication keeps them in contact with the main urban centers. But these types of communication links are not sufficient to encourage the establishment of industries and trade.

It is interesting to note that the communities without access to any kind of telephone are not necessarily concentrated in a specific area of a developing country, nor are they only small hamlets of a few hundred inhabitants. To illustrate these points Table 2 shows, for Peru, the number of communities of over 1000 and over 5000 inhabitants that have no access to any

kind of telephone communications, and their geographical distribution.

TABLE 2

| DEPARTAMENTO (Province) | Number of communities of more than: 1000 inhabitants | 5000 inhabitants |
|---|---|---|
| Tumbes | 10 | 1 |
| Piura | 44 | 4 |
| Lambayeque | 23 | - |
| La Libertad | 33 | - |
| Ancash | 29 | 4 |
| Lima | 62 | 10 |
| Ica | 29 | 1 |
| Arequipa | 41 | 3 |
| Moquegua | 1 | - |
| Tacna | 3 | - |
| Cajamarca | 63 | - |
| San Martin | 19 | 2 |
| Huanuco | 13 | 4 |
| Pasco | 19 | - |
| Junin | 56 | 33 |
| Huancavelica | 12 | 2 |
| Apurimac | 18 | - |
| Ayacucho | 31 | 1 |
| Cuzco | 51 | 4 |
| Puno | 21 | 4 |
| Madre de Dios | 1 | 1 |
| Amazonas | 12 | - |
| Loreto | 9 | 2 |
| Prov.Constitut.del Callao | 10 | - |

Fig. 1 in the next page shows a map of Peru with the geographical location of the areas mentioned in Table 2.


Amazonas
Tumbes
Loreto
Piura
San
Martin
Lambayeque
Cajamarca
La
Libertad
Ancash
Huanuco
Pasco
Lima
Junin
Madre
de Dios
Provincia
Constitucinal
del Callao
Cuzco
Huancavelica
Ayacucho
Puno
Ica
Arequipa
Apurimac
Moquegua
Tacna

Fig. 1.

## I.2 ENVIRONMENTAL CONDITIONS

The major constraints in the design of a telecommunications network are those imposed by the environment of the area in which the network is going to be operating. Here an outline of the general characteristics of such an area is presented. However, it is outside the scope of this study to find a precise model of a developing area, one including all the parameters which would be relevant to the planning of the whole economy. Thus, here, for the sole purpose of considering the best suited type of telephone network, some very general characteristics are outlined.

a- The majority of the population in underdeveloped or developing countries is rural. Most of the rural population is geographically distributed in a semi-dispersed way, that is, in many small communities, more or less evenly scattered on the fertile plains or valleys.

b- There is little or no industry besides that at the level of artisanry and crafts.

c- Communication between the individuals for domestic business or official purposes is very limited. " In developing countries no need is expressed for a telephone system , but this does not mean that a rapid means of communication is unnecessary."[C1]

d- In the first stage of development, priority will have to be given to securing the basic needs for the subsistence of the population in the community. This will

call for vast improvements in the exploitation and distribution of agricultural products.[B1,B3]

e- As development progresses, rural areas may be transformed into industrial, commercial or mining centres, particularly when improvement in the agricultural output is due to increased farming efficiency. Then, some human resources will be employed in industry, if the infrastructural conditions are appropriate. This will result in the concentration of population in larger centres, and the need of expanding the telephone service.

In planning a telephone network for an area with these characteristics and possibilities of development, certain features of the network will become more important, whereas others, such as security*, will be less so. This will be seen in Chapter II.

## I.3 NEED AND DEMAND OF TELEPHONES

The two means of telecommunications of most relevance are the telephone and the telegraph. The telephone network is the more important considering its penetration and accesibility, although it is the more expensive to install and maintain.

When assessing the needs of telephone service for underdeveloped areas one must not follow, as a general rule, the approach applied to developed countries, nor that used when those countries were in the stage of development. The differences between today's world and

* i.e. reliablity

the nineteenth century, when industrialization began in the developed countries of today, are so significant that the possibility of drawing any similarity is very small indeed.[S4] On the other hand, data which is usually very valuable in forecasting future demand, such as past statistics and present demand, are often inaccurate or non-existent in the developing countries.

The standing applications for telephone service do not always reflect the actual demand of service, and much less the need of service. B. Wellenius and J. Potocnjak show that, in a case study of Chile, there is more than one type of demand.[W2] They were able, for 1967, to survey and discriminate the following categories of potential subscribers:

1- those that had filed an application and were on a waiting list;

2- those that had not bothered to file an application because the waiting time is too long, and the telephone administration may not transfer the telephone if the subscriber moves to another address (waiting time had a median of 3 years, 20% was 5.5 years, and 10% was 7.5 years or more);

3- those who do not know the procedure to apply for telephone service.

The unexpressed demand originated by the last two groups (considering only those who can afford to pay for the service) is <u>four times</u> as large as the expressed

demand shown by group 1, that is, the waiting applications. This study was made for the suburban area of Santiago de Chile. For rural areas the numerical results may be somewhat different. Nevertheless, for the same reasons, a large unexpressed demand may exist.

Both the expressed and the unexpressed demand can be called a priori demands, that is, demands that exist before the service is available. There is still another type of demand, which one could call a posteriori demand, that is the one which is created by the existence of the facility. This type of demand is very difficult to estimate even in the countries where accurate past statistics exist and will consequently be more difficult where these do not exist.

But even if the actual demand were given by the expressed demand, these statistics would still not be sufficient to forecast the needs of the developing countries, since that would imply the assumption of a similar trend of progress, whereas, what the developing country would be trying to achieve is a faster rate of economic growth in order to begin the journey to development.

# CHAPTER II

# SELECTION OF A NETWORK

## II.1 PROBLEMS IN THE SELECTION OF THE TYPE OF NETWORK

As has been mentioned, the economic problem is the most difficult obstacle to be overcome by the governments in order to build the infrastructure. In the case of telephone services, the problem is further complicated by the fact that there are many different parameters affecting the selection of any specific type of network. Since there is no ideal telephone plant for all conditions, the administrative body will have to select the system best suited to their needs. When doing so, and particularly when considering the rural areas, the administrations are confronted with the following situations.

1- In the rural areas, there will usually be no telephone network, and if there is one, it is probably obsolete and not suitable for the encouragement of economic growth. Hence, in these cases it is not a question of improving or expanding the existing network, but instead of the necessity of installing a new one.

2- The actual telephone traffic and number of lines, or even the foreseen potential traffic and number of lines, may be too small to justify the installation of automatic exchanges.

3- In the early stages of development, even if the population were relatively dense, the telephones will be thinly distributed. Hence, the alternatives are either to install very long lines from subscribers to a central automatic exchange, or to install many uneconomic self-controlled exchanges. The possibility of using concentrators as a solution to the problem could be considered, although these have the disadvantage of creating the less reliable 'star' type of network.

4- Manual exchanges may be out of the question when it might not be possible to find the personnel to attend them, particularly if a 24-hour service is desired.

5- In cases where manual exchanges could be installed, it may be uneconomic to do so, if in the near future, due to a growing demand caused by economic development, an automatic exchange is needed.

Due to these conditions, it seems that a possible solution to the problem of providing an efficient telephone network at low cost and with the capability of easily expanding in size and security, is the use of a dispersed switching network, remotely controlled by a central processor.[A2] In this case, the processor is situated at a processing centre, and exercises control of both, adjacent and remote switches. In this way advantage can be taken of the control capability of the processor by controlling a relatively large number of exchange lines.

In the initial phases of developing the communication networks of rural areas, no excessive effort (in terms of cost) should be made to provide a high degree of security (reliability). Instead, it is a better policy to provide service to a more extensive area, covering a larger number of people and increase the security of the system progressively once the basic needs for communications are satisfied. This is because: "Reliability becomes increasingly important as the message capacity of systems increases. If a communication link carries only 5 or 10 messages once, then a breakdown may not be too serious... But if 10,000 are carried, then the disruption caused by a breakdown might be disastrous".[32] Naturally, the plants and network should be of such nature that improvements in security can be carried out with additional costs that are somewhat proportional to the improvement.

## II.2 PROPOSITION FOR A DEVELOPING AREA

Keeping in mind what has been said so far, a general recommendation for a type of telephone network suitable for the developing areas would be a network of small, but easy to expand, automatic exchanges remotely controlled by central processor(s) with the provision for subscriber trunk dialling. The system is thus geared to the largest sector of the population of most developing areas, namely the rural sector. The reasons for the selection of this type of network structure are

many, and the following can be mentioned:

1- A dispersed switching network with a central processor to control a large number of small unattended offices, allows the implementation of a simple system which will provide the most essential telephone service.

2- A stored-program control system, concentrating most of the control functions in the main office, in the form of software, reduces the maintenance requirements, both in the central office and in the satellite exchanges.[A2] This has shown itself to be an important consideration in allowing an expanding telecommunication service tc be economically viable.[M2]

3- The proposed system can be easily expanded in size, reliability and the number of services. The expansion in size requires few alterations in the controlling program, and additional hardware. The expansion in the number of services requires mainly software changes and little additional equipment. And the improvement of reliability consists mainly in duplicating equipment.

4- It has all the advantages of the automatic systems, one of which is the 24-hour continuous unattended service. This could otherwise be difficult to achieve, particularly in small villages, for the lack of operators.

5- Remote controlled satellite exchanges allow in many cases the use of transverse lines between the satellites, leading to the more secure 'mesh' type of network.

6- The computing facilities offered by the central processor, in the main office, could be used, at low

traffic hours, for purposes other than control, such as: accounting, customers billing, pay-roll computations, routine faults checking, print-out of faults on type-writer terminal, etc.

7- The advantage of having direct trunk dialling is significant to the users as well as to the administration. On the one hand this provides the majority of the subscribers with a most needed service (penetration of official telephones in rural areas is higher than that of residential telephones)[C6]. On the other hand, it is of great economic advantage to the administration since the income derived from trunk traffic grows much faster than the number of subscribers when direct trunk dialling is available.

Fig.2 shows the percentage increase of income derived from long distance traffic, from 1952 to 1961, when subscriber trunk dialling was introduced in West Germany.[H2]

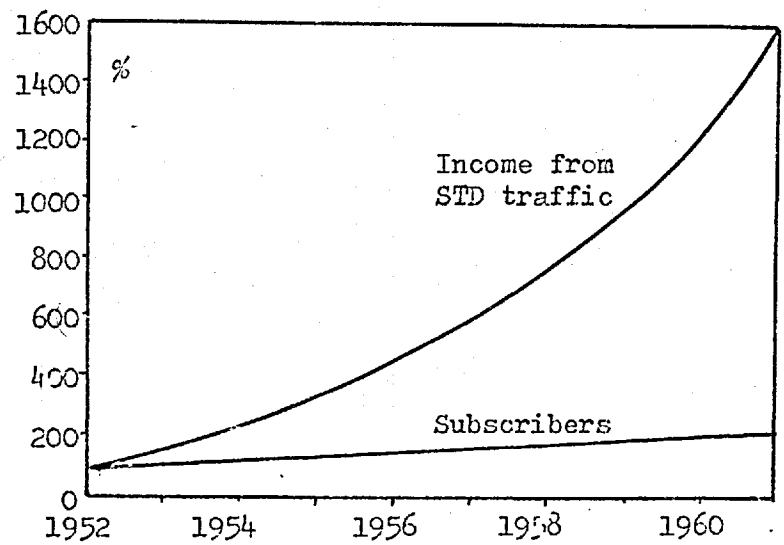



Fig. 2.

## II.3 THE CONTROL OF SWITCHING NETWORKS

### Background

Most of the early automatic switching systems used direct dial control from the subscriber's equipment as a way to interconnect two subscribers. The equipment for this method of switching control (Strowger step-by-step) consists in a set of two-motion rotary switches driven by electric pulses generated by the subscriber's dialling. Although this method was adequate for its time, its lack of flexibility and the multiplicity of equipment required for expansion, restricted its subsequent use.

An improvement in the cost of expanding such system was achieved by the storage of the dialled digits and the introduction of common control. The stored information (digits) could then be used to drive the switching mechanism directly or indirectly. In the second case, the electric pulses generated by the switch are compared to the ones stored to halt the moving mechanism of the rotary switches. Furthermore, the stored information (subscriber's number) could be translated to another numbering system (equipment number) allowing a greater flexibility in the expansion of the network and making the trunk hunting action independent of the time interval between the dialled digits.

Thus, in the new developments the dialled digits were stored and translated into a code-number. A device called 'marker' used this code-number to switch (to mark)

the connecting element, usually by placing a voltage on the corresponding wires.

The separation of the switching function from its control fostered the introduction of new technology in both the common control and the switching mechanism. In the area of common control a combination of electronic and electromechanical devices was first introduced and then a completely electronic system. As a consequence the transfer of the control from a hardware to a software mechanism was facilitated.

In the switching mechanisms, the wiper motion switches have limitation in the flexibility of their application and speed of connection. Thus, the introduction of the crosspoint type of switching, first in the form of a bank of relays and later with the crossbar and reed relay switches rendered the ideal partnership to the common control technique.

Fig. 3 shows a schematic form of the crosspoint type of switching and control. The subscribers (numbered 1 to 5 on the left of the diagram) are interconnected by means of two switching matrices, the input-matrix and the output-matrix. When a calling subscriber (i.e., 3) picks up his hand set, an off-hook condition is detected in his line circuit. This information causes the subscriber's line to be connected to a bridge in the input-matrix through which the subscriber obtains dial tone. The bridge then connects through a marker control matrix to a free register. The digits dialled by the calling subscriber

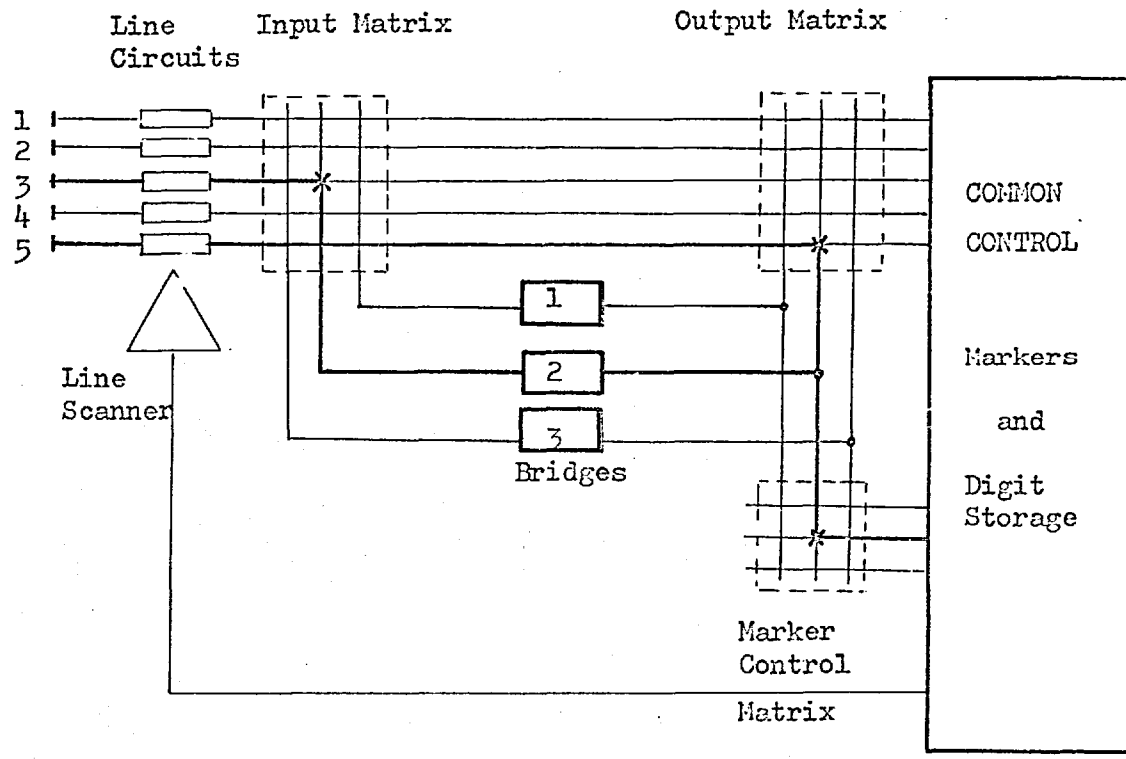



Fig. 3.

are then stored in the register. When the dialling is completed, the connection in the marker control matrix is dropped and the connection of the called subscriber (i.e., 5) is performed by the marker control acting on the output matrix. Thus, a path from the calling to the called subscriber is established through the input matrix, the bridge, and the output matrix.

## Common Control Principles

Three control techniques are applied[P6] in modern telephone exchanges with common control: wired logic control, programmable control, and stored-program control.

Wired logic is the most widely used. The control actions are determined by the interconnection between the different control devices. Thus, the sequence of actions for every call follows a pattern rigidly established by the structure of the system. Changes in the system generally involve the changing of equipment and a considerable amount of rewiring.

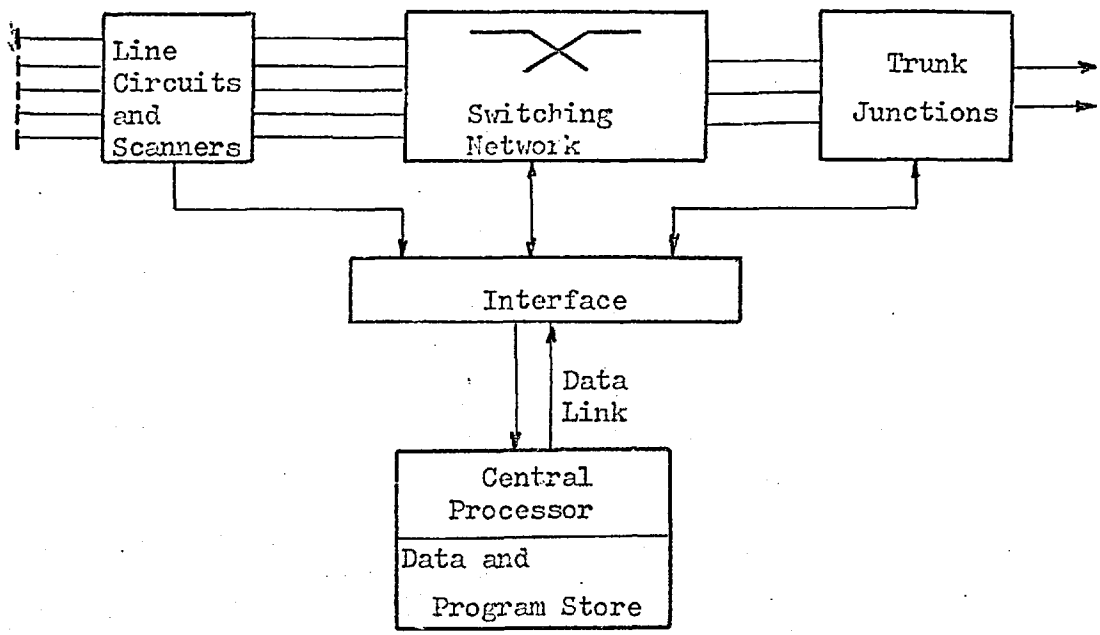
In a programmable system the control actions can be altered within certain limits since its logic sequence is stored in some form of wired memory. Thus, many changes can be performed by just rewiring a frame with ferrite-cores or modifying a translation device. Several systems with programmable control are in use in the UK and the USA.

The highest degree of flexibility can be obtained with a stored-program control where the logic of control is almost entirely determined by a computer-like program permanently stored in a processor.

## Stored-Program Control

Figure 4 shows a block diagram of a stored-program control system. The line circuits, switching network and trunk junctions are sometimes grouped together under the

Fig. 4.




name of telephony unit. The block labelled 'switching network' includes the input and the output matrices as well as any intermediate switching stage necessary to interconnect two subscribers of the same exchange or a local subscriber with a trunk circuit. The line testing units and the switching and relay operating units together with the data buffering elements are sometimes grouped together under the name of interface or transfer unit. The processing unit consists of the central processor(s) and the data and program store(s). The operation of the line circuits, scanner and the switching network

is in principle the same as the one explained previously for a common control system. The basic differences here are that: 1- the logic sequence as well as information about the state of the different parts of the exchange and the dialled digits are kept in the central processor's store; and that, 2- the state changes of the lines, switches, etc. are converted at the interface in binary digital messages. These messages are transferred to the central processor through a data link. In the data link, messages flow in two directions. Messages are received by the central processor informing events in the network, and messages are sent by the central processor to the network to produce new events, like the connection or disconnection of lines or devices.

The interfacing of a central processor with the controlled exchange can be done in several ways ranging from the provision of separate interfaces for each specific task to a single multifunction interface, as in the case of remote controlled exchanges.

Markers and registers as those used in non-stored-program control systems are no longer required since their operations have been taken over by the processing unit.

One main concern of the designers of stored-program control systems has been security. The term <u>security</u> is used in the field of telephone engineering to mean safety from a total breakdown. The reason for this concern is

implicit in the use given to the system. A highly industrialized and complex social group is extremely dependent on its communication links. Therefore, security is of prime consideration. Even in the rural areas of the most developed countries, where telephone traffic is low, a high degree of security is usually maintained as a matter of standards.

Several techniques have been adopted to overcome this problem of security. One is the so-called synchronous duplicated processor, where two equal processors and input-output interfaces are working simultaneously and in parallel, while only one of them is performing the control operation, and the other taking over the control only in case the first fails. To expand this system everything has to be expanded in pairs. This type of system has already been working satisfactorily for several years (ESS1, ESS2, AKE 120).[P6]

Another approach to the problem has been the share-load technique, where two processors, each capable of handling all operations, work simultaneously sharing the message traffic. Examples of such systems are the ITT-10C (Metaconta) system and the TXE4 exchanges. This type of system requires an interprocessor communication link between the processors which allows them to coordinate their share of the traffic.

Yet another technique has been suggested to solve the same problem. It is called multiprocessor control,

where a set of central processor units have access to several stores as shown in Fig.5.

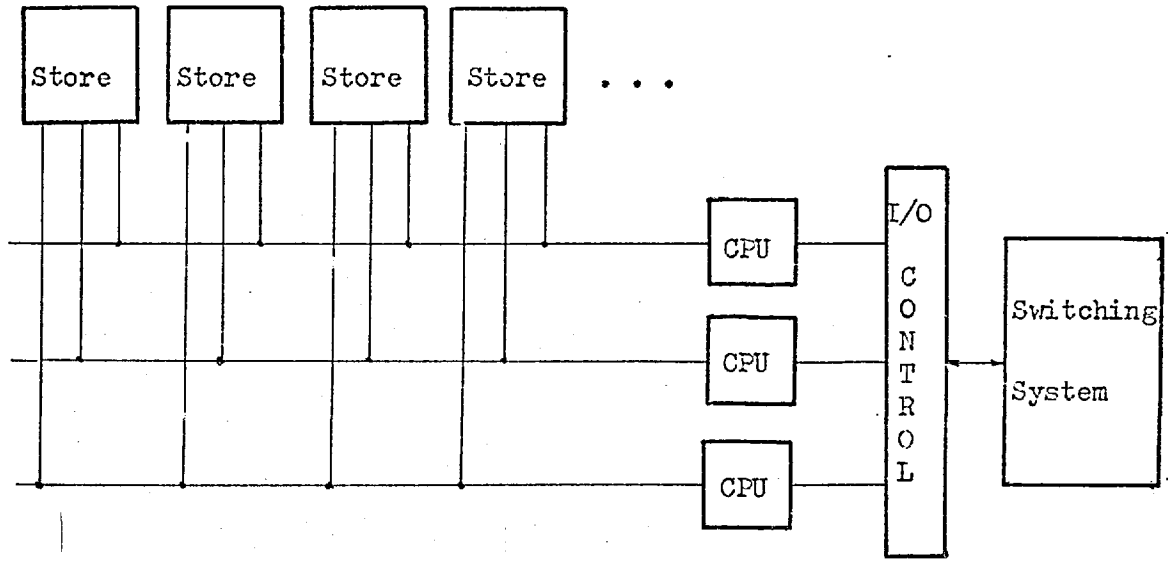



Fig. 5.

As in the previous ones, this arrangement presents some advantages in reliability but also some disadvantages, particularly in the synchronization of the central processors and in the complexity of the interfaces.

In general, it can be said that stored-program control allows a much more diversified operation of the exchanges than any previous type of control, since the number of services and the number of alternatives to deal with the calls are only limited by the size of the processor's storage and the designer's imagination.

## II.4 SOME EXISTING SYSTEMS

In the last 15 years a number of telephone switching systems with electronic control and with stored-program control have been constructed and some have been commercially manufactured. Due to the rapid expansion imposed upon the telephone companies of the large urban communities, the aims in the design of these systems have been:

1- to be able to control a very large number of switches;

2- to be able to perform the controls in a much shorter time than is possible with the electromechanical devices;

3- to reduce space and power consumption with respect to the classical electromechanical systems;

4- to be able to perform more types of services due to the increasing demand for these by businesses and industries, and the sophistication of modern society;

5- to make use of new types of hardware and techniques that will enable the administration to provide less expensive and more reliable service.

These five aims and the concern for security have guided the most modern designs of telephone systems. Geared thus to the needs of developed areas, they are often not suitable to the conditions and needs of underdeveloped or developing areas.[W3] Nevertheless, there is nothing intrinsic in electronic controls or in stored-

program control techniques that makes them unsuitable for rural areas of developing countries. On the contrary, it can be pointed out that these techniques have some advantages that make them particularly suitable, especially in economic terms, for application in those areas.

The number of stored-program control telephone exchanges actually in operation and production is not very large, but the number of systems proposed and in the stage of experimentation or trial is considerable.

In this section a review of only some of these systems is considered, specifically of those that have some relevance or application to the type of system proposed in the present work.

## The SP-1 Switching System *

The SP-1 switching system is a stored-program control exchange with a capacity that ranges from 2,000 to 10,000 lines.[W5,I1] The central processors (2) and software are organized in the form of time-sharing multiprogrammed machines. Figure 6 shows a simplified block diagram of the system.

The telephony unit, the interface and the central processor functions are similar to the ones described previously in Section II.3. The Program Stores contain the operating programs and data about the lines and trunks used by the CPU. The store consists of 4 modules of 4096 words each. The word's length is 51 bits. The Call

* Northen Electric Lab.

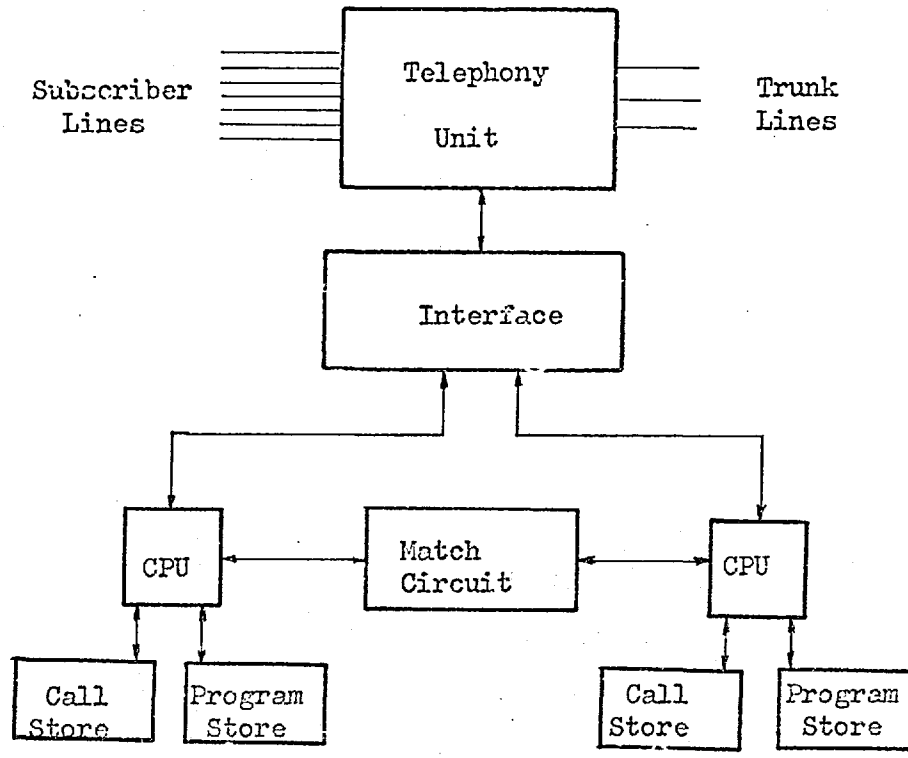



Fig. 6.

Store contains the temporary information about the state of the switches, lines and other devices required for the calls being processed. The Match Circuit (MC) is used to detect mismatching in the information processed by the CPU's.

When mismatching occurs the MC produces an interrupt and provides the control program with the store address of the information producing the mismatching.

This system provides a variety of services like automatic transfer of incoming calls to another number

(call forwarding), abbreviated dialling for frequently called numbers (speed calling), indication of another call waiting while a conversation is in progress (call waiting), transfer to the waiting call and back to the previous call again several times. It also allows a third party to be dialled and included in a conversation (three-way calling) and several other services. The facility of offering so many services and the possibility of adding many more without incurring in extra heavy expenses lies mainly in the fact that the control is in the form of a stored-program.

The N°2 Electronic Switching System

This system* was designed to operate networks that range between 1000 and 10,000 lines in a nonmetropolitan exchange office.[S2,V3] The service features offered to the subscribers are the same as those offered with the earlier and much larger system, the N°1 ESS.[K4]

Figure 7 shows a block diagram of the 2ESS. Its principle of operation is similar to the systems described previously. The controls are duplicated, but working with a common maintenance center. One distinguishable feature is a semi-autonomous input-output section for each processor. This I/O section works in conjunction with the program to 1) scan the lines, and 2) collect dialled digits, tone signals and transmit data. The basic program size for a small exchange is of 75,000 words of 22 bits each.[B5]

*Bell Telephone

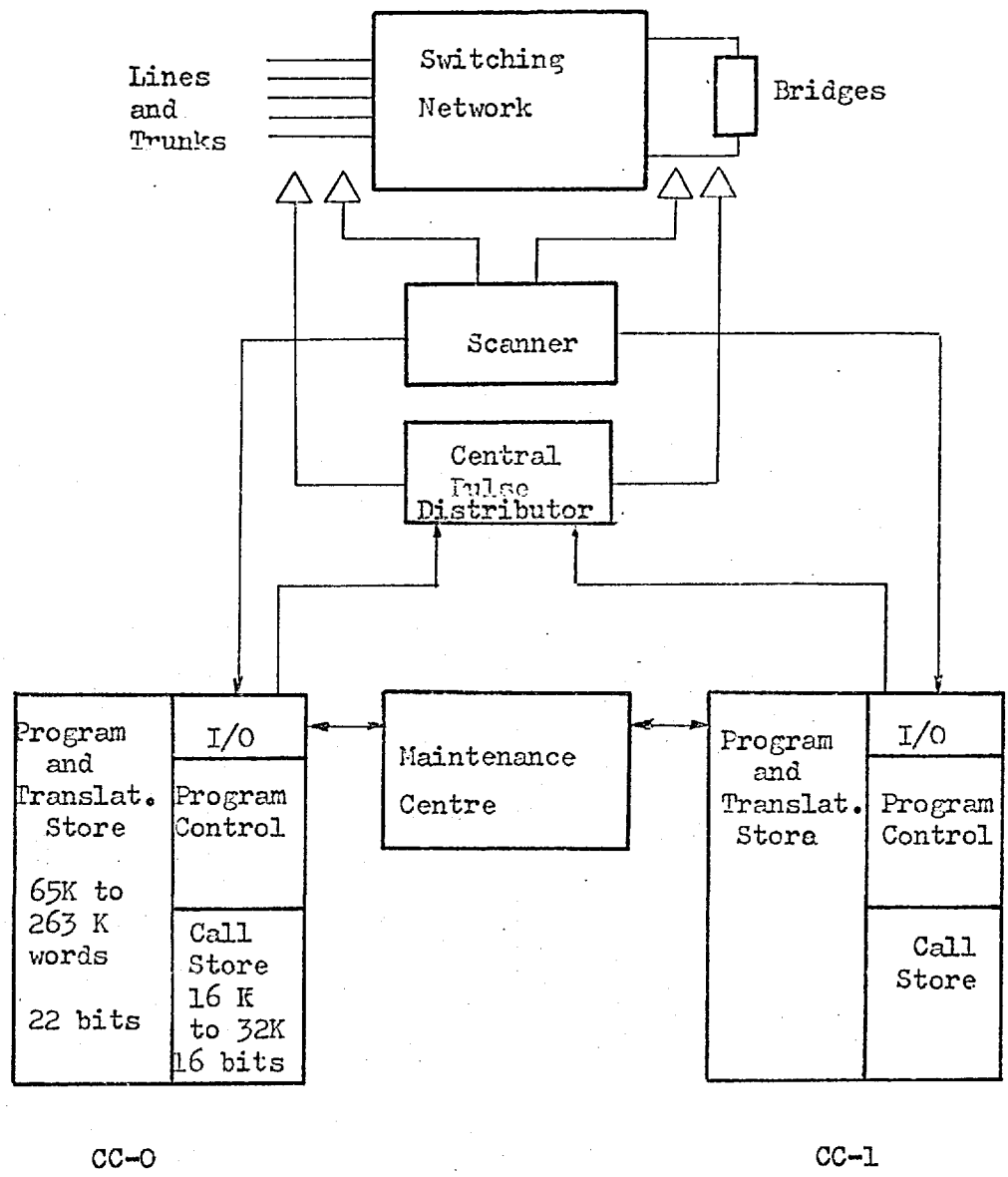

Lines
and
Trunks
Switching
Network
Bridges
Scanner
Central
Pulse
Distributor
Program
and
Translat.
Store
65K to
263 K
words
22 bits
I/O
Program
Control
Call
Store
16 K
to 32K
16 bits
Maintenance
Centre
Program
and
Translat.
Stora
I/O
Program
Control
Call
Store
CC-0
CC-1

Fig. 7.

## The 10 C Switching System

The 10 C switching system* is a stored-program controlled system with a reed relay switching network.[B6]

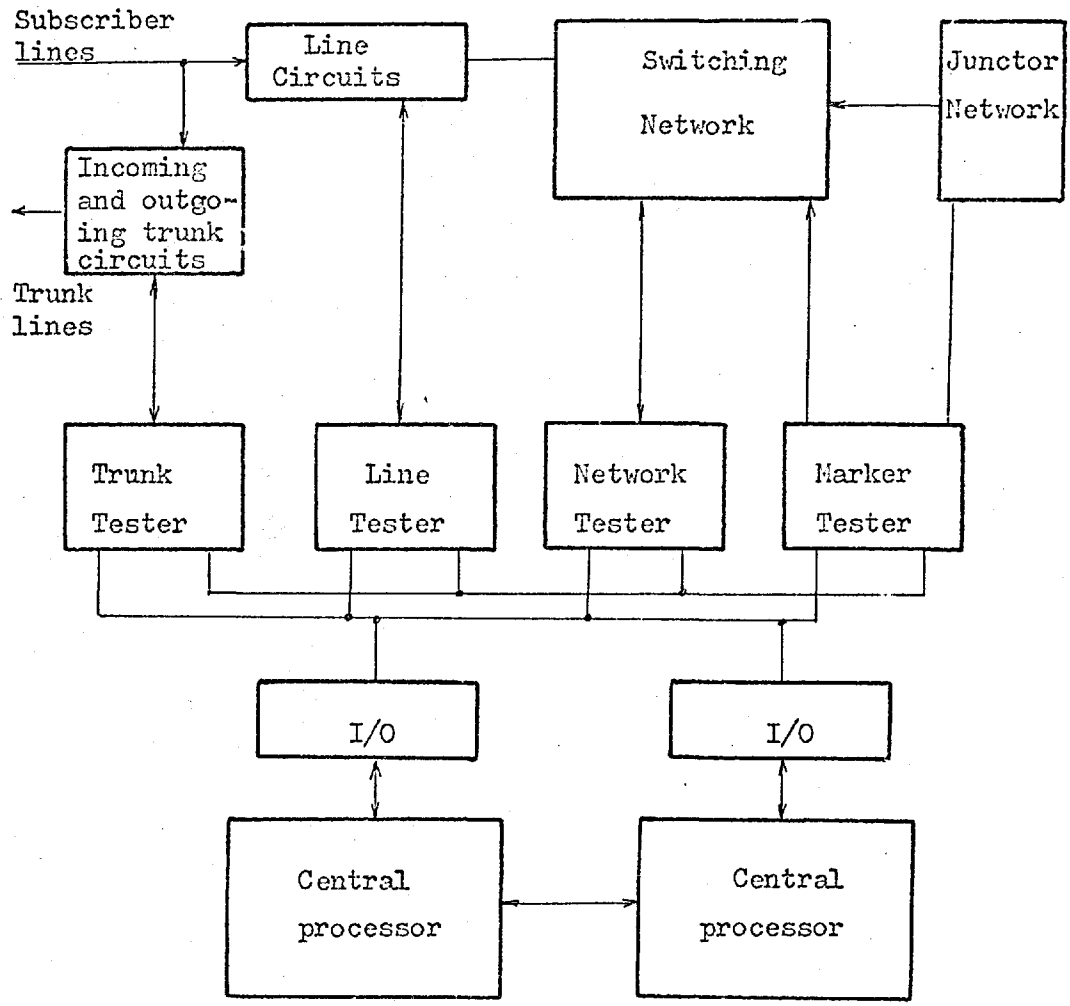



Fig. 8.

*Bell Telephone

The design objectives were based on the use of miniature dry-reed sealed contacts, a stored-program control to increase the flexibility of the system, computer-aided maintenance, and a modular structure. Fig.8 shows a block diagram of the system.

The control is exercised by two processors and duplicated peripheral equipment (I/O), operating on a load-share basis. A 1000 lines system was installed in Antwerp, Belgium in 1967. Its call-processing program contains 13,200 instructions. The on-line test programs and man-machine communication program have 2100 and 3900 instructions respectively. Another 1100 and 750 instructions were needed for start-up and recovery, and routine test programs.

A Satellite Electronic Exchange with Programmed Common Control

This satellite exchange designed jointly by two Danish organizations[01,02*] contains all electronic programmable controls. It is built up in groups of 500 lines each, and uses crossbar switches for the voice paths. The design objectives have been to simplify maintenance by concentrating as much as possible the registering and controlling functions at the main exchange.

The exchange has no registers and the switching is controlled by an electronic marker which is connected to the main exchange by a data link. The subscriber cir-

*Jutland Telephone Company

cuits have only a small number of registers and a small capacitor per subscriber. The exchange is entirely controlled from the main exchange through the data link. The connection of two satellites with the main exchange is shown in Fig.9.

