LEAD TIME REDUCTION FOR NEW PRODUCTS

By

PAUL HEATON DAWKINS

Thesis submitted for the degree of Doctor of
Philosophy
The University of Aston in Birmingham

June 1983

THE UNIVERSITY OF ASTON IN BIRMINGHAM

LEAD TIME REDUCTION FOR NEW PRODUCTS

Submitted by Paul Heaton Dawkins for consideration for the degree of Doctor of Philosophy in 1983.

SUMMARY

The research, which was given the terms of reference, 'To cut the lead time for getting new products into volume production', was sponsored by a company which develops and manufactures telecommunications equipment.

The research described was based on studies made of the development of two processors which were designed to control telephone exchanges in the public network. It was shown that for each of these products, which were large electronic systems containing both hardware and software, most of their lead time was taken up with development. About half of this time was consumed by activities associated with redesign resulting from changes found to be necessary after the original design had been built.

Analysing the causes of design changes showed the most significant to be Design Faults. The reasons why these predominated were investigated by seeking the collective opinion from design staff and their management using a questionnaire.

Using the results from these studies to build upon the works of other authors, a model of the development process of large hierarchical systems is derived. An important feature of this model is its representation of iterative loops due to design changes.

In order to reduce the development time, two closely related philosophies are proposed:

- By spending more time at the early stages of development (detecting and remedying faults in the design) even greater savings can be made later on,

- The collective performance of the development organisation would be improved by increasing the amount and speed of feedback about that performance.

A trial was performed to test these philosophies using readily available techniques for design verification. It showed that about an 11 per cent saving would be made on the development time and that the philosophies might be equally successfully applied to other products and techniques.

DESIGN-METHODS: DEVELOPMENT-TIME: ITERATIVE-LOOPS:
DESIGN-REVIEWS: DESIGN-VERIFICATION

AKNOWLEDGEMENTS

I should like to thank the following for the support and help they gave me during the project.

- My Supervisors........Dr.G.Beaumont,
  Mr.G.A.Montgomerie,
  Mr.P.Sigrist.

- Mrs Sharon Dowson and Mrs Brenda Spittle for typing this thesis.

- The staff in the GEC Telecommunications Limited Telephone Switching Group Development Laboratories (particularly those working in the Processors Group).

- GEC Telecommunications Limited and the Science Research Council for sponsoring the project financially.

CONTENTS

PART I

PAGE

PART II

PAGE

PART III

PAGE

APPENDICES

LIST OF TABLES AND ILLUSTRATIONS

FIGURES

PAGE

PAGE

TABLES

PAGE

PREFACE

The project described in this thesis was performed under the auspices of the Interdisciplinary Higher Degree (IHD) Scheme of the University of Aston in Birmingham. This scheme has a number of characteristics which make its projects rather different from those for traditional PhD's.

IHD projects aim to solve problems which were originated in Companies, Commerce or Public Bodies by making use of expertise and resources from whatever faculties of the University may be necessary, and also from the Company. There is, thus, a benefit for the University, which is new knowledge, and a benefit for the Sponsoring Organisation which is that it has the problem investigated and, hopefully, solved in a dispassionate way. The projects are, therefore, more than just consultancy investigations.

The interdisciplinary approach has two major effects on the research. Firstly, the problem is examined from more than one view point or discipline and, secondly, the scope of the research tends to be wider than for the traditional PhD.

A further characteristic of IHD projects, which results from them taking place 'in the real world', is that they are often unclearly or ambiguously defined at the beginning and a significant part of the research may be spent identifying the problem. Also, the research may change direction several times while it is being carried out.

This thesis will show how the characteristics of IHD projects, outlined above, were exhibited during the research.

The work was carried out under a project sponsored by the Science Research Council and GEC Telecommunications Limited who initiated it.

PART I

CHAPTER 1

INTRODUCTION

Research workers and R & D managers have in recent years been concerned about their ability to forecast accurately the duration and costs of development projects. They have published many papers on the subject and in these they have concentrated on the psychology and mechanics of the forecasting and planning, rather than examining the way project durations get built-up. <1, 2, 3>.

There have, however, been a number of studies made of the process of new product innovation but these have concentrated more on the mechanics of R & D projects <4, 5, 6, 7, 8, 9>. They have identified factors which have been shown to increase the chance of success of new products by considering aspects of project management, such as project selection, planning, control and employment of resources.