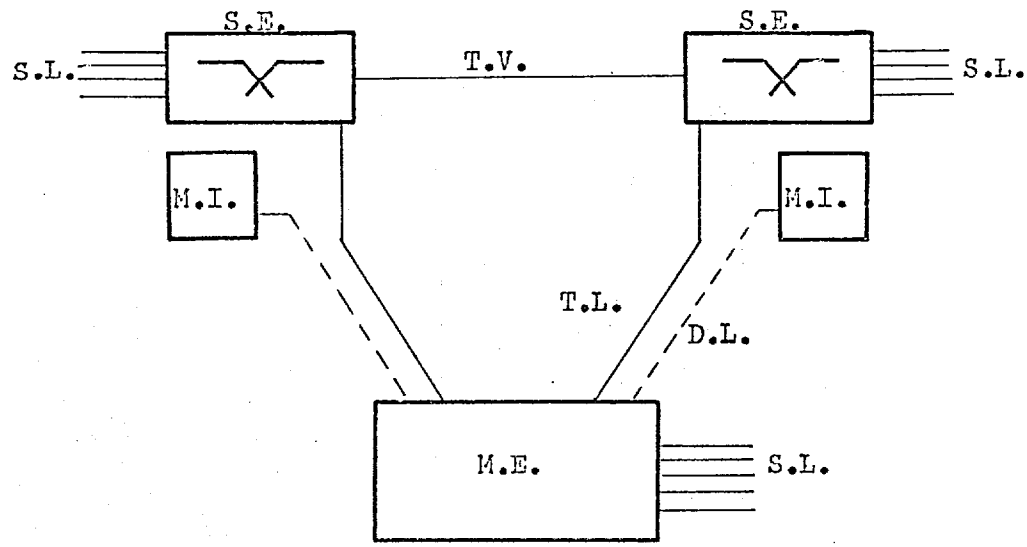



Fig. 9.

The meaning of the symbols used is:

M.I. - marker and interface
M.E. - main exchange
S.E. - satellite exchange
S.L. - subscriber's lines
D.L. - data link
T.L. - trunk line
T.V. - transverse line

Some of the attractive features of this system are: 1- the possibility of using the first switching

stage separately as a line connector for up to 200 subscribers, 2- the use of the common control line data link for several rural exchanges, and 3- the possibility of using transverse lines as shown in Fig.9. The transverse line feature applies also when the calling and the called subscribers belong to satellites that do not have a common main exchange. However, in each satellite exchange there still exist a number of control devices and memories, a calling subscriber number-memory and a signal memory, which could be removed from them if the main exchange were stored-program controlled. The CGS number-memory is used in the case a call is initiated when another one is being established at the exchange. Similarly, the signal memory is used to receive a call when the marker is engaged.

Nevertheless, by placing most of the control in the main exchange and by using an electronic subscriber circuit, it was possible to develop a rural exchange of simple design.

A 10 Line Semi-Electronic Exchange

Many remote areas of Scotland, especially some of the small islands, share the same problem of the developing countries in the field of telecommunications. Small communities far from each other, with a low capacity for generating telephone traffic create the problem of whether to provide the area with some sort of switchboard or a

public telephone. When public telephones are installed, potential users are often hindered to reach the telephone box because they have to travel a considerable distance, and only in extreme cases do they make use of telephone communication. To solve this problem in the Scotland Telecommunications Headquarters[K1] a 10 line semi-electronic exchange for remote areas was designed.

This unit permits the establishment of local and long distance calls. The long distance connection with the main exchange consists of a single channel v.h.f. radio link. The maximum capacity of ten lines is due to the single digit numbering scheme necessary to maintain simplicity in the equipment.

When this system was initially designed (1962), a semi-electronic device was considered as the best arrangement, where electromechanical relays perform switching and control functions. Solid state elements are used only for the tone generators and the line finder unit.

A new design of this equipment with a minicrossbar switch and an all-electronic control should reduce the power consumption and allow simultaneous calls. As it is now, in the system, a trunk call to the main exchange cannot be established if a local call is in progress. This limitation was imposed to reduce the power consumption of the batteries.

This type of exchange does not require any special housing and can be replaced by a larger one without real capital loss because it can be easily installed in another place when needed.

# CHAPTER III

# A NETWORK WITH STORED-PROGRAM CONTROL EXCHANGES

## III.1 SYSTEM DESCRIPTION

The system proposed here provides for automatic switching of voice paths connecting subscribers located in small communities scattered over a wide area and a larger group of subscribers located in a more densely populated area. The switching network for the voice paths is a dispersed one, as shown with solid lines in Fig.10, where every remote exchange (satellite) has its own switching system for local calls. The main switching centre, located in the larger town, provides switching of the local subscribers for local calls and for long distance calls to the remote exchanges or to other switching centres.

The block diagram of the signalling system is shown with dotted lines in Fig.10. As can be seen, the signalling network is a centralized one. Every remote exchange is connected to a centralized control by means of a two-way data link (including the main switching centre). The centralized control consists of a central processor and an interface between the data links and the processor. The data that is transferred in the two-way data link, between the exchanges and the interface,

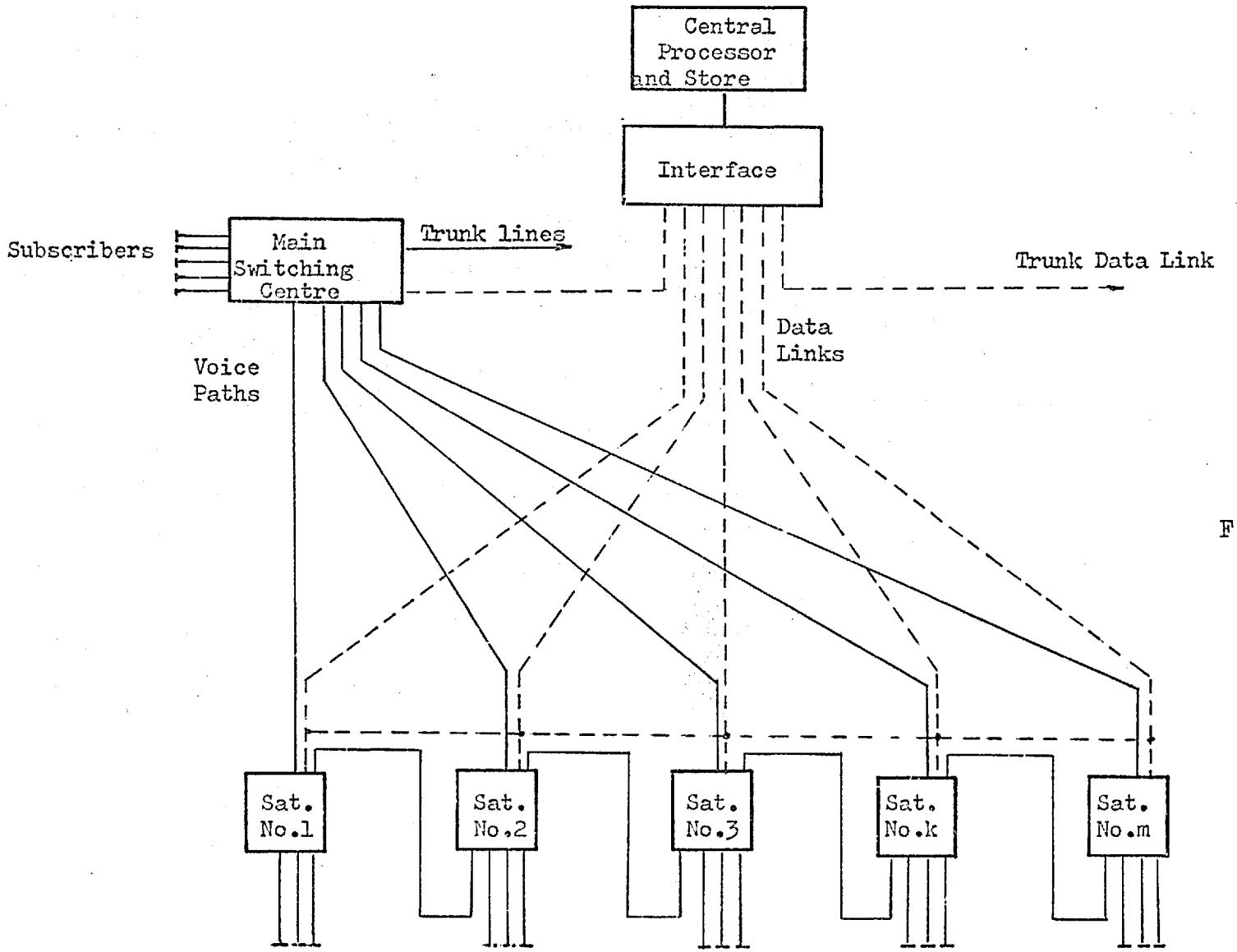

Central Processor and Store
Interface
Subscribers
Main Switching Centre
Trunk lines
Trunk Data Link
Data Links
Voice Paths
Sat. No.1
Sat. No.2
Sat. No.3
Sat. No.k
Sat. No.m

Fig. 10.

is in the form of serial binary digits. If the central processor is chosen to be a parallel machine, the interface will also act as a serial-parallel converter and viceversa.

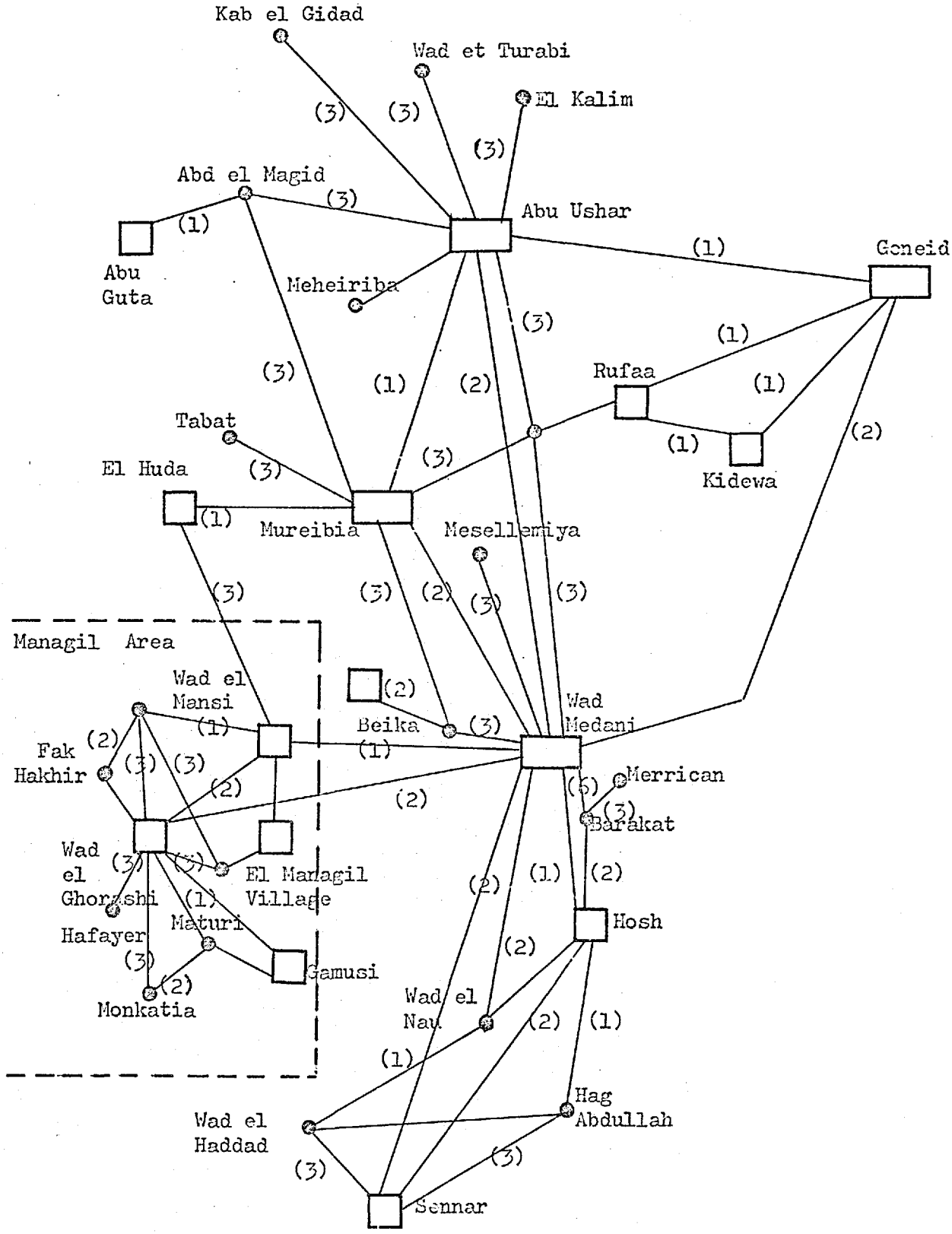
The main exchange will be that of the more densely populated town or city within the region. In most cases it will be the provincial capital. It will most probably be housed in the same building as the central processor. All those exchanges within the controlled zone other than the main exchange, will be known as satellite exchanges. Both the main exchange and the satellite exchanges within a controlled zone will be regarded as remote exchanges. The word 'remote' refers here to the way the control is exercised by the central processor, and not to the physical distance from it.

The aim of this system is to keep the equipment necessary to provide telephone connection for decentralized groups of subscribers (rural areas) at a minimum. Only one central processor and only one link between each satellite exchange and the main exchange is used. But when the economic situation of the particular administration allows it, the system can be expanded either in size or in security, by duplicating the processor and the links with the satellite exchanges. Security can also be improved by providing a link between the satellite exchanges, as shown in Fig.10 with dotted lines. In this way, if the link between the central office and a satellite exchange

is accidentally interrupted, the exchange control is exercised through the data link of a neighbour satellite exchange. The voice path is also provided through the neighbour satellite exchange, but all the traffic of the first is handled as if it were a single subscriber of the latter. In such a case the grade of service of the satellite exchange with disrupted link will be reduced but the exchange does not remain isolated.

The links interconnecting the satellites have the double purpose of being auxiliary routes during a temporary disruption of the main links and lateral paths for calls between satellite exchanges. In this way a 'mesh' type of network is obtained, in contrast to the 'star' type of structure. Advantage is thus taken of the security and reduced junction costs characteristic of the first type, when large areas have to be covered. To illustrate this point next page shows a diagram of the trunking network of the Gezira and Managil areas in Sudan where the network is arranged in this form.[P7]

The switching control and signalling of the exchanges is exercised by a processor with a stored-program. The program is permanently present in the main core memory of the processor and can be easily altered by changing instructions or storing new ones. This is one of the most attractive features of the system, which allows the changing of priorities according to the needs, the expansion of the system, the addition of new subscribers and




Trunk and Junction Network of the Gezira and Managil Areas

testing procedures, by making changes only in the software of the system.[W4]

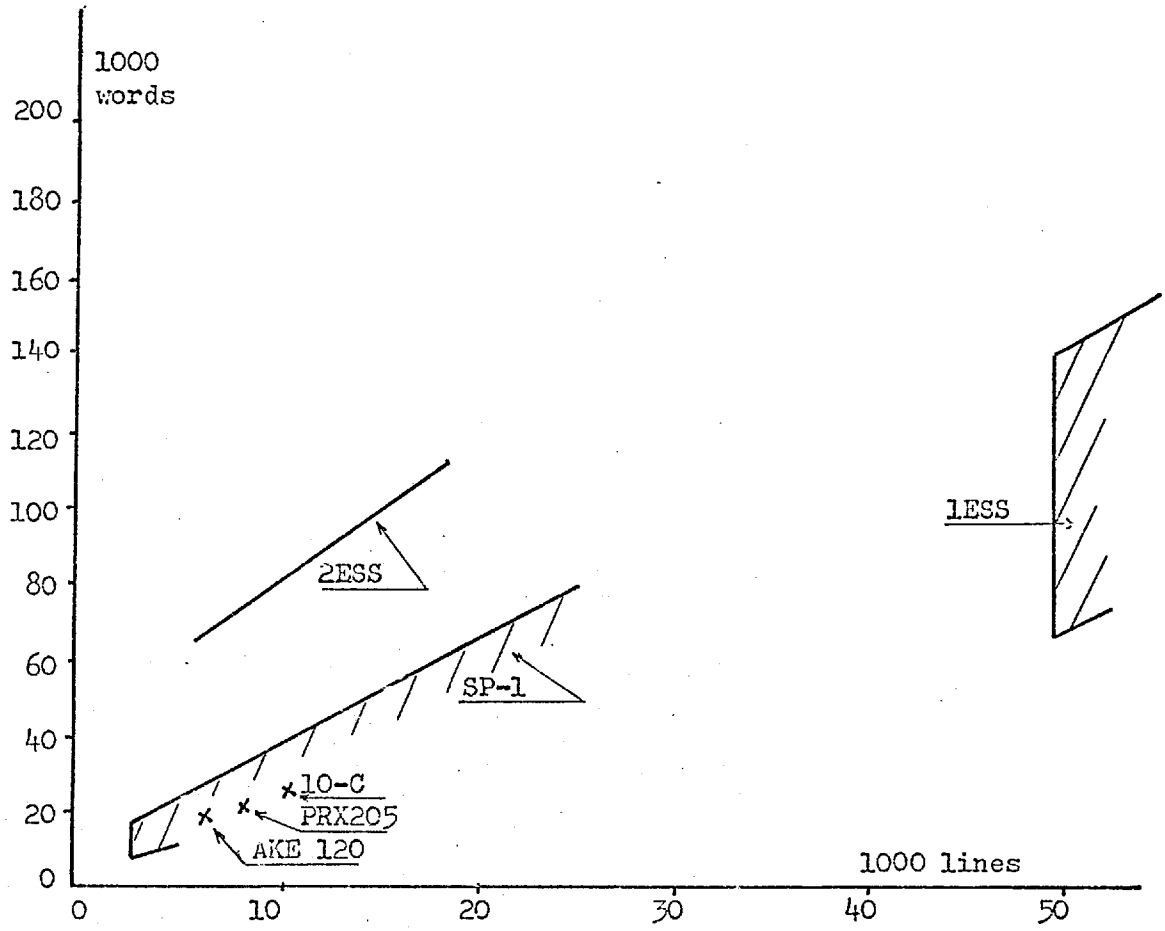
The program can be written either in assembly language or in a higher level language. Most of the stored-program controlled systems, either installed or being developed, are programmed with assembly languages or some sort of a low level language. The use of high level language has been investigated[C4,K3], but up to the present moment, the main difficulty lies in the requirement of excessively large memory capacities.

In Section III.2 the strategy of operation of a controlling program is shown. Statements and instructions were written in the standard form of FORTRAN IV to allow its simulation in the University of London computers.

The Central Processor

This is basically a digital computer of types normally available in the market for general purposes. A simple input/output device to store and modify the program in the memory is necessary. The least expensive device is a typewriter terminal with a paper tape puncher and reader. The central processor unit (CPU) organization is that of a general purpose computer and variation between one manufacture and another will not affect the structure of the whole system, since the control strategy of the exchanges is concentrated in the stored program.

Fig. 11.




References to Fig.11:

SP-1[W5,I1]

PRX 205[V2]

AKE 120 (Tumba)[K2]

1ESS[K3]

10-C[B6]

2ESS[S2,B5]

The memory capacity necessary will depend mainly on three factors: the number of subscribers and of satellite exchanges controlled by the processor, the traffic handled and the number of services provided. It is very difficult to draw a function of the memory size in function of the above mentioned variables. Nevertheless, an idea of the memory capacity can be obtained from some existing system. Fig.11 shows the approximate number of memory words used in six known systems. It is important to point out that these systems are designed to operate with sophisticated networks and therefore require a memory size much above a minimum requirement.

The Data Link

The transfer of controlling data between the remote exchanges and the interface of the central processor is carried out by a data link in the form of serial binary information. Data incoming to the central processor carry information about off-hook conditions, trunk calls, dialled digits, etc. Outgoing data carry information interrogating the state of the various devices, or instructions to connect or disconnect them.

In a simple network the voice path and the signalling can be carried out on the same pair of wires. The control signal can be sent on multi-frequency carriers. This mode of operation permits the use of data link for control purposes and for supervision while the voice channel

is being used.

In relation to the speed of signalling, no real problem should be encountered since the signalling traffic is expected to be low. The signalling speed on the data link is mainly determined by the following factors:

1- the number of subscribers on the exchange;

2- the average calling rate per subscriber;

3- the number of messages per call;

4- the number of bits per message;

5- the time constraints imposed;

6- the propagation time through the link; and

7- the average processing time of the messages.

The following figure (Fig.12) is a representation of the message handling during the set-up of a call. On the left is the satellite exchange and on the right the central processor. The vertical axis represents time.

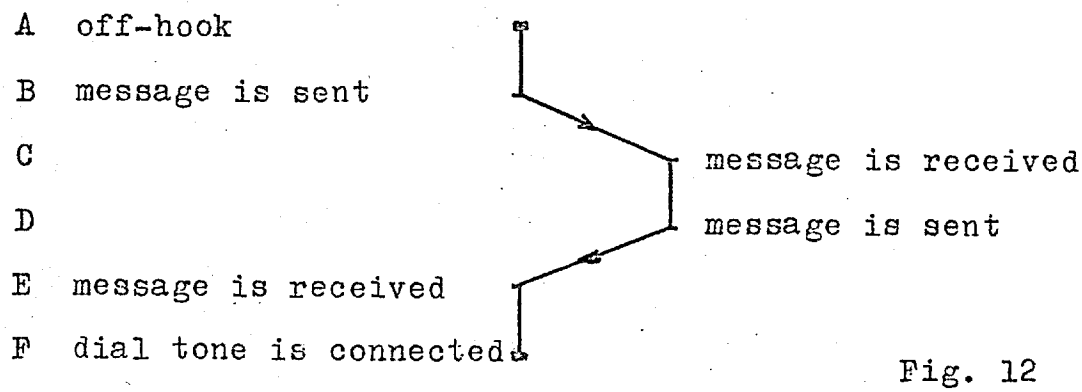



Fig. 12

At time A a subscriber off-hooks his telephone; at time B an off-hook message is sent to the central processor. B-A is the scanning time of the line units. At

time C the message is received at the central processor. The time B-C is the propagation time through the data link. The time D-C is the processing time of the message and the generation of a message to instruct the satellite to connect dial tone to the calling subscriber. At time D the instruction is sent; at time F the dial tone generator is connected. The time F-A is the time necessary for the subscriber to obtain dial tone. This is usually a constraint set by the design objectives.

The number of subscribers (1) and the average calling rate (2) will determine the total calling rate. To the designer the total calling rate is a given parameter. The propagation time (6) is also a parameter which depends on the distance between the main processor and the satellite exchange, and usually negligible. Thus, only the number of messages per call (3) and the number of bits per message (4) are left to the designer to optimize the signalling system. Let us consider the following example:

| | |
|---|---|
| the average calling rate per subscriber is | C= 3 calls/B.H.x subs. |
| the propagation time is | $t_g$= 0.5 ms |
| the average processing time is | $t_p$= 1.0 ms |
| the times (B-A) and (F-E) are | $t_1$=$t_2$= 2.0 ms |
| the number of subscribers is | n = 500 subscribers |
| the number of messages in each direction per call is | MC= 10 messages/call |
| the number of bits per message is | b = 32 bits/message |

the number of messages per set-up in each direction is $MU = 1$ mess/call

The average rate of information R(bits/sec) transmitted on the data link in one direction is:

$$R = \frac{MC \cdot b \cdot n \cdot C}{3600} \quad \text{bits/second}$$

The number of bits per call required to set-up a call is:

$$S = MU \cdot b \quad \text{bits/set-up}$$

If the rate of transmission adopted for the data link is R, then the time to send the messages serially in one direction is:

$$T_t = \frac{S}{R} = \frac{MU \cdot b \cdot 3600}{MC \cdot b \cdot n \cdot C} = \frac{MU}{MC} \times \frac{3600}{n \cdot C}$$

In this example $T_t = 0.24$ sec., and,

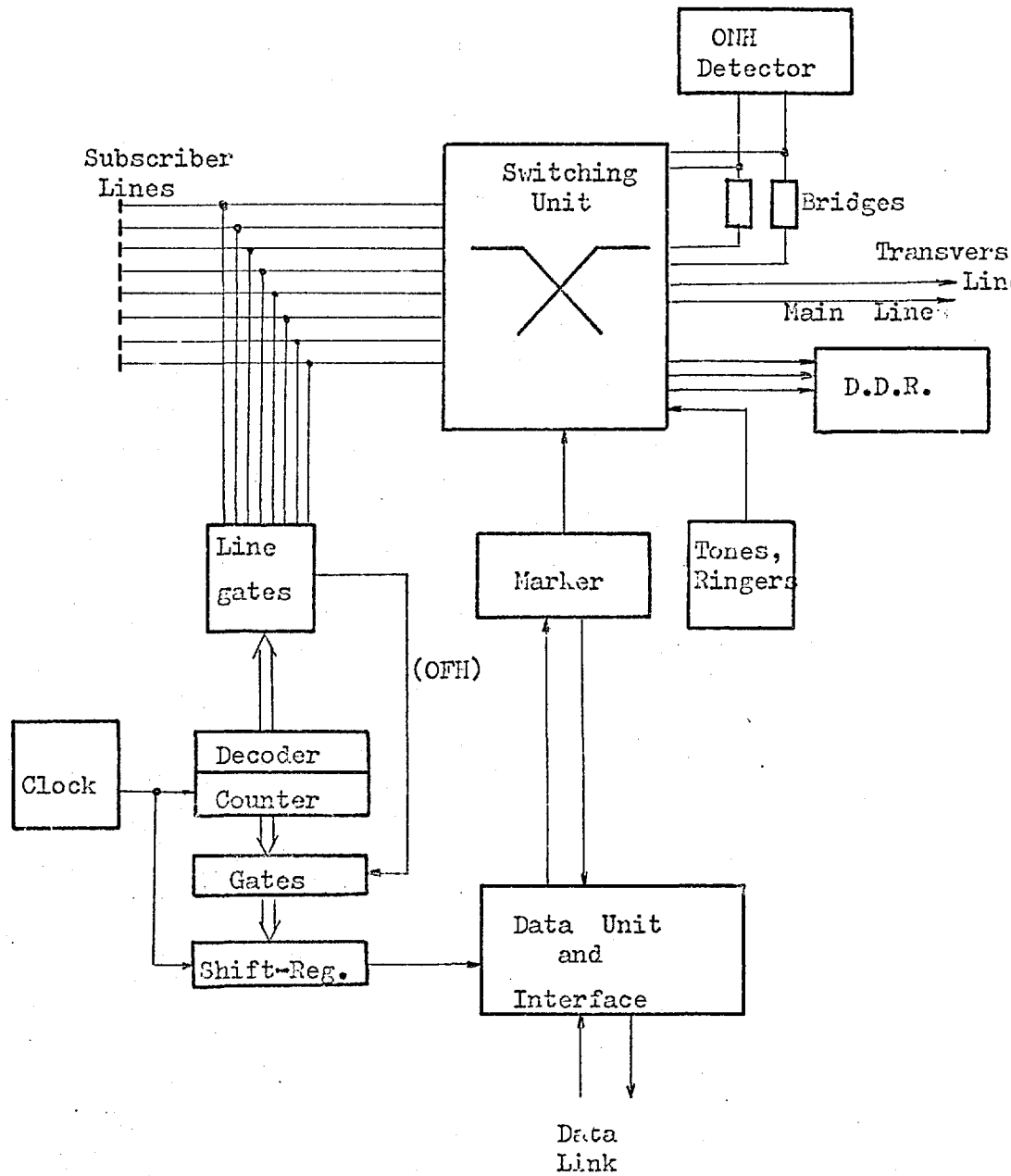
$T_s = 2t_g + t_1 + t_2 + t_p + 2T_t = 0.6 + 4.0 + 1.0 + 480 = 485.6$ ms

that is, in less than half a second the calling subscriber will receive dial tone. For larger traffic, a higher rate of transmission through the data link would be necessary. But, a rate of transmission of 500 bits/sec. will be sufficient for the type of network and traffic considered in this work.

The Satellite Exchange

Fig. 13 shows a diagram of a satellite exchange to be remotely controlled by a central processor.

Fig. 13.



The switching unit, preferably of the reed-relay or mini-crossbar type, contains one or two switching stages according to the number of lines in operation.[E1] The on-hook detectors can either be attached to the bridges or to the line gates (units). In the first case only one detector per bridge is necessary; in the latter case, one per line. The line unit operates in the following way. The counter, triggered by the clock, produces a binary number of M digits. With M bits, it is possible to scan up to $N=2^M$ subscribers' lines. The counter cycles from: 000...0 to 111...1 and back to 000...0 . This binary number M is decoded and a 1 out of N outputs is obtained to gate an AND-circuit. In this way when the line K produces an off-hook condition, this information is gated together with the decoder output and an off-hook signal is generated. This signal gates into a shift register the value of the counter at that instant (K), that is, the subscribers' number. An off-hook message sent to the central processor will generate there an instruction to the satellite to connect the subscriber to the dial tone generator and to a single digit buffer-register, called Dialled Digit Receiver (DDR). Dialled digits will be sent to the DDR and from there to the central processor through the data unit. Every dialled digit will release from the data unit a message to the central processor. When the dialling is completed an instruction to operate the marker will be sent by the processor to connect the calling subscriber to the called subscriber.

The scanning device of the line unit can be used to detect whether a called subscriber is free or engaged, that is, whether the subscriber's equipment is on-hook or off-hook. When this facility is incorporated in the satellite, prior to the connection of the ringing current, an interrogation is made (a message is sent by the central processor) on whether the called subscriber is engaged or free. Upon receiving a reply message at the central processor, instructions (messages) are sent to the satellite to connect ringing current to the called subscriber and ringing tone to the calling subscriber, or otherwise, busy tone to the calling subscriber.

When an on-hook signal is detected at the bridge, a message is sent to the central processor with this information to clear the call.

The satellite exchanges considered here are of a capacity ranging from 50 up to 500 subscribers' lines. For smaller units, automatic service can be provided with an extension of the first switching stage used as a line connector.[01,02] In case a small group of subscribers is very far from the nearest satellite exchange, a 10-line type automatic exchange[K1] could be used.

## III.2 THE STORED PROGRAM

Fig. 14 shows a simplified flow chart of the stored program. The computer, controlled by this program, will be idling if no call is in process. Only when an

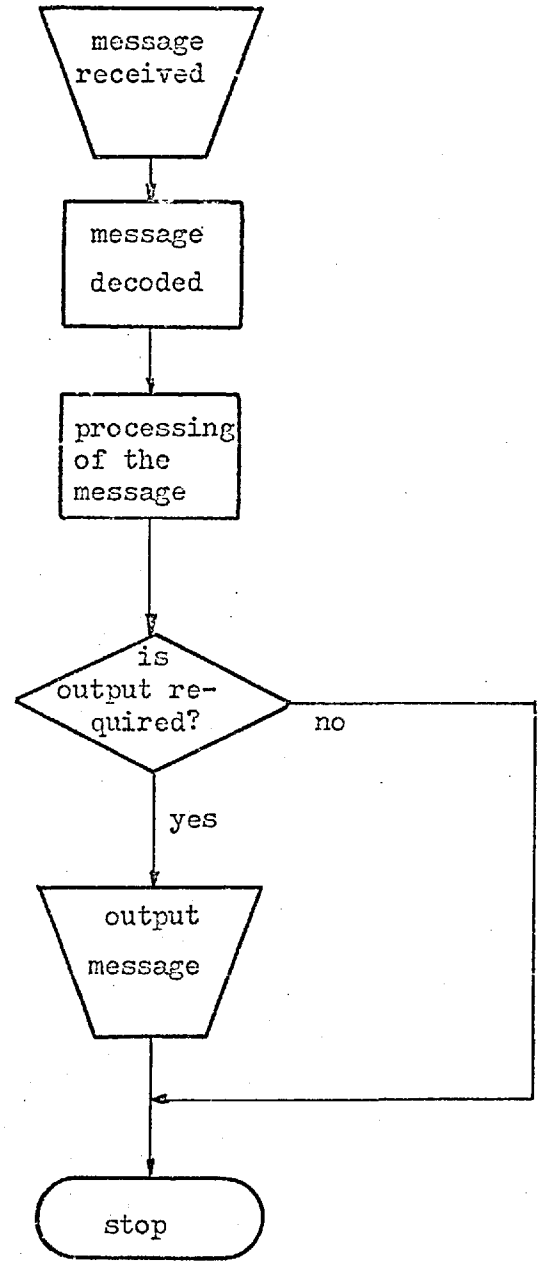

message received
message decoded
processing of the message
is output required?
no
yes
output message
stop

Fig. 14.

interrupt is originated by an incoming message, the machine starts to operate. Messages arriving from the exchanges are handled one at a time. A priority scheme in the interface prevents conflicts when two or more exchanges signal at the same time (try to produce an interrupt). This priority scheme is a standard feature of most I/O interfaces for teletype connections.

The stored program is organized in the form of blocks that perform separate tasks. Each of these blocks may comprise one or more subroutines.

A message (calling signal) arriving from one of the controlled exchanges into the interface, will originate an interrupt at the central processor. This message carries with it the address of origin and information about the type of call that has been sent. If the interrupt is accepted, the information of the message is analysed to determine the block to be called in order to process the message.

At this point, it is convenient to define the terms message, signal and call, since they are the main subjects of this section. The terms signal and message are used interchangeably here, since signal is the electrical change in the wires, and message, the information carried by the signals.

From the processing point of view, a call can be defined as the handling, by the central processor, of a set of messages, some of which arrive to it from the

remote exchanges, and others are generated and sent by the central processor to the remote exchanges.

Calls also fall in two categories: local and long distance, or trunk calls. For the purpose of this work, local calls are all those within a zone controlled by a central processor, and long distance calls are those between subscribers that do not belong to the same zone. The consideration of traffic between the satellite exchanges of the same controlled zone as local, is due to the nature of the areas considered here. Typically, they would be agricultural areas where the subscribers, private or official, are scattered and not concentrated in densely populated urban areas. If the geographical distance were the criteria to determine whether a call is local or long distance, one would find in a sparsely populated area that the calls are all long distance, making the tariff structure more complicated and defeating the purpose of promoting the service.

Long distance calls originating at any of the satellite exchanges or at the main switching centre of a controlled zone are called outgoing calls or outgoing trunk calls. Calls originating outside a controlled zone are called incoming calls, or incoming trunk calls.

A message consists of three fields. The first field is a code number indicating the origin of the message. The meaning of the information of the other two fields depends on the nature of the message.

The messages arriving to and being sent from the central processor have in general the following format:

| CODE | ADDRESS | INSTRUCTION |
|---|---|---|

Table 3 shows the meanings of the code numbers and of the address field. The code structure shown here serves only the purpose of demonstrating the mechanism of the stored-program and may not be the most adequate. The size and format of the messages will largely depend on the type of central processor adopted and on the network structure.

The third field can be considered as the actual message. Table 4 shows the use of this field. The symbols used in that table are:

C = connected
NC= not connected
DT= dial tone
DD= dialled digit
R = ringing
B = busy
U = unobtainable
G = granted
EN= exchange number
RN= buffer-register number
TN= tag number
DC= disconnected from subscriber
RC= ringing current
RT= ringing tone
BT= busy tone
UT= unobtainable tone

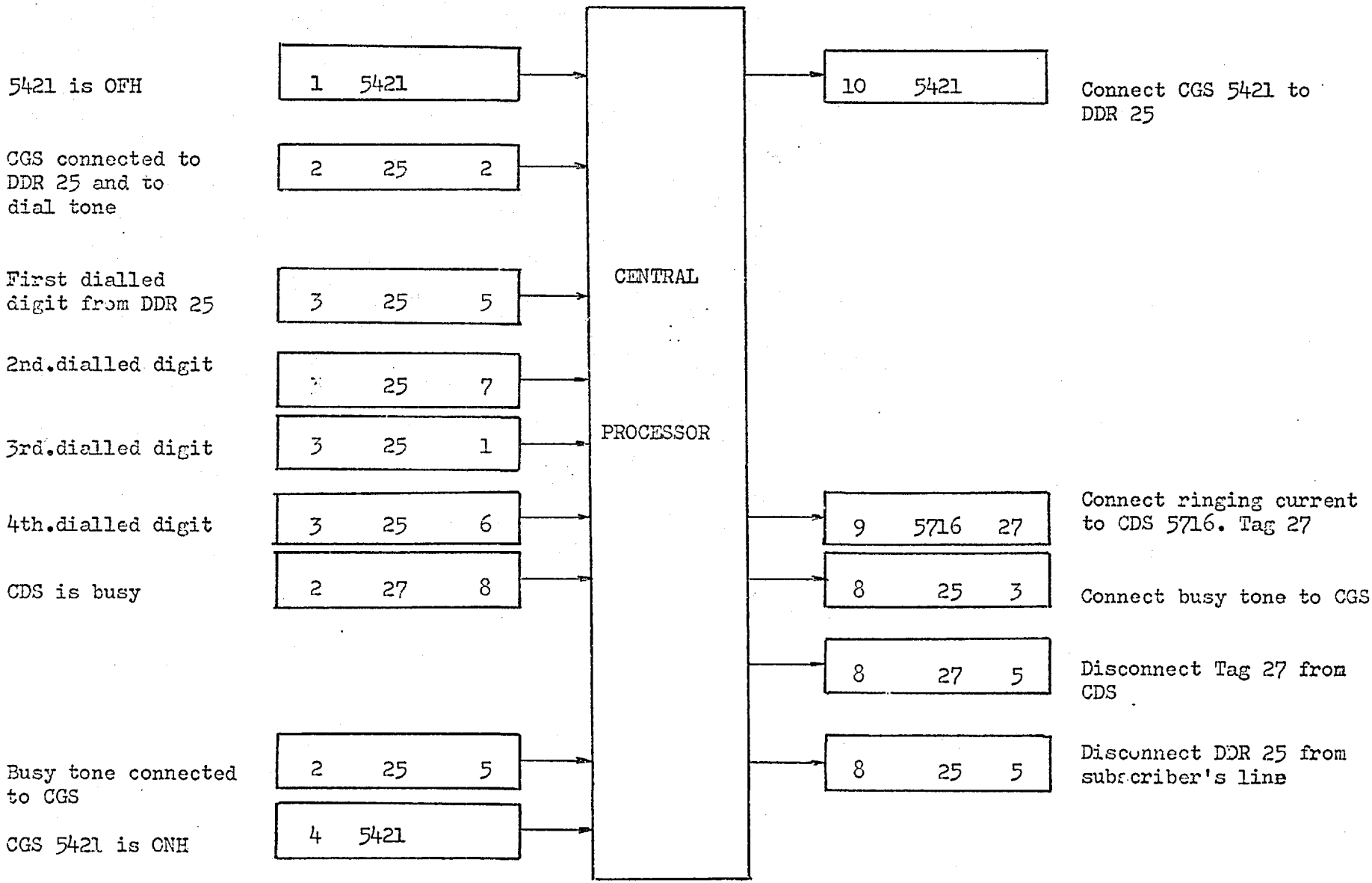
TABLE 3

| CODE | Meaning<br>This message comes from: | ADDRESS | Meaning |
|---|---|---|---|
| 1 | an OFH condition | CGS | a calling subscriber's number |
| 2 | a remote exchange informing the status of a subscriber's equipment | TAG | an identification number |
| 3 | a dialled digit | DDR | a dialled digit receiver number |
| 4 | an ONH condition | CGS | a calling subscriber's number |
| 5 | an incoming call requesting store space | FILE | a file number |
| 6 | a processing centre answering a trunk call | FILE | a file number |
| 7 | an incoming call dialled digit | FILE | a file number |
| 8 | a processing centre instructing the connection or disconnection of a device | TAG | an identification number |
| 9 | a processing centre requesting ringing current for a CDS | CDS | a called subscriber's number |
| 10 | a processing centre requesting that a dialled digit receiver and dial tone | CGS | a calling subscriber's number |

TABLE 4

| When CODE | Then, the meaning of Instruction field is: 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ... |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | - | - | - | - | - | - | - | - | - | - | - |
| 2 | - | OFH | C+DT | NC | RT | BT | UT | R | B | U | - |
| 3 | DD | DD | DD | DD | DD | DD | DD | DD | DD | DD | - |
| 4 | - | - | - | - | - | - | - | - | - | - | - |
| 5 | EN | EN | EN | EN | EN | EN | EN | EN | EN | EN | ... |
| 6 | - | R | B | U | G not G | | - | - | - | - | - |
| 7 | DD | DD | DD | DD | DD | DD | DD | DD | DD | DD | - |
| 8 | - | RC | RT | BT | UT | DC | - | - | - | - | - |
| 9 | - | TN | TN | TN | TN | TN | TN | TN | TN | TN | ... |
| 10 | - | RN | RN | RN | RN | RN | RN | RN | RN | RN | ... |

To illustrate the application of the different codes, the example in the next page shows a simplified case of the handling of messages in the processing of a call. On the left are the messages sent by a remote exchange and arriving at the central processor. On the right are the messages sent by the central processor to the remote exchange. The call commences with an off-hook condition's message sent by the subscriber 5421. The sequence of messages is self explanatory from the chart.