Although profitability is often used for judging success, a number of other criteria can be used. These are mostly commercial and include the cost of the product (including the cost of research and development), when the product is available for the market (which is usually dependent on the development time), the technical performance <10> and the suitability of the product to the innovating organisation. There are also a number of indirect criteria, such as how the product conforms to social standards, law and safety practices.

Usually, when deciding how successful a new product is, one considers a balance between these criteria but sometimes one or two of them may be far more important than the others. It was because of an overwhelming requirement to reduce development times that this project was initiated.

Effects of Extended Project Duration.

Although the length of the development period is one of several criteria that determine the success of a product, it is probably the most important in more cases than any of the others. Gee <11> highlights a general reason why the time period for innovation and thus development should be as short as possible,

> "The time span for technological innovation, is important in that it directly affects the rate at which innovations are produced. A higher rate of

technological innovation is desired for increased competitiveness and time response to market needs"

The duration also directly affects the cost of a project and its financial returns <12>. The longer it continues without a product being available for sales, the greater will be the cumulative cash flow into the project and more of the end product must be sold to create a profit. This may be exasperated if market conditions are changing rapidly and the product life cycle is limited.

The profitability of a product for a company often depends on whether the company is the market leader. The work by the Boston Consulting Group <13, 14> into experience curves shows that the competitor who has the largest cumulative market share has the lowest costs and probably the greatest profits. If the product is late on to the market then the company may lose the opportunity of becoming the market leader, and its profits may be diminished.

The length of one project may also affect others within a company if resources of cash, people and equipment are limited. These facilities may be tied up for longer than was planned, thus preventing the other projects, which may be just as important for the company's profitability and survival, from progressing.

The duration of an R&D project can also affect the success of the project indirectly by putting a strain on other criteria which would have enabled it to be successful. The longer the project, the more difficult it will be to plan, and cost estimating of the project and the end-product will be more difficult. Also, the longer the project, the harder it will be to maintain the team working on it and new people, requiring training, may need to be brought in. The psychological group structure of the team may be destroyed, adversely affecting morale and motivation and resulting in the project time increasing because of poor workmanship.

Although the project duration is usually of overriding importance, there are circumstances where it may be fixed (by market conditions or a customer) and what is important is the level of performance of the product that can be achieved within that timescale. Since the objective of techniques employed to reduce the duration of development is to increase the overall amount of work done in a given time, those same techniques would be effective in maximising the performance that can be achieved in a set time. Whether the overriding need is to maximise performance or to minimise development time, it is necessary to consider the mechanisms by which a new product is created.

The Innovation Process.

There are a number of recognised stages through which most innovations progress <15, 16>. The first of these is 'pure research' in which the basic laws of nature are formulated. These laws are then utilised, through a process of 'applied research', to produce the basis of the new product. The product is then developed to give its final form, which must satisfy customer needs, be compatible with company requirements and be capable of being manufactured economically. The design is documented and communicated to the organisation's non-development departments. Precisely to which departments the design is communicated, and what information is given, will depend on the type of product and the structure of the company.

Normally, this flow of information will take place during the course of development, which is the phase of a new product on which this thesis will concentrate.

The Development Process.

For a simple product, the development process starts by defining the requirements of the product; its desired performance, how it will be constructed, how it may be manufactured, how much it should cost and when it should be available for the market. These requirements are collated to form a specification from which the product is designed. This design is then tested, either by simulation or by operating prototypes, and an iterative optimisation of the product takes place.

For large, complex products the design may take place in stages, each with a different level of detail. Usually, the design starts at the system level and proceeds down through the subsystems to the basic components. The product is then tested, often starting at the lowest level and proceeding up to the system level. The iterative optimisation will take place at each of the different levels of the product hierarchy.

The development process thus comprises a number of stages which can be considered as being distinct, and which may be performed by different groups of people. The lead time will be affected either as a result of factors acting on each of the stages independently, or as a result of their inter-relationships.

In the former case, if the durations of the stages do not directly affect each other, then making one longer will make the lead time longer. Similarly, making one or more shorter will shorten the lead time. The speed at which each stage is completed will depend on general managerial skill, the techniques employed and the level of resources of manpower and equipment applied at that stage.

Usually, however, the duration of one stage is affected by other stages and it may be possible to shorten the overall time by increasing the length of some stages.

Iterations Within The Development Process.

Apart from a mention of the iterative optimisation of the product, the development process so far described has been assumed to be linear; that is the project passes from one stage to the next without returning to the first. This is not usually the case and it is often necessary to return to some stages already completed. The reasons for this are many but include changing customer requirements, design errors and mistakes, the availability of a new technology, social pressures and Government legislation.

There are a number of effects of the iterative process of returning to earlier stages of development. Firstly, the product lead time will lengthen. Secondly, the development cost will increase. Thirdly, documentation control will be more difficult and prone to mistakes because of the design changes and the number of different document issues generated. Fourthly, as engineers tend to prefer to create new designs, rather than modify existing ones, morale and motivation may suffer with further unfortunate consequences for the project. Fifthly, the quality of the design, and in particular reliability and failure modes, may be less satisfactory if parts of it are changed. Sixthly, limited resources may be tied up for longer than would otherwise have been the case, and seventhly, the iterations will add uncertainty to the project making it more difficult to plan.

It is the first of these effects, the increased length of the product lead time, on which this thesis concentrates.

The Research Described in This Thesis.

The need to reduce the development time could, therefore, be met in one of two ways; either by reducing the duration of each stage of development or by reducing the contribution of iterations to the overall development time.

The research described in this thesis concentrates on the latter approach. It examines the causes of the iteration, the mechanisms by which they take place and their effects. Philosophies for reducing these effects are then discussed and a practical trial is described and assessed.

CHAPTER 2

BACKGROUND OF THE PROJECT

2.1. Chapter Preview.

This Chapter describes the background of the project, why it was initiated and its importance to the Sponsoring Company. It sketches:

- The recent historical development of the Public Switching Sector of the British telecommunications industry (Section 2.2.),
- The industry's current structure (Section 2.3),
- The activities and organisation of the Sponsoring Company (Section 2.4.),
- The structure of the product,
- The non-development engineering activities required to supply the equipment.

2.2. The Recent Historical Development of the Public Switching Sector of the British Telecommunications Industry.

For many years, from the early 1920's, the only type of automatic telephone switching equipment produced by the British telecommunications manufacturers was Strowger. This was based on electro-mechanical step-by-step switches. It offered very restricted network facilities and the equipment took up a large amount of space and required considerable maintenance. It, nevertheless, was sold in considerable quantities, both to the British Post Office and to telecommunications authorities abroad.

In the 1950's and 60's, however, improved types of telephone switching equipment were developed by foreign competitors. The most important was Crossbar, the main feature of which was its register control. It offered the major advantage of flexible network routing of calls and, less significantly, greater reliability and reduced maintenance costs. It was, therefore, bought abroad in preference to the U.K. produced Strowger product.

At this time, contracts to supply the British Post Office with telephone switching equipment were placed on a non-competitive basis with the principal U.K. manufacturers. This was carried out under the "Bulk Supply Agreement" and the contract prices included a sum of money for new product development which was undertaken by the manufacturers. The first significant development under this arrangement resulted from the introduction of Subscriber Trunk Dialling (STD) which required Register Translator Equipment for the translation of dialled numbers. This had to work with the Strowger equipment, of which the network was constructed.

In the late 1950's, the Joint Electronic Research Committee (JERC) was set up between the Post Office and the manufacturers to co-ordinate the development of electronic switching equipment. The first achievement of this was the design and installation at Highgate Wood of the first all electronic telephone exchange. Although the principle behind it was shown to work, the technology - thermionic valves - was unsuitable. When the problems were recognised, sometime before the project was completed, development started on three other systems. Two of them, using time-division multiplexing, were unsuccessful, again because suitable technology was not then available, but the third, employing Reed relays as the speech switch, led to the development of a number of different types of exchanges. The most important of these are:

- The TXE2 small local exchange, the first of which was opened in 1966 and over 1000 of which have been installed in the U.K,

- The TXE4 large local exchange, the first of which was opened in 1976,
- And TXE4A, of which production started in 1979.

For several reasons, these types of exchanges were, and are, unsuitable for export. The most significant reason was that only subscriber exchanges were produced, as opposed to a complete family including trunk exchanges. The TXE' equipment was also surpassed by more advanced products produced overseas.

In the early 1960's, as TXE4 was not yet available, the British Post Office decided to install Crossbar exchanges for medium/large local applications, and in 1964 the first was opened. Crossbar was also employed for Group Switching Centres and Sector Switching Centres which used Stored Program Control (SPC) for the first time in the U.K.

Stored Program Control is the control of a Telephone Exchange, or part of it, using a central processor. Its development meant that greater flexibility could be achieved in controlling the exchange and in performing administrative functions. The manufacturers, however, were still unable to export the equipment in significant quantities because still no complete family of exchanges with a common technology was developed.