5421 is OFH
1 5421
CGS connected to DDR 25 and to dial tone
2 25 2
First dialled digit from DDR 25
3 25 5
2nd.dialled digit
25 7
3rd.dialled digit
3 25 1
4th.dialled digit
3 25 6
CDS is busy
2 27 8
Busy tone connected to CGS
2 25 5
CGS 5421 is ONH
4 5421
CENTRAL
PROCESSOR
10 5421
Connect CGS 5421 to DDR 25
9 5716 27
Connect ringing current to CDS 5716. Tag 27
8 25 3
Connect busy tone to CGS
8 27 5
Disconnect Tag 27 from CDS
8 25 5
Disconnect DDR 25 from subscriber's line

## III.3 THE PROGRAM'S BLOCKS

The processing of a message is performed by first decoding the message and then referring it to the proper block. These blocks are in essence subroutines and the number of them will depend largely on the number of services available to the subscribers. However, a minimum number of them is necessary to handle the basic services. In the subsequent sections seven of such blocks are briefly explained. A flow chart of those blocks and a basic FORTRAN IV listing of them is shown in Appendix V.

### The off-hook block (OFH)

When an 'off-hook' condition is detected by a scanning device, at a remote exchange, a message is sent to the central processor. It is decoded by the stored program and referred to the OFF-HOOK block. In this block, an enquiry is made, on whether or not the service is to be granted to the calling subscriber (CGS), i.e., normal subscriber (NS) versus temporarily out of service (TOS).

In a very simple system, the subscriber's number (SN) is at the same time the equipment number (EN). In such cases, in order to have a line temporarily out of service, it would be necessary to disconnect the line physically. Otherwise, a record should be kept in the memory of those subscriber's numbers not allowed to have service.

This would increase the size of the memory and the processing time of a call, and hence the cost of the

exchange. For a more elaborate system, a translation of the subscriber's number to the equipment number is necessary. Then the information of a buscriber TOS may be obtained from the translator.

If the service is to be granted, the number of the calling subscriber is stored in one of the memory allocations intended for it.

In the satellite exchanges there are dialled digit receivers to store temporarily the dialled digits. The number of these receivers at every satellite will depend on the magnitude of the traffic handled by the exchange.

In the main storage, an area is reserved to store the current calls. The following Table 5 shows the arrays kept in that area, which are necessary to process a call.

As an example, let us assume that a calling subscriber with the number 5421 corresponds to an exchange that has two DD receivers, and let these receivers be identified with the numbers RN = 24 and RN = 25 . Upon receiving the off-hook message:

| | | |
|---|---|---|
| 1 | 5421 | |

and having confirmed that 5421 is a valid and normal subscriber, a place in the storage is allocated for the call and a DDR in the remote exchange is connected to the subscriber's line.

TABLE 5

| I | LINK | DDR | TAG | FILE | CGS | EXNU | DD | ND | BUSY | RING |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 25 | 0 | 0 | 5421 | 00 | 0000 | 0 | 0 | 0 |
| 2 | 3 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 3 | 4 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 4 | 5 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| . | . | . | . | . | .... | .. | .... | . | . | . |
| . | . | . | . | . | .... | .. | .... | . | . | . |

| I | RN | IRN |
|---|---|---|
| 1 | 0 | 0 |
| 2 | 0 | 0 |
| 3 | 0 | 0 |
| . | . | . |
| . | . | . |
| 25 | 1 | 1 |
| . | . | . |

| I | FN | IFN |
|---|---|---|
| 1 | 0 | 0 |
| 2 | 0 | 0 |
| 3 | 0 | 0 |
| . | . | . |
| . | . | . |
| . | . | . |
| . | . | . |

To allocate a storage and a DDR to the call, the subroutine ALLOC is used. This subroutine searches for the 'next' available space in the list of calls. The next available space is not necessarily the one after the last in the list, since, after a call has terminated, the

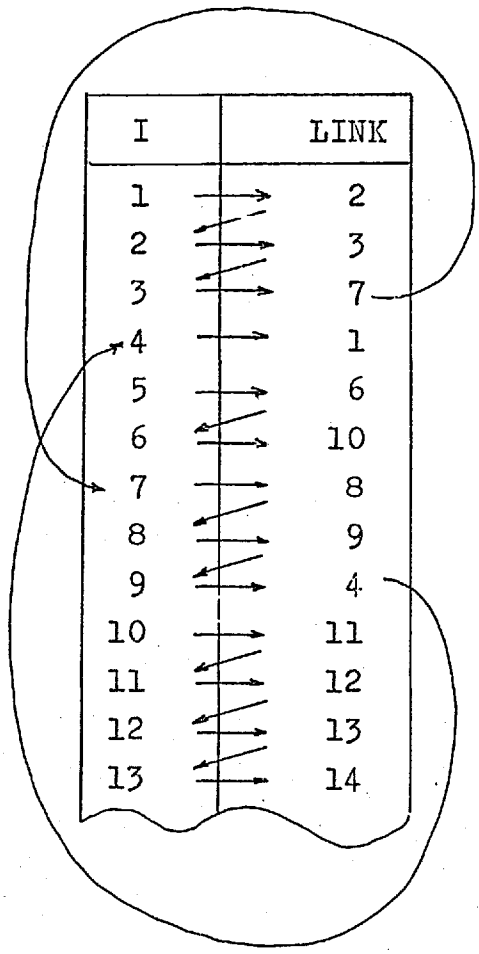
information stored for that call is deleted and the space becomes available for a future call. Therefore, the next available space may be a space in the middle of the list. An array called LINK keeps the addresses of the calls in process in such a way that in any given moment all addresses are 'linked' together. This allows the program to know at any given moment which is the next available space.

Table 6 shows an example of the use of this array. The array and a variable called PROX, containing the address of the next available storage space, allow the subroutine ALLOC to allocate a call in the memory without delay. An alternative method would be to have an array with a 'flag bit' to indicate if either the space is occupied by a call in process or if it is free. Then, to allocate a call, the program has to scan the list of 'flag bits' to find a free space, causing every time a corresponding delay. This waiting time can be avoided by using the LINK mechanism. Every time a call is deleted the numbers in the LINK array are reordered and PROX is set to a new value.

After having allocated a space for the call, the subroutine RENU is used to find a free DD receiver in the remote exchange for the calling subscriber's line. This subroutine searches a list of receivers' numbers to find which one is free and assigns it to the CGS. For this purpose, three arrays called IRN, RN, and DDR are used.

Let for instance the receiver 25 be free and be

TABLE 6



| I | LINK | |
|---|---|---|
| 1 | 2 | call in process |
| 2 | 3 | call in process |
| 3 | 7 | call in process |
| 4 | 1 | call in process |
| 5 | 6 | free space |
| 6 | 10 | free space |
| 7 | 8 | call in process |
| 8 | 9 | call in process |
| 9 | 4 | call in process |
| 10 | 11 | free space |
| 11 | 12 | free space |
| 12 | 13 | free space |
| 13 | 14 | free space |

PROX = 5

assigned to a CGS allocated in the address I. Then, RN(25) = I , IRN(25) = 1 and DDR(I) = 25 . See example on Table 5, where I = 1 .

If no receiver is free the call is considered a lost call. The information about the off-hook condition kept in the memory, is deleted by means of a subroutine called DELETE and the subscriber's line receives no dial tone. A new off-hook message has to be received by the central processor in order to initiate the call again.

The dialled digits block (DD)

This block handles the messages carrying dialled

digits from calling subscribers. When such a message is received, the calling subscriber's memory allocation is found through the second field of the message which is a receiver number. Let that receiver number be 25 . Then, RN(25) contains the address of the calling subscriber's allocation in the main storage. Next, the ordinal of the just received digit is determined, i.e., the first digit, the second digit, and so on, and stored in the proper place. After receiving the last digit, and when the subscriber number (SN) is different from the equipment number (EN), the called SN is translated to the corresponding EN. If the call is an outgoing long distance call a file number is assigned to the call and a message is sent to the processing centre of the CDS. The FILE and the FN arrays, kept in the main storage, are similar in nature to the DDR and the RN arrays and are used as a reference number between central processors to handle trunk calls. If it is not an outgoing trunk call, but a call to a normal subscriber within the same control zone, then a tag number is assigned to the called subscriber's line and connected to the ringing current generator.

The Data Unit block (DAUNIT)

Messages sent by and to the data units in the remote exchanges concerning the connection of DD receivers, ringing current or tone generators and markers, are handled by the block called DAUNIT. This block decodes the

arriving messages and proceeds according to the corresponding instruction. One type of instruction (2 and 3) is to inform the central processor whether a DD receiver has been connected or not to a calling subscriber's line to receive dialled digits, and also whether the subscriber has obtained the dialled tone or not. The OFH instruction (1) is to inform the central processor that a CDS is OFH and the information kept in the main store about the call can be deleted.

Other instructions refer to the connection of ringing, busy or unobtainable tone (4, 5 and 6) to a calling subscriber. And finally there are three other instructions (7, 8 and 9) from a called subscriber indicating the status of his equipment, either ringing, busy or unobtainable. Upon processing the last type of messages the central processor proceeds to instruct the remote exchanges to connect the corresponding tone to the calling subscriber.

The Trunk Calls' Blocks

Three blocks called TRUNKCD, INCOM and INCOMDD blocks are used to handle the incoming and outgoing long distance calls. The operations performed by these blocks will be described using the example shown in the diagram on the following page:

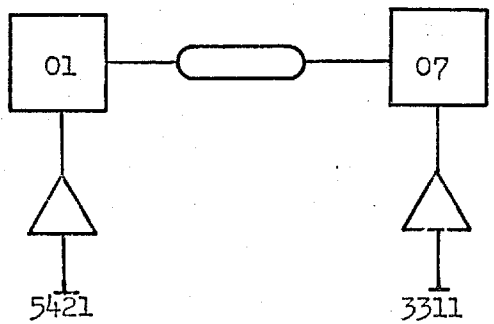
1- Similarly to the example of Section III.2, the subscriber 5421 is connected to the DD receiver 25 and had

1) CGS 5421 of zone 01 is connected to DDR 25 and has dialled 6 digits

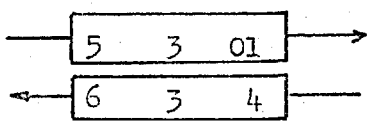
2) The information of this call when stored in the memory looks as follows:

| I | FILE | CGS | EXNU | DD |
|---|---|---|---|---|
| 9 | 3 | 5421 | 07 | 3311 |

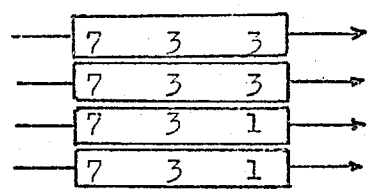
| I | FN | IFN |
|---|---|---|
| 3 | 9 | 1 |




3) A message is sent




5) Under the control of the TRUNKCD-block four messages are sent with the dialled digits




4) The message is stored in CP 07 as follows:

| I | FILE | CGS | EXNU | DD | |
|---|---|---|---|---|---|
| 15 | 3 | | 01 | 3311 | under the control of the INCOM-block |

Then, a message is sent.

6) The INCOMDD-block stores the dialled digits, then assigns the Tag number 27 to the CDS and instruct to connect the ringing current.

A message is sent to the CDS

| 9 | 3311 | 27 |
|---|---|---|

A message is received from the CDS informing that it is busy

| 2 | 27 | 8 |
|---|---|---|

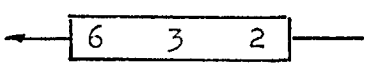
7) A message is sent with the information that the CDS is busy.

A message is sent to disconnect the Tag 27

| 8 | 27 | 5 |
|---|---|---|

The call is deleted.

8) Under control of the TRUNKCD-block messages are sent to the CGS to inform the status of the CDS and to disconnect the DDR 25




| 8 | 25 | 3 |
|---|---|---|
| 2 | 25 | 5 |
| 8 | 25 | 5 |
| 4 | 5421 | - |

dialled six digits corresponding to the subscriber 3311 in the zone 07.

2- Under the control of the DD block a store space with address 9 has been allocated to this call. It has also allocated the file number 3 by making use of the subroutine FILENU, which operates in the same way as the subroutine PENU.

3- A message | 11 | 3 | 01 | is sent to the central processor CP 07 requesting store space for this call.

4- In the CP 07 under the control of the INCOM block, a space is allocated in the main memory for the call and a message is sent to the CP 01 with the reply that a space has been granted | 6 | 3 | 4 | .

5- In the CP 01 after receiving the last message, four consecutive messages are sent to the CP 07 with the CDS number.

6- In the CP 07 the INCOMDD block handles the messages of the trunk dialled digits, find out whether the CDS number is normal, assigns a tag number to it and instructs to connect ringing current to the CDS. If the connection is unsuccesful as in the case of this example, the CP 07 receives a message indicating that the CDS number is busy.

7- The CP 07 sends a message to the CP 01 informing of the status of the CDS.

8- The CP 01 under control of the TRUNKCD block

proceeds to inform the CGS about the status of the CDS and to clear the call.

The On-Hook Block (ONH)

The on-hook message indicates the termination of a call and this may happen at any moment after a call has been initiated by an off-hook condition. When an on-hook message arrives at a central processor, the ONH-block takes care of the message. After it has been found whether the subscriber's number that caused the on-hook was a valid number, the arrays keeping the CGS' numbers are scanned to find the location where the call is stored. Once the location is found the call is deleted and the space made available to another call.

# CHAPTER IV

# COMPUTER SIGNALLING SIMULATION

## IV.1 SIGNALLING SYSTEM

The large number of combinations of messages flowing through the network and their complexity make it convenient to have a tool to study the interplay of signals between the central processor and any of the remote exchanges, or other central processors. For the present work the tool has been a computing machine programmed to simulate the signalling system.

This way of analyzing the signalling system allows one to obtain, if desired, values for the traffic handled and the grade of service, as a function of different parameters such as: number of DD receivers, size of the memory to store the call in progress, delay of the different parts of the system, etc. Another use of the simulation system is for the elaboration and test of the controlling program. That is, given the network as a parameter, with the number of subscribers, size and capacity of the processing equipment and numbering scheme, one can write a program to control the network, modify and test it with the help of the simulation system.

This type of simulation can have extensive use considering the advent of the more complex electronic

exchanges and computer controlled networks and traffic. It is commonly believed that the future exchanges will become more communication exchanges than just telephone exchanges. This is due to the increasing amount of data being transmitted today through the telephone wires, i.e., video-phone, facsimile, computer data, and so on.[W4]

Thus, the study of message handling between senders, receivers and control centres is relevant not only for the purpose of developing the system considered in this work.

## IV.2 A LANGUAGE FOR SIGNALLING SYSTEMS SIMULATION (SSS)

In order to test the controlling program of the exchanges and their handling of messages, the dynamic behaviour of the system was simulated with a computing machine.

In the course of the present study, no simulation language has been found, in the available program libraries, suitable to simulate the type of system being considered here. So, a language was developed to simulate the signalling traffic of the telephone network interplaying with the controlling program. The basic mechanism of the S.S.S. is shown in Fig.15 on the next page.

For every message coming into the processing centre a three stage cycle takes place. The first stage

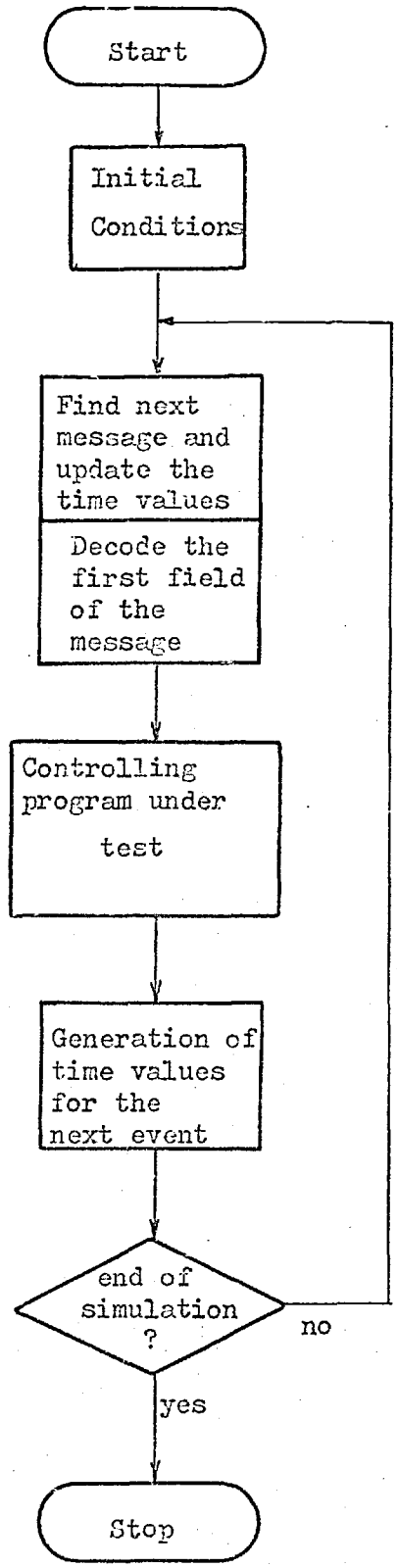

Start
Initial Conditions
Find next message and update the time values
Decode the first field of the message
Controlling program under test
Generation of time values for the next event
end of simulation ?
no
yes
Stop

Fig. 15.

is the simulation of the arrival of the message to the processing centre. A table, containing the next event of all calls being processed, is searched in order to find which one will occur first. This table is updated every cycle at this stage.

The second stage is the processing of the message just arrived. This stage is carried out under the control of the program being tested.

The third stage consists in the generation of the arrival of the next message concerning the present call, and of a subsequent message if necessary. That is, if the present message is the first dialled digit sent by a calling subscriber, the next message, concerning the present event, will be either a second dialled digit or an on-hook condition. After the arrival of each off-hook message, a time value is generated for the next off-hook. After completion of the three stages, the system is recycled and the program returns to the first stage.

The Subroutines

The S.S.S. consists of a main program and five subroutines. It is set by defining several arrays or tables and by the initialization of the variables. The arrays and most of the variables are common to all the subroutines in the system. The five subroutines of the S.S.S. are called:

XCHANGE

FIRST

SCAN

ERROR

CHART

To start the S.S.S. the following subroutines are used sequentially:

1- XCHANGE: The system is initialized by using this subroutine. All arrays are set to zero except the array called LINK, which is set to LINK(J) = J+1 .

Next table shows all the arrays including the ones described in Section III.4. The values of the first two columns are part of an example and are used to explain the mechanism of the S.S.S.

TABLE 7

| S | NEXT | I | LINK | DDR | TAG | FILE | CGS | EXNU | DD | ND | BUSY | RING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 15 | 1 | 1 | 25 | 00 | 00 | 5421 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 6 | 2 | 3 | 0 | 00 | 00 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 0 | $10^5$ | 3 | 4 | 0 | 00 | 00 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 0 | $10^5$ | 4 | 5 | 0 | 00 | 00 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| . | . | . | . | . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . | . | . | . | . |

2- FIRST: This subroutine initiates the S.S.S. cycle by setting up in the table an arbitrary complete first cycle. A calling subscriber (5421) has produced an OFH condition and an OFH message was sent to the processing centre. The DD receiver (DDR(1) = 25) has been assigned to it. Thus, RN(25) = 1 and IRN(25) = 1 .

The automatic assignment of a DD receiver ( if available) to an off-hooked subscriber has been described in Section III.4 and is part of the operation of the controlling program (Step 2).

To complete the cycle the next event of the present message is determined by setting S(1) = 2 , and its arrival time is obtained from a random number generator (Step 3). This value of the S-code means that the next event is a reply indicating whether the DD receiver has been connected and that the CGS has dial tone. Simultaneously, the times in which this and the next OFH will occur are also generated.

The following list shows the meaning of the code numbers S.

| S | Meaning- the next event of the present call will be: |
|---|---|
| 1 | an off-hook message; |
| 2 | a message informing whether a DD receiver has been connected and the CGS has dial tone; |
| 3 | a message with a dialled digit; |
| 4 | a message informing that a CDS is either ringing or busy; |

| S | Meaning |
|---|---|
| 5 | a message informing that a CDS is OFH; |
| 6 | a message informing that a long distance CDS is either ringing or busy; |
| 7 | a message from a long distance processing centre informing whether a space has been allocated for a trunk call; |
| 8 | a message informing that a CGS has ringing tone; |
| 9 | a message informing that a CGS has busy tone; |
| 10 | a message informing that a CGS has unobtainable tone; |
| 11 | an incoming trunk call message requesting a space for a call; |
| 12 | a message with an incoming trunk call dialled digit; |
| 13 | an on-hook message. |

3- SCAN: This subroutine is operated in the first stage of every cycle and its objective is to scan the NEXT-array and update it. The NEXT-array contains the time-value assigned to the next event of every call, that is, to the next message associated with each call being processed or to be processed. Thus, with the SCAN subroutine, the NEXT-array is scanned and the message with the smallest time-value is selected for the present cycle. Once the next message has been found, the NEXT-array is updated, that is, the time-values of all the next events are reduced by the amount of the time-value just selected during the scanning.

4- CHART: This subroutine, when used, produces a print-out of all the arrays in the main storage of the program under test and those of the simulation system. Thus, at any given instant a state of all calls being processed can be obtained.

The generation of random arrival time

The arrival time of the next event of each call being processed is kept in the array called NEXT. This time is generated in the third stage of the S.S.S. cycle, in some cases with a random number generator, in others, by assigning a fixed time value.

Under certain assumptions,* the initiation of calls arrive to an exchange in accordance with a Poisson probability law at a rate of $\lambda$ initiated calls per unit time. Thus, the waiting time for the $k^{th.}$ call, after the last call has been initiated, obeys a gamma probability law with parameters k and $\lambda$ . Thus, the probability density function is:[P2,F3]

$$p(t) = \frac{\lambda}{(k-1)!} (\lambda t)^{k-1} e^{-\lambda t} \quad \text{for } t \geq 0$$

$$= 0 \quad \text{for } t < 0$$

---

* The assumptions are that the number of subscribers is infinite, and that there is independence between the calls. The first assumption is valid in the sense that the number of subscribers, and thus the potential number of calls at any given instance, is many times larger than the traffic, that is, the number of calls in progress. The second assumption is valid in any particular instance of time, but not when a period of several minutes is considered, since the occurence of a second call may be affected by a first one.

Here, the interarrival time of such events is of particular interest, that is, the case when $k = 1$ . Then,

$$p(t) = \lambda e^{-\lambda t} \qquad \text{for } t \geq 0$$
$$= 0 \qquad \text{for } t < 0$$

which is usually called the negative exponential density function.

Most computing machine libraries (subroutine libraries) contain random number generators with uniform distribution. To obtain a sequence of random digits with a negative exponential distribution one can use random digits $\left(0 \text{ and } 1\right)$, with a uniform distribution and a transformation function.

Let the transformation function be:

$$t = g(T) = -\frac{1}{\lambda} \log_e (1-T)$$

where T is a random variable with a uniform density function and values between 0 and 1 . That is:

$$p(T) = 1 \qquad \text{for } 0 \geq T \geq 1$$

Then, the probability density function of t is:[Pl]

$$p(t) = \frac{p(T)}{\left| g'(T) \right|} = \frac{1}{\frac{1}{\lambda(1-T)}}$$

$$p(t) = \lambda (1-T) = \lambda e^{-\lambda t}$$

Thus, using the sequence of random numbers T and the transformation function $t = g(T)$ , one can obtain a stream of random numbers with negative exponential

distribution.

The Time Structure in S.S.S.

When the simulation is started the array NEXT contains a stream of random bumbers with negative exponential distribution and $\lambda = 1$ , that is, at the average arrival rate of one initiated call per second. In the simulation program one second is represented by 100 units of time, that is, every unit of simulated time is 10 milliseconds. At the same time, the value of S is made equal to one in all locations. This means that a stock of 100 potential calls is kept in the storage and will be handled by the program under test in accordance with the time value stored in NEXT.

Every time a call is terminated and a space is vacant in the main storage a new OFH or incoming trunk call is generated. In this way a constant stream of calls is maintained and the stored-program can be tested.

# CHAPTER V

# OPTIMUM INVESTMENT STRATEGY

## V.1 PLAN OF INVESTMENT

A telephone network, as most infrastructural elements, can be planned and designed in a relatively short period of time, but its implementation will undoubtedly demand several years. Therefore, decisions have to be made as to which parts of the network will be built (i.e., invested in) in the first, second and subsequent stages of construction. Such a set of decisions is called a plan of investment.

A given plan of investment that produces a maximization of profits, as regards those factors concerned, will be regarded here as an optimum plan of investment. However, this does not necessarily imply that economic profits have to be made. Particularly during the periods of building up the infrastructure, the telephone network may have only economic losses. In times of no economic profit, similarly, the criteria will be that of minimizing losses.

Needless to say, in assigning priorities to the construction of the different parts of the network, the number of alternatives will be very large and the selection of them can be affected by many factors of social,

political, military or economic character. In general, the main interest of the administrative body is to maximize the rate of return on invested capital.

Nevertheless, the administration may include in the optimization procedure the non-economic factors, (that is, the social profit) since the ultimate goal of the network is a socio-economic one.

When evaluating the optimum program, a certain profit or pay-off is assigned to each investment. In this way social, military, political or other types of profit can be conveniently scaled in monetary units and added as if they were part of the economic profit of that particular investment. Or, alternatively, a weight can be assigned to the projects if they are ranked in order of importance.

## V.2 STAGES OF OPTIMIZATION

The problem of optimizing investments in telephone networks can be approached in two different ways. The first one is the determination of the optimum investment plan for a given telephone network, when the final form of the network is known and fixed, that is, when the final system structure is a parameter. The second one is the determination of the optimum investment plan to be followed when, due to unexpected demand or to unexpected economic or social developments, the network structure has to be changed in one or more of the periods of investment.

For obvious reasons the second approach will seem to be more appropriate for a developing society than the first one. This will require that in every investment period a revision of the state of the network be made considering the new circumstances.

It may seem inconsistent to design, let us say, a ten-year plan, and then, after one or two years, to modify the plan. Nevertheless, this is precisely what is needed when no accurate forecast can be made of the socio-economic behaviour of the country or society in consideration.[11] Therefore, a flexible way of optimization is more adequate than a rigid one, and that policy is the one followed in the present study.

Moreover, optimization can be carried out either in one single stage, in which the whole system is considered in detail, or in several stages. In deciding the stages to be adopted a value judgment is introduced which may alter the results, and this value judgment will affect the results more critically, according to the complexity of the problem. On this subject Duncan Luce's comments are particularly relevant:[13]

> To deal with complicated decisions it is usual to subdivide them into two or more stages: the alternatives are grossly categorized in some fashion and a first decision is made among these categories; the one chosen is further categorized and a second division is made, etc. It is commonly accepted, and is probably true, that when such a multistage process is needed the overall result depends significantly on which partitionings are employed. One senses, however, that if the decision situation is quite simple-- so that a two stage process is not really needed-- then the intermediate categorization, if used, will not matter.

In this thesis, both ways, the stagewise optimization and a single stage optimization are shown. For the first case the solution will be subdivided into two stages. In the first stage, (Section V.6), which can be called the macro-program, the best set of projects for the coming period of investment, will be selected. This first stage takes into consideration all environmental constraints, among them the final structure of the network, but does not consider the budget as a constraint.[L1] In the second stage, or subprogram, (Section V.8), the selected projects for one period will be optimized, since the budget alloted for investment might be less than that required to implement the selected set of projects. In other words, with the subprogram one can find how much of each of the selected projects can be built in the period being considered.

The two stages of optimization are carried out by using in each case a different algorithm. This general approach has also been followed by M. Soewarso[S1] in a case study of Indonesia. The main problem in that study lies in the fact that the algorithm used in the first stage cannot guarantee an optimum and the procedure followed in the second stage may lead to a very complicated situation where it may become necessary to approximate the results.

In Sections V.6 and V.8 this type of optimization is used but important modifications are introduced in each stage. The first stage is carried out with a

modified version of Reiter's algorithm[R3] for the selection of independent projects where, even though the absolute optimum cannot be guaranteed, the probability of reaching it is considerably increased. The second stage is carried out by considering the problem as a knapsack problem and is solved with a dynamic program, thus avoiding the problem encountered in the work of Soewarso.

In Section V.9, instead of considering the problem in two stages, the problem has been solved in a single stage. The algorithm used in this case guarantees the optimum solution, reduces the over-all computing time and allows the observation of a wide range of solutions simultaneously for different budgets.

During the present work this approach has been found more satisfactory in many ways than the multi-stage approach. But the latter should nevertheless be considered as a useful method, for even if an absolute maximum cannot be guaranteed, the partition of the solution in several stages allows the problem to be solved with a small size computer, whereas the one-stage optimization requires in some cases a considerably larger memory.

At the end of Section V.9 the results obtained with both methods are discussed.

## V.3 THE INVESTMENT MODEL

Following the techniques used in operations research, the solution to the problem of optimizing an

investment plan requires a model. This model is a mathematical description of the investments, incomes and running costs of the different parts of the network under study. The model and the algorithm used here are very general in structure and could be used for other types of network, although it was specially intended for telephone communication networks. To solve a particular problem using this model, the number and value of the different parameters, as well as the number of variables have to be fitted to the particular case.

Planning time

The total cost of equipment and installation of the telephone network will amount to C monetary units. But funds for only an amount of $f_t$ units are available from the budget in a given period. This amount could be the same or different from period to period. If the total number of periods is T, the total cost of the network is:

$$C = \sum_{t=1}^{T} f_t$$

Then, assuming that in every period the same amount F is available, the construction time will be extended over at least:

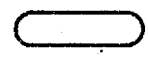
$$T \geq \frac{C}{F} \quad \text{periods.}$$

Here, since government budgets are usually done on a yearly basis, and also for the sake of simplicity, the period of planning is assumed to be of one year. But it could, of course, be of any appropriate length of time.

The number of periods T is not only limited in the lower bound by the availability of funds to pay for equipment and installation, but also by the capacity of the administration to install the equipment and train the service personnel, and also by the capacity of the supplier to deliver the equipment. Consequently, the administration should not try to invest, within a given period, in more equipment than it can install.

## V.4 SUBDIVISION OF THE NETWORK

For the purpose of this model the network structure is subdivided as follows:

| | Graphic Symbol | Size given by: |
|---|---|---|
| 1- processor system |  | amount of traffic to be handled |
| 2- satellite exchanges (local wiring, switching equipment, interfacing) |  | number of (100) lines |
| 3- trunk circuits |  | number of channels |

The following diagram shows an example of the type of network considered here.

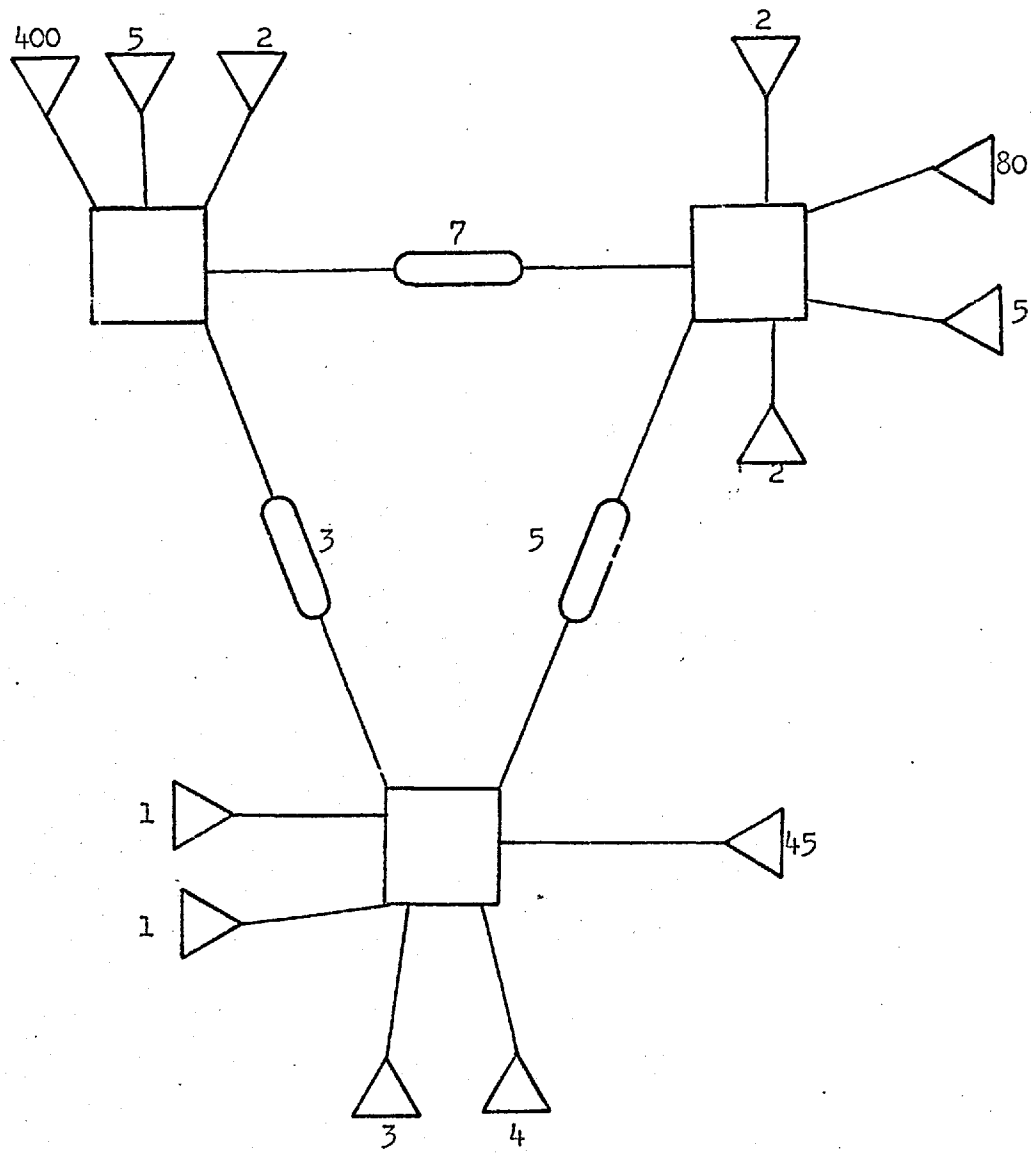



Fig. 16.

The whole network is subdivided in zones, each one consisting of a main processing centre and a number of satellite exchanges attached to it. The long distance connections are symbolized by the links (trunk circuits) between these zones, each containing a certain number of channels.

This representation of the network serves only the purpose of the present study. Physically the different parts of the network may not be grouped as they are shown in the diagram. For instance, the processing centres are physically located near of or within the same building of the largest switching centre (main exchange) of the zone. But from the point of view of the processing centre, the main exchange is attached to it as any other satellite exchange, since all switches are remotely controlled by the central processor. In the above diagram the main exchanges can be distinguished from the other satellites merely by the number of lines.

## V.5 VARIABLES OF THE MODEL

This section describes most of the variables used in the model for the economic planning of the network, and, when necessary, an explanation of their use is also given. These variables and parameters will be used in Sections V.6, V.8 and V.9 for the optimization procedures.

Commissioning costs

Processing Centres.- The cost of the processing centres is subdivided in the cost of the processing machines, building sites, etc., and that of the additional facilities necessary to enlarge the centre.

The cost of the processing machines and their software's I/O devices is not dependent, from period to period, on the number of lines in operation, because the processors' capacity is intended for the handling of future demands. However, the cost of the interface equipments and the additional core memories are proportional to the number of lines controlled by the processing centres. Therefore, additional cost per line will not be considered as part of the processing centres' costs but as part of those of the satellite exchanges.

Then, we define: $q_{io}$ = fixed cost of processing centre i (ƒ)*

Satellite Exchanges.- The pace of wiring the local network is in general different from that of the construction or expansion of the satellite exchanges' equipment. The cost of the wiring of the local network is minimized if it is done all at once in each local area. (See Appendix IV.) On the other hand, the exchange equipment can be expanded many times without incurring extra heavy expenses. Thus, the cost of the wiring of the local network will be considered as a fixed cost and the expansion costs of the

---

* The symbol ƒ is used to indicate an arbitrary monetary unit.

satellite equipment will be considered as proportional to the number of (100) lines. The cost, per 100 lines, of the satellite equipment includes the cost of the necessary interface in the processing centre and the additional core memory facility (if necessary).

The following variables express the costs to commission the satellite exchanges:

$q_{io}$ = fixed cost of the wiring of the local network i (\$) ,

$q_i$ = cost of exchange equipment per 100 lines for the satellite i (\$/100 lines) .

Trunk Circuits.- For the trunk circuits a difference has to be made between the cost of the station equipment (i.e., multiplex equipment, etc.) and the transmission equipment.[F2] In general it can be stated that the cost of transmission equipment is more or less proportional to the length of the link and independent of the number of channels, since the installation is made for a future demand. However, the cost of the station equipment is in a way proportional to the number of channels. The following variables define the commissioning costs of the trunk circuits.

$q_{io}$ = cost of transmission equipment for the trunk i (\$) ,

$q_i$ = cost of station equipment for the trunk i, per channel (\$/channel).

Thus, the cost variables for any project i can be

generalized as follows:

$q_{io}$ = fixed cost of project i ($)

$q_i$ = cost of project i per item ($/item) ,

where the items are either 100 lines in the case of the satellite exchanges, or channels in the case of trunk circuits.

Running costs

The definitions of the following variables are made under the assumption that the running costs of the different parts of the network start immediately after their installation.

Processing Centres.- The running costs of the processing centres are relatively independent of the number of satellites connected to them or to the traffic handled, so that a fixed running cost for each processing centre per year is assumed.

$P_{io}$ = running cost of processing centre i ($/year) .

Satellite Exchanges.- The satellite exchanges are subject to a fixed annual running cost for the main equipment, building site, link with the processing centre, interface equipment, etc., and a running cost that depends on the size of the exchange, which is mainly for the maintenance of the local network and subscriber's equipments. Thus, the running costs for the satellites are:

$P_i$ = running cost of the local network of the satellite i, per 100 lines, per year ($/100 lines . year) ,

$P_{io}$ = fixed running cost of satellite exchange i, per year ($/year) .