In 1969, under the Post Office Act, the Post Office became a nationally owned corporation rather than a Department of State. The Bulk Supply Agreement was abolished and emphasis was placed on competitive tendering for supply of equipment. Collaboration in switching system development, which had been co-ordinated through the Joint Electronic Research Committee, gave way to an arms length relationship on systems development both between the producers and with the Post Office. With the prospect of these changes about to take place, in 1968 an Advisory Group on Systems Definitions (AGSD) was set up between the Post Office and the manufacturers. The purpose of the AGSD was to produce definitions and specifications for new systems, but with the individual firms being left to meet the requirements in their own way. This led to divergent private venture development which produced a prolification of equipment types. Although there was an abundance of inventive ideas, the individual companies did not have the resources to carry them through to successful conclusions. The Advisory Group on Systems Definitions, however, undertook studies to attack two specific tasks. The first was essentially concerned with technical matters and was to advise the Post Office on systems and subsystems which should be developed for the 1980's. The other was to advise on how to reconcile innovation and competition in design, with standards necessary for inter-working and effectiveness. The more important areas of study are summarised by Harris <17>.

In 1975, the Joint Telecommunications Systems Strategy Committee (JTSSC) was set up, between the Post Office and the three main manufacturers, to build on the work of the Advisory Group on System Definitions and to bring the companies and the Post Office closer together to undertake the development of a major new telecommunications system. This new system, known as System X, is based on the findings of the Advisory Group on Systems Definitions and on the work which had been undertaken independently by the separate companies.

The discussion above shows why the British telecommunications manufacturers became unable to export telephone switching equipment. There is, however, another important factor affecting their operations.

There is an ever decreasing added value content of the product being produced. There are two reasons for this: firstly, the product is becoming physically smaller with higher levels of 'integration' and, secondly, a high proportion of the costs are due to bought-in items.

The Post Office has committed itself to an extensive modernisation programme, but even so, if the manufacturers are to continue to produce telephone switching equipment, without further reducing the size of their operations, they must be able to export their products.

In order to do this they must be able to offer up to date technology. In common with other electronic products, the technology of telephone switching equipment is advancing faster and faster. Development of new equipment must, therefore, be rapid, not only to produce competitive products, but also because the life cycle of each product is being reduced. This is shortening the potential period when development costs can be recovered from sales.

It was with this background that the Sponsoring Company decided to initiate the project.

2.3. Current Structure of the Telephone Switching Sector of the British Telecommunications Industry.

This section is concerned with the current structure of the public telephone switching sector of the British telecommunications industry but concentrates on the development of new products and, in particular, on the organisation which has been set up to develop System X.

There are four main organisations concerned with telephone switching equipment in the U.K.; The British Post Office, GEC Telecommunications Ltd., which is the Company sponsoring this project, Standard Telephones and Cables Ltd (STC), and Plessey Telecommunications Ltd

(PTL). In order to market System X abroad, however, another company has been set up. This is British Telecommunications Systems Ltd (BTS) and is jointly owned by the other four organisations.

The British Post Office is responsible for planning and running the public telecommunications network in Britain. It also undertakes some research and development in the communications field. Although it manufactures a small percentage of the equipment it uses, it purchases most of it from the three principal producers; GEC, STC and PTL.

These companies are the main producers of telephone switching equipment in the U.K. and both manufacture and install it. They also undertake research and development, either through private venture projects financed by themselves, or through projects under contract from the Post Office. The largest development project currently being carried out is for the family of telephone exchanges known as System X. This is being financed by the Post Office and undertaken jointly by all four of the organisations. At the time of writing, there were over 1000 engineers working on it.

System X has a modular construction: that is, each of the subsystems is self contained and can be developed independently of the others, provided the interfaces are adequately defined. Each of the participating firms is responsible for the development of particular subsystems and also one or two of the exchanges. An exchange being developed by one of the companies may be constructed of subsystems developed by any of them.

The design authority for the whole project is vested in the Post Office which approves all designs and standards and vets changes. It also maintains overall control of the project and carries out planning and monitoring functions, but at a high level. Each individual work area, which is a section within one of the participating companies and is responsible for one subsystem or a system, produces its own plans within the framework of the total plan, and monitors its own progress, which it reports to the Post Office.

A number of working parties, incorporating representatives from the four organisations, have been set up to co-ordinate various aspects of the project. These include Standards, Documentation, Design Procedures, Computer Aided Design Techniques and Configuration Management.

Clearly, with a project of this complexity and with so many organisations participating, the exchange of information must be rigorously defined. Two important principles govern the approach which has been adopted.

1. Each design must be capable of being manufactured by any of the participating firms and,

2. The product must be identical, irrespective of which firm made it.

As a result of these, all product information must be interchangeable between the firms. This is co-ordinated through the System X Information System (SXIS) which is described by Price and Allen <18>. The documentation computer databases of the three participating firms are linked to that of the Post Office so that they are all synchronised. To achieve this co-ordination, a carefully defined, comprehensive documentation structure has been developed.

The telephone switching sector of the British telecommunications industry has been homogonized in order to be able to undertake the major project of developing System X. Although the participating organisations are still independent of each other, they are no longer working at arms length from each other. They are working together, but as a result of this some of the development overheads are higher than they would have been if the project were undertaken by just one company. More time and effort has had to be spent on activities supporting design and development than would be the case for a simpler project.

2.4. Activities and Organisation of the Sponsoring Company.

This Section describes the Company environment in which this project was undertaken. An outline is given of the activities and organisation of the Sponsoring Company. It is, however, not the intention to discuss here in great detail the organisational structure as the relevant parts are described in Appendix A.

The project was sponsored by GEC Telecommunications Ltd. which is a Management Company of the GEC Group of Companies. It employs about 19,000 people and is responsible for manufacturing telecommunications equipment and systems. It has eleven factories spread over the country; five in Coventry, three in the North East, two in Scotland and one in South Wales. The headquarters and development laboratories are situated in Coventry.

The Company is divided into four divisions:

- Telephone Switching Group (TSG) which produces public telephone switching equipment (telephone exchanges within the Post Office's network),

- Telephone Division which produces subscribers equipment,

- Transmission Division which produces the equipment which links exchanges together,

- Private Systems Division which produces private exchanges for companies, hotels, universities, etc.

Although there are a number of central departments, such as the financial, computer, personnel and commercial departments, the four divisions operate largely independently of each other within their defined areas. This project was sponsored by Telephone Switching Group. The activities of the other divisions are not covered further in this thesis.

Telephone Switching Group is responsible for the development, manufacture, testing, installation and sales of public telephone switching equipment. The organisational structure of the Division is described in detail in Appendix A.

Telephone Switching Group has one main customer, the British Post Office. Although some equipment is sold to telecommunications authorities abroad, about 90 per cent of the output is supplied to the Post Office. The Company has, in effect, a monopoly customer which has two important consequences for it.

Firstly, the monopoly customer can insist that contracts contain special clauses which would not be included for other customers.

Secondly, any changes in the fortunes or policies of the Post Office are directly reflected in the Company.

2.5. Structure of the Product.

This Section gives a brief, non-technical description of the system structure and the physical construction of typical equipment currently under development.

The descriptions given apply in particular to System X although there is a similarity with other new products. The reasons for using System X for this discussion are:

- That it is by far the most important product under development,

- It is typical of the next generation of telephone switching equipment, and

- Development of parts of it are used as case studies in this thesis.

System Structure.

Within a public telephone network there are a number of different types of telephone exchange. These range from small subscriber exchanges for rural areas, through to large trunk non-subscriber exchanges for switching calls between subscriber exchanges. There may be as many as twelve exchange types in the family necessary to cover this range, and each of these would be developed as a separate system, but using a number of common subsystems. The system structure outline is shown in Figure 2-1. Technical details of the System X structure are described by Kirtland and Tippler <19> and in the brochure, "System X, The Complete Approach to Telecommunications" <20>.

Each system is constructed of standard subsystems which may be either hardware or software. 'Hardware Subsystems' are constructed from 'Hardware Functional Entities'. These carry out defined functions within the subsystem and are in turn made up of 'Slide-in-units' (SIU's) or special equipment such as disc drives. These are the lowest level of assembled equipment. Each Slide-in-unit contains part of its Functional Entity circuit.