Trunk Circuits.- In the case of the trunk circuits, the fixed running costs are much more important than those given by the number of channels. Therefore only a fixed yearly running cost per trunk circuit will be assumed.

$P_{io}$ = fixed annual running cost of the trunk circuit i ($/year) .

The running cost variables for any project i can be generalized as follows:

$P_{io}$ = fixed running cost of project i ($/year) ,

$P_i$ = running cost of project i per item ($/item . year) .

Incomes

The following variables define the income obtained from operating the telephone network.

Satellite Exchanges.- The yearly income derived from the local calls and from the monthly basic rate paid by the subscriber is:

$I_i$ = income from satellite i, per 100 lines, per year ($/100 lines . year) .

Trunk Circuits.- The yearly income obtained from

operating the long distance network can either be given per channel, that is, between two adjacent zones, or for each route, that is, between any two satellite exchanges. Both are possible, especially during the time of commissioning the network, since the number of channels in operation will barely cover the traffic demand.

In the case of dealing with single trunk circuits the yearly income is:

$I_i$ = income from the trunk circuit i, per channel, per year (Ø/channel . year) .

In the case of dealing with the traffic from routes between any two satellite exchanges, the incomes are given in the $|A|$-matrix. The element $A_{ij}$ is the income obtained from the traffic between the satellites i and j. These incomes can only be considered when a route between the satellites is completed. To satisfy this constraint the $|\lambda|$-matrix is used.

Route Income Matrix:

$$|A| = \begin{array}{c|cccccc} & 1 & 2 & \cdot & \cdot & j & \cdot & \mu \\ 1 & & & & & \cdot & & \\ 2 & & & & & \cdot & & \\ \cdot & & & & & \cdot & & \\ i & \cdot & \cdot & \cdot & \cdot & A_{ij} & & \\ \cdot & & & 0 & & & & \\ \cdot & & & & & & & \\ \mu & & & & & & & \end{array} .$$

$A_{ij}$ is the income per year obtained from the traffic of route i-j.

Route Constriction Matrix:

$$|\lambda| = \begin{array}{c} \\ 1 \\ 2 \\ \cdot \\ i \\ \cdot \\ \cdot \\ \mu \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & \cdot & \cdot & j & \cdot & \mu \end{array} \\ \begin{vmatrix} \ddots & & & & \vdots & & \\ & \ddots & & & \vdots & & \\ & & \ddots & & \vdots & & \\ \cdots & \cdots & 0 & \ddots & \lambda_{ij} & & \\ & & & & & \ddots & \end{vmatrix} \end{array}$$

where: $\lambda_{ij} = \begin{cases} 1 & \text{if a route between i and j is completed,} \\ o & \text{otherwise.} \end{cases}$

The entries of this matrix insure that no income $A_{ij}$ enters the calculation of the economic profit, if the route is not completed.

Constriction on the Number of Items.- Each project consists of a number of items, i.e., the satellite exchange i planned for a total of 700 lines, will consist of 7 items. The number of items of project i at the completion of the network is $b_i$ . The number of items to be commissioned in a given period is $b_i^*$ .

Thus, in a given period, the cost of commissioning the project i is:

$$Q_i = q_{io} + b_i^* \cdot q_i \quad ,$$

the running cost of project i is:

$$P_i = P_{io} + b_i^* \cdot P_i \quad ,$$

and the net income from project i is (excluding that from long distance calls) :

$$N_i = b_i^* \cdot I_i - P_i \quad .$$

## V.6 MACRO-PROGRAM: AN ALGORITHM FOR THE SELECTION OF THE BEST SET OF PROJECTS

This section deals with the first of the two stages of optimization of an investment plan for a telephone network, as indicated in Section V.2. The idea behind the macro-program is to select those parts of the network which are of more profit or importance for the whole area.

To optimize, in the sense of searching for the best investment, one has to fragment the network in smaller parts. In this case, the different pieces of hardware for the processing centres and satellite exchanges as well as the installation of the local networks and the trunk circuits are broken into a number of discrete parts which cannot be arbitrarily small. For example, a central processor or the building site of a satellite exchange are discrete elements which must be considered as indivisible parts. Thus, the network and its installation is subdivided into many projects, each of which can be independently built or installed.

The projects are independent of each other in the

sense that any one can be selected together with any other one, and the profits from two or more projects taken together may be less, equal or larger than any of the two profits taken separately or the sum of the individual profits. The projects are numbered 1, 2, .., $\mu$ and for each project i a profit can be calculated.

A selection of projects can then be represented by a vector

$$\{ V_k(v_1, v_2, \ldots, v_\mu) \mid H_1 \}$$

where the elements $v_i$ form an ordered set with the following values:

$$v_i = \begin{cases} 1 & \text{if the project i is selected} \\ 0 & \text{if the project i is not selected} \end{cases}$$

$$\text{for } i = 1, \ldots, \mu$$

and $H_1$ is the set of constraints imposed to $V_k$ . These constraints determine when a project is 'acceptable' and can be commissioned, since it can be put in operation only if some other projects are, or have been, already commissioned.

Every vector $\{V_k(v_i) \mid H_1\}$ will have a ranking $R(V_k \mid H_1)$ given partially by the individual profits of the projects in the vector and partially by other factors. Now the optimization consists in finding the vector $\{V_{ko}(V_k \mid H_1)\}$ that has the maximum ranking $R(V_{ko})$ . In order to do this several algorithms have been studied and compared, and these results are shown in Appendix I. Algorithm C has been worked out to improve Reiter's algorithms (A and B in Appendix I) in increasing the proba-

bility of reaching the absolute maximum.

It is perhaps important to remember that the constraints $H_1$ do not include the budget constraint, since the first stage of the optimization consists in finding the best selection of projects without regard to whether they can be financed. In many cases the best selection of projects may involve an investment larger than $f_t$, the available share of the budget for that period. It will then be necessary to find how many of the projects can be commissioned. This is done with the subprogram in Section V.8.

The Algorithm C

A thorough discussion of this algorithm can be found in Appendix I. The following steps describe the algorithm.

I- A random vector $\{V_k | H_1\}$ is selected and its ranking $R(V_k | H_1)$ is calculated.

II- All possible vectors $\{V_k | H_1\}$, resulting from deleting or adding one project at a time are searched and their respective rankings $R(V_k | H_1)$ are calculated. The search stoped when the first positive gradient is found by:

$$G(k, x+) = R(V_k | H_1) - R(V_{x+} | H_1) > 0$$

III- The vector $V_{x+}$ thus obtained is taken as the new starting combination and the procedure is repeated again as from (I), until a vector $V_{ko}$ is found, such that:

$$G(ko, x) \leqq 0 \quad , \text{ for all } x$$

IV- Next, the algorithm is reinitiated, from (I), but instead of a random vector, the vector $\{V_{ks}|H_1\}$ is taken as the starting vector. Here $V_{ks}$ is as $V_{ko}$ but with two projects changed (deleted or added).

V- This procedure is continued several times, finding each time a vector $V_{ko}$ that satisfies the inequality of Step III. The vectors $V_{ko}$ may all be different, representing each of them a local maximum. The probability of reaching the absolute maximum, in a given number of iterations, is proportional to the number of local maxima and to the distribution of the starting points with respect to the local maxima.

VI- Finally, the ordered set $v_i$ , corresponding to the vector $\{V_{ko}|H_1\}$ with the highest rank, is taken as the best selection of projects.

The Ranking of $V_k$:

In what follows, and for the sake of simplicity, a vector $\{V_k|H_1\}$ will simply be referred as $V_k$ . The overall ranking of a vector $V_k$ is:

$$R(V_k) = R_e(V_k) \cdot R_f(V_k)$$

where: $R_e(V_k)$ is the economic ranking of $V_k$ ,

and : $R_f(V_k)$ is the 'functional' ranking assigned to $V_k$. By functional ranking we understand a weight-value which is a function of the criterion not included in the economic ranking. This functional ranking, as will be seen in Chapter VII, follows from the

social welfare function.

In every period the ranking of the projects has to be done again. This is so because alterations and unforeseen developments may affect the economic profit of the projects as well as the social profits, as was previously indicated in Section V.2.

## V.7 ECONOMIC PROFIT

In national planning, complications arise when, in the optimization process, multiple objectives are present and some of them are conflicting, that is, when the attainment of one objective precludes or impairs that of the others. To this situation two approaches have been taken by operation researchers. The first is the method of pre-assigned weights, where relative numerical weights are assigned to the projects in order to scale the various objectives. Thus, the value judgments are introduced at the beginning of the optimization. The second approach is the efficient surface method, where the value judgments are introduced at the end of the optimization process.*

F. Hanssman[H1] suggests the efficient surface method as more suitable for the national planning of the economy of underdeveloped countries, when all commodities

---

* The introduction of value judgments can be simply done, either in the form of weights or by using the technique called Goal Programming, where instead of weights, minimum requirements are imposed in the programming procedure.

and services are simultaneously taken into account, as he argues that the pre-assigned weights may loose their meaning through the elaborated processes of optimization.

Clearly in the case Hanssmann describes, there is a large number of conflicting objectives. But the problem treated here is different in that the optimization is for only one element of the infrastructure. In this case the final objective of the administration is the completion of the network. The different projects in which the network is subdivided do not have conflicting objectives in the sense that the commissioning of one projects does not preclude or impair the commissioning of another; it may only postpone it to the next period.

Here, a modified version of the method of pre-assigned weights will be applied to give to the economic ranking of each project the possibility of being weighted according to other factors, i.e., social profit.

The economic profit

In general, $R_e(V_k)$ will be defined as the relation between the contribution of $V_k$ to the desired objective and the amount of scarce resources necessary to commission $V_k$. This ratio is called the 'rank' of $V_k$. That is:

$$R_e(V_k) = \frac{\text{contribution of } V_k \text{ to objective}}{\text{scarce resources to commission } V_k}$$

In addition to the variables defined in Section V.5 the following matrix will be used to insure that the constrictions $H_1$ are met.

System Constriction Matrix (Acceptance Matrix)

This matrix contains the information required to decide if vector $V_k$ is acceptable, that is, if $V_k$ meets the constrictions $H_l$ .

$$|X| = \begin{array}{c|cccccccc} & 1 & 2 & . & . & j & . & . & \mu \\ 1 & & & & & \cdot & & & \\ 2 & & & & & \cdot & & & \\ . & & & & & \cdot & & & \\ i & . & . & . & . & x_{ij} & & & \\ . & & & & & & & & \\ \mu & & & & & & & & \end{array}$$

where:

$$x_{ij} = \begin{cases} 1 \text{ if project } j \text{ is required to commission project } i \\ 0 \text{ otherwise} \end{cases}$$

for : $i, j = 1, \ldots, \mu$

A vector $V_k$ belongs to the set of vector $\{V_k \mid H_l\}$ if and only if

$$\sum_{j=1}^{\mu} x_{ij} \cdot v_j \geq 1$$

for all i such that $v_i = 1$ , where $i = 1, \ldots, \mu$

In other words, a selected project $v_i$ is acceptable if $x_{ij} \geq v_j$ for all j, and a vector $V_k$ is acceptable if all selected projects ($v_i = 1$) are acceptable.

The meaning of the entries of this matrix is the following: if a selected project i corresponds to a satellite exchange, it is necessary to find if the corresponding processing centre j has also been selected. If a selected project i corresponds to a processing centre, it is necessary

to find if at least one of the connected satellites j has also been selected. If a selected project i corresponds to a trunk connecting the processing centres j and k, it is necessary to find if <u>both</u> processing centres j and k have been selected. In order to ensure this condition in the system constriction matrix, the entries $x_{ij}$ and $x_{ik}$ are not set equal to one, but equal to $v_k$ and $v_j$ respectively. The example in the next page will illustrate this point.

With the variables defined in Sections V.5, V.6, and in this one, it is possible to write an expression for the

$$\left.\begin{array}{l}\text{contribution of } V_k \\ \text{to the objective}\end{array}\right\} = \sum_{i=1}^{\mu} N_i \cdot v_i + \sum_{i,j=1}^{\mu} A_{ij} \cdot \lambda_{ij} \cdot v_j$$

and the

$$\left.\begin{array}{l}\text{scarce resources} \\ \text{to commission } V_k\end{array}\right\} = \sum_{i=1}^{\mu} Q_i \cdot v_i$$

Then, the economic ranking of the vector $V_k$ is:

$$R_e(V_k) = \frac{\sum_{i=1}^{\mu} N_i \cdot v_i + \sum_{i,j=1}^{\mu} A_{ij} \cdot \lambda_{ij} \cdot v_i \cdot v_j}{\sum_{i=1}^{\mu} Q_i \cdot v_i}$$

or:

$$R_e(V_k) = \frac{\sum_{i=1}^{\mu} b_i \cdot I_i - (p_{io} + b_i \cdot p_i) \cdot v_i + \sum_{i,j=1}^{\mu} A_{ij} \cdot v_i \cdot v_j \cdot \lambda_{ij}}{\sum_{i=1}^{\mu} (q_{io} + b_i \cdot q_i) \cdot v_i}$$

Example:

The next figure shows a very simple network with five satellite exchanges, three processing centres and two trunk circuits. In this over-simplified case, there are $2^{10} = 1024$ possible vectors $V_k$ , but not all of them are acceptable. The numbers inside the blocks in the figure indicate the project number and the numbers encircled indicate the number of items for each project. Thus, project 1, for instance, consists of a satellite exchange with 30 times 100 lines, that is, 3000 lines.

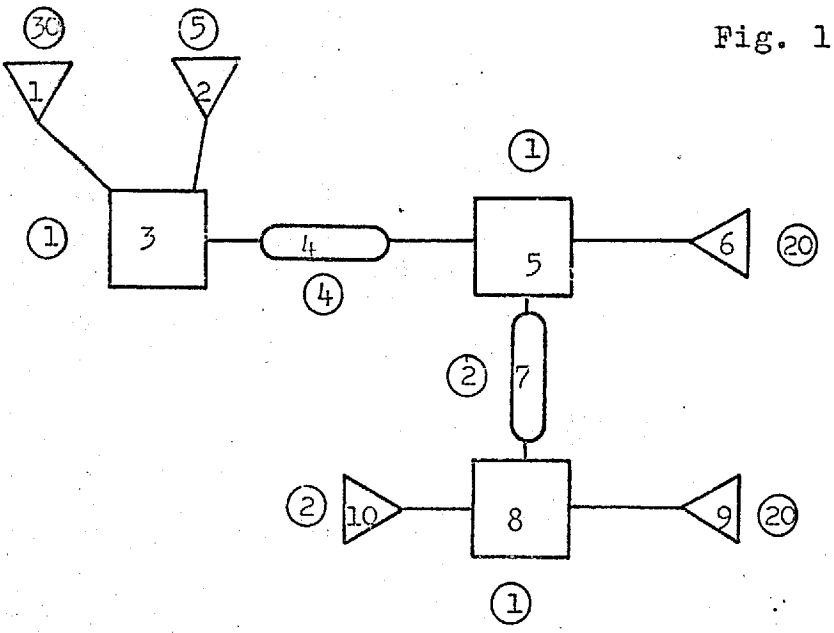



Fig. 17.

From the network structure and the previous definitions, the following three matrices can be written. All the costs and incomes in this example are completely

arbitrary and have the sole purpose of demonstrating the application of the algorithm.

The System Constriction Matrix:

$|X| =$

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | 1 | | | | | | | |
| 2 | | | 1 | | | | | | | |
| 3 | 1 | 1 | | | | | | | | |
| 4 | | | $v_5$ | | $v_3$ | | | | | |
| 5 | | | | | | | 1 | | | |
| 6 | | | | | 1 | | | | | |
| 7 | | | | | $v_8$ | | | $v_5$ | | |
| 8 | | | | | | | | | 1 | 1 |
| 9 | | | | | | | | 1 | | |
| 10 | | | | | | | | 1 | | |

The entries of the matrix, in the case of a satellite exchange or a central processor, are '1' . The meaning of this is as follows. When the algorithm has to determine whether project 3 is acceptable or not, the condition $x_{31} \cdot v_1 + x_{32} \cdot v_2 \geq 1$ has to be met. Since $x_{31} = x_{32} = 1$ then, either $v_1$ or $v_2$ has to be equal to one. In the case of a trunk circuit, (number 4, for instance), instead of 1 in $x_{43}$ and $x_{45}$ , the values of $v_5$ and $v_3$ are inserted. Thus, if in a given set of projects, projects 4 and 5 or 4 and 3 are included but not all three of them, the condition :

$x_{43} \cdot v_3 + x_{45} \cdot v_5 = v_5 \cdot v_3 + v_3 \cdot v_5 \geq 1$ will not be met.

In the matrices, for simplicity, only the values

that may be different from zero have been entered.

The Route Income Matrix:

$|A| =$

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | 3.0 | | | | 1.8 | | | 1.25 | 1.0 |
| 2 | | | | | | 1.45 | | | 1.37 | 0.3 |
| 3 | | | | | | | | | | |
| 4 | | | | | | | | | | |
| 5 | | | | | | | | | | |
| 6 | | | | | | | | | 2.5 | 0.6 |
| 7 | | | | | | | | | | |
| 8 | | | | | | | | | | |
| 9 | | | | | | | | | | 3.0 |
| 10 | | | | | | | | | | |

The Route Constriction Matrix:

$|\lambda| =$

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | $\lambda_{12}$ | | | | $\lambda_{16}$ | | | $\lambda_{19}$ | $\lambda_{1\,10}$ |
| 2 | | | | | | | | | | |
| 3 | | | | | | $\lambda_{26}$ | | | $\lambda_{29}$ | $\lambda_{2\,10}$ |
| 4 | | | | | | | | | | |
| 5 | | | | | | | | | | |
| 6 | | | | | | | | | $\lambda_{69}$ | $\lambda_{6\,10}$ |
| 7 | | | | | | | | | | |
| 8 | | | | | | | | | | |
| 9 | | | | | | | | | | $\lambda_{9\,10}$ |
| 10 | | | | | | | | | | |

where:

$\lambda_{12} = v_2$ $\quad$ $\lambda_{29} = v_3 \cdot v_4 \cdot v_5 \cdot v_7 \cdot v_8$

$\lambda_{16} = v_3 \cdot v_4 \cdot v_5$ $\quad$ $\lambda_{2\,10} = v_3 \cdot v_4 \cdot v_5 \cdot v_7 \cdot v_8$

$\lambda_{19} = v_3 \cdot v_4 \cdot v_5 \cdot v_7 \cdot v_8$ $\quad$ $\lambda_{69} = v_5 \cdot v_7 \cdot v_8$

$\lambda_{1\,10} = v_3 \cdot v_4 \cdot v_5 \cdot v_7 \cdot v_8$ $\quad$ $\lambda_{6\,10} = v_5 \cdot v_7 \cdot v_8$

$\lambda_{26} = v_3 \cdot v_4 \cdot v_5$ $\quad$ $\lambda_{9\,10} = v_8$

The following table shows the incomes, running costs and installation costs of every project. The values shown in the table are $\times 10^3$ $.

| i | $I_i$ | $p_{io}$ | $p_i$ | $q_{io}$ | $q_i$ | $b_i$ |
|---|---|---|---|---|---|---|
| 1 | 4.0 | 4.0 | 1.0 | 40.0 | 16.0 | 30 |
| 2 | 4.0 | 4.0 | 1.0 | 36.0 | 14.0 | 5 |
| 3 | 0.0 | 10.0 | 0.0 | 60.0 | 0.0 | 1 |
| 4 | 2.0 | 3.0 | 0.0 | 16.0 | 1.0 | 4 |
| 5 | 0.0 | 6.0 | 0.0 | 21.0 | 0.0 | 1 |
| 6 | 4.0 | 1.0 | 1.5 | 14.0 | 10.0 | 20 |
| 7 | 1.0 | 1.0 | 0.0 | 8.0 | 1.0 | 2 |
| 8 | 0.0 | 6.0 | 0.0 | 21.0 | 0.0 | 1 |
| 9 | 4.0 | 2.0 | 1.0 | 20.0 | 12.0 | 20 |
| 10 | 4.0 | 2.0 | 1.0 | 14.0 | 7.0 | 2 |

In the following page a list of all acceptable projects and their ranking is shown. In this example it is possible to carry out the total enumeration of the acceptable set of projects due to the limited number of

them and the structure of the network.

| Project Combination | Ranking | Total Cost |
|---|---|---|
| 13 | 0.1310 | 580 |
| 23 | 0.0060 | 166 |
| 123 | 0.1312 | 686 |
| 56 | 0.1830 | 235 |
| 1356 | 0.1460 | 815 |
| 2356 | 0.1097 | 401 |
| 12356 | 0.1444 | 921 |
| 13456 | 0.1507 | 835 |
| 23456 | 0.1198 | 421 |
| 123456 | 0.1501 | 941 |
| 89 | 0.1851 | 281 |
| 1389 | 0.1487 | 861 |
| 2389 | 0.1186 | 447 |
| 12389 | 0.1468 | 967 |
| 5689 | 0.1841 | 516 |
| 135689 | 0.1560 | 1096 |
| 235689 | 0.1408 | 682 |
| 1235689 | 0.1539 | 1202 |
| 1345689 | 0.1593 | 1116 |
| 2345689 | 0.1459 | 702 |
| 12345689 | 0.1581 | 1222 |
| 56789 | 0.1873 | 526 |
| 1356789 | 0.1578 | 1106 |
| 2356789 | 0.1438 | 692 |
| 12356789 | 0.1555 | 1212 |
| 13456789 | 0.1621 | 1126 |
| 23456789 | 0.1507 | 712 |
| 123456789 | 0.1618 | 1232 |
| 810 | -0.0408 | 49 |
| 13810 | 0.1176 | 626 |
| 23810 | -0.0047 | 215 |
| 123810 | 0.1197 | 735 |
| 56810 | 0.1444 | 284 |
| 1356810 | 0.1354 | 864 |
| 2356810 | 0.0933 | 450 |
| 12356810 | 0.1351 | 970 |
| 13456810 | 0.1400 | 884 |
| 23456810 | 0.1031 | 470 |
| 123456810 | 0.1407 | 990 |
| 567810 | 0.1449 | 294 |
| 13567810 | 0.1357 | 874 |
| 23567810 | 0.0948 | 460 |
| 123567810 | 0.1353 | 980 |
| 134567810 | 0.1414 | 894 |
| 234567810 | 0.1049 | 480 |
| 1234567810 | 0.1421 | 1000 |
| 8910 | 0.1909 | 309 |
| 138910 | 0.1519 | 889 |

| Project Combination | Ranking | Total Cost | |
|---|---|---|---|
| 258910 | 0.1263 | 475 | |
| 1238910 | 0.1497 | 995 | |
| 568910 | 0.1875 | 544 | |
| 13568910 | 0.1584 | 1124 | |
| 23568910 | 0.1451 | 710 | |
| 123568910 | 0.1561 | 1230 | |
| 13456891C | 0.1615 | 1144 | |
| 234568910 | 0.1499 | 730 | |
| 1234568910 | 0.1602 | 1250 | |
| 5678910 | 0.1915 | 554 | ** |
| 135678910 | 0.1606 | 1134 | |
| 235678910 | 0.1487 | 720 | |
| 1235678910 | 0.1581 | 1240 | |
| 1345678910 | 0.1656 | 1154 | |
| 2345678910 | 0.1557 | 740 | |
| 12345678910 | 0.1653 | 1260 | |

The maximum ranking is obtained with the vector $V_k(0000111111)$, that is, with the combination of projects 5, 6, 7, 8, 9, 10. This combination requires an investment of:

$$\sum_i (q_{io} + b_i \cdot q_i) \cdot v_i = \$\ 554{,}000.-$$

The net cost of the network (disregarding interests) is:

$$C = \sum_{i=1}^{\mu} q_{io} + b_i \cdot q_i = \$\ 1{,}260{,}000.-$$

If the network is to be completed in 4 periods, with an equal investment in each of them, then in each period there will only be $ 315,000.- available. This implies that only part of these projects can be constructed in the first period. In Section V.8 the second stage of optimization is carried out, where it will be determined how much of each project can be commissioned in the first period. That part that cannot be commissioned

in the first period plus the rest of the projects, are again subdivided in projects in the same manner as has been done with the whole network. Thus, the algorithm can be applied again to find the best set of projects for the second period, and then the third. In the fourth period no optimization is required since the yearly budget will be the cost of the completion of the network.

## V.8 PROGRAM WITHIN ONE PERIOD

Once the best selection of projects, regardless of the budget constriction, has been obtained by means of macro-optimization, it is then necessary to optimize within the selected projects the best investment considering the budget constraint. In other words, it is now necessary to find, with the available resources, how many lines or channels it is possible to commission during the first period and in which satellites and trunk circuits from the selected ones.

What follows is the description of a dynamic program to solve the type of problems which in the field of operations research is called 'the knapsack problem' or 'the cutting stock problem'.[G1]

The optimization consists in finding the maximum value of the following expression:

$$F(f_t) = \max_{n} \left\{ \sum_{i=1}^{\mu} \pi_i(b_i^*) \right\}$$

subject to:
$$\sum_{i=1} l_i(b_i^*) \leqq f_t$$

and:
$$b_i^* \leqq b_i$$

where: $b_i^*$ is the number of items that project i will include to achieve optimum;

$b_i$ is the maximum number of items that project i can include;

$\pi_i(n) = \left[ n.I_i - p_{io} + n.p_i \right] \cdot R_i$ , is the contribution of the project i, with n items, to the intended objective;

$l_i(n)$ is the cost of project i with n items;

$f_t$ is the budget constriction or the amount of funds available in period t;

and: $R_i$ is the normalized value of social ordering (see Section VII.2).

The solution to this problem is sought by an algorithm based on the following function (see Appendix II):

$$G_{i+1}(x) = \max_{n} \left\{ G_i \left[ x - l_{i+1}(n) \right] + \pi_{i+1}(n) \right\}$$

where: $G_i(x)$ is the optimum value obtained when including projects 1, 2, ..., i with funds available up to the amount of x, and subject to:

$$x \geqq l_{i+1}(n)$$

and:

$$b_{i+1} \geqq n \geqq 0 \quad .$$

This function is modified to suit the mechanism of the algorithm by changing the variable $x$ to $x+l_{i+1}(n)$ , resulting in:

$$G_{i+1}(x+l_{i+1}(n)) = \max_{n} \left\{ G_i(x) + \pi_{i+1}(n) \right\} \quad .$$

A detailed explanation of the algorithm in its general form can be found in Appendix II. Here the application of the algorithm to solve the second stage of optimization is shown. A general outline of the features of the algorithm is first given, and then it is followed by its formal description.

1- The projects are numbered, their profits and costs tabulated as in the example in Section V.7, assigning project zero to an imaginary project with zero profit and unity installation cost. After each step of optimization some funds may remain unused and should be 'invested' in this project.

2- Then, one project after the other is allocated without exceeding the available budget x. But first it is necessary to find whether the project under consideration is acceptable for investment at this stage. That is, for instance, if the project is a trunk circuit one, it must be ascertained that investments in the corresponding terminating exchanges have already been made.

Otherwise the project is not acceptable and its consideration has to be postponed until later on. For this purpose the X-matrix (see Section V.7) and a table described below (E-Table) furnish the relevant information.

3- Thus, starting with project zero (i=0), x=0 and n=1 , the values of $G_{i+1}(x+1_{i+1}(n))$ are compared with $G_i(x) + \pi_{i+1}(n)$ and replaced by it if the latter is larger than or equal to the former. Moreover, each time a project i is added to the process of optimization, a set of values: $b_0^*(i,x)$, $b_1^*(i,x)$, ..., $b_i^*(i,x)$ is obtained.

4- The final result is a table of the values of $G_\mu(x)$, that is, of the optimum values that can be obtained with projects 1, 2, ...,$\mu$ and an amount of x monetary units, and a table of the number of items in each case: $b_0^*(\mu,x)$, $b_1^*(\mu,x)$, ..., $b_\mu(\mu,x)$ .

5- In order to reduce the size of the memory needed when a computing machine is programmed with this algorithm, the values of $G_j(x)$ and those of $b_i^*(j,x)$ are kept in two tables, one called the 'zero-table', containing the values of: $G_0(x)$, $b_0^*(0,x)$, $b_1^*(0,x)$, ..., $b_i^*(0,x)$ and the other called the 'one-table', containing the values of: $G_1(x)$, $b_0^*(1,x)$, $b_1^*(1,x)$, ..., $b_i^*(1,x)$. The 'zero-table' represents, at any given instant of the optimization, the table for the i-th. project; and the 'one-table', the table for the i+1-th. project.

The Algorithm

I.1 Initialize:

$$\left.\begin{aligned} G_i(x) &= 0 \\ b_j^*(i,x) &= 0 \\ b_o^*(0,x) &= x \\ b_o^*(1,x) &= x \end{aligned}\right\} \text{for: } \begin{cases} x = 0, 1, \ldots, L \\ i = 0, 1 \\ j = 0, 1, \ldots, \mu \end{cases}$$

let: $i = 0$, $k = 0$, and go to II.1 .

II.1 Let $x = 0$ and go to II.2 .

II.2 If $b_{i+1} \geqq 1$ go to II.3. Otherwise, go to IV.1.

II.3 Let $n = 1$ and go to II.4.

II.4 If $x+l_{i+1}(n) \leqq L$ , then let $V = G_o(x) + \pi_{i+1}(n)$ and go to II.5. Otherwise, go to III.2.

II.5 If $V \geqq G_1(x+l_{i+1}(n))$ , let $G_1(x+l_{i+1}(n)) = V$ , $b_{i+1}^*(1,x+l_{i+1}(n)) = n$ , and:
$b_j^*(1,x+l_{i+1}(n)) = b_j^*(0,x)$ , for $j = 0, 1, \ldots, i$ .
Let $k = 1$ and go to III.1. Otherwise go to III.1 .

III.1 If $n \geqq b_{i+1}$ go to III.2. Otherwise let $n = n+1$ and go to II.4.

III.2 If $k = 1$ go to III.3. Otherwise go to IV.1.

III.3 (Do III.3 for all t from 1 to L)
If $G_1(t) < G_1(t-1)$ let: $G_1(t) = G_1(t-1)$ ;
$b_j^*(1,t) = b_j^*(1,t-1)$ for $j = 1, 2, \ldots, i-1$;
$b_o^*(1,t) = b_o^*(1,t-1)+1$ and go to III.4. Otherwise, go to III.4.

III.4 If $t \leqq L$ go to III.3. Otherwise let $k = 0$ and go to IV.1.

IV.1 Let $x = x+1$ and go to IV.2.

IV.2 If $x+l_{i+1}(1) > L$ go to IV.3. Otherwise go to II.2.

IV.3 If $b_{i+1} \geqq 1$ go to V.1. Otherwise go to IV.4

IV.4 Reorder projects: advance the projects by one in the list of projects, from $i+2$ to $\mu$. Put project $i+1$ at the end of the list and go to V.3.

V.1 Let $i = i+1$ and go to V.2.

V.2 If $i \geqq \mu$ stop. Otherwise go to V.3.

V.3 Let $s = 1$ and go to V.4.

V.4 Let $G_0(s) = G_1(s)$, $b_j^*(0,s) = b_j^*(1,s)$ for $j = 1, \ldots, i$ and go to V.5.

V.5 If $s < L$, let $s = s+1$ and go to V.4. Otherwise go to II.1.

The E-Table

The E-Table which was mentioned in the outline of the algorithm is as follows:

| | 1 2 3 . . | i | . μ |
|---|---|---|---|
| 1 | | . | |
| 2 | | . | |
| . . | | . . | |
| r | . . . . . | $e_{r,i}$ | . |
| . . ε | | | |

where:

$e_{r,i}$ is the number of channels that can be commissioned in the trunk circuit i, when r is known.

To obtain r one proceeds as follows. Let i be a trunk circuit between the centres j and k. The total number of subscribers in j and k, when the network is completed, will be $b_j$ and $b_k$ respectively, and the number of channels will be $b_i$. Therefore at any given period of construction to avoid a larger investment than necessary, one would like to impose a bound on $b_i$ which should be a function of the product $(b_j^* . b_k^*)$ ,that is, of the number of lines already commissioned or to be commissioned in the present period. Thus, for instance if $b_i = 4$ , $b_j = 2$ and $b_k = 4$ , the products r are:

| $b_j^*$ | $b_k^*$ | r | $e_{r,i}$ |
|---|---|---|---|
| 2 | 4 | 8 | 4 |
| 2 | 3 | 6 | 3 |
| 2 | 2 | 4 | 2 |
| 1 | 4 | 4 | 2 |
| 1 | 3 | 3 | 2 |
| 1 | 2 | 2 | 1 |
| 2 | 1 | 2 | 1 |
| 1 | 1 | 1 | 1 |
| | | 0 | 0 |

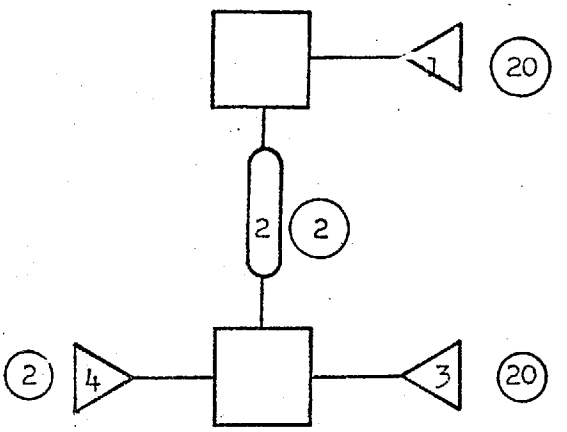
If it be further assumed that one desires to have a number of channels somehow proportional to the product of the number of lines already commissioned, then the values of $e_{r,i}$ should be like the ones shown in the

table above.

Example:

The following diagram shows the project combination that has been selected during the first stage of optimization in the example of the previous section.

Fig. 18.




The table below shows the incomes, running costs and installation costs of the projects. Note that the projects have been renamed, that is, the project numbers do not correspond to the project numbers in the previous section. The central processors have been amalgamated with the 'largest satellites', which are no other than the main exchanges. Since the commissioning of a main exchange centre is associated with the central processor and viceversa, their incomes and costs have been added together and considered as a single project.

TABLE 9

| $i$ | $I_i$ | $p_{io}$ | $p_i$ | $q_{io}$ | $q_i$ | $b_i$ |
|---|---|---|---|---|---|---|
| 0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 315 |
| 1 | 4.0 | 7.0 | 1.5 | 35.0 | 10.0 | 20 |
| 2 | 1.0 | 1.0 | 0.0 | 8.0 | 1.0 | 2 |
| 3 | 4.0 | 8.0 | 1.0 | 41.0 | 12.0 | 20 |
| 4 | 4.0 | 2.0 | 1.0 | 14.0 | 7.0 | 2 |

A computing machine has been programmed with the above described algorithm and data. The following table shows the results obtained.

TABLE 10

| $x$ | $G(x)$ | $b_0^*(x)$ | $b_1^*(x)$ | $b_2^*(x)$ | $b_3^*(x)$ | $b_4^*(x)$ |
|---|---|---|---|---|---|---|
| 0 | 0.0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0.0 | 1 | 0 | 0 | 0 | 0 |
| 2 | 0.0 | 2 | 0 | 0 | 0 | 0 |
| . | . | | | | | |
| . | . | | | | | |
| . | . | | | | | |
| 65 | 0.5 | 0 | 3 | 0 | 0 | 0 |
| . | . | | | | | |
| . | . | | | | | |
| 75 | 3.0 | 0 | 4 | 0 | 0 | 0 |
| 85 | 5.5 | 0 | 5 | 0 | 0 | 0 |
| 95 | 8.0 | 0 | 6 | 0 | 0 | 0 |
| 105 | 10.5 | 0 | 7 | 0 | 0 | 0 |
| 115 | 13.0 | 0 | 8 | 0 | 0 | 0 |
| 125 | 15.5 | 0 | 9 | 0 | 0 | 0 |
| 135 | 18.0 | 0 | 10 | 0 | 0 | 0 |
| 145 | 20.5 | 0 | 11 | 0 | 0 | 0 |
| 155 | 23.0 | 0 | 12 | 0 | 0 | 0 |
| 165 | 25.5 | 0 | 13 | 0 | 0 | 0 |
| 175 | 28.0 | 0 | 14 | 0 | 0 | 0 |
| 185 | 30.5 | 0 | 15 | 0 | 0 | 0 |

TABLE 10 (Cont.)

| x | G(x) | $b_0^*(x)$ | $b_1^*(x)$ | $b_2^*(x)$ | $b_3^*(x)$ | $b_4^*(x)$ |
|---|---|---|---|---|---|---|
| 195 | 33.0 | 0 | 16 | 0 | 0 | 0 |
| 205 | 35.5 | 0 | 17 | 0 | 0 | 0 |
| 215 | 38.0 | 0 | 18 | 0 | 0 | 0 |
| 225 | 40.5 | 0 | 19 | 0 | 0 | 0 |
| 235 | 43.0 | 0 | 20 | 0 | 0 | 0 |
| 245 | 43.0 | 0 | 0 | 0 | 17 | 0 |
| 257 | 46.0 | 0 | 0 | 0 | 18 | 0 |
| 269 | 49.0 | 0 | 0 | 0 | 19 | 0 |
| 281 | 52.0 | 0 | 0 | 0 | 20 | 0 |
| 297 | 53.0 | 0 | 0 | 0 | 19 | 2 |
| 302 | 53.0 | 0 | 0 | 0 | 20 | 1 |
| 309 | 56.0 | 0 | 0 | 0 | 20 | 2 |
| 310 | 56.0 | 1 | 0 | 0 | 20 | 2 |
| 311 | 56.0 | 2 | 0 | 0 | 20 | 2 |
| 312 | 56.0 | 3 | 0 | 0 | 20 | 2 |
| 313 | 56.0 | 4 | 0 | 0 | 20 | 2 |
| 314 | 56.0 | 5 | 0 | 0 | 20 | 2 |
| 315 | 56.0 | 6 | 0 | 0 | 20 | 2 |

The above table gives the number of items $b_1^*(x)$ as a function of the budget constraint x;and the largest possible value G(x) obtained with them.

As can be seen, for x = $ 315,000.- the optimum combination is: $b_0^*(315) = 6$ , $b_1^*(315) = 0$ , $b_2^*(315) = 0$ , $b_3^*(315) = 20$ and $b_4^*(315) = 2$ . That is, the optimum is obtained when the main exchange (3) and the satellite exchange (4) are selected. The maximum net income profit that can be obtained per year is G(315) = $ 56,000.- and the commissioning cost is:

$$\sum_{i=3,4} q_{io} + b_i^*(315).q_i = \$\ 309,000.-$$

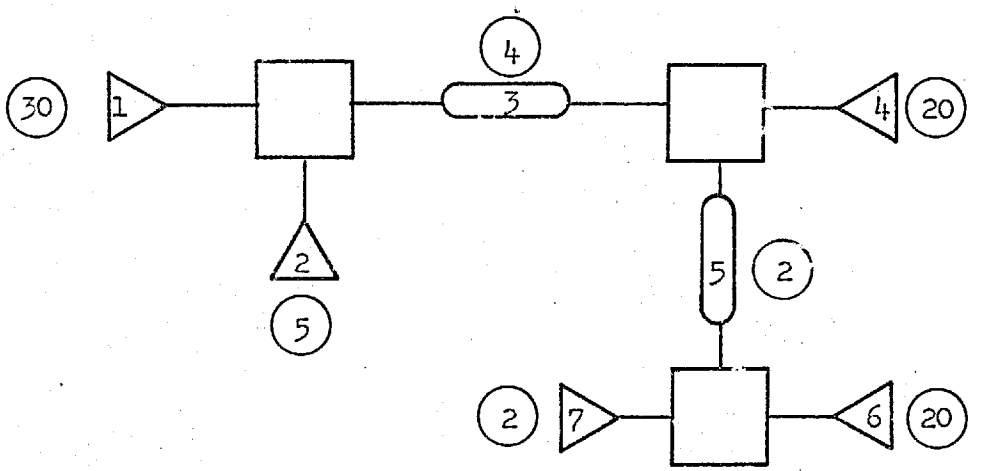
There is a remaining amount of $ 6,000.- from project zero that will be added to the budget of the next period.