The 'Software Subsystems' are made up of 'Software Functional Entities' which are analogous to the hardware

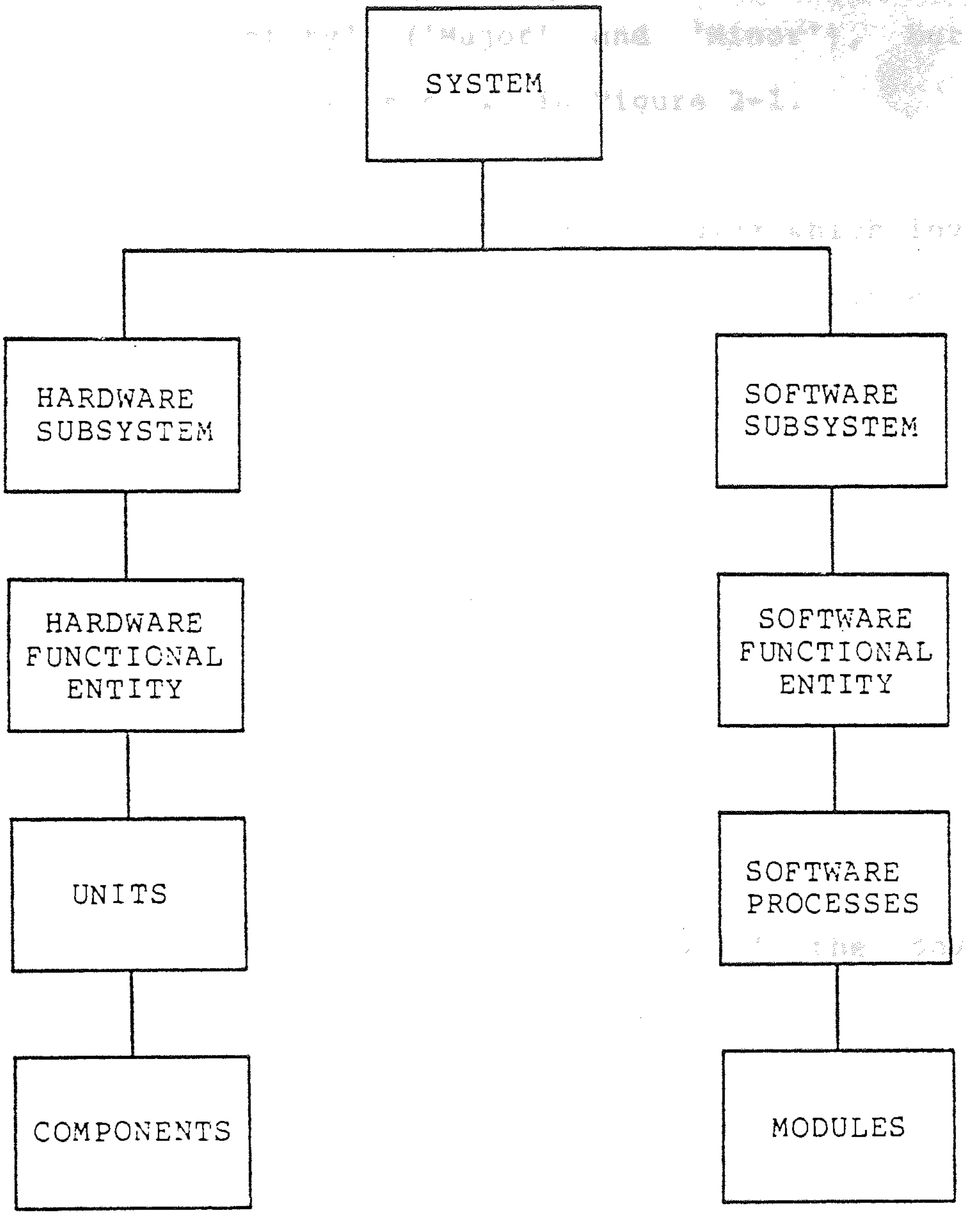


Figure 2-1. System Structure of a Typical Telecommunications System.

ones. They consist of 'Software Processes' which are subdivided into 'Modules'.

Some subsystems contain both hardware and software 'Functional Entities'. There may be two levels of

'Functional Entity' ('Major' and 'Minor'), but for simplicity only one is shown in Figure 2-1.

This thesis will consider studies which involved the development of two exchange subsystems. Since these were developed largely independently of other subsystems, although with clearly defined interfaces, they may be considered as being systems in their own right. They will be treated in this way throughout the rest of this thesis. The 'Functional Entities', then, will be treated as subsystems.

Physical Equipment Construction.

A description is given below of the physical construction of typical equipment. This is shown diagrammatically in Figure 2-2.

In the discussion of the system structure it was shown that the lowest assembled level of hardware is the Slide-in-unit. Each of these is constructed of a Printed Circuit Board, a front panel and a number of components ranging from simple passive ones to Large Scale Integrated Circuits. A number of Slide-in-units plug into a Wired Shelf Group which contains the necessary wired connections to make a complete circuit. The assembly of a Wired Shelf Group and its associated Slide-in-units is known as an 'Equipped Shelf Group',

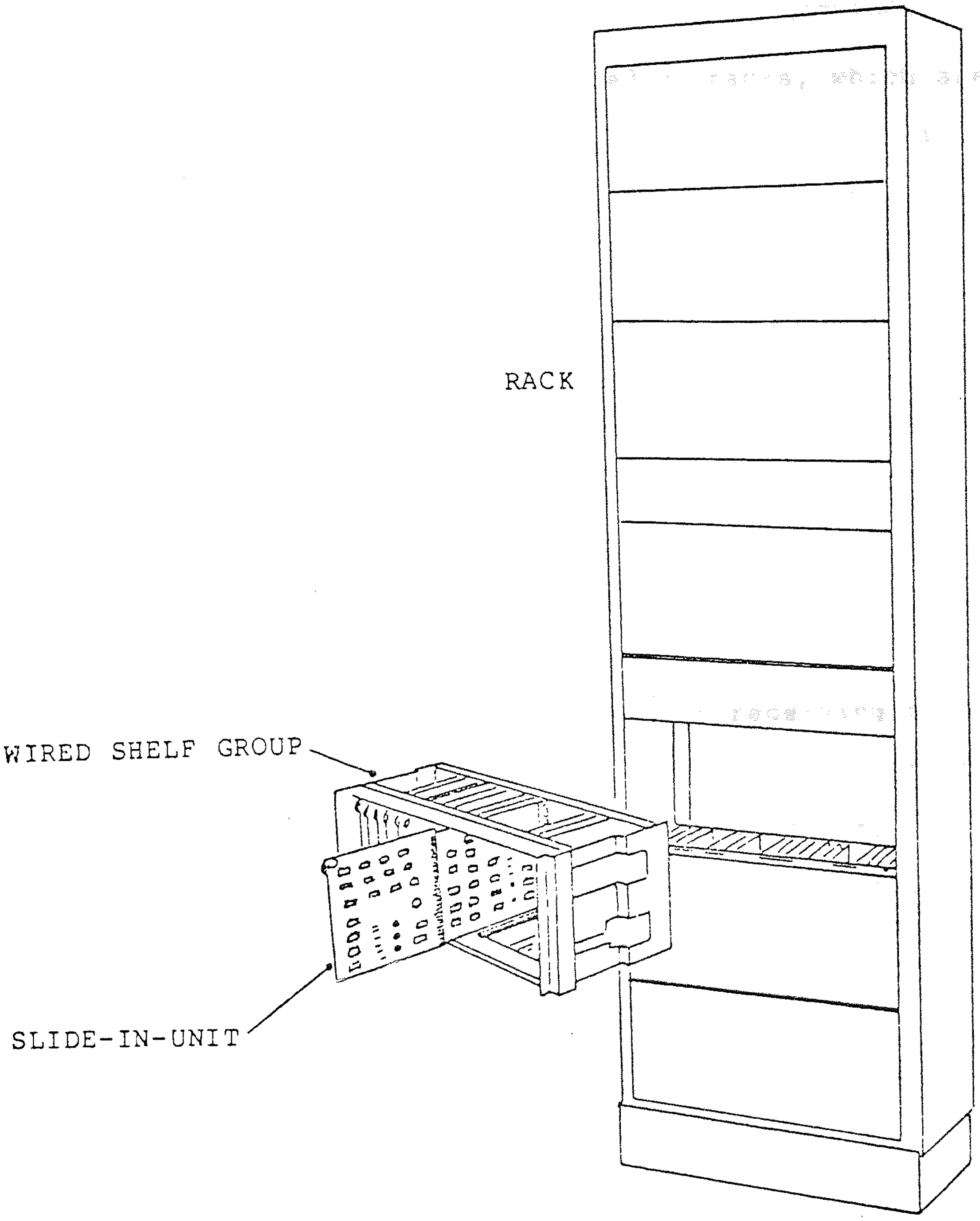


Figure 2-2. The Physical Construction of Typical Telecommunications Equipment.

which is often the physical realisation of a 'Functional Entity'.

Wired Shelf Groups are mounted on racks, which are purely mechanical and have no electrical function. Shelves within one rack, or between racks, are connected together by plug terminated cables.