## V.9 ONE STAGE OPTIMIZATION

The one stage optimization is carried out with the algorithm described in Section V.8, but considering the whole network at once. The example of Section V.7 and Section V.8 will be used here and the results compared.

The network is to be commissioned in four periods. For each period there will be funds available of $f_t$ = $ 315,000.- . The following diagram shows the network where the projects have been renamed for the same reason as in Section V.8.

Fig. 19.




The following table shows the incomes and costs to calculate the optimum investments for the first period.

TABLE 11

| $i$ | $I_i$ | $p_{io}$ | $p_i$ | $q_{io}$ | $q_i$ | $b_i$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 315 |
| 1 | 4 | 14 | 1 | 100 | 16 | 30 |
| 2 | 4 | 4 | 1 | 36 | 14 | 5 |
| 3 | 2 | 3 | 0 | 16 | 1 | 4 |
| 4 | 4 | 7 | 1.5 | 35 | 10 | 20 |
| 5 | 1 | 1 | 0 | 8 | 1 | 2 |
| 6 | 4 | 8 | 1 | 41 | 12 | 20 |
| 7 | 4 | 2 | 1 | 14 | 7 | 2 |

A computer has been programmed with the algorithm of Section V.8 to solve this problem and the results are in the following table.

TABLE 12

| $t$ | $G(x)$ | $x$ | $b_1^*(x)$ | $b_2^*(x)$ | $b_3^*(x)$ | $b_4^*(x)$ | $b_5^*(x)$ | $b_6^*(x)$ | $b_7^*(x)$ |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 56.0 | 309 | 0 | 0 | 0 | 0 | 0 | 20 | 2 |
| 2 | 44.0 | 245 | 0 | 0 | 0 | 20 | 2 | - | - |
| 3 | 40.0 | 388 | 18 | 0 | 0 | - | - | - | - |
| 4 | 52.0 | 318 | 12 | 5 | 4 | - | - | - | - |
| Totals | | 1260 | 30 | 5 | 4 | 20 | 2 | 20 | 2 |

The result for the first period is the same as the one obtained with the two stage optimization of Sections V.7 and V.8. This is not surprising because in the first stage of optimization the optimum was taken from total enumeration and the algorithm of the second stage (the same as the one used in this Section) guarantees the optimum. If the optimization of the first stage had been done by the algorithm described in Section V.6, in this particular case the result would have been exactly

the same as the one obtained with total enumeration, but in a more complex case the result could have been quite different and not the absolute maximum.

The projects left for the second period are the (1), (2), (3), (4) and (5) with a budget of £ 315,000.- plus the £ 6,000.- remaining from the first period, that is, a total of £ 321,000.- . The optimum result, as can be seen from the table above, is the commissioning of projects (4) and (5), leaving £ 76,000.- unused.

The projects left for the third period are the (1), (2), and (3) with a budget of £ 391,000.- (315,000.- + 76,000.-). The optimum is achieved by commissioning 18 items of project (1) at a cost of £ 388,000.- , remaining £ 3,000.- .

The projects left for the fourth period are: 12 items of project (1), and projects (2) and (3). The total budget for the commissioning of these projects is £ 318,000.- and this is the last period. Therefore, no optimization is needed.

The memory capacity of the computing machine to run the program when the optimization is carried out in two stages is one half of that necessary to process the one stage optimization with the knapsack type of solution.

The main advantage of solving the problem as a knapsack problem is that the optimum is guaranteed. Another advantage is that the solution is obtained in the form of a table where the maximum value of G(x) and the

number of items for each project is given. This may be useful if one wants to 'adjust' the budget constriction according to the results.

# CHAPTER VI

## A MODEL CASE PLANNING

In Chapter V two methods have been shown to optimize investments in a telephone network when the network structure and the budget constrictions are known. The problems of weighing the social preferences of the projects, according to the desires of different agencies, and a method to accomplish this is suggested in the next chapter.

In this chapter a model case will be studied. The model is an imaginary country called Teleria.* It has all the main features of a developing nation. In the following section a brief description of the country

---

* In 1964, the United Nations organized, through its Training Programme in Development Financing, a series of seminars on the technical and financial aspects of telephone projects. Those seminars were held in Geneva in the summer of 1964, and attended by senior officials from economic and planning organizations of twenty countries in Africa and the Middle East. At that occasion lectures were given by Mr. Sven Lönnström and Mr. Ingemar Moo dealing broadly with the problems of planning telephone projects. Based on those lectures and with the contribution of Mr. Folke Marklund they published in 1965 a book entitled A Telephone Development Project, where they consider a hypothetical country called Teleria. In this study, and with permission from the authors, the findings of the present work will be applied to the network planned for Teleria. The attractiveness of Teleria for this purpose is that its description has been carefully made to suit many of the important characteristics of the developing countries.

is given to illustrate the case. Part of the information on Teleria has been obtained from the literature,[L2,*] but it has been here expanded to complete the model.

## VI.1 THE COUNTRY

Figure 20 shows a general map of Teleria. The country is situated within the tropical zone. Lowlands are to be found along the coast and in the northern part of the country. The eastern and southern parts are mountainous.

Teleria has a Capital with 400,000 inhabitants, and it is the main administrative centre and a natural centre of an agricultural area.

The other major cities are:

Northtown: an old commercial centre in the coffee area with 85,000 inhabitants,

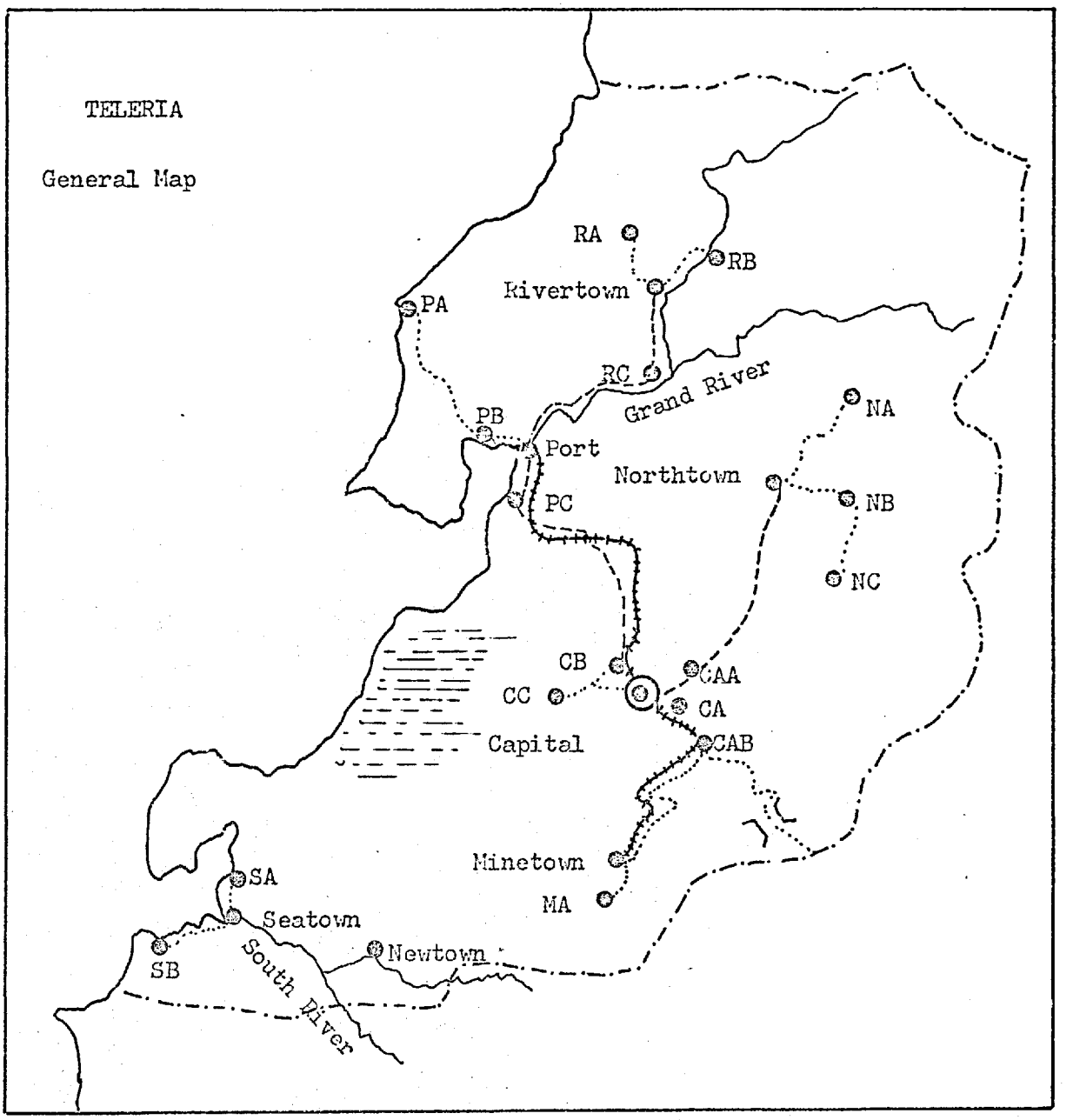
Port : largest harbour of the country, with 200,000 inhabitants, and the most important commercial centre since all imports and exports are traded here,

Rivertown: situated in the lowlands on a tributary of the Grand River. It has 250,000 inhabitants and is the commercial and manufacturing centre of the cotton-growing area. It is also the most industrialized city. Cotton is the main

---

* The author wishes to make full acknowledgment to L. M. Ericsson, and to Messrs. Lönnström, Marklund, and Moo for granting permission to reproduce here the map of Teleria and Tables 13 and 14.

Fig. 20.


TELERIA
General Map
RA
RB
Rivertown
PA
RC
Grand River
NA
PB
Port
Northtown
NB
PC
NC
CB
CAA
CC
CA
Capital
CAB
Minetown
SA
MA
Seatown
Newtown
SB
South River
Legend
Capital
Other towns
Railways
Mayor roads
Minor roads

export of Teleria.

Minetown : is in the mountain area. It has 70,000 inhabitants, and its main resources come from the iron ore mines nearby.

Seatown : has 60,000 inhabitants. It is the main centre of an area partially isolated from the rest of the country by marshlands and mountains.

Teleria has a total population of 7.2 millions, of whom 1.5 millions are urban or suburban. The rest, over 70% are mainly peasants. There has been a population increase of approximately 1.5% per year, and it is thus assumed that in the next decade (planning horizon for the telecommunications network) the population will grow by 16% . Therefore, in ten years time the total population of Teleria will be approximately 8.3 millions.

## VI.2 THE TELEPHONE NETWORK AND THE FORECASTED NEEDS

The present telephone network is unsatisfactory for the development of Teleria. The main needs in communications are from the interior to the coast of the country. Only in the Capital and Port is there automatic telephone service.

The long distance network consists of the following trunk circuits:

Capital -- Port 9

Capital -- Minetown 3

Capital -- Northtown 3

Port -- Rivertown 3

and is entirely manual, causing waiting time of several hours for a call.

Since here we are not concerned with the study of the methods of forecasting needs, we assume that such a study has been made. The desired number of lines for each exchange and the desired number of channels for each route, is known. Furthermore, we assume that although the precarious existing network will still be operating during the next few years, it will be gradually replaced. Hence, it will have very little influence in the technical and economic planning of the new network.

Table 13, in the next page, shows, for every city and town, the present and future population, the estimated telephone density and the planned number of subscribers' lines.

The long distance network has also been planned to be completed in ten years time. Table 14 shows the number of channels for each route. Here, only the main long distance routes are considered.

TABLE 13

| Town | Present population | Expected increase over next decade % | Population 10 years ahead (approx.) | Estimated telephone density % | Number of subscriber lines |
|---|---|---|---|---|---|
| Capital | 400,000 | 60 | 650,000 | 6 | 39,000 |
| CA | 50,000 | 40 | 70,000 | 2 | 1,400 |
| CAA | 30,000 | 20 | 35,000 | 1 | 350 |
| CAB | 20,000 | 25 | 25,000 | 1 | 250 |
| CB | 40,000 | 40 | 55,000 | 2 | 1,100 |
| CC | 20,000 | 50 | 30,000 | 1 | 300 |
| Minetown | 70,000 | 60 | 110,000 | 4 | 4,400 |
| MA | 10,000 | 100 | 20,000 | 1 | 200 |
| Seatown | 60,000 | 65 | 100,000 | 4 | 4,000 |
| SA | 15,000 | 30 | 20,000 | 1 | 200 |
| SB | 20,000 | 25 | 25,000 | 1 | 250 |
| Newtown | 5,000 | 400 | 25,000 | 1 | 250 |
| Northtown | 85,000 | 40 | 120,000 | 4 | 4,800 |
| NA | 20,000 | 30 | 25,000 | 1 | 250 |
| NB | 15,000 | 30 | 20,000 | 1 | 200 |
| NC | 10,000 | 50 | 15,000 | 1 | 150 |
| Port | 200,000 | 60 | 320,000 | 6 | 19,200 |
| PA | 40,000 | 30 | 50,000 | 2 | 1,000 |
| PB | 30,000 | 30 | 40,000 | 1 | 400 |
| PC | 10,000 | 30 | 13,000 | 1 | 130 |
| Rivertown | 250,000 | 60 | 400,000 | 4 | 16,000 |
| RA | 60,000 | 30 | 80,000 | 2 | 1,600 |
| RB | 10,000 | 20 | 12,000 | 1 | 120 |
| RC | 30,000 | 30 | 40,000 | 1 | 400 |
| Others | 5,700,000 | 5.6 | 6,019,000 | --- | --- |
| Total | 7,200,000 | | 8,325,000 | | 95,950 |

TABLE 14

| Route | A | E | AE | n |
|---|---|---|---|---|
| C-M | 13.8 | 1 | | 23 |
| C-S | 8.3 | 1 | | 16 |
| C-N | 10.2 | 1 | | 18 |
| C-P | 24.0 | 1 | | 35 |
| C-R | 12.7 | 10 | 1.3 | 15 |
| P-R | 5.6 | 1 | | 12 |

where:

A = Traffic flow in Erlangs

E = Grade of Service in percentage

AE= Traffic rejected from direct route (alternative routing)

n = Required number of channels

C = Capital

M = Minetown

S = Seatown

N = Northtown

P = Port

R = Rivertown

Fig.21 shows a map of Teleria with the planned network. Fig.22 shows a diagram of the network with the numbered projects and the number of items in each case; and Table 15 shows the incomes, running costs and commissioning costs of the projects.

Fig. 21.

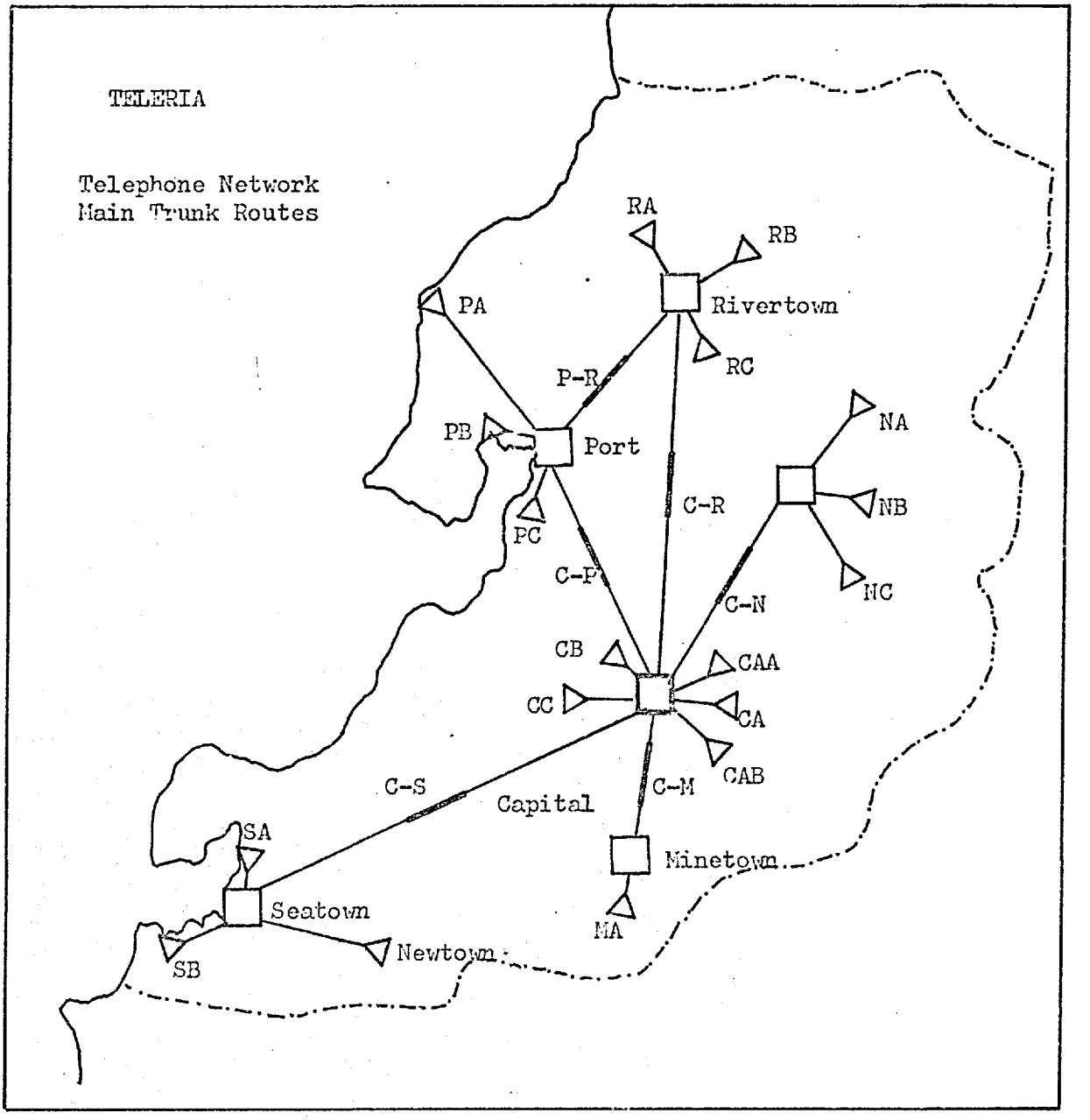

TELERIA
Telephone Network
Main Trunk Routes
RA
RB
PA
Rivertown
RC
P-R
PB
Port
NA
NB
NC
PC
C-R
C-P
C-N
CB
CAA
CC
CA
CAB
C-S
Capital
C-M
SA
Minetown
Seatown
MA
SB
Newtown
Legend
Zone Exchange..
Satellite .....
Exchange
Trunk .....

Fig. 22.

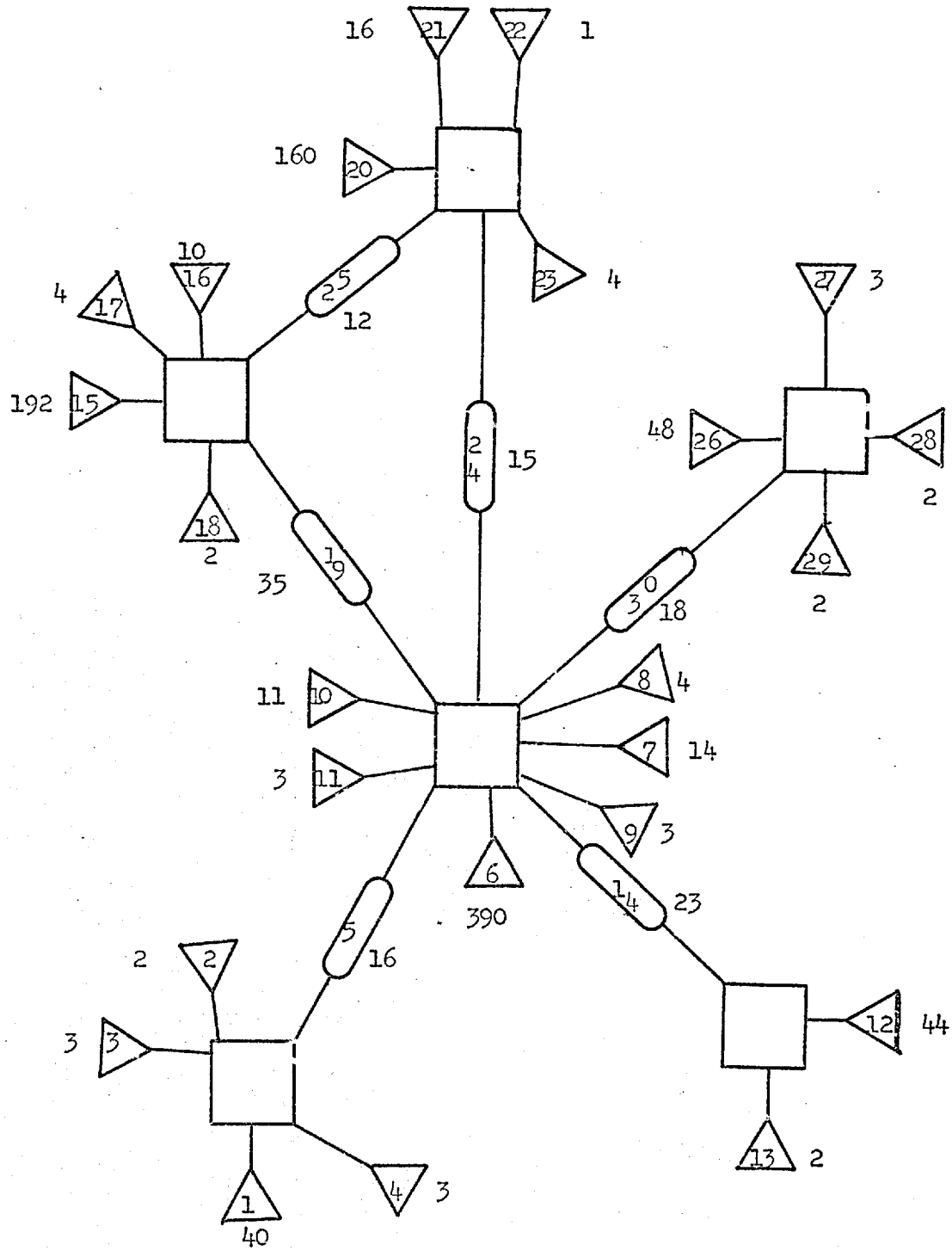

16
21
22
1
160
20
23
4
10
16
4
17
25
12
27
3
192
15
24
15
48
26
28
2
18
2
29
2
35
19
30
18
11
10
8
4
7
14
3
11
9
3
6
390
14
23
5
16
2
2
3
3
12
44
13
2
1
40
4
3

TABLE 15

| i | $I_i$ | $p_{io}$ | $p_i$ | $q_{io}$ | $q_i$ | $b_i$ | Name | Total Cost |
|---|---|---|---|---|---|---|---|---|
| 1 | 1.0 | 2.4 | 0.3 | 35.0 | 2.5 | 40 | Seatown | 1350 |
| 2 | 1.0 | 0.1 | 0.3 | 5.4 | 2.6 | 2 | SA | 106 |
| 3 | 1.0 | 0.1 | 0.3 | 5.4 | 2.5 | 3 | SB | 129 |
| 4 | 1.0 | 0.1 | 0.4 | 5.9 | 2.8 | 3 | SC(Newtown) | 143 |
| 5 | 1.0 | 0.5 | 0.0 | 30.0 | 2.0 | 16 | Trunk C-S | 620 |
| 6 | 1.0 | 20.0 | 0.3 | 180.0 | 2.3 | 390 | Capital | 10,770 |
| 7 | 1.0 | 1.2 | 0.4 | 25.0 | 2.8 | 14 | CA | 642 |
| 8 | 1.0 | 0.1 | 0.4 | 5.2 | 2.9 | 4 | CAA | 168 |
| 9 | 1.0 | 0.1 | 0.4 | 5.0 | 2.9 | 3 | CAB | 137 |
| 10 | 1.0 | 1.2 | 0.4 | 18.0 | 2.8 | 11 | CB | 488 |
| 11 | 1.0 | 0.2 | 0.3 | 5.0 | 2.7 | 3 | CC | 131 |
| 12 | 1.0 | 2.4 | 0.3 | 35.0 | 3.0 | 44 | Minetown | 1670 |
| 13 | 1.0 | 0.1 | 0.5 | 5.2 | 3.0 | 2 | MA | 112 |
| 14 | 1.0 | 0.5 | 0.0 | 30.0 | 2.3 | 23 | Trunk C-M | 829 |
| 15 | 1.0 | 8.5 | 0.3 | 140.0 | 2.3 | 192 | Port | 5816 |
| 16 | 1.0 | 2.0 | 0.4 | 25.0 | 3.2 | 10 | PA | 570 |
| 17 | 1.0 | 0.2 | 0.4 | 5.2 | 3.0 | 4 | PB | 172 |
| 18 | 1.0 | 0.1 | 0.4 | 5.2 | 3.0 | 2 | PC | 112 |
| 19 | 1.0 | 0.5 | 0.0 | 32.0 | 2.2 | 35 | Trunk C-P | 1090 |
| 20 | 1.0 | 8.0 | 0.3 | 120.0 | 2.5 | 160 | Rivertown | 5200 |
| 21 | 1.0 | 0.8 | 0.4 | 32.0 | 2.5 | 16 | RA | 720 |
| 22 | 1.0 | 0.1 | 0.3 | 5.0 | 2.6 | 1 | RB | 76 |
| 23 | 1.0 | 0.1 | 0.3 | 5.2 | 2.5 | 4 | RC | 152 |
| 24 | 1.0 | 0.5 | 0.0 | 32.0 | 2.0 | 15 | Trunk C-R | 620 |
| 25 | 1.0 | 0.5 | 0.0 | 26.0 | 2.2 | 12 | Trunk P-R | 524 |
| 26 | 1.0 | 2.4 | 0.3 | 45.0 | 2.4 | 48 | Northtown | 1602 |
| 27 | 1.0 | 0.5 | 0.2 | 5.0 | 2.4 | 3 | NA | 122 |
| 28 | 1.0 | 0.4 | 0.3 | 5.4 | 2.4 | 2 | NB | 102 |
| 29 | 1.0 | 0.5 | 0.2 | 5.4 | 2.4 | 2 | NC | 102 |
| 30 | 1.0 | 0.5 | 0.0 | 31.1 | 2.3 | 18 | Trunk C-N | 725 |

The values of $I_i, p_{io}, p_i, q_{io}$ and $q_i$ are in £ x $10^4$ .
The total costs are in £ x $10^3$ .

## VI.3 THE PLAN OF INVESTMENT

A computing machine has been programmed with the algorithm of Section V.8 in order to find the optimum investment strategy for Teleria. Table 16 shows the results obtained from such optimization when neither the budget available nor any environmental condition has changed during the whole ten periods, and assuming a functional rank equal to 1 for each project.

From Table 16 one can observe that in the first period investments are made in Seatown, Northtown and their satellite exchanges. In the second and third periods investments are made in Rivertown, and so on.

The main concern of this and the previous chapter has been to formulate a method of optimization of investments. The numerical values assumed in the case of Teleria were arbitrary and the normalized values of social ordering (or social ranking) were also arbitrarily set equal to 1 . The results are thus optimum with respect to those values.

The optimization procedure has been performed by running ten times (ten periods) the same computer program, changing only a few data values every time to eliminate from the optimization the projects already selected in previous periods. In a real case, a revision of the costs and/or socio-economic weights may be made every period in order to adjust to unforeseen events or developments.

As a concluding remark on Chapters V and VI, it

TABLE 16

Projects
Number of items commissioned during the period t

| t | x | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 345.2 | 40 | 2 | 3 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 352.5 | - | - | - | - | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 349.7 | - | - | - | - | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 44 | 0 | 0 | 0 |
| 4 | 351.6 | - | - | - | - | 0 | 0 | 0 | 0 | 0 | 0 | 0 | - | 0 | 0 | 92 |
| 5 | 349.4 | - | - | - | - | 0 | 0 | 0 | 0 | 0 | 0 | 0 | - | 0 | 0 | 100 |
| 6 | 350.2 | - | - | - | - | 0 | 74 | 0 | 0 | 0 | 0 | 0 | - | 0 | 0 | - |
| 7 | 349.6 | - | - | - | - | 0 | 152 | 0 | 0 | 0 | 0 | 0 | - | 0 | 0 | - |
| 8 | 349.6 | - | - | - | - | 0 | 152 | 0 | 0 | 0 | 0 | 0 | - | 0 | 0 | - |
| 9 | 350.4 | - | - | - | - | 0 | 0 | 0 | 0 | 0 | 0 | 0 | - | 0 | 20 | - |
| 10 | 351.8 | - | - | - | - | 16 | 12 | 14 | 4 | 3 | 11 | 3 | - | 2 | 3 | - |
| TOTALS | 3500.0 | 40 | 2 | 3 | 3 | 16 | 390 | 14 | 4 | 3 | 11 | 3 | 44 | 2 | 23 | 192 |

TABLE 16 --Continued.

| t | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 48 | 3 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 93 | 0 | 0 | 0 | 0 | 0 | - | - | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 67 | 0 | 0 | 4 | 0 | 0 | - | - | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | - | 0 | 0 | - | 0 | 0 | - | - | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | - | 14 | 0 | - | 0 | 12 | - | - | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | - | 0 | 0 | - | 0 | - | - | - | 0 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | - | 0 | 0 | - | 0 | - | - | - | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | - | 0 | 0 | - | 0 | - | - | - | 0 | 0 | 0 |
| 9 | 0 | 4 | 0 | 34 | - | 2 | 1 | - | 15 | - | - | - | 0 | 2 | 15 |
| 10 | 10 | - | 2 | 1 | - | - | - | - | - | - | - | - | 2 | - | 3 |
| TOTALS | 10 | 4 | 2 | 35 | 160 | 16 | 1 | 4 | 15 | 12 | 48 | 3 | 2 | 2 | 18 |

can be said that the algorithm of Section V.8 could be easily applied to solve similar problems involving other types of networks and not only of telecommunications, as long as the data and constrictions of the model can be presented inatable such as Table 15.

# CHAPTER VII

## SOCIAL PROFIT AND OTHER PRIORITIES

As Merriman has stated, "the task of the engineer in telecommunications is not only to create the facility, but also to concern himself with the possible impact of that facility upon social structure and individual reactions".[M2] It is within this context that the present study has concerned itself also with the formulation of a method for ranking various alternative projects according to their social profit and their incorporation to the overall ranking.

### VII.1 THE RANKING OF PROJECTS

In Chapter V the economic profit was expressed as an absolute and as a relative value. The absolute economic profit was given by the difference between the income or revenues derived from operating the project concerned, and the running costs. This value gives the net monetary income. Another expression used was the relative economic profit, or ranking, of the project, where the net income is related to the cost to commission the project.

One could immediately attempt to do the same type of ranking when considering profits other than the

purely economic ones. For instance, one may rank a certain project considering the health care of the population of a particular area. It is a well known fact that medical doctors are a scarce resource in the developing areas. In rural areas one could measure how many people or how many hamlets could be assisted with one doctor if there was an efficient means of communication. Thus, the Ministry of Health of the country concerned could rank the projects as follows:

$$\text{Rank for project } i = \frac{\text{net profit (i)}}{\text{scarce resources (i)}} = \frac{\text{possible number of people assisted (i)}}{\text{number of doctors(i)}}$$

This way of ranking projects on a non-monetary basis can be used in some cases; in others, it may seem quite difficult to find a numerical expression that reflects the net profit and/or the scarce resources to be invested.

An attempt was made by Wellenius[W3] to develop a criteria to assign priorities to telephone projects for developing countries. He does this by using a 'synthetic indicator of socio-economic situation' of the towns concerned. The indicators are obtained from the degree of institutionalization achieved by those towns. That is to say, the more established institutions they have (police stations, airports, hospitals, ports, banks, etc.), the higher will the indicator be, and hence the higher will be the priority given it to obtain attention to its tele-

phone needs. This criterion introduces a value judgment prior to any decision making process and will most probably lead to further the development of the most developed areas of the country. Furthermore, in case a national plan for development exists, the method does not give room to weigh the objectives of such a plan.

Other methods of assigning ranks to independent projects are the two described by W. Churchman, R. Ackoff and E. Arnoff,[C3] and more particularly the one by P. Stillson.[S3] All three methods are very similar in nature. Here one example of the basic steps used in all of them will be presented.

A government agency (not the one responsible for the telephone network) is asked to rank three different telephone projects according to the benefit each project would bring to the objectives of that agency if it were commissioned. One of these projects might be the extension of one automatic exchange in a suburban area; another might be the installation of a new long distance trunk between two towns; and another one might be an increment in the number of channels of an existing long distance trunk, and so on. The government agency is represented by a board of planners that makes decisions on a democratic basis, that is, averaging votes. The steps of the procedure to be followed by the board of planners would be as follows:

1- Make a tentative listing of all objectives of

the agency in the order of importance, giving to the objective at the top of the list the value of 100 and assigning to the others arbitrary values that indicate approximately the relevant importance of the objectives. If the agency concerned were the Department of Agriculture, the objectives might be:

$O_1$ Increment of agricultural output by 10%

$O_2$ Optimization of the use of existing agricultural machinery

$O_3$ Improvement in the use of the existing irrigation system

$O_4$ Provision of technical assistance where and when needed

$O_5$ Raising the education and social level of farmers

and the tentative assignment of values might be:

to objective $O_1$ : $v_1 = 100$

to objective $O_2$ : $v_2 = 77$

to objective $O_3$ : $v_3 = 70$

to objective $O_4$ : $v_4 = 43$

to objective $O_5$ : $v_5 = 25$

2- Each member of the board of planners, separately, proceeds to re-adjust the assigned values as follows. $O_1$ is compared to $O_2$ plus $O_3$ and if the importance of the objective $O_1$ is greater than that of the objectives $O_2$ and $O_3$ combined, then the sum of $v_2$ plus $v_3$ should be

less than $v_1$. Otherwise, the values of $v_2$ and/or $v_3$ should be re-adjusted conveniently. Conversely, if the importance of $O_1$ is less than that of $O_2$ and $O_3$ combined, then $v_1$ should be less than $v_2$ plus $v_3$.

3- Next, a comparison is made between $O_1$ and the combination of $O_2$ and $O_4$. The following list indicates the set of comparisons to be made. The symbol (:) will be used to mean 'compared with' and the symbol (&) to mean 'combined with'.

$O_1 : O_2 \,\&\, O_3$
$O_1 : O_2 \,\&\, O_4$
$O_1 : O_2 \,\&\, O_5$
$O_1 : O_2 \,\&\, O_3 \,\&\, O_4$
$O_1 : O_2 \,\&\, O_3 \,\&\, O_5$
$O_2 : O_3 \,\&\, O_4$
$O_2 : O_3 \,\&\, O_5$
$O_2 : O_3 \,\&\, O_4 \,\&\, O_5$
$O_3 : O_4 \,\&\, O_5$

After each comparison is made, the corresponding decision is noted and, if necessary, a correction made such that the values are consistent with the present and previous decisions. If during the comparisons a large discrepancy exists among most of the new values and the initial tentative ranking, then it is advisable to repeat the procedure starting with the ranking just obtained as if it were the initial tentative ranking.

4- The last step of the iterative procedure is to

normalize the result by dividing each value by the sum of all the resulting values.

5- Once each member of the board has arrived at a final normalized ranking, it is possible to obtain a common ranking by averaging the results. For instance, let the final rankings be:

for the objective $O_1$ the value, $v_1 = 0.45$
for the objective $O_2$ the value, $v_2 = 0.20$
for the objective $O_3$ the value, $v_3 = 0.25$
for the objective $O_4$ the value, $v_4 = 0.05$
for the objective $O_5$ the value, $v_5 = 0.05$

6- Now, a table can be formed having on top of each column one of the objectives and letting the rows indicate the telephone projects to be ranked, as shown below. The table should be filled, if possible, by a committee of technical advisors. The committee's task is to evaluate the utility that each project is going to give to each objective. The committee's judgment in each case should be represented by a number between zero and one, being the meaning of '1' that the project guarantees the objective (or that the utility of the project to the objective is 100%), '0' that the project is totally disadvantageous to the objective and '0.5' that the project is indifferent to the objective. Whenever possible the members of the board should not be members of the committee. The members of the committee and the board should not know the numerical results arrived at by the

other group. Let us assume that the committee's judgments are:

| | $O_1$ | $O_2$ | $O_3$ | $O_4$ | $O_5$ |
|---|---|---|---|---|---|
| Project I | 0.4 | 0.2 | 0.8 | 0.8 | 0.3 |
| Project II | 0.9 | 0.9 | 0.2 | 0.3 | 0.8 |
| Project III | 0.7 | 0.7 | 0.4 | 0.3 | 0.7 |

Then, the expected relative utility or ranking of the projects is obtained as follows:

$$R_i = u_{i1} \cdot v_1 + u_{i2} \cdot v_2 + u_{i3} \cdot v_3 + u_{i4} \cdot v_4 + u_{i5} \cdot v_5$$

for i = I, II, III,

and where $u_{ij}$ is the relative utility of project i to the objective j. In the present example the rankings are:

$$R_I = 0.475$$
$$R_{II} = 0.690$$
$$R_{III} = 0.605$$

## VII.2 THE SOCIAL WELFARE FUNCTION

With one of the above mentioned methods the different government agencies may be able to express numerically the relative advantage of each project to their objectives. Here the problem is only halfway solved since we still have to 'combine' the results given by the various agencies, in order to obtain a single list of rankings.

This is a problem which has been theoretically

studied mainly by social scientists and more recently by political theorists. The most thorough analytical studies have been made by Kenneth J. Arrow[A1] and Duncan Black.[B4] The simplest form of the problem is the so-called 'paradox of voting'. This happens when three voters have to compare three social alternatives in pairs and the resulting social preferences of the three voters cannot be combined in a transitive set of social preferences, even if the individual preferences are transitively ordered. The paradox is best illustrated by the following classical example.

Let us say that there are three alternatives to choose from: a, b, and c. The expression 'aPb' is read as: a is preferred to b. The voters are (1), (2), and (3). Assume that the outcome is as follows:

for voter (1) aPb , bPc and aPc
for voter (2) bPc , cPa and bPa
for voter (3) cPa , aPb and cPb

This is a circular result without any majority selection since:

two voters ( 1 and 3) prefer a to b;
two voters ( 1 and 2) prefer b to c;
two voters ( 2 and 3) prefer c to a.