2.6. Non-Development Engineering Activities Required to Supply The Equipment.

Although the equipment is produced in standard modules, these must be selected and arranged to meet a particular customer's requirements. On receiving these requirements, therefore, it is necessary to consider the number of lines, the traffic, charging procedures, etc., to decide which modules to use and in what quantities. This applies to software as well as hardware.

A second engineering activity is to design the inter-rack cabling and the ironwork to hold it and support the racks. This is necessary because of the various shapes of exchange buildings. Power requirements must also be established.

A third engineering activity may be necessary to develop special equipment or software for a particular unique application.

Efficiency of manufacture, then, is maximised by producing standard modules, but equipment versatility is achieved by varying the configurations in which they are assembled.

CHAPTER 3

NARROWING DOWN THE PROBLEM

3.1. Chapter Preview.

When this project was initiated, it was given terms of reference which were general and wide ranging. This Chapter describes how the problem, defined by the terms of reference, was narrowed down to a specific aspect of the product development process.

The Chapter opens with a discussion of the terms of reference (Section 3.2). It goes on to describe a study of the formal organisational structure of the Sponsoring Company and how information concerning new products is disseminated through that organisation (Section 3.3).

This is followed by descriptions of the development and product transfer processes (Section 3.4). The next Section, (3.5), describes how the formal, official documentation information is generated, processed, distributed and used. The duration of each stage is indicated and it is shown how the most significant contribution to the total lead time is taken up by development. More specifically, it is shown that design change and redesign durations account for a major part of the development time. For this reason, it was decided that the project should concentrate on the phenomenon of redesign, its causes and its effects.

The Chapter concludes with a summary of the narrowing-down process and how the project boundaries were erected.

3.2. Terms of Reference.

The terms of reference of the project, given at its conception by the Sponsoring Company, were:

> "TO CUT THE LEAD TIME NECESSARY FOR GETTING NEW PRODUCTS INTO VOLUME PRODUCTION"

Although this set the task to be carried out, it was extremely general and needed clarification in a number of areas in order to reduce the problem to something tangible and manageable. In order to do this, it was necessary to restrict the project through the erection of project boundaries. These are listed and discussed below:

a) Products to be covered by the Project.

Whilst it was desirable that the conclusions of this thesis should cover as wide a range of products as possible, the Sponsoring Company was specifically concerned with modern electronic telephone switching equipment.

As was outlined in Chapter 2, a characteristic of this type of equipment is that it takes the form of a large electronic system which has strong similarities with other products, such as computers.

The first restriction put on the project, therefore, was that it should concentrate on large, electronic systems. This would, nevertheless, encompass a number of different products but conclusions should still, where possible, apply to an even wider range of products.

b) Hardware/ Software/ System Design.

The terms of reference do not specify whether the conclusions should apply to hardware, software, systems design or all three. As large electronic systems, and in particular Telephone Exchanges, are constructed of hardware and software, the project had ideally to cover both. It was also required to include Systems Design.

c) Definition of "Lead Time".

"Lead Time" was not defined, although it was essential to know what had to be cut. The Company, however, agreed that it should cover the period from the conception of an idea for a new product, through the definition, design, development and production engineering stages, to the time when the product is introduced into volume production.

This begged a definition of what was meant by the expression, "introduced into volume production". In the past, prototype models of some products had been made in volume production before they had been fully developed and when they were some way short of market introduction.

d) Definition of Lead Time End Point.

The end point for the lead time was defined as the time when production, test, installation and commissioning could take place with no further support required from the development laboratories. The product documentation must be complete, the manufacturing processes developed and accepted, and test programs written.

Whilst this definition was adopted, it was recognised that so long as production continued there would, from time to time, be some laboratory intervention. This would be because of the product being changed to include new technology, because of experience of operating the product in the field, or because of cost reduction exercises.

e) Size of Development Project.

At any one time, the development department of the Sponsoring Company runs a number of projects of different sizes. These range from small product modifications to the development of a whole family of telephone exchanges. The Company specified that the conclusions of this project should apply to all these development project sizes, but, in particular, to the large scale developments of new products.

f) Size of Cut Required.

This should be as large as possible but the Company would be pleased with a 20% reduction.

g) Cost of Improving Lead Time.

The Company placed no constraints on the amount of money which might be spent to improve the lead time. They did, however, make the proviso that any proposed expenditure would have to be justified according to "normal company practice". This was to apply both to once-off expenditure and to possibly increased