This paradox is sometimes called l'effet Condorcet after the Marquis de Condorcet, who was the first to recognize the problem in his Essai sur l'Application de l'Analyse á la Probabilité des Décisions Rendues a la

Pluralité des Voix in 1785. The problem has aroused much interest because, so far, in the fields of sociology, politics and economics, the problem of combining individual choices has been solved by using either the criteria of equality or fairness, or that of compromise, and by always introducing some kind of constraints in the choices to avoid paradoxical (conflicting) results.

In the language of sociology the alternatives to choose from are usually called 'social states', the individual rankings are 'individual orderings', the resulting combined ranking is 'social ordering' and the method or function to combine the individual orderings is 'social welfare function'.

In Appendix III the most relevant part of the axiomatic analysis introduced by Arrow is summarized. The essence of Arrow's theorem is that if certain reasonable conditions are placed on the function, then no function exists that can produce a unique, ordered set of social preferences. This conclusion may seem to have a negative character, but it really implies that somehow the set of 'reasonable conditions' has to be violated or restricted in order to avoid the paradox.

The importance of this result becomes more apparent when a social welfare function is adopted, because then, although not all reasonable conditions can be met, it is possible to know in a formal way the type of restrictions imposed on the decision mechanism.

Recent discussions have led some authors to consider dropping the third condition of Arrow's theorem (independence from irrelevant alternatives, see Appendix III), which avoids interpersonal comparison of utility in the ordering process. If this is done many satisfactory social welfare functions are possible.

On this basis Goodman and Markowitz[G2] introduced the concept of irrelevant alternatives given by the interpersonal comparison of tastes.

This argument can be presented by a simple example. Let there be only two alternatives: a and b, to choose from in a society of two persons, and the outcome of the orderings be:

| Person | Rankings for a | b |
|---|---|---|
| 1 | 1 | 2 |
| 2 | 2 | 1 |
| Sum of Ranks | 3 | 3 |

By simply adding the rankings one can find that the best ranked alternative is the one whose sum is less than any other. In the example just mentioned there is no majority decision. But if the preferences of the individuals were of different intensity, and this were taken into account, the outcome would be:

Person 1 prefers strongly a to b

Person 2 prefers slightly b to a

Then, in this case it is argued that it would be reasonable that the society would choose alternative a. This is done in Goodman and Markowitz' work by introducing

'irrelevant' alternatives c, d, e, etc. If this is done, adding one irrelevant alternative, the outcome could have been:

| Persons | Rankings for a | b | c |
|---|---|---|---|
| 1 | 1 | 3 | 2 |
| 2 | 2 | 1 | 3 |
| Sum of Ranks | 3 | 4 | 5 |

In this case it is clear that the alternative a obtained the majority. The weakness of this method may lie in that the irrelevant alternatives may become at once relevant.

In the course of the present work it has been found that in the studies mentioned above there are two underlying assumptions, whose primary importance is not sufficiently taken into account. One of these is the assumption that the orderings made by the members of the society or committee concerned are transitive. This has to be assumed in order to obtain a transitive social ordering, but in many real life cases, it may not be true.

The other assumption, and perhaps this is the most ignored in the literature, is that information about the alternatives is presented to the individuals in a suitable way, which in many instances is hardly true. One may think that good decisions can only be made if information about the alternatives is presented in a suitable way.

One way of solving this problem is to present the information, whenever possible, in different forms and in this way let the individuals order the alternatives based solely on the information presented in one form. Then, do the same again with information presented in another form, and so on. In this way one could see that if there are wide discrepancies among the decisions, one or more forms of presenting the information are not suitable.

Some authors, mainly in the field of economics, are questioning the relevance of Arrow's theorem on social choices as a whole,[B2] while others recognize its relevance, but remark, as W. Vickery does, that "problems of social choice will not vanish merely because the tools available to handle them are declared to be imperfect".[V1] On this subject there is an excellent exposition and summary by W. H. Riker.[R4]

Having pointed out most of the difficulties in arriving at a social welfare function and paying attention to the remarks of Vickery, one should attempt to use the solution that best fits the problem concerned.

The work of Goodman and Markowitz could offer a satisfactory compromise if the idea of introducing irrelevant alternatives is disregarded. Instead, another concept, the concept of intensity of preference could be introduced. One way to do this would be to give to the individual voters the task of ordering alternatives, assigning

simultaneously a degree (or intensity) of preference. This can be easily done if, when ranking the projects (individual ordering), the agencies obtain orderings like the one shown in the example in Section VII.1. In that example it is clear that the Department of Agriculture prefers $R_{II}$ to $R_{III}$, $R_{III}$ to $R_I$, and $R_{II}$ to $R_I$. The following example will illustrate the proposed method.

Three government agencies (e.g., Department of Agriculture, Department of Health, Department of Defense) are participating in the social ordering of three telephone projects called I, II, and III, as in the example of the previous section. Let us assume that the ordering of the Departments is as follows:

| | $R_I$ | $R_{II}$ | $R_{III}$ |
|---|---|---|---|
| Agriculture : | 0.475 | 0.690 | 0.605 |
| Health : | 0.801 | 0.511 | 0.202 |
| Defense : | 0.423 | 0.218 | 0.541 |
| Sums : | 1.699 | 1.419 | 1.348 |
| Normalized Sums $R_i$ : | 0.38 | 0.319 | 0.301 |

Thus, the social ordering is: $R_I \; P \; R_{II}$, $R_{II} \; P \; R_{III}$ and $R_I \; P \; R_{III}$. But, if the individual (agency) orderings had been presented without indicating the intensity of preference, the result would have been the classical paradox.

Therefore, a satisfactory social ordering of projects can be obtained by using the method outlined for the last example. The next step consists in incorporating these social orderings or rankings to the optimization procedures shown in previous sections.

The method by which a committee or group of people have to evaluate projects has to be of such a nature that the evaluation is done for each individual project and not for every combination of them, since, although it would be ideal to have such an evaluation, it would imply the total enumeration of combinations of projects.

In Section V.6, the overall ranking of a vector $V_k$ was defined as:

$$R(V_k) = R_e(V_k) \cdot R_f(V_k)$$

where: $R_e(V_k)$ is the economic ranking of $V_k$

and: $R_f(V_k)$ is the functional ranking of $V_k$

which is a weight-value of the projects included in $V_k$ .

Thus, we define:

$$R_f(V_k) = \sum_{i=1} R_i \cdot v_i$$

where: $v_i = 0$ or 1, as defined in Section V.6,

and: $R_i$ are the normalized values of social ordering.

# CONCLUDING REMARKS

The present work has been primarily concerned with some of the problems encountered by the developing countries when, as part of a development program, have to build a telephone network. The problems regarding the use of technology are different from those faced by the developed countries, presenting a new historic situation. Modern technological knowledge is already available to the developing nations, creating the conflict of selecting, among a wide variety, the most adequate hardware.

Many of today's technological designs are geared towards the richer and more developed countries. This fact leaves the developing ones with the unfavourable alternatives of either choosing a modern device that is not designed for their needs, or a device that may suit their needs more closely, but is considered obsolete elsewhere and will probably become obsolete for them in the foreseeable future.

In this work, a solution to such a problem has been attempted, by proposing a telephone network specifically designed for the developing countries. It can be built entirely with existing technological knowledge, and most of its components are available in the market.

The system description includes the satellite

exchanges and the controlling mechanism which is in the form of a stored-program. A simulation language has been written to operate the system under simulation conditions in a large computer. The network has a centralized control in contrast with earlier systems with decentralized control, giving a different pattern to its implementation.

Then, the problem encountered is that of deciding where and when the investments have to be made in order to optimize the socio-economic effect of the network. A method to find such an optimum plan was designed. It optimizes the invested capital, maintaining flexibility and admitting modifications when the circumstances so require it.

The problem of assigning social weights to the different telephone projects in order to take into account their social impact, have been discussed. A satisfactory social welfare function for this purpose is given.

The fields of study covered in this work could each constitute an area of concentration worth pursuing further. The author has chosen to interrelate these fields because he feels that the type of problem considered here cannot be looked at from a single discipline.

# APPENDIX I

## AN ALGORITHM FOR THE SELECTION OF INVESTMENT PROJECTS

The problem of selecting the best set of independent projects out of a large number of them has been treated previously by S.Reiter.[R1,R3] The problem can be formulated as follows: a decision making body has to select, among many, a set of independent projects each of which will return to the investor a certain profit, generally called pay off, and measured in any appropriate unit. The projects are independent of each other in the sense that each one can be selected together with any other one, and the pay off of two projects taken together may be less, equal or larger than any of the two pay offs taken separately, or the sum of the individual pay offs.

The projects are numbered 1,2,...,n and for each pair of projects, the pay offs $a_{ii}$, $a_{jj}$ and the enhancement $a_{ij}$ are determined for i,j=1,2,...,n . A rectangular matrix $|A|$ , for the pay offs, can be written, where the elements $a_{ij}$ are the pay offs obtained when projects i and j are selected simultaneously.

Then, the pay off matrix can be written as shown in the next page.

$$|A| = \begin{vmatrix} a_{11} & \cdot\cdot & a_{1i} & \cdot\cdot & a_{1n} \\ \cdot & \cdot & & & \cdot \\ \cdot & \cdot & & & \cdot \\ 0 & & a_{ii} & & a_{in} \\ \cdot & & & \cdot & \cdot \\ \cdot & & & \cdot & \cdot \\ 0 & \cdot\cdot & 0 & \cdot\cdot & a_{nn} \end{vmatrix}$$

A selection of investment projects, or program, is represented by a vector $V_k(v_1, v_2, \ldots, v_n)$ where:

$$v_i = \begin{cases} 1 & \text{if the project i is selected} \\ 0 & \text{if the project i is not selected} \end{cases}$$

for $i=1,\ldots,n$ .

$V_k$ is then a vector representing a combination of projects out of a large number of them. The set of all allowed vectors will depend on the constrictions Z imposed by the specific problem. For an unconstricted case: $k=1,2,\ldots,2^n$. But in most problems there will be only a set of vectors $V_k$ which will satisfy the constrictions Z. The set of these vectors is represented by $\{V_k \mid Z\}$ .

Every vector $V_k$ will produce a total pay off $P(V_k)$ . The problem of optimization consists in finding a vector $V_{ko} \in \{V_k \mid Z\}$ such that maximizes the profit $P(V_{ko})$ .

The pay off of a vector is a function of the projects included in it. This function may be, as in the

problem solved by Reiter,[R3] the sum of the individual pay offs: $P(V_{ko}) = \sum_{i,j=1}^{n} a_{ij} \cdot v_{ij}$, or a weighted pay off, or a ranking of them,[H1] or a more complex function.

The possibility of complete enumeration, to find $V_{ko}$, can only be considered when n is small. For large values of n a systematic method should be applied.[R3]

Here several algorithms are shown and numerical results are compared.

The Algorithms

A- Maximum Gradient.[R3] This algorithm consists first in selecting a random vector $\{V_k \mid Z\}$ and calculating its pay off $P\{V_k \mid Z\}$. Then, all possible vectors are formed, by either deleting or adding one project, and their respective pay offs $P\{V_t \mid Z\}$ are calculated.

The gradient function defined by:

$$G(k,t) = P\{V_k \mid Z\} - P\{V_t \mid Z\}$$

is applied to each vector in order to find the vector $\{V_{to} \mid Z\}$ that gives the largest gradient.

The vector $V_{to}$ thus obtained is taken as a new starting combination and the procedure repeated again. The algorithm stops when a vector $V_{ko}$ is found such that:

$$G(ko,t) \leq 0 \text{ , for all } t.$$

Thus, the set of all pay offs form a 'surface', where several local maxima and minima may be found. The probability of reaching the absolute maximum, when

starting with a random selection of projects, will depend on the fraction of starting combinations that lead to it.

With this algorithm, once a maximum has been reached, the whole procedure is repeated by starting again with a new randomly selected combination of projects and the whole procedure has to be repeated several times. Therefore it is convenient to find a way of incrementing the probability of reaching the absolute maximum. The next algorithm is a modified version of the first, which has been shown to reduce the computation time and increase the chances of reaching the absolute maximum in the same number of trials.

The last two algorithms (C and D) differ from the first two in that once a local maximum has been reached the next starting combination is not selected at random but in a more convenient way, showing a reduction in the number of random starts.

B- Positive Gradient.[R1,R2] This algorithm is similar to the previous one, but instead of finding the pay offs $P\left\{V_t \mid Z\right\}$ and the gradient $G(k,t)$ for all t, the computation is carried out until a vector $\left\{V_{t+} \mid Z\right\}$ is found that produces a positive gradient:

$$G(k,t) = P\left\{V_k \mid Z\right\} - P\left\{V_{t+} \mid Z\right\} > 0$$

The vector $V_{t+}$ thus obtained is taken as a new starting combination. The procedure is repeated again as in the previous algorithm.

By stopping the search with the first positive gradient found, the algorithm does not follow the 'steepest slope' but any positive one. The way of reaching the local maximum may seem longer, but a large amount of calculation is saved by avoiding the 'complete enumeration' of all combinations resulting from adding or deleting one project to every starting combination.

C- One at a Time, Plus Two at a Time. This algorithm starts in the same way as the previous one but differs in the last step. The steps are as follows:

I- Select a random starting vector $\{V_k \mid Z\}$ and calculate its pay off $P\{V_k \mid Z\}$ .

II- All possible vectors $\{V_t \mid Z\}$ are tried by deleting or adding one project at a time and their respective pay offs $P\{V_t \mid Z\}$ are calculated. The procedure is stopped when the first positive gradient

$G(k,t+) = P\{V_k \mid Z\} - P\{V_{t+} \mid Z\} > 0$ , is found.

III- The vector $V_{t+}$ thus obtained is taken as a new starting combination and the procedure is repeated again as from (I), until a vector $V_{ko}$ is found, such that:

$G(ko,t) \leqq 0$ , for all t.

IV- Select: $\{V_{ks} \mid Z\}$ , where $V_{ks}$ is as $V_{ko}$ but with two projects changed (added or deleted), and go to I.

Thus the main feature here is that once a local maximum has been reached (by changing one project at a time), the algorithm 'jumps out' of the 'surrounding vectors' by changing two projects at a time. This, as we shall see, will most probably lead to a different local maximum.

A simple problem will be used to illustrate graphically step IV. The pay off matrix as well as the pay off function is similar to the one used by Reiter,[R3] but it has been reduced to four projects, to simplify the illustration.

$$|A| = \begin{vmatrix} 0 & -2 & -3 & 5 \\ 0 & 0 & 6 & -3 \\ 0 & 0 & 0 & -2 \\ 0 & 0 & 0 & 0 \end{vmatrix}$$

The resulting 'trees' of vectors leading to the maxima are shown in Fig.30. Fig.31 shows a map, where it is possible to represent each vector by a cell. The columns of the map represent two projects, for example the 1 and 2, and the rows another two projects, 3 and 4. The four columns and the four rows intersect in 16 cells that represent each of all possible combinations of projects. For example, column 01 and row 11 intersect in the cell 0111, which represents the vector V(2,3,4).

An important property of this map is that any move either vertical or horizontal of one cell involves the change of only one project. Thus, the step II

consists of moving from one cell to one of the n adjacent ones. This property is not limited by the edges of the map since the top cell is adjacent to the bottom cell of the same column. Also the leftmost cell is adjacent to the rightmost of the same row. For example, the four adjacent cells to the 1010 are:

1011
1110 1010 0010
1000

The numbers in the center of each cell are the corresponding combination of projects and the arrows show how the algorithm 'moves' from one vector to the adjacent one for this specific problem.

From the map and the algorithm it can be seen that: two maxima cannot occupy adjacent cells, the maxima are at least at a 'distance' of two cells and there can be a maximum of $2^{n-1}$ local maxima. These facts have suggested the present algorithm. In the example shown here as well as in the one used by Reiter[R3] the present algorithm represents an improvement in the probability of reaching another local maximum once the first one has been obtained.

For a problem of 5 or even 6 projects a similar map can be drawn to observe the pattern of the arrows and the location of the maxima, but over 6 projects the map loses its simplicity. Fig.32 shows the map corresponding to the problem given by Reiter.[R3]

The pay off matrix is:

$$|A| = \begin{vmatrix} 0 & -2 & -3 & 5 & -1 \\ 0 & 0 & 6 & -3 & 4 \\ 0 & 0 & 0 & -2 & -5 \\ 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

The encircled dots represent arrows arriving to the cell, while the encircled crosses are arrows departing from the cell, (see Fig. 32).

D- One at a Time, Plus N at a Time. This algorithm follows the same steps I, II and III of the previous one but in step IV instead of starting with $V_{ks}$ Z , the new starting vector is $V_{kN}$ Z , where $V_{kN}$ is as $V_{ko}$ but with N projects changed. Thus, algorithm C is equivalent to algorithm D for N=2 .

Results of Calculated Problems

Many problems with different pay off matrices and different number of projects, were tried. Here, the result of ten of those problems is shown. The result is typical of all the problems tried.

All ten problems have a 10x10 pay off matrix and the entries were numbers between -9 and +9, obtained from a random number generator with a uniform distribution. A 10x10 matrix contains a maximum of 1024 vectors $V_k$. Thus, complete enumeration and the a priori knowledge of the absolute maximum was possible.

The algorithms were applied until the absolute maximum was reached. This was done 100 times for each

problem and the execution time, the number of pay off calculations and the number of random starts were recorded. Table 17 shows, for the algorithms A, B and C, the average execution time (ET) in seconds,* the average number of pay off calculations (PO) and the average number of times (ST) the algorithms had to be restarted with a random vector.

Table 18 shows the percentage of 'improvement' of ET, PO and ST obtained with the algorithm C in comparison with B. As can be seen, in average, there is a 47.48% reduction in the number of random starts (ST). This is due to the fact that after reaching one maximum the next start is not selected at random. The improvements in PO and ET are in average 10.2% and 11.5% respectively.

The execution time ET of the different algorithms should be compared considering that the computer programs differ in complexity from algorithm to algorithm. The number of basic instructions of the different computer programs used are:

| A | B | C | D |
|---|---|---|---|
| 35 | 37 | 51 | 51 |

The improvement of only 11.5% in the average execution time is mainly because the program for the algorithm C is more complex than the one used for B, having a larger number of basic instructions. The improvement in the number of random starts with algorithm C compared

*The CDC 6600 computer of the University of London was used.

to B is very significant when it is related to the probability of reaching the absolute maximum with one random start.

Fig.33 is a plot of the improvement in percentage in the number of random starts as a function of the probability of reaching the absolute maximum $p(V_k)$ with one random selected vector $V_k$. As can be seen, algorithm C is better than B when the probability of finding the absolute maximum in one trial is small.

A comparison has been made between the results obtained with algorithm C and D, for N=3, 4 and 5. Algorithm D has shown no significant improvement in comparison to C. The comparisons with algorithm A are not shown because the differences in ET, PO and ST are so large that it was not considered worthwhile.

In the light of the above results it can be said that algorithms B, C and D are many times more efficient than algorithm A and also that algorithm C is an improvement over B. It is significant to note the large improvement obtained when the probability of reaching the absolute maximum with one random start is small.

TABLE 17

| Problem | A ET | A PO | A ST | B ET | B PO | B ST | C ET | C PO | C ST |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.513 | 859 | 86 | 0.451 | 749 | 16.56 | 0.065 | 105 | 1.13 |
| 2 | 0.606 | 1080 | 98 | 0.029 | 57 | 1.73 | 0.034 | 71 | 1.00 |
| 3 | 0.506 | 879 | 88 | 0.287 | 516 | 11.62 | 0.043 | 75 | 1.00 |
| 4 | 0.486 | 846 | 85 | 0.054 | 99 | 3.02 | 0.075 | 134 | 1.36 |
| 5 | 0.491 | 880 | 88 | 0.090 | 145 | 3.73 | 0.055 | 87 | 1.00 |
| 6 | 0.627 | 1060 | 106 | 0.024 | 41 | 1.04 | 0.023 | 40 | 1.00 |
| 7 | 0.276 | 482 | 48 | 0.056 | 106 | 3.06 | 0.093 | 181 | 2.12 |
| 8 | 0.594 | 1002 | 100 | 0.026 | 40 | 1.00 | 0.025 | 40 | 1.00 |
| 9 | 0.499 | 877 | 79 | 0.055 | 82 | 1.89 | 0.042 | 63 | 1.00 |
| 10 | 0.577 | 1026 | 93 | 0.022 | 40 | 1.00 | 0.023 | 40 | 1.00 |

TABLE 18

| Problem | C vs. B in % ET | PO | ST |
|---|---|---|---|
| 1 | 86 | 86 | 93.18 |
| 2 | -17 | -25 | 42.20 |
| 3 | 85 | 85 | 91.40 |
| 4 | -39 | -35 | 54.97 |
| 5 | 39 | 40 | 73.20 |
| 6 | 4 | 2 | 3.85 |
| 7 | -66 | -74 | 69.00 |
| 8 | 4 | 0 | 0 |
| 9 | 24 | 23 | 47.09 |
| 10 | -5 | 0 | 0 |
| | 11.5 | 10.2 | 47.48 |

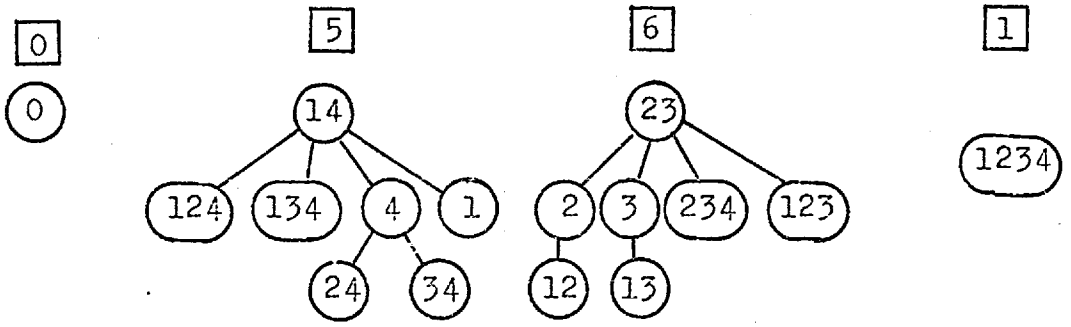

0
0
5
14
124
134
4
1
24
34
6
23
2
3
234
123
12
13
1
1234

Fig. 30.

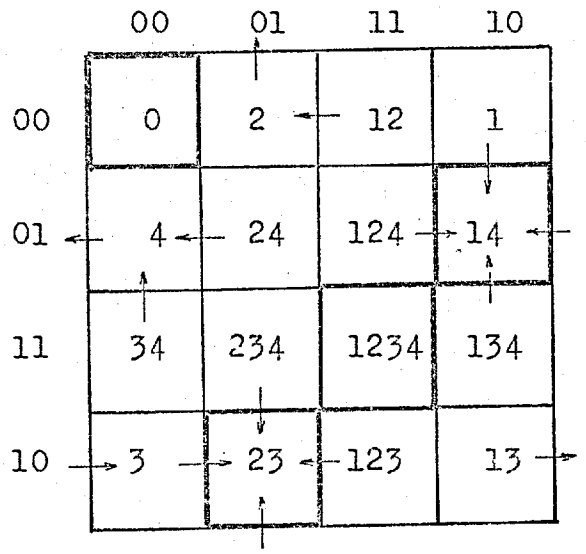

00
01
11
10
00
0
2
12
1
01
4
24
124
14
11
34
234
1234
134
10
3
23
123
13

Fig. 31.

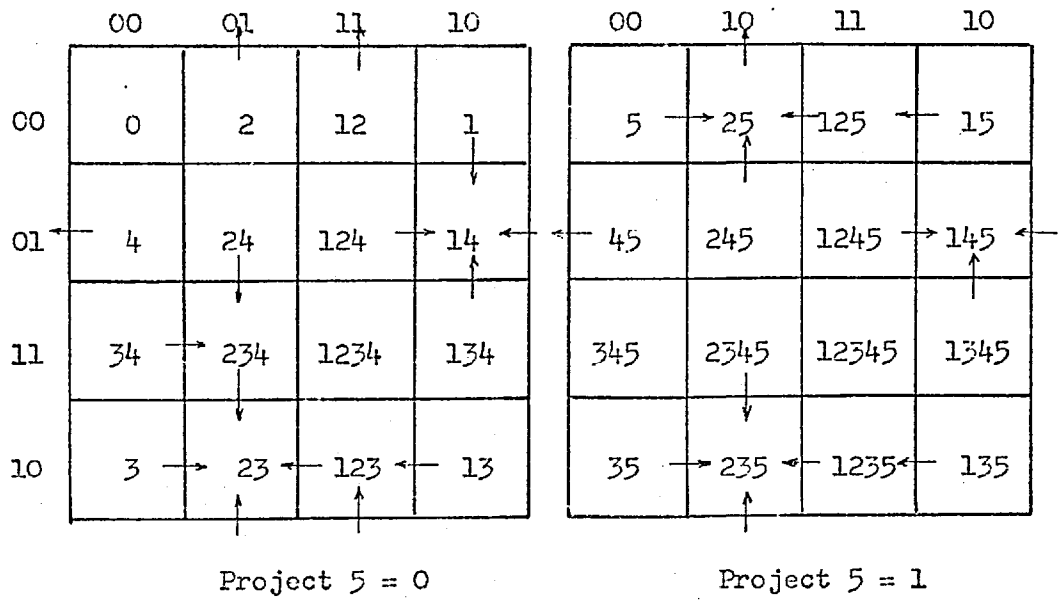

00 01 11 10
00 0 2 12 1
01 4 24 124 14
11 34 234 1234 134
10 3 23 123 13
Project 5 = 0
00 10 11 10
5 25 125 15
45 245 1245 145
345 2345 12345 1345
35 235 1235 135
Project 5 = 1

Fig. 32.

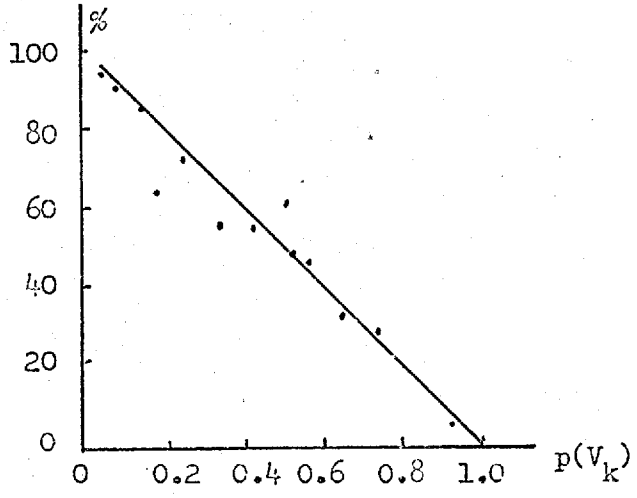

%
100
80
60
40
20
0
0 0.2 0.4 0.6 0.8 1.0
p(V_k)

Fig. 33.

# APPENDIX II

## AN ALGORITHM FOR THE KNAPSACK PROBLEM WITH CONSTRAINTS IN EACH VARIABLE

The knapsack problem in the general form has been treated extensively in a paper by Gilmore and Gomory,[G1] where solutions are provided for some problems. Among others they treat the knapsack problem with bound in the number of each type of item and a solution is proposed but no algorithm is shown for it. Here, an algorithm for the solution of such problem is given. The output is a set of arrays giving the value of the optimum combination in each case and the number of each type of item.

In order to summarize the functions and be consistent with the notation used by Gilmore and Gomory the problem is stated as follows:

$$F(x) = \max \sum_{i=1}^{m} \pi_i \cdot Z_i$$

subject to:

$$\sum_{i=1}^{m} Z_i \cdot l_i \leqq x$$

and:

$$Z_i \leqq b_i$$

A function $G(b_1, \ldots, b_m, x)$ is used to solve the

knapsack problem. This function is defined as follows:[G1]

$$G(0,\ldots,0,x) = 0 \text{ for all } x \tag{1}$$

$$G(b_1,..,b_{i+1},0,..,0,x) = \max_n \left\{ G(b_1,..,b_i,0,..,0,x-n.l_{i+1})+n.\pi_{i+1} \right\} \tag{2}$$

where:

$$x \geqq n.l_{i+1} \quad \text{and} \quad b_{i+1} \geqq n \geqq 0 \tag{3}$$

Equality (2) can also be expressed as follows:

$$G(b_1,..,b_{i+1},0,..,0,x+n.l_{i+1}) = \max_n \left\{ G(b_1,..,b_i,0,..,0,x)+n.\pi_{i+1} \right\} \tag{4}$$

The functions used in the algorithm are (1) and (4) and constraints (3). An explanation of the algorithm based on (2) is as follows. Assume that for the items $1,\ldots,i$, we know the optimum subdivision of the length x, for x between 0 and L, and that these optima are all tabulated. Then, the optimum value for a certain length x will be $G(b_1,\ldots,b_i,0,\ldots 0,x)$. Now, for the same length x we would like to know if the optimum value still improves when a new item i+1 of value $\pi_{i+1}$ and length $l_{i+1}$ is introduced. In order to do this we shorten the length x by $l_{i+1}$ and find in the table what is the optimum value with all items up to the i-th. Then, we prolong the length $x-l_{i+1}$, by $l_{i+1}$ introducing one of the items of value $\pi_{i+1}$. If the value of $G(b_1,..,b_i,0,..,0,x-l_{i+1}) + \pi_{i+1}$ is larger than the

one for the length x and items 1,...i, then, we note this new value and the new subdivision for the length x and items 1,...i+1 . Then, we try by shortening x by two times $l_{i+1}$ and adding these two items of the value $_{i+1}$ , etc. This procedure continues until x is shortened by $n.l_{i+1}$ , where $n=b_{i+1}$ (the bound imposed on the number of items of type i+1). Thus, we obtain a new column for the values of the function (2). Simultaneously with this procedure, we keep track of the number of each type of item that is needed to achieve each optimum. In the algorithm we call these numbers $b_i^*$ , where $b_i^*$ $b_i$ .

In order to simplify the notation in the algorithm and not to confuse the interim values obtained with the final values, the following is defined:

$b_j^*(i,x)$ (for $j \leqq i$) is an array where each point represents the number of items of type j that have been selected when items up to the type i and length x are considered. From $j \leqq i$ it follows for example that if i=2 then, there are only points in the array for $b_0^*(2,x)$ , $b_1^*(2,x)$ and $b_2^*(2,x)$ .

$G_i(x)$ is the interim value of the function $F(b_1,..,b_i\ 0,..0,x)$ during the computation. The final value of $G_i(x)$ will be equal to $F(b_1,..,b_i,0,..,0,x)$ .

Instead of having an item m+1 of length $l_{m+1}=1$ and value $\pi_{m+1}=0$ , here the corresponding item is the item 0 with length $l_0=1$ and value $\pi_0=0$ . The rest of

the notation follows closely the one used by Gilmore and Gomory.

The main features of the algorithm are as follows, (see Algorithm below):

a- From I.1 to I.3 the variables are initialized.

b- From II.1 to III.1 the introduction of the new item i+1 is performed and the values for $G_{i+1}(x+n.l_{i+1})$ and $b^*_j(i+1,x+n.l_{i+1})$ are found.

c- From III.2 to III.4 the rest of the arrays are updated.

d- From III.5 to IV.2 the value of x is incremented.

e- From V.1 to V.2, i is incremented and the values of the arrays i+1 are transferred to the arrays i.

Only two sets of arrays are needed at one time. Therefore, the algorithm requires at least two sets of arrays to work with. One set is called: $G_o(x), b^*_o(o,x), b^*_1(o,x), \ldots, b^*_i(o,x)$ and the other is called: $G_1(x), b^*_o(1,x), b^*_1(1,x), \ldots, b^*_{i+1}(1,x)$ .

At any time during the computation, the first set corresponds to the set i and the second one to the set i+1 . When the computation for a certain i is completed, the values of the set i+1 are transferred to the set i. Thus, an automatic renaming of the sets, for the new i and i+1, takes place.

The number of memory locations needed is $2(m+2).L$ .

The Algorithm

I.1 Reorder the items such that $\pi_{i+1}/l_{i+1} \geqq \pi_i/l_i$ , for $1 \leqq i \leqq m$ .

Initialize $G_a(x)=0$ and $b_j^*(a,x)=0$ for: $0 \leqq x \leqq L$, $a=0,1$ and $0 \leqq j \leqq m$ .

Initialize $b_0^*(0,x)=x$ and $b_1^*(0,x)=x$ for $0 \leqq x \leqq L$ .

Let $i=0$ and $k=0$ .

I.2 Let $x=0$ .

I.3 Let $n=1$ .

II.1 If $x+n.l_{i+1} \leqq L$, then let $V=G_0(x)+n.\pi_{i+1}$ and go to II.2.

Otherwise go to III.1.

II.2 If $V \geqq G_1(x+n.l_{i+1})$ let $G_1(x+n.l_{i+1})=V$ , $b_{i+1}^*(1,x+n.l_{i+1})=n$ and $b_j^*(1,x+n.l_{i+1}) = b_j^*(0,x)$ , for $j=0,\ldots,i$ . Let $k=1$ and go to III.1.

Otherwise go to III.1

III.1 If $n \geqq b_{i+1}$ go to III.2.

Otherwise let $n=n+1$ and go to II.1.

III.2 If $k=1$ go to III.3.

Otherwise go to III.5.

III.3 Do to III.4 for all $t$ from 1 to $L$:

if $G_1(t) < G_1(t-1)$ let:

$$G_1(t) = G_1(t-1) \ ,$$
$$b_j^*(1,t) = b_j^*(1,t-1) \text{ for } 1 \leqq j \leqq i+1 \ ,$$
$$b_0^*(1,t) = b_0^*(1,t-1)+1$$

and go to III.4.

Otherwise, go to III.4.

III.4 If $t \leqq L$ go to III.3.

Otherwise let k=0 and go to IV.1.

IV.1 Let x=x+1 and go to IV.2.

IV.2 If $x+l_{i+1} > L$ ,let i=i+1 and go to V.1.

Otherwise go to I.3.

V.1 If $i \geqq m$ stop.

Otherwise go to V.2.

V.2 Let s=1 and go to V.3.

V.3 Let $G_0(s)=G_1(s)$ , $b_j^*(0,s)=b_j^*(1,s)$ for $0 \leqq j \leqq i$ and go to V.4.

V.4 If $s < L$ let s=s+1 and go to V.3. Otherwise go to I.2.

## APPENDIX III

## ARROW'S AXIOMATIC FORMULATION OF THE SOCIAL WELFARE FUNCTION'S THEOREM

This appendix contains a summary of K. Arrow's axiomatic formulation and conditions for a social welfare function.[A1] The definitions used here are:

xPy is read as : x is preferred to y

xIy is read as : x is indifferent to y

xRy is read as : x is preferred or indifferent to y

The concept of social state is defined by a complete description of the following:

1- the amount of each type of commodity in the hands of each individual,

2- the amount of each productive resource invested in each type of productive activity,

3- the amount of various types of collective activities (municipal, diplomatic, etc.), and

4- the amount of labour supplied by each individual.

In other words, a social state is a vector with many components, some of which are collective activities. The expression $R_i$ will here be read as the ordering of individual i for the alternative social states.

The problem considered here is how one can combine individual preferences when making social choices in a rational way. K. Arrow's formulation is only concerned with the formal procedure, if such thing is possible. Starting with two axioms, he defined the notion of preference.

Axiom I : For all x and y, either xRy or yRx.

That is to say, that is should always be possible to compare any pair of alternatives (x and y).

Axiom II: For all x, y and z, xRy and yRz imply xRz.

This means that the preferences are transitive, e.g., if xRy and yRz, then it would be inconsistent for the voter to select zRx.

Observe that axioms I and II do not exclude the possibility xRy and yRx. This implies that xIy and yIx. If axioms I and II are met, it is said that the individuals are rational.

The set of conditions required for a social welfare function are:*

Condition 1: Universal admissibility of individual ordering

Any individual may choose any one order of alternatives. This condition is very important because one may not like to impose restrictions on individual choices

---

* Following W.H. Riker[R4], a rather less formal language than Arrow's will be used here.

simply to make a social welfare function work.

Condition 2: Positive association of social values

Social ordering must follow positively the individual orderings. That is, that if in the established social order xRy and a group of individuals raises x still further in the ordering, then it will still be xRy (x will not fall below y).

Condition 3: Independence from irrelevant alternatives

This condition means that if in a social ordering including alternatives x, y and z, xRy, and then z is eliminated as an alternative, then it should still remain xRy.

Condition 4: The social welfare function is not to be imposed

This condition is related to Condition 1, and states that the social welfare function must not be given independently of individual preferences.

Condition 5: The social welfare function is not to be dictatorial

Here, dictatorial means that some individual or group of individuals' preference is xRy, and it is then decided that the social preference is xRy, regardless of the preferences of other individuals.

At this point, it may be useful to quote Arrow's

remarks on these conditions:[A1]

> We have now imposed five apparently reasonable conditions on the construction of a social welfare function. These conditions are, of course, value judgments and could be called into question; taken together they express the doctrines of citizens' sovereignty and rationality in a very general form, with the citizens being allowed to have a wide range of values. The question raised is that of constructing a social ordering of all conceivable alternative social states from any given set of individual orderings of those social states, the method of construction being in accordance with the value judgments of citizens' sovereignty and rationality...

Having stated these conditions, Arrow shows his fundamental theorem with a detailed proof.[A1] The theorem is stated thus:

> Every social welfare function satisfying Conditions 1 to 3 and producing a social ordering satisfying Axioms I and II must violate Condition 4 or Condition 5.

# APPENDIX IV

## LOCAL NETWORK OPTIMIZATION

The wiring of the local network is an important part to be considered in the optimization procedure since it accounts for approximately 35 to 45% of the total investment in the network.[L1,C5]

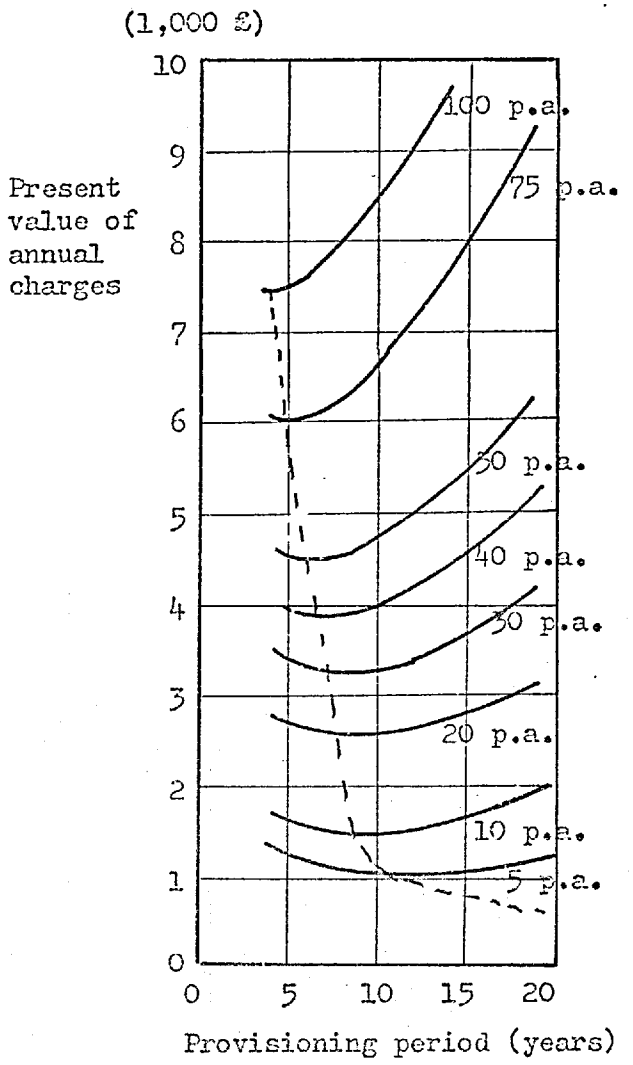
Swain has shown in a study of the costs of the local networks, that the present value of annual charges involved are a function of the rate of growth of the number of lines installed and of the provisioning period.

In the next page, Fig. 34, taken from Swain,[S6] illustrates this function, having as a parameter the rate of growth in the number of lines.

From this function one can observe that if the rate of growth in the number of lines is ten per year, the optimum provision period would be 9 years. Fig. 35 shows the optimum provision period in function of the rate of growth of the subscriber lines.

The numerical values for these functions were obtained by the British G.P.O. from statistics for the United Kingdom. The actual values for another country will probably vary from this. It would then be advisable, whenever possible, to make this type of calculation before deciding the installation of the local network.

Fig. 34


(1,000 £)
Present value of annual charges
100 p.a.
75 p.a.
50 p.a.
40 p.a.
30 p.a.
20 p.a.
10 p.a.
5 p.a.
0
1
2
3
4
5
6
7
8
9
10
0
5
10
15
20
Provisioning period (years)

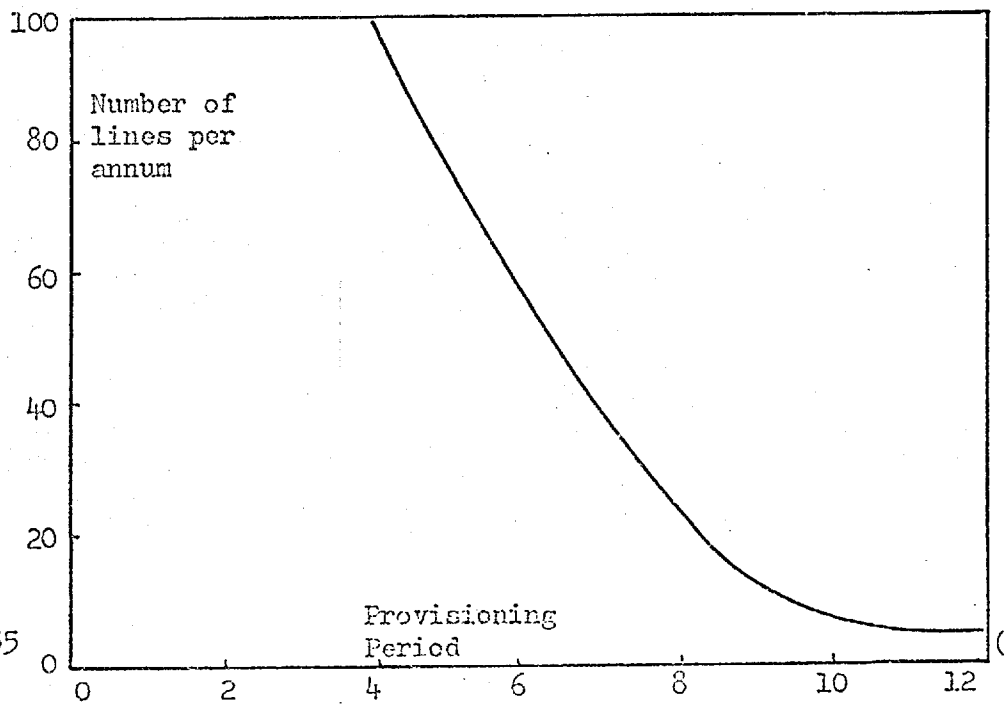

Number of lines per annum
100
80
60
40
20
0
0
2
4
6
8
10
12
Provisioning Period
(years)

Fig. 35

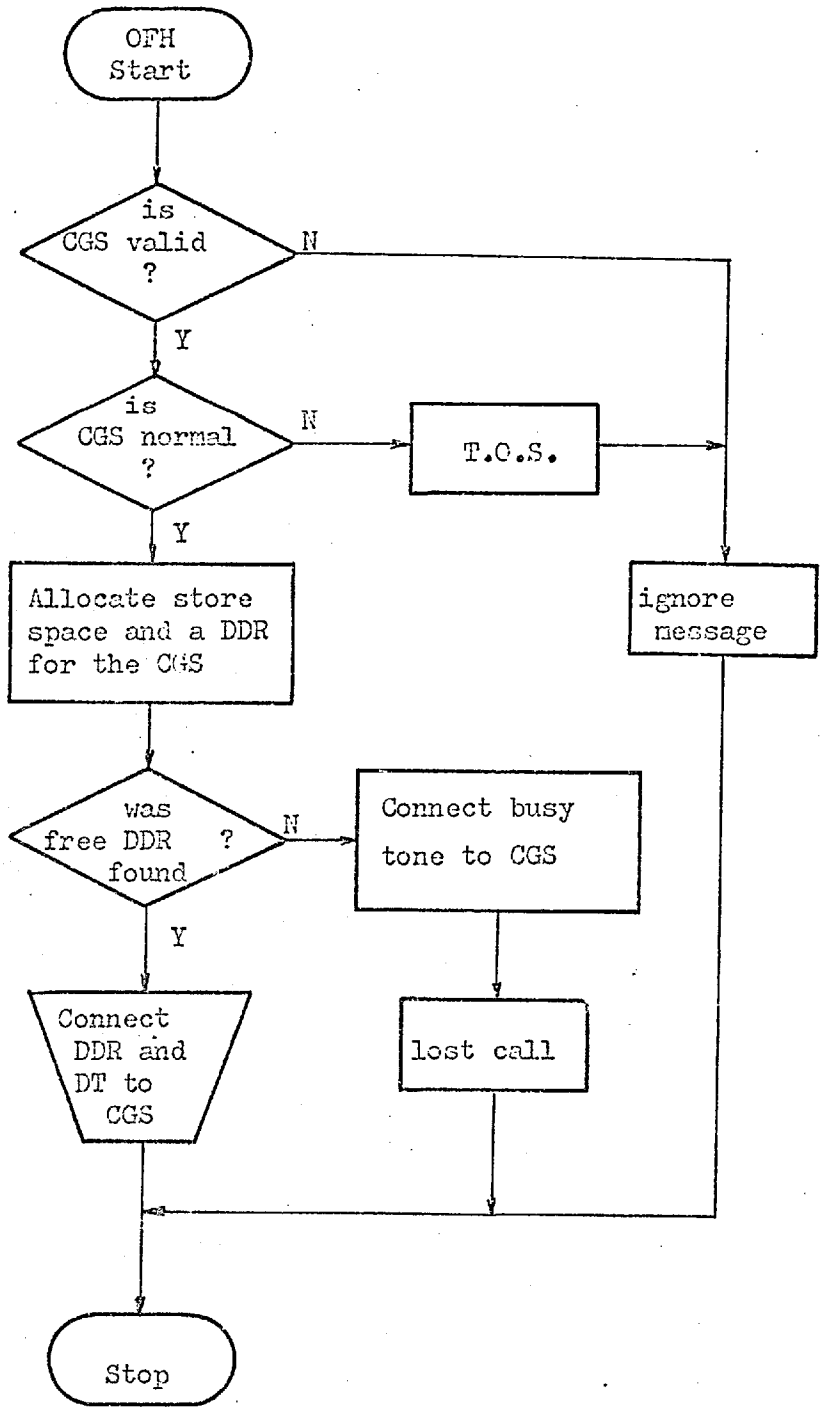
# APPENDIX V

## PROGRAM LISTINGS AND FLOW-GRAPHS OF CHAPTERS III AND IV

The following six figures, Figs. 23 to 28 are the flow-graphs of the different blocks of the controlling program. Fig. 29 shows a flow-graph of the controlling program working within the simulation program.

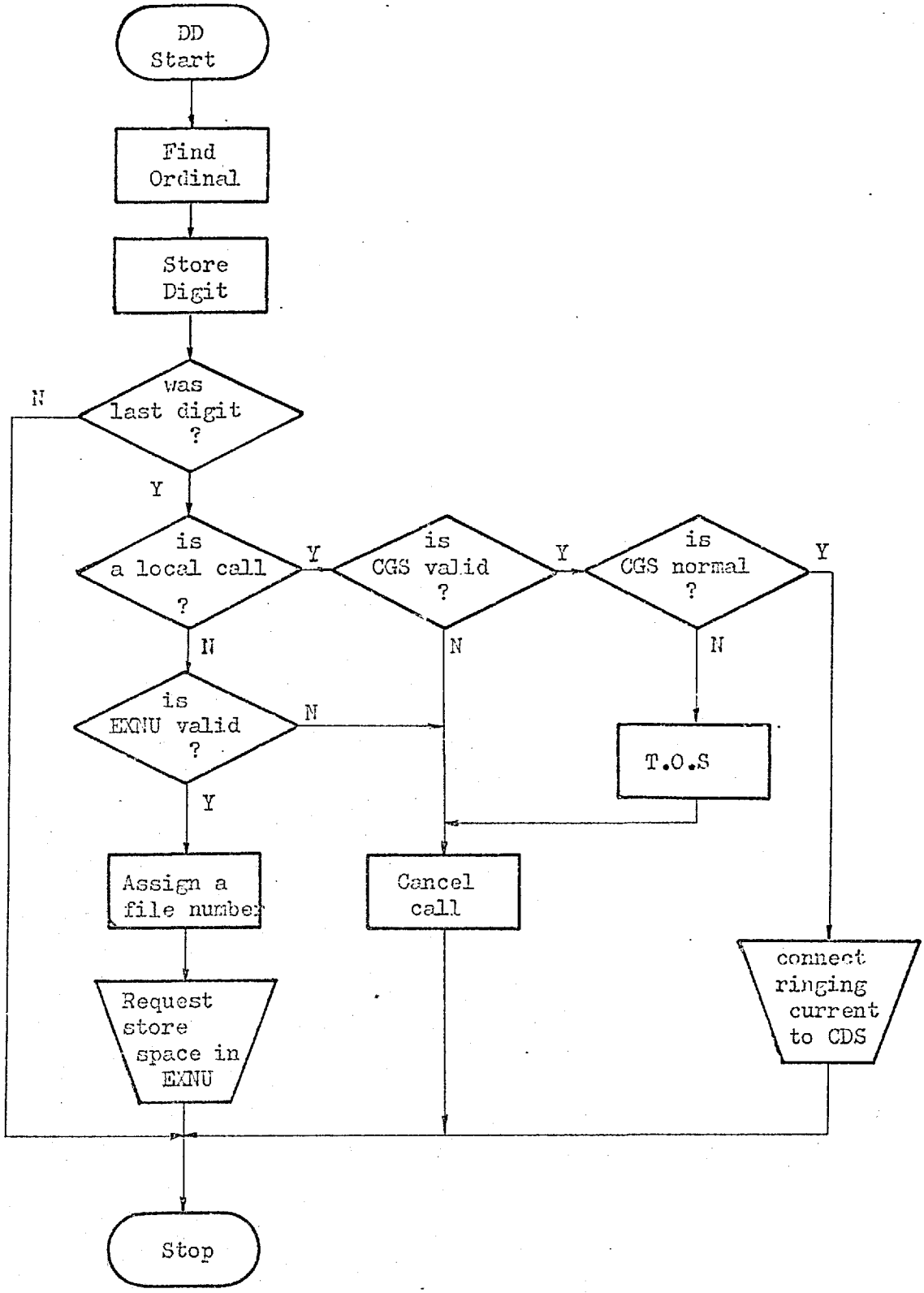
The flow-graphs are followed by a listing of the simulation and controlling program, and by a sample of a typical simulation output. First the initial state of the storage is shown; then a sample of the output and finally the state of the storage after 40 complete cycles of simulation. A listing of the programs used in this work appears on pages 188 - 199

Fig. 23.



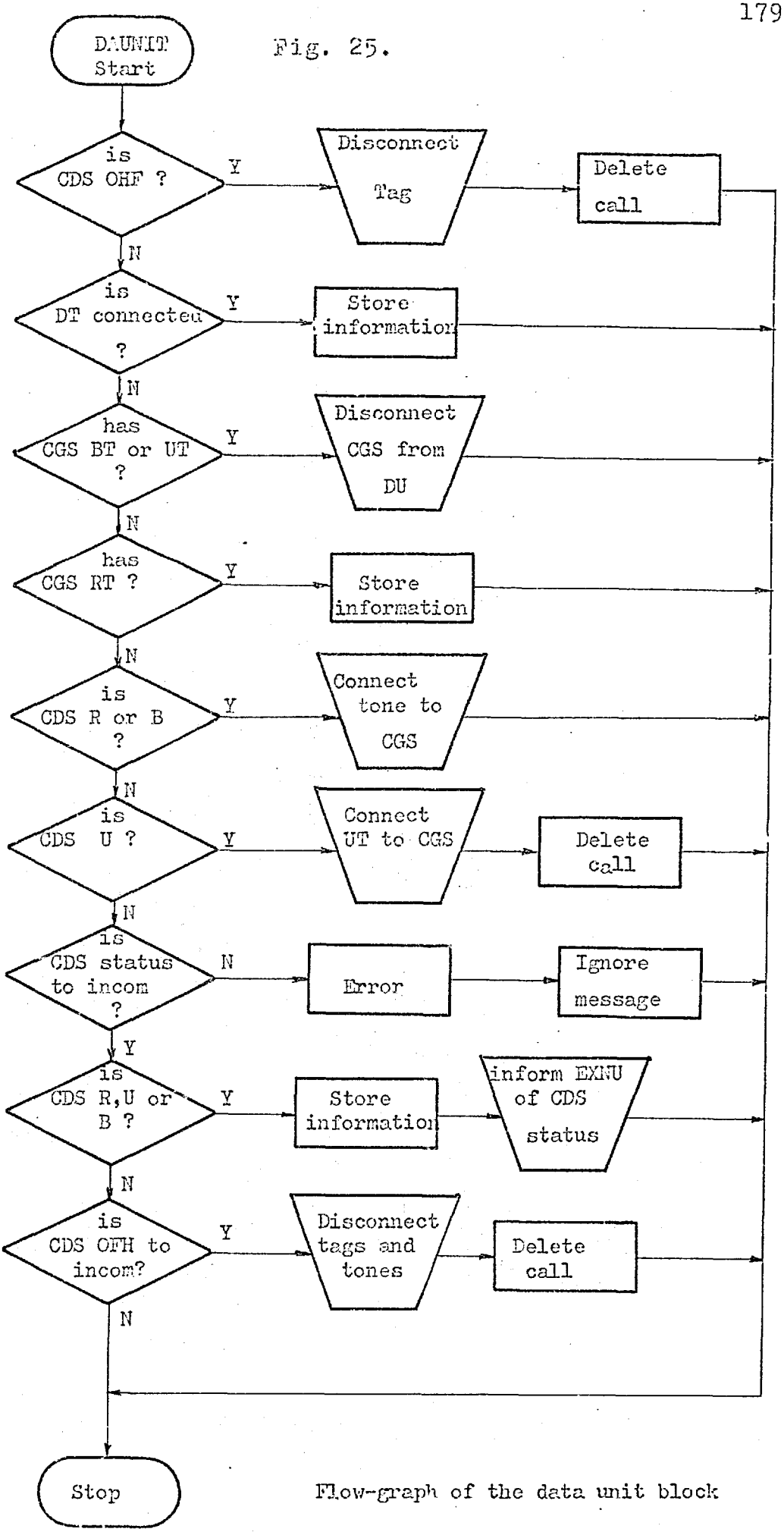

Flow-graph of the off-hook block

Fig. 24.



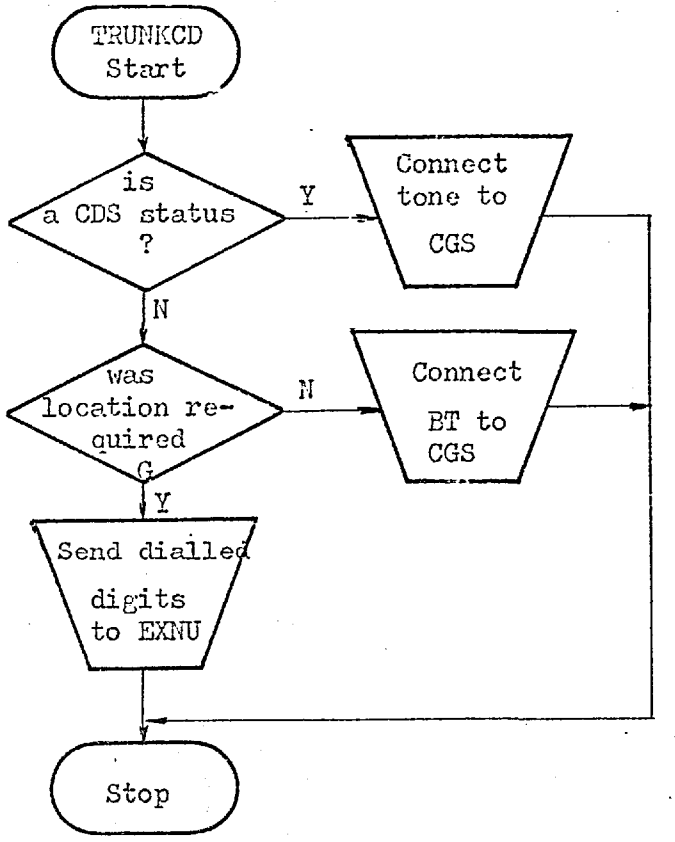

Flow-graph of the dialled digits block

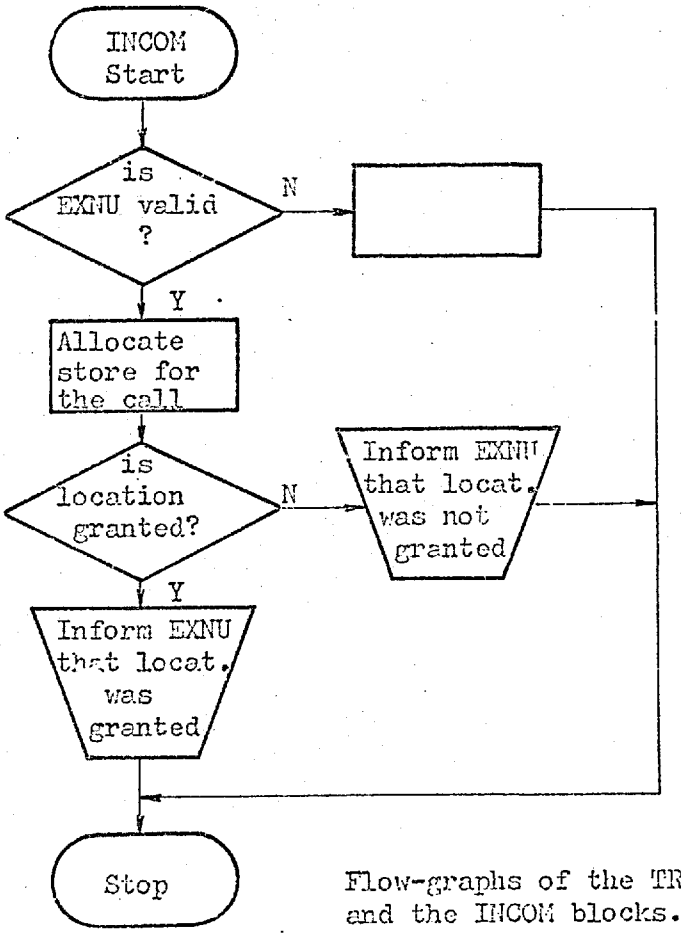
Fig. 25.




Flow-graph of the data unit block

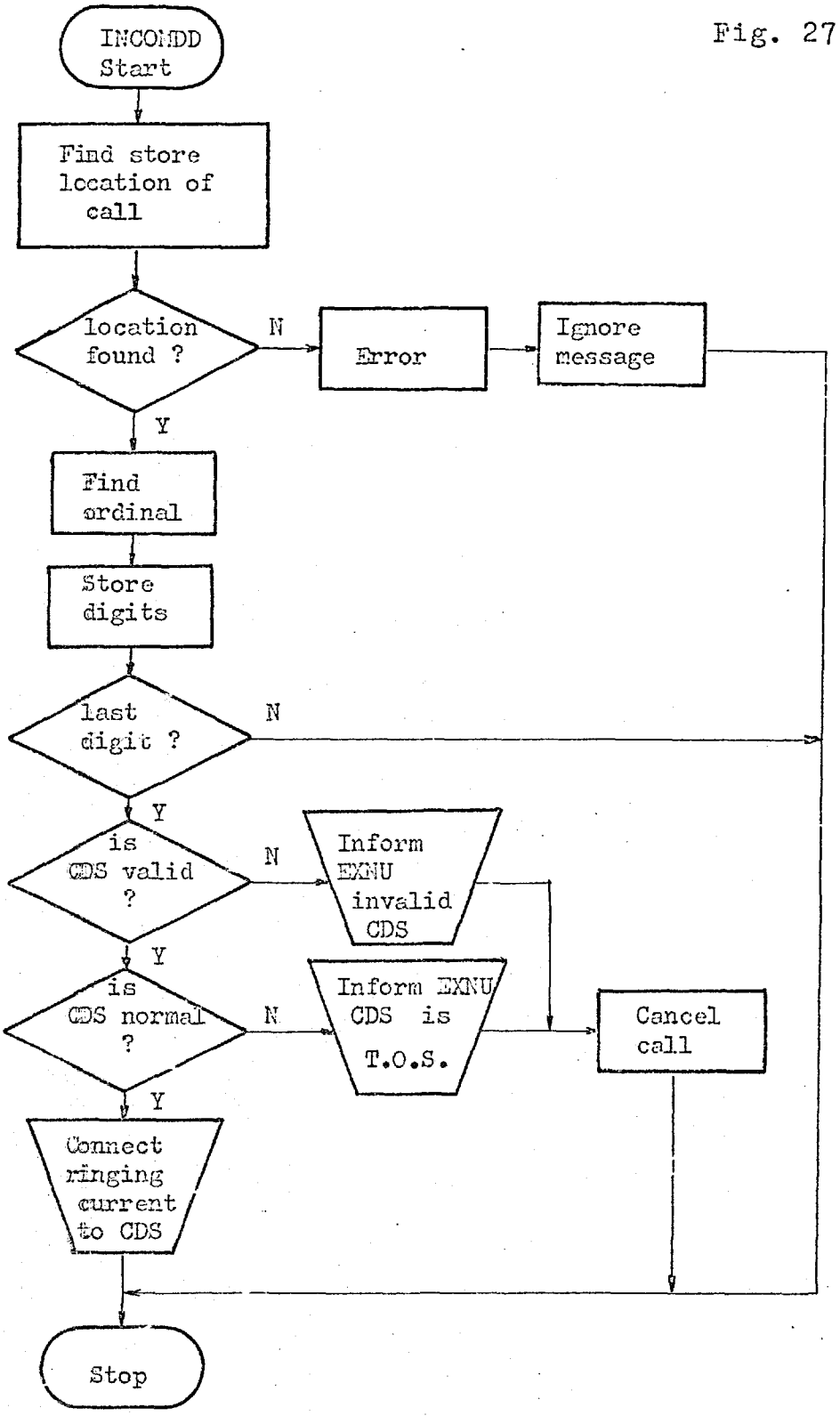
Fig. 26.



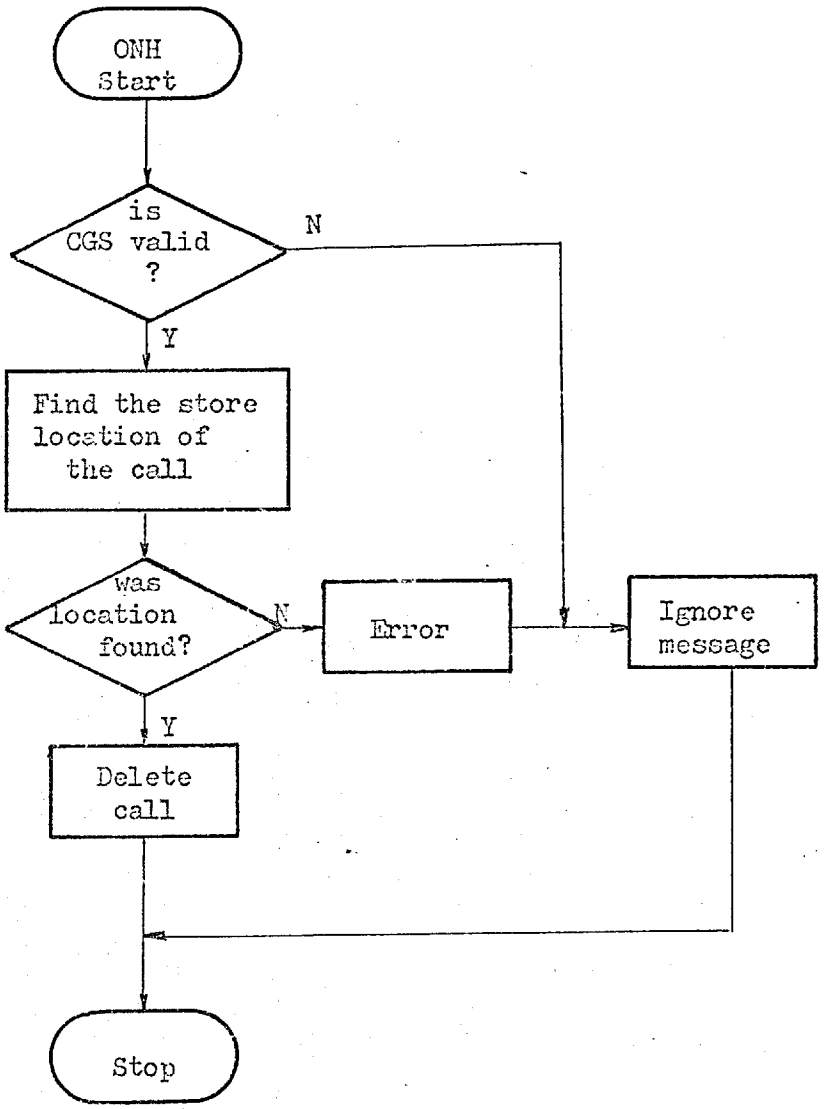

Flow-graphs of the TRUNKCD and the INCOM blocks.

Fig. 27.




Flow-graph of the incoming dialled digits block.

Fig. 28.




Flow-graph of the on-hook block.

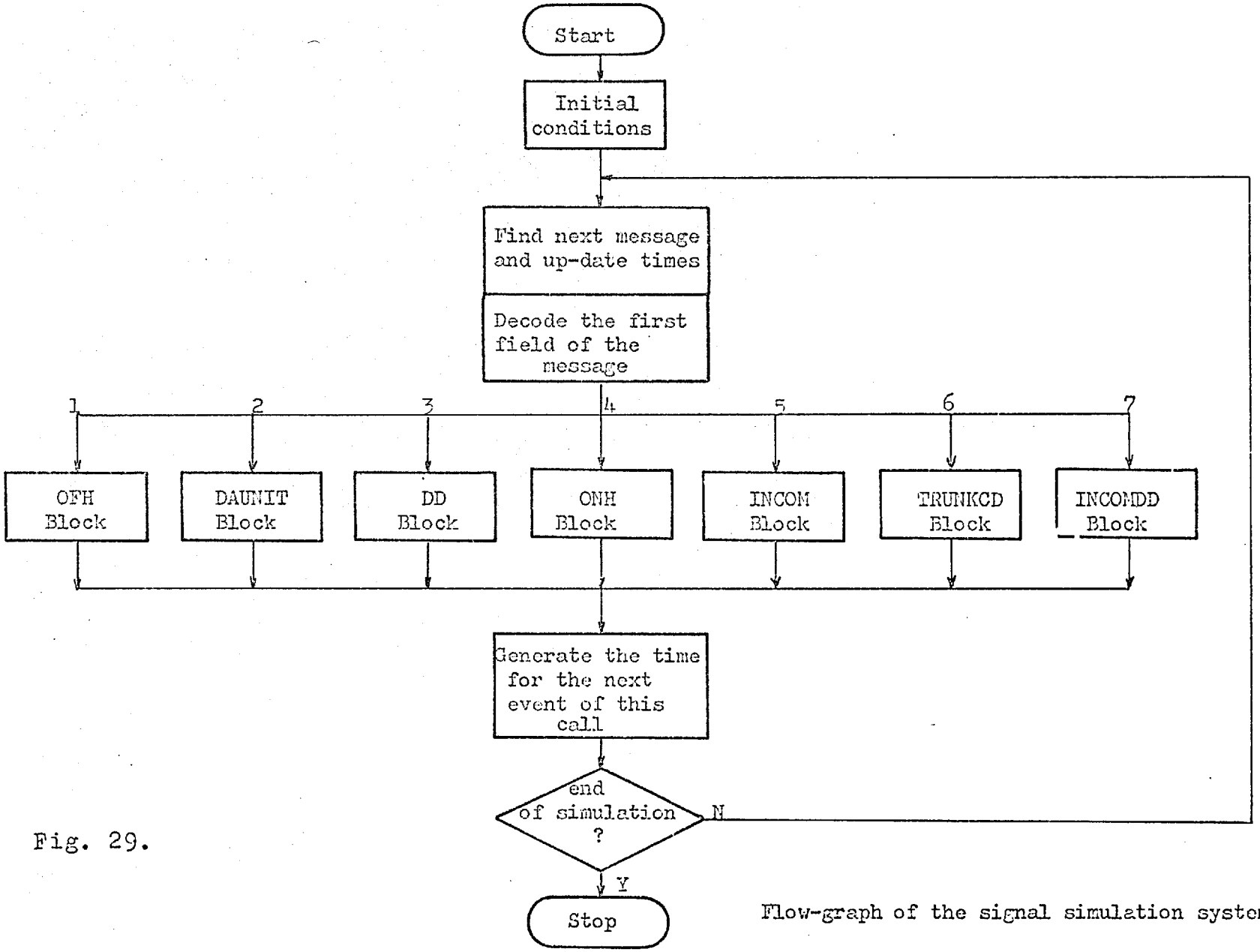



Fig. 29. Flow-graph of the signal simulation system.

| S | NEXT | I | LINK | DDR | TAG | FILE | CGS | EXNU | DD | ND | BUSY | RING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 89 | 1 | 2 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 183 | 2 | 3 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 41 | 3 | 4 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 84 | 4 | 5 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 104 | 5 | 6 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 3 | 6 | 7 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 47 | 7 | 8 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 108 | 8 | 9 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 41 | 9 | 10 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 6 | 10 | 11 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 10 | 11 | 12 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 18 | 12 | 13 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |

| Message | | | | Comments: |
|---|---|---|---|---|
| RECEIVED, | 1 | 3126 | 0 | |
| SENT , | 10 | 3126 | 15 | |
| RECEIVED, | 2 | 15 | 2 | CGS 3126 has Dial Tone |
| RECEIVED, | 3 | 15 | 6 | |
| RECEIVED, | 1 | 3956 | 0 | |
| SENT , | 10 | 3956 | 14 | |
| RECEIVED, | 3 | 15 | 6 | |
| RECEIVED, | 2 | 14 | 2 | CGS 3956 has Dial Tone |
| RECEIVED, | 1 | 3409 | 0 | |
| SENT , | 10 | 3409 | 13 | |
| RECEIVED, | 3 | 15 | 7 | |
| RECEIVED, | 3 | 15 | 7 | |
| INVALID CDS = 00 6677 | | | | |
| CALL LOCATED IN INDEX = 1 | | | | DELETED |
| RECEIVED, | 3 | 14 | 5 | |
| RECEIVED, | 3 | 14 | 5 | |
| RECEIVED, | 1 | 3476 | 0 | |
| SENT , | 10 | 3476 | 15 | |
| RECEIVED, | 2 | 15 | 2 | CGS 3476 has Dial Tone |
| RECEIVED, | 2 | 13 | 2 | CGS 3409 has Dial Tone |
| RECEIVED, | 3 | 13 | 2 | |
| RECEIVED, | 3 | 13 | 5 | |
| RECEIVED, | 3 | 15 | 8 | |
| RECEIVED, | 3 | 15 | 2 | |
| RECEIVED, | 3 | 13 | 4 | |
| RECEIVED, | 3 | 13 | 4 | |
| SENT , | 9 | 2544 | 10 | |
| RECEIVED, | 2 | 10 | 7 | |
| SENT , | 8 | 13 | 2 | |
| RECEIVED, | 3 | 14 | 6 | |
| RECEIVED, | 2 | 13 | 4 | |
| RECEIVED, | 3 | 14 | 3 | |
| SENT , | 9 | 5563 | 25 | |
| RECEIVED, | 2 | 25 | 8 | |
| SENT , | 8 | 25 | 5 | |
| SENT , | 8 | 14 | 3 | |
| RECEIVED, | 2 | 14 | 5 | |
| SENT , | 8 | 14 | 5 | |
| RECEIVED, | 3 | 15 | 2 | |
| RECEIVED, | 3 | 15 | 8 | |
| INVALID CDS = 00 8228 | | | | |
| CALL LOCATED IN INDEX = 1 | | | | DELETED |
| RECEIVED, | 1 | 4917 | 0 | |
| SENT , | 10 | 4917 | 20 | |
| RECEIVED, | 1 | 2931 | 0 | |
| SENT , | 10 | 2931 | 9 | |
| RECEIVED, | 2 | 20 | 2 | CGS 4917 has Dial Tone |
| RECEIVED, | 2 | 9 | 2 | CGS 2931 has Dial Tone |
| RECEIVED, | 3 | 9 | 8 | |
| RECEIVED. | 3 | 9 | 0 | |
| RECEIVED, | 3 | 20 | 5 | |
| RECEIVED, | 3 | 20 | 1 | |
| RECEIVED, | 3 | 9 | 8 | |

(Continued)

| Message | Comments: |
|---|---|
| RECEIVED, 1 3393 0 | |
| SENT , 10 3393 15 | |
| RECEIVED, 3 20 1 | |
| RECEIVED, 3 9 1 | |
| INVALID CDS = 00 8081 | |
| CALL LOCATED IN INDEX = 4 | DELETED |
| RECEIVED, 2 15 2 | |
| RECEIVED, 3 20 2 | |
| SENT , 9 5112 25 | |
| RECEIVED, 2 25 8 | |
| SENT , 8 25 5 | |
| SENT , 8 20 3 | |
| RECEIVED, 3 15 7 | |
| RECEIVED, 3 15 1 | |
| RECEIVED, 2 20 5 | |
| SENT , 8 20 5 | |
| RECEIVED, 3 15 6 | |
| RECEIVED, 3 15 6 | |
| INVALID CDS = 00 7166 | |
| CALL LOCATED IN INDEX = 5 | DELETED |
| RECEIVED, 1 3681 0 | |
| SENT , 10 3681 15 | |
| RECEIVED, 1 3833 0 | |
| SENT , 10 3833 14 | |
| RECEIVED, 2 14 2 | |
| RECEIVED, 2 15 2 | |
| RECEIVED, 3 15 8 | |
| RECEIVED, 3 15 6 | |
| RECEIVED, 3 15 3 | |
| RECEIVED, 1 2925 0 | |
| SENT , 10 2925 9 | |
| RECEIVED, 2 9 2 | |

| S | NEXT | I | LINK | DDR | TAG | FILE | CGS | EXNU | DD | ND | BUSY | RING |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 5 | 1 | 9 | 19 | 0 | 0 | 4136 | 00 | 0000 | 0 | 0 | 0 |
| 13 | 6 | 2 | 3 | 0 | 0 | 0 | 3956 | 00 | 5563 | 4 | 1 | 0 |
| 13 | 13 | 3 | 1 | 13 | 10 | 0 | 3409 | 00 | 2544 | 4 | 0 | 1 |
| 13 | 46 | 4 | 6 | 0 | 0 | 0 | 3833 | 00 | 5903 | 4 | 1 | 0 |
| 13 | 6 | 5 | 4 | 0 | 0 | 0 | 3701 | 00 | 3878 | 4 | 1 | 0 |
| 3 | 7 | 6 | 8 | 20 | 0 | 0 | 4937 | 00 | 8000 | 1 | 0 | 0 |
| 13 | 10 | 7 | 2 | 0 | 0 | 0 | 2198 | 00 | 2670 | 4 | 1 | 0 |
| 3 | 2 | 8 | 7 | 9 | 0 | 0 | 2337 | 00 | 6200 | 2 | 0 | 0 |
| 3 | 2 | 9 | 10 | 14 | 0 | 0 | 3238 | 00 | 2700 | 2 | 0 | 0 |
| 3 | 2 | 10 | 11 | 18 | 0 | 0 | 4645 | 00 | 0000 | 1 | 0 | 0 |
| 2 | -0 | 11 | 5 | 15 | 0 | 0 | 3886 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 63 | 12 | 13 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 346 | 13 | 14 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |
| 1 | 18 | 14 | 15 | 0 | 0 | 0 | 0000 | 00 | 0000 | 0 | 0 | 0 |

```
C
C     ********************      MAIN PROGRAM      ********************
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
   23 FORMAT(//31H OFH,S ARE EXCEEDING NEXT(ARRAY)
   73 FORMAT (/12H NEXT EVENT=,I2)
   74 FORMAT (/4H SS=,I2,6H NEXT=,I6,5H IND=,I2,4H RR=,I2)
   75 FORMAT(/46H                                    MESSAGE RECEIVED,I4,I6,I4)
C     ********************     INITIAL CONDITIONS     ********************
      CALL XCHANG
C     SET FIRST CALL TO START THE SYSTEM
      CALL FIRST
      CALL CHART
      N = 0
C
C     ********************   NEW MESSAGE (INTERRUPT)   ********************
C
   97 CALL SCAN(1,SS,IND,RR,FLNM)
      GO TO  (51,52,53,54,55,56,57,58,59,60,61,62,63),SS
   51 CONTINUE
C-----NEXT MESSAGE IS AN OFF HOOK
      IF(IND.EQ.PROX) GO TO 25
      NP = NEXT(PROX)
      NEXT(PROX) = NEXT(IND)
      NEXT(IND) = NP
      ISP = S(PROX)
      S(PROX) = S(IND)
      S(IND) = ISP
      IND = PROX
   25 CONTINUE
      A1=3.*RECDIS(X) +2.
      A2 = 10.*RECDIS(X)
      A3 = 10.*RECDIS(X)
      A4 = 10.*RECDIS(X)
      ICODE = 1
      IADD=A1*1000+A2*100+A3*10+A4
      MESS=0
      GO TO 96
   52 CONTINUE
C-----NEXT MESSAGE IS FROM A REGISTER
C        MESS= 2(CONN+DT), 3(UNABLE TO CONN.)
      ICODE =2
      IADD=RR
      MESS = 2
      GO TO 96
   53 CONTINUE
C-----NEXT MESSAGE IS A DIALLED DIGIT
      ICODE = 3
      DIGIT=10.*RECDIS(X)
      IADD=RR
      MESS=DIGIT
      GO TO 96
   54 CONTINUE
C     NEXT MESSAGE IS FROM A REGISTER ABOUT CD, IF RQ,MESS= 7(R), 8(B), 9(U)
```

```
      ICODE = 2
      IADD=REG2(IND)
      RR=REG2(IND)
      ROB=RECDIS(X) - 0.5
      IF(ROB) 954,955,956
  954 MESS=7
      GO TO 96
  955 MESS=9
      GO TO 96
  956 MESS=8
      GO TO 96
   55 CONTINUE
C        NEXT MESSAGE IS FROM A CDS OFH
      ICODE=2
      IADD=REG2(IND)
      MESS=1
      GO TO 96
   56 CONTINUE
C        NEXT MESSAGE IS FROM A TRUNKCD ANSWER,MESS= 1(R), 2(B), 3(C)
      ICODE = 6
      IADD=FLNM
      ROB = RECDIS(X) - 0.5
      IF(ROB) 854,855,856
  854 MESS=1
      GO TO 96
  855 MESS=3
      GO TO 96
  856 MESS=2
      GO TO 96
   57 CONTINUE
C        NEXT MESSAGE IS FROM A TRUNKCD ANSWER
      ICODE=6
      IADD=FLNM
C       MESS= 4(GRANTED), 5(NOT GRANTED)
      MESS=4
      GO TO 96
   58 CONTINUE
C     NEXT MESS IS FROM A REG INFORMING THAT CGS HAS RT
      MESS=4
      GO TO 591
   59 CONTINUE
C     NEXT MESS IS FROM A REG INFORMING THAT CGS HAS BT
      MESS=5
      GO TO 591
   60 CONTINUE
C     NEXT MESS IS FROM A REG INFORMING THAT A CGS HAS UT
      MESS=6
  591 ICODE=2
      IADD=RR
      GO TO 96
   61 CONTINUE
C    NEXT MESSAGE IS AN INCOMING TRUNK CALL
      IF(IND.EQ.PROX) GO TO 625
      NP=NEXT(PROX)
      NEXT(PROX)=NEXT(IND)
      NEXT(IND)=NP
      ISP=S(PROX)
      S(PROX)=S(IND)
      S(IND)=ISP
```

```
      NEXT(I)=-100.*ALOG(T)
      TOL=RECDIS(X)-0.1
      IF(TOL) 654,655,655
  654 S(I)=11
      GO TO 99
  655 S(I)=1
      GO TO 99
C     GENERATE FIX TIME FOR REGISTER ANSWER
   82 CONTINUE
      NEXT(IND) = NEXT(IND) + 1
      S(IND) = 2
      GO TO 81
C        GENERATE TIME FOR A DD
   83 CONTINUE
      T=RECDIS(X)
      NEXT(IND)=-5.*ALOG(T)
      S(IND) = 3
      GO TO 99
C        REGISTER ANSWER ON CDS STATUS
   84 CONTINUE
      NEXT(IND) = NEXT(IND) + 1
      S(IND) = 4
      GO TO 99
   85 CONTINUE
C        GENERATE TIME FOR CDS OFH
      T=RECDIS(X)
      NEXT(IND)=-200.*ALOG(T)
C     CDS DOES NOT ANSWER RINGING
      IF(NEXT(IND).GE.100) GO TO 885
      S(IND) = 5
      GO TO 99
  885 GO TO 73
   86 CONTINUE
C     INVALID OFH
      S(IND)=0
      NEXT(IND)=100000
      GO TO 81
C       GENERATE TIME FOR TRUNKCD ANSWER
   87 CONTINUE
      S(IND)=7
      NEXT(IND) = NEXT(IND) + 1
      GO TO 99
C     GEN TIME FOR A REG TO ANSWER THAT CGS HAS RT
   88 CONTINUE
      NEXT(IND) = NEXT(IND) + 1
      S(IND)=8
      GO TO 99
   89 CONTINUE
      GO TO 99
C     GEN TIME FOR A REG TO ANSWER THAT CGS HAS RT
   90 CONTINUE
      NEXT(IND)=NEXT(IND)+1
      S(IND)=9
      GO TO 99
C     GEN TIME FOR A REG TO ANSWER THAT CGS HAS UT
   91 CONTINUE
      NEXT(IND) = NEXT(IND) + 1
      S(IND)=10
      IND=PROX
  625 CONTINUE
      ICODE = 5
      IADD=10.*RECDIS(X)
      LE1 = 0.0
      LE2 = 6.*RECDIS(X)
      MESS=LE2
      FLNM=IADD
      GO TO 96
   62 CONTINUE
C     NEXT MESSAGE IS AN INCOMING DD
      ICODE = 7
      IADD=FLNM
      MESS=10.*RECDIS(X)
      GO TO 96
   63 CONTINUE
C     NEXT MESSAGE IS A CGS ONH
      ICODE=4
      A1=C1(IND)
      A2=C2(IND)
      A3=C3(IND)
      A4=C4(IND)
      IADD=1000*A1+100*A2+10*A3+A4
      MESS=0
      GO TO 96
C     ********************    PROCESSING OF MESSAGE    ******************
C
C A NEW MESSAGE HAS BEEN RECEIVED,IS DECODED AND TRANSFERRED TO THE PROPER BLOC
   96 CONTINUE
      WRITE(6,75) ICODE,IADD,MESS
      GO TO(1,2,3,4,5,6,7),ICODE
    1 CALL OFH   (A1,A2,A3,A4)
      GO TO 72
    2 CALL DAUNIT (RR,MESS)
      GO TO 72
    3 CALL DD    (RR,DIGIT)
      GO TO 72
    4 CALL ONH(A1,A2,A3,A4)
      GO TO 72
    5 CALL INCOM (FLNM,LE1,LE2)
      GO TO 72
    6 CALL TRUNKC  (FLNM,MESS)
      GO TO 72
    7 CALL INCOMD  (FLNM,MESS)
      GO TO 72
C     ********************     DECIDE NEXT EVENT    *********************
   72 CONTINUE
      GO TO (81,82,83,84,85,86,87,88,89,90,91,92,93,94),NXEVNT
C     GEN TIME AND FIND SPACE FOR OFH OR INCOMING CALL
C     S=1 (OFH)      S=11 (INCOM)
   81 CONTINUE
      DO 22  I=1,MAX
   22 IF(S(I).EQ.0) GO TO 24
      MAX = MAX + 1
      IF(I.LE.100) GO TO 24
      WRITE(6,23)
      GO TO 99
C GENERATE TIME FOR NEXT DD OR AN INCOMING TRUNK CALL
   92 CONTINUE
      NEXT(IND) = NEXT(IND) + 1
      S(IND) = 12
      GO TO 99
C     GEN TIME FOR A CGS ONH
   93 CONTINUE
      T=RECDIS(X)
      NEXT(IND)=-100.*ALOG(T)
      S(IND)=13
      GO TO 99
C     GEN TIME FOR AN FXNU ANSWER IF CDS IS ROB
   94 NEXT(IND)=NEXT(IND)+2
      S(IND)=6
      GO TO 99
C     ********************    OUTPUT AND RECYCLING    ******
   99 CONTINUE
      N = N+ 1
      IF(N-100) 97,98,98
   98 CONTINUE
      CALL CHART
      STOP
      END
```

```
      SUBROUTINE FIRST
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
C
      DO 10 I=1,100
      T=RECDIS(X)
      NEXT(I)=-100.*ALOG(T)
   10 S(I)=1
      COLA = 1
      RETURN
      END
```

```
      SUBROUTINE SCAN(II,SS,IN,RR,FLNM)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
      II = 9999
      DO 1091  J=1,100
      IF(NEXT(J).GE.II) GO TO 1091
      II = NEXT(J)
      IN=J
 1091 CONTINUE
      SS = S(IN)
      RR = REG1(IN)
      FLNM=FILE(IN)
      DO 2001  K=1,MAX
 2001 NEXT(K) = NEXT(K) - II
      RETURN
      END
```

```
      SUBROUTINE XCHANG
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
   52 FORMAT(//3H X=,F5.2,30H     THE FIRST RANDOM NUMBER IS,F8.6)
      DO 50 J = 1,100
      D1(J) =0
      D2(J) =0
      D3(J) =0
      D4(J) =0
      ND(J) =0
      C1(J) =0
      C2(J) =0
      C3(J) =0
      C4(J) =0
      E1(J) =0
      E2(J) =0
      NEXT(J) = 100000
      REG1(J)=0
      REG2(J)=0
      S(J)=0
      FILE(J) = 0
      BUSY(J) = 0
      RING(J) =0
   50 LINK(J) = J+1
C
      DO 51 I = 1,45
      FN(I)=0
      IFN(I)=0
      RN(I) =0
   51 IRN(I) =0
      MAX =0
      PROX = 1
      X = 12.78
      XX=RECDIS(X)
      WRITE(6,52) X,XX
      LENGTH = 0
      COLA =0
      RETURN
      END
```

```
      SUBROUTINE OFH(A1,A2,A3,A4)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
   76 FORMAT(/46H                                MESSAGE SENT     ,I4,I6,I4)
  399 FORMAT(26H ALL REGS ARE ENGAGED,CGS ,4I1,14H LOST THE CALL)
  821 FORMAT(//13H INVALID CGS=,4I1,16H MESSAGE IGNORED).
C     TEST IF CGS IS A VALID NUMBER
      IA = A1 + 1
      GO TO (814,814,813,813,813,813,814,814,814,814),IA
  813 CONTINUE
C         VALID NUMBER,T,O,S, TEST HAS TO BE INTRODUCED HERE
      CALL ALLOC (A1,A2,A3,A4,0,0,0,0,0,0)
      I = COLA
      CALL RENU(A1,LLENO,LL)
      IF(LLENO.EQ.1) GO TO 800
      REG1 I)=LL.
      IRN(LL) = I
      RN(LL) = I
      ICODE=10
      IADD = A1*1000+A2*100+A3*10+A4
      MESS = LL
      WRITE(6,76) ICODE,IADD,MESS
      NXEVNT =.2
      GO TO 803
C     ALL REGS ARE ENGAGED
  800 CALL DELETE(I)
      WRITE(6,399) A1,A2,A3,A4
      NXEVNT = 1
      GO TO 803
C     CGS IS NOT VALID
  814 CONTINUE
      WRITE(6,821) A1,A2,A3,A4
      NXEVNT=6
  803 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ALLOC (A1,A2,A3,A4,B1,B2,B3,B4,LE1,LE2)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,F1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
 2020 FORMAT(//25H MEMORY CAPACITY EXCEEDED)
      J = PROX
      K = COLA
      PROX = LINK(J)
      IF(PROX.GE.100) GO TO 2001
      MAX=MAX+1
      LINK(J) = LINK(K)
      LINK(K) = J
      COLA = J
      LENGTH = LENGTH + 1
      C1(COLA) = A1
      C2(COLA) = A2
      C3(COLA) = A3
      C4(COLA) = A4
      D1(COLA) = B1
      D2(COLA) = B2
      D3(COLA) = B3
      D4(COLA) = B4
      E1(COLA) = LE1
      E2(COLA) = LE2
      GO TO 2999
 2001 CONTINUE
      WRITE(6,2020)
      CALL ERROR
 2999 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE RENU(JAB,LLENO,LL)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
  350 FORMAT(//16H FIRST DIGIT IS ,I1,24H, RENU CALLED ERRONEOUSLY)
C     JAB= FIRST DIGIT OF CGS,CDS.    LL= REG FOUND FREE
      KK =0
      LLENO =0
      GO TO (381,382,383,384,385,386,387,388,389),JAB
C   FIND A FREE REG
  381 CONTINUE
      GO TO 390
  382 CONTINUE
      IF(IRN(8).EQ.0)KK=1
      IF(IRN(9).EQ.0)KK=2
      IF(IRN(10).EQ.0)KK=3
      IF (KK.EQ.0) GO TO 360
      LL = 7 + KK
      GO TO 311
  383 CONTINUE
      IF(IRN(13).EQ.0)KK=1
      IF(IRN(14).EQ.0)KK=2
      IF(IRN(15).EQ.0)KK=3
      IF (KK.EQ.0) GO TO 360
      LL = 12 + KK
      GO TO 311
  384 CONTINUE
      IF(IRN(18).EQ.0)KK=1
      IF(IRN(19).EQ.0)KK=2
      IF(IRN(20).EQ.0)KK=3
      IF (KK.EQ.0) GO TO 360
      LL = 17 + KK
      GO TO 311
  385 CONTINUE
      IF(IRN(23).EQ.0)KK=1
      IF(IRN(24).EQ.0)KK=2
      IF(IRN(25).EQ.0)KK=3
      IF (KK.EQ.0) GO TO 360
      LL = 22 + KK
      GO TO 311
  386 CONTINUE
  387 CONTINUE
  388 CONTINUE
  389 CONTINUE
  390 CONTINUE
      WRITE(6,350) JAB
      GO TO 311
  360 LLENO = 1
  311 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FILENU(IE2,LLENA,LX)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
      KK=0
      LLENA = 0
      GO TO (1,2,3,4,5) ,IE2
    1 CONTINUE
      IF(IFN(11).EQ.0) KK=1
      IF(IFN(12).EQ.0) KK=2
      IF(IFN(13).EQ.0) KK=3
      IF(KK.EQ.0) GO TO 160
      LX=10+KK
      GO TO 111
    2 CONTINUE
      IF(IFN(21).EQ.0) KK=1
      IF(IFN(22).EQ.0) KK=2
      IF(IFN(23).EQ.0) KK=3
      IF(KK.EQ.0) GO TO 160
      LX=20+KK
      GO TO 111
    3 CONTINUE
      IF(IFN(31).EQ.0) KK=1
      IF(IFN(32).EQ.0) KK=2
      IF(IFN(33).EQ.0) KK=3
      IF(KK.EQ.0) GO TO 160
      LX=30+KK
      GO TO 111
    4 CONTINUE
      IF(IFN(41).EQ.0) KK=1
      IF(IFN(42).EQ.0) KK=2
      IF(IFN(43).EQ.0) KK=3
      IF(KK.EQ.0) GO TO 160
      LX=40+KK
      GO TO 111
    5 CONTINUE
      IF(IFN( 3).EQ.0) KK=3
      IF(IFN( 1).EQ.0) KK=1
      IF(IFN( 2).EQ.0) KK=2
      IF(KK.EQ.0) GO TO 160
      LX=KK
      GO TO 111
  160 LLENA=1
  111 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DD(IDDR ,DIGIT)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
   76 FORMAT(/46H                                    MESSAGE SENT     ,I4,I6,I4)
C     SET ORDINAL OF DIGIT
      I= RN(IDDR)
      ND(I) = ND(I) + 1
      NDD = ND(I)
      GO TO (901,902,903,904,905,906),NDD
  901 D1(I) = DIGIT
      GO TO 990
  902 D2(I) = DIGIT
      GO TO 990
  903 D3(I) = DIGIT
      GO TO 990
  990 CONTINUE
      NXEVNT=3
      GO TO 999
C     FOURTH DIGIT
  904 D4(I) = DIGIT
C    TRUNK CALL TEST
      IF(D1(I).EQ.0) GO TO 920
C     TEST IF VALID NUMBER
      JB = D1(I) + 1
      GO TO (911,911,913,913,913,913,914,914,914,914),JB
C     OPERATOR REQUESTED
  911 CALL OPERAT(I)
      NXEVNT = 9
      GO TO 999
  913 CONTINUE
C         TOS TEST HAS TO BE INTRODUCED HERE,CONNECT RINGING CURRENT TO CDS
      ID1=D1(I)
      CALL RENU(ID1,LLENO,LL)
      IF(LLENO.EQ.1) GO TO 980
      REG2(I)=LL
      IRN(LL)=I
      RN(LL)=I
      ICODE=9
      IADD=1000*D1(I)+100*D2(I)+10*D3(I)+D4(I)
      MESS=REG2(I)
      WRITE(6,76) ICODE,IADD,MESS
      NXEVNT=4
      GO TO 999
C     INVALID CDS
  914 CALL INVALI (I)
      NXEVNT = 9
      GO TO 999
C     TRUNK CALL
  920 CONTINUE
      E1(I) = D1(I)
      E2(I) = D2(I)
      D1(I) = D3(I)
      D2(I) = D4(I)
      D3(I) =0
      D4(I) = 0
      GO TO 990
C     FIFTH DIGIT
  905 CONTINUE
      D3(I) = DIGIT
      GO TO 990
C     SIXTH DIGIT
  906 CONTINUE
      D4(I) = DIGIT
C     OUTGOING TRUNK CALL DIALLING COMPLETED,SEND MESS,TO EXNU
      IE2=E2(I)
      IE=IE2+1
      GO TO (9,1,1,1,1,1,9,9,9,9),IE
    9 CONTINUE
      GO TO 914
    1 CALL FILENU(IE2,LLENA,LX)
      IF(LLENA.EQ.1) GO TO 970
      FILE(I) = LX
      IFN(LX)=I
      FN(LX)=I
      ICODE=5
      IADD=FILE(I)
      MESS=1
      WRITE(6,76) ICODE,IADD,MESS
      NXEVNT = 7
      GO TO 999
  970 CONTINUE
  980 CONTINUE
      ICODE=8
      IADD=REG1(I)
      MESS=3
      WRITE(6,76) ICODE,IADD,MESS
      NXEVNT=10
      GO TO 999
  999 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INCOMD (FLNM,MESS)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
   76 FORMAT(/46H                                MESSAGE SENT     ,I4,I6,I4)
  625 FORMAT (/20H INCOMING CALL,EXNU=,I1,5H CDS=,4I1,12H INVALID CDS)
  626 FORMAT (/29H INCOMDD MESSAGE,FILE NUMBER=,I2,10H NOT FOUND)
C     SCAN FILE
      DO 621  K=1,MAX
      IF(FILE(K).EQ.FLNM) GO TO 622
  621 CONTINUE
      WRITE(6,626) FLNM
      CALL ERROR
C     FIND ORDINAL OF DIGIT
  622 CONTINUE
      ND(K) = ND(K) + 1
      NN = ND(K)
      GO TO (601,602,603,604),NN
  601 D1(K)=MESS
      GO TO 690
  602 D2(K)=MESS
      GO TO 690
  603 D3(K)=MESS
      GO TO 690
  690 CONTINUE
      NXEVNT = 12
      GO TO 699
C     FOURTH DIGIT,DIALLING HAS BEEN COMPLETED,TEST IF VALID NUMBER
  604 D4(K)=MESS
      JB = D1(K) + 1
      GO TO(611,611,613,613,613,613,614,614,614,614),JB
C     OPERATOR REQUESTED
  611 CONTINUE
      CALL OPERAT(K)
      NXEVNT = 9
      GO TO 699
C     VALID CDS,TOS TEST TO BE INTRODUCED HERE,CONN RC TO CDS
  613 CONTINUE
      ID1=D1(K)
      CALL RENU(ID1,LLENO,LL)
      IF(LLENO.EQ.1) GO TO 680
      REG2(K)=LL
      IRN(LL)=1
      RN(LL)=K
      ICODE=9
      MESS=REG2(K)
      IADD=1000*D1(K)+100*D2(K)+10*D3(K)+D4(K)
      WRITE(6,76) ICODE,IADD,MESS
      NXEVNT=4
      GO TO 699
C     NO REG FOR CDS AVAILABLE,CGS GETS BT
  680 CONTINUE
      ICODE =6
      IADD=FLNM
      MESS=2

      CALL DELETE(K)
      NXEVNT = 9
      GO TO 699
C     INVALID CDS
  614 CONTINUE
      WRITE(6,625) E1(K),E2(K),D1(K),D2(K),D3(K),D4(K)
      CALL INVALI (K)
      NXEVNT = 9
      GO TO 699
  699 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DELETE(IN)
      INTEGER COLA,REGNO,RR,SS,DIGIT,PROX,A1,A2,A3,A4,N1,R2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NALVNT,FILE,REG1,REG2,FN
1070  FORMAT(//23H CALL LOCATED IN INDEX=,I3,17H HAS BEEN DELETED)
1084  FORMAT(//24H DURING DELETE LOCATION=,I3,11H ,NOT FOUND)
C     FIND INDEX OF PREVIOUS LOCATION
      K = COLA
      DO 1080  I=1,LENGTH
      LAST = K
      K = LINK(K)
1080  IF(K.EQ.IN) GO TO 1081
C     LOCATION NOT FOUND
      WRITE(6,1084) IN
      CALL ERROR
      GO TO 1090
C     LOCATION FOUND
1081  CONTINUE
C     MODIFY LINK OF PREVIOUS LOCATION
      LINK(LAST) = LINK(IN)
C     MODIFY LINK OF DELETED LOCATION
      LINK(IN) = PROX
C     SET NEW PROX,LENGTH AND COLA
      PROX = IN
      LENGTH = LENGTH - 1
      COLA = LAST
C     CLEAR LOCATION AND REG
1099  CONTINUE
      N1=REG1(IN)
      IF(N1.EQ.0) GO TO 1097
      RN(N1)=0
      IRN(N1)=0
1097  N2=REG2(IN)
      IF(N2.EQ.0) GO TO 1098
      RN(N2)=0
      IRN(N2)=0
1098  J1=FILE(IN)
      IF(J1.EQ.0) GO TO 1096
      FN(J1)=0
      IFN(J1)=0
1096  CONTINUE
      NEXT(IN) = 100000
      S(IN) = 0
      C1(IN) =0
      C2(IN) =0
      C3(IN) =0
      C4(IN) =0
      D1(IN) =0
      D2(IN) =0
      D3(IN) =0
      D4(IN) =0
      E1(IN) =0
      E2(IN) =0
      ND(IN) =0
      BUSY(IN) =0

      RING(IN) =0
      REG1(IN)=0
      REG2(IN)=0
      FILE(IN) =0
      WRITE(6,1070)IN
1090  CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DAUNIT(ITAG,MESS)
      INTEGER COLA,REGNU,RP,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
C     MESS= 1(CGS OFH), 2(C+DT), 3(NOT CONN), 4(CGS RT), 5(CGS BT), 6(CGS UT),
C     MESS= M(CDS R), 8(CDS B), 9(CDS U)
   76 FORMAT(/46H                                    MESSAGE SENT     ,14,16,14)
  801 FORMAT (/5H CGS=,411,33H IS OFF HOOKED AND WITH DIAL TONE)
 1010 FORMAT(4HREG ,I4,18H UNABLE TO CONNECT)
      I= RN(ITAG)
      IF(REG1(I).EQ.0) GO TO 210
      ICODE =8
      GO TO(201,202,203,204,205,206,207,208,209),MESS
  201 CONTINUE
C         CDS IS OFH
      MESS=5
      IADD=REG1(I)
      WRITE(6,76) ICODE,IADD,MESS
      IADD=REG2(I)
      WRITE(6,76) ICODE,IADD,MESS
      CALL DELETE(I)
      NXEVNT=1
      GO TO 300
  202 CONTINUE
C     THE REG AND THE DIAL TONE HAVE BEEN CONNECTED
      WRITE(6,801)C1(I),C2(I),C3(I),C4(I)
      NXEVNT = 3
      GO TO 300
  203 CONTINUE
C     REG UNABLE TO CONNECT
      WRITE(6,1010) REGNU
      CALL OPERAT(I)
      GO TO 299
  204 CONTINUE
C     CGS HAS RINGING TONE
      RING(I) = 1
      NXEVNT=5
      GO TO 300
  205 CONTINUE
C     CGS HAS BUSY TONE
      BUSY(I) = 1
  206 CONTINUE
C     CGS HAS UNOBTAINABLE TONE
      IADD=REG1(I)
      MESS=5
      WRITE(6,76) ICODE,IADD,MESS
      REG1(I)=0
      RN(IADD)=0
      IRN(IADD)=0
      NXEVNT=13
      GO TO 300
  207 CONTINUE
C     CDS IS RINGING,CONN RINGING TONE TO CGS
      MESS=2
      NXEVNT=8
      GO TO 301
  208 CONTINUE
C     CDS IS BUSY,CONN BT TO CGS
      IADD=REG2(I)
      MESS=5
      WRITE(6,76) ICODE,IADD,MESS
      MESS=3
      NXEVNT=10
      GO TO 29
  209 CONTINUE
C     CDS IS UNOBTAINABLE,CANCEL CALL,CONN UT TO CGS
      IADD=REG2(I)
      MESS=5
      WRITE(6,76) ICODE,IADD,MESS
      MESS=4
      NXEVNT=11
   29 REG2(I)=0
      RN(IADD)=0
      IRN(IADD)=0
      GO TO 301
  210 CONTINUE
C     ANSWER STATUS OF CDS TO AN INCOMING TRUNK CALL
      IF(MESS.EQ.1) GO TO 214
      MESS=MESS-6
      ICODE =6
      IADD=FILE(I)
      GO TO (211,212,213),MESS
  211 CONTINUE
C     CDS RINGING TO AN INCOMING TRUNK CALL,CONN RT TO EXNU
      RING(I)=1
      MESS=1
      NXEVNT=5
      GO TO 302
  212 CONTINUE
C     BT TO EXNU
      BUSY(I)=1
      MESS=2
      GO TO 302
  213 CONTINUE
C     UT TO EXNU
      MESS=3
      GO TO 302
  302 WRITE(6,76) ICODE,IADD,MESS
  214 CONTINUE
C     CDS IS OFH TO AN INCOMING CALL
      ICODE=8
      IADD=REG2(I)
      MESS=5
      WRITE(6,76) ICODE,IADD,MESS
      CALL DELETE(I)
      NXEVNT=1
      GO TO 300
  299 NXEVNT=9
      GO TO 300
  301 IADD=REG1(I)
      WRITE(6,76) ICODE,IADD,MESS
      GO TO 300
  300 CONTINUE
      RETURN
```

```
      SUBROUTINE TRUNKC  (FLNM,MESS)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
   76 FORMAT(/46H                                MESSAGE SENT     ,I4,I6,I4)
C     MESS=1 RINGING,MESS=2 BUSY,MESS=3 UNOBTAINABLE,MESS=4 GRANTED,MESS=5 NOT G
      I= FN(FLNM)
      GO TO(501,502,503,504,505),MESS
C     CDS IS RINGING
  501 CONTINUE
      MESS=2
      NXEVNT=8
      GO TO 590
C     CDS IS BUSY
  502 CONTINUE
      MESS=3
      NXEVNT=10
      GO TO 590
C     CDS UNOBTAINABLE
  503   CONTINUE
      MESS=4
      NXEVNT=11
      GO TO 590
C     LOCATION IN EXNU GRANTED,SEND DIGITS TO EXNU
  504 CONTINUE
      ICODE=12
      IADD=FLNM
      WRITE(6,76) ICODE,IADD,D1(I)
      WRITE(6,76) ICODE,IADD,D2(I)
      WRITE(6,76) ICODE,IADD,D3(I)
      WRITE(6,76) ICODE,IADD,D4(I)
      NXEVNT=14
      GO TO 599
C     LOCATION IN EXNU NOT GRANTED
  505 CONTINUE
      MESS=3
      NXEVNT=10
  590 CONTINUE
      ICODE = 8
      IADD=REG1(I)
      WRITE(6,76) ICODE,IADD,MESS
  599 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INCOM(FLNM,LE1,LE2)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,RE
   76 FORMAT(/46H                                MESSAGE SENT     ,I4,I
C     TEST IF VALID EXNU
      IE = LE2 + 1
      GO TO (101,101,103,103,103,103,101,101,101,101),IE
C     IGNORE MESSAGE
  101 CONTINUE
      NXEVNT=6
      GO TO 199
C     EXNU IS VALID
  103 CONTINUE
C     ALLOCATE STORE FOR EXNU
      CALL ALLOC(0,0,0,0,0,0,0,0,LE1,LE2)
      FILE(COLA) = FLNM
      ICODE=6
      IADD=FLNM
      MESS=4
      WRITE(6,76) ICODE,IADD,MESS
      NXEVNT=12
      GO TO 199
  199 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ONH(A1,A2,A3,A4)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FI
  840 FORMAT(//17H INVALID ONH,CGS=,4I1)
  841 FORMAT(//15H ONH,VALID CGS=,4I1)
  842 FORMAT(//5H CGS=,4I1,18H FOR ONH NOT FOUND)
  843 FORMAT(//22H CGS FOUND,CALL DELETE)
C     TEST IF CGS IS A VALID NUMBER
      IA = A1 +1
      GO TO(814,814,813,813,813,813,814,814,814,814),IA
C     INVALID CGS IGNORE MESSAGE
  814 CONTINUE
      WRITE (6,840) A1,A2,A3,A4
      GO TO 899
C     VALID CGS ONH
  813 CONTINUE
      WRITE(6,841) A1,A2,A3,A4
      DO 890  I=1,MAX
      IF(A1.EQ.C1(I)) GO TO 801
  880 CONTINUE
  890 CONTINUE
      WRITE(6,842) A1,A2,A3,A4
      CALL ERROR
  801 CONTINUE
      IF(A2.EQ.C2(I)) GO TO 802
      GO TO 880
  802 CONTINUE
      IF(A3.EQ.C3(I)) GO TO 803
      GO TO 880
  803 CONTINUE
      IF(A4.EQ.C4(I)) GO TO 804
      GO TO 880
  804 CONTINUE
      WRITE(6,843)
      CALL DELETE(I)
      NXEVNT=1
  899 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INVALI (I)
      INTEGER COLA,REGNU,RR,SS,DIGIT,PROX,A1,A2,A3,A4,B1,B2,B3,B4
      INTEGER S(100),NEXT(100),LINK(100),FILE(100),RN(45),FLNM
      INTEGER ND(100),D1(100),D2(100),D3(100),D4(100),IRN(45),BUSY(100)
      INTEGER C1(100),C2(100),C3(100),C4(100),E1(100),E2(100),RING(100)
      INTEGER REG1(100),REG2(100),FN(45),IFN(45)
      COMMON LENGTH,COLA,PROX,MAX,NEXT,LINK,S,RN,IRN,ND,RING,X,IFN
      COMMON D1,D2,D3,D4,E1,E2,C1,C2,C3,C4,BUSY,NXEVNT,FILE,REG1,REG2,FN
  660 FORMAT(5H CGS=,4I1,21H INVALID CALL     CDS=,2I1,3I11)
      WRITE(6,669)C1(I),C2(I),C3(I),C4(I),E1(I),E2(I),D1(I),D2(I),D3(I),
     5D4(I)
      CALL DELETE(I)
      RETURN
      END
```

# SELECTED REFERENCES


A1. Arrow, K. J. Social Choice and Individual Value. Cowles Foundation Monograph 12. John Wiley, New York, 1963.

A2. Adelaar, H. H. "Influence of Stored-Program Control On the Development of Communication Networks", Electrical Communication. Vol. 45, No. 3, 1970, pp. 184.

B1. Ralogh, Thomas. The Economics of Poverty. Weidenfeld and Nicolson, London, 1966. Section Two.

B2. Bergson, Abram. "On the Concept of Social Welfare", Quarterly Jounal of Economics. Vol. 74, 1960, pp. 507-535.

B3. Bhagwati, Jagdish. The Economics of Underdeveloped Countries. World University Library, London, 1966, Part 3, Section 17.

B4. Black, Duncan. The Theory of Committee and Elections. Cambridge University Press, Cambridge, 1958.

B5. Barton, M. E. , N. M. Haller and G. W. Ricker. "Service Program", BSTJ. Vol. 48, 1969, pp. 2865-2891.

B6. Broux, J. A. "Initial Experience with the 10C Switching System", Electrical Communication. Vol. 43, No. 4, 1968, pp. 323-328.

C1. CCITT. "Factors Affecting the Supply of and Demand for Telecommunication Facilities", Economic Studies at the National Level in the Field of Telecommunications: 1964-1968. I.T.U., Geneva, July, 1968.

C2. Cherry, Colin. World Communication-Threat or Promise. John Wiley, UK, to be published, July 71 Chapter 5.

C3. Churchman, C. W., R. L. Ackoff, and E. L. Arnoff. Introduction to Operations Research. John Wiley, London, 1957.

C4. Constantine, H. "Towards a Telecommunications Oriented Programming Language", D.I.C. dissertation. Imperial College of Science and Technology, London, 1967.

C5. CCITT. "Economic Factors chich determine the Development of Telephone Traffic", Economic Studies at the National Level in the field of Telecommunications: 1964-1968. I.T.U., Geneva, July, 1968.

C6. Cherry, Colin. "Report of a Preliminary Statistical Survey of Telephone Penetration in Various Countries of the World", Imperial College of Science and Technology, Report. 1963.

E1. Ekbergh, E. "Pentaconta 52 Telephone Switching System for Rural Networks", Electrical Communication. Vol. 42, No. 1, 1967, pp. 56-61.

F1. Feeney, G. J. "A Basis for Strategic Decisions on Inventory Control Operations", Management Science 2. No. 1, 1955.

F2. Franklin, R. H. "Transmission Systems for Trunk and Junction Circuits", Proceedings of IEE. April, 1964, pp. 700-712.

F3. Feller, W. An Introduction to Probability Theory and its Applications. Vol. 1, pp. 423. John Wiley, London, 1962.

G1. Gilmore, P. C. and R. E. Gomory. "The Theory and Computation of Knapsack Functions", Operations Research, Vol. 14, No. 6, November-December, 1966, pp. 1045-1074.

G2. Goodman, L. A. and H. Markowitz. "Social Welfare Functions Based on Individual Rankings", American Journal of Sociology. Vol. 58, 1952, pp. 257-262.

H1. Hanssman, Fred. "Operations Research in National Planning of Underdeveloped Countries". Operations Research. No. 9, 1961, pp. 230-248.

H2. Hettwig, E. Direct Distance Dialling. Siemens and Halske A.G., Munich, 1966.

I1. Ives, W. J. "The SP-1 Electronic Switching System-Hardware", IEE Conference on Switching Techniques for Telecommunications Networks. London, April, 1969.

J1. Jones, N. J. H. "People and Telephones", D.I.C. dissertation. Imperial College of Science and Technology, London, 1963.

K1. Knight, A. V. and C. J. Macpherson. "A 10-Line Semi-Electronic Exchange for Remote Areas". POEEJ. Vol. 62, Part 4, January, 1970, pp. 232-238.

K2. Katzeff, K. and U. Jerndal. "General Presentation of L. M. Ericsson's Stored-Program Controlled System AKE", IEE Conference on Switching Techniques for Telecommunications Networks. London, April, 1969.

K3. Ketchledge, R. W. "Administration of Generic Programs", IEE Conference on Switching Techniques for Telecommunications Networks. London, April, 1961.

K4. Keister, W. , R. W. Ketchledge and H. E. Vaughan. "No. 1 ESS: System Organization and Objectives", BSTJ. Vol. 43, September, 1964, pp. 1831-1844.

L1. Lavi, Y. "Planning Israel's Telecommunication System: a Review of Planning Activities for a 15-years Plan (1966-1980)", Telecommunications Journal. Vol. 35, August, 1968, pp. 262-269.

L2. Lonnstrom, S., F. Marklund and I. Moo. A Telephone Development Project. L. M. Ericsson, Stockholm, 1965.

L3. Luce, R. Duncan. Individual Choice Behavior. John Wiley, New York, 1959, p. 7.

M1. Maddison, Angus. Economic Progress and Policy in Developing Countries. George Allen and Unwin, London, 1970.

M2. Merriman, J. H. H. "Men, Circuits and Systems in Telecommunications", POEEJ. Vol. 60, Part 4 January, 1968, pp. 241-251.

O1. Ortuad, A. "Rural Automatic Exchange with Electronic Control Circuits". IEE Journal. Vol. 109, Part B, No. 43, January, 1962.

O2. ----------. "Principles of an Automatic Rural Exchange for 1500 Subscribers with Electronic Control Circuits", Second International Conference on Electronic Switching. Paris, 1966.

P1. Papoulis, Athanasios. Probability, Random Variables and Stochastic Processes. Mc Graw Hill, New York, 1965, pp. 234-235.

P2. Parzen, Emanuel. Modern Probability Theory and its Applications. John Wiley, London, 1960, Chapter 6, pp. 260-261.

P3. Personal Correspondence from Mr. Adolfo Loza Arguello. Director of Telecommunications of Ecuador, 1969.

P4. Personal Correspondence from Mr. Alfonso Pereyra. Director of Telecommunications of Peru, 1969.

P5. Personal Correspondence from Mr. Alfonso Pineda Gutierrez. Director of Telecommunications of El Salvador. 1969.

P6. Pollard, J. R. "Technological Progress in Telecommunications Switching", Electronics and Power. August, 1970, pp. 305-308.

P7. Personal Correspondence from Mr. L. E. Scammell. Project Manager, I.T.U., United Nations office, Kartoum, Sudan.

R1. Reiter, S. and G. Sherman. "Allocating Indivisible Resources Affording External Economies or Diseconomies", International Economic Review. Vol. 3, No. 1, January, 1962, pp. 108-135.

R2. ----------. "Discrete Optimizing". Journal for the Society of Industrial and Applied Mathematics. Vol. 13, No. 3, September, 1965, pp. 864-889.

R3. Reiter, S. "Choosing an Investment Program among Independent Projects", Review of Economic Studies. January, 1963, pp. 32-36.

R4. Riker, William H. "Voting and the Summation of Preferences: An Interpretative Bibliographical Review of Selected Developments During the last Decade", The American Political Science Review. Vol. 55, December, 1961, pp. 900-911.

S1. Soewarso, M. "Optimale Investitionsstrategie beim Ausbau des Landestelefonnetzes in Entwicklungslandern, dargestellt in einer Fallstudie fur Indonesien", Dr. Ing. Dissertation. Technische Hochschule Munchen, 1967.

S2. Spencer, A. E. and F. S. Vigilante. "N°2 ESS: System Organization and Objectives", BSTJ. Vol. 43, September, 1964, pp. 1831-1844.

S3. Stillson, Paul. "A Method for Defect Evaluation", Industrial Quality Control. Vol. XI, No. 1, July, 1954, pp. 9-12.

S4. Spencer, D. L. and A. Woroniak. The Transfer of Technology to Developing Countries. Frederick A. Praeger Publishers, New York, 1967, p. 23.

S5. ----------. Op. cit., p. 6.

S6. Swain, E. C. "Local Lines--Past, Present and Future", The Institution of Post Office Electrical Engineers, Printed Paper No. 226, November, 1965, G.P.O., London.

V1. Vickery, William. "Utility, Strategy, and Social Decision Rules", Quarterly Journal of Economics. Vol. 74, 1960, pp. 507-535.

V2. Van't, Slot. "The Semi-Electronic Switching System PRX-205", IEE, Conference on Switching Techniques for Telecommunications Networks. London, April, 1969.

V3. Vigilante, F. S. "General Description of No. 2 Electronic Switching System", IEE, Conference on Switching Techniques for Telecommunications Networks. London, April, 1969.

W1. Ward, Barbara. Las Naciones Ricas y las Naciones Pobres. Mexico, 1963.

W2. Wellenius, B. and J. Potocnjak. Analysis of the Telephone Demand in the Residential Great Santiago. Report No. 20. Department of Electrical Engineering, University of Chile, December, 1968.

W3. Wellenius, B. "Estimating Telephone Necessities in Developing Nations: A New Approach", IEEE Telecommunications Conference. San Francisco, June, 1970.

W4. Whyte, J. S. "Telecommunications in the Next Thirty Years", P.O.E.E.J. 19 , pp. 137-141.

W5. Webster, H. L. "The SP-1 Electronic Switching System", IEE Conference on Switching Techniques for Telecommunications Networks. London, April, 1969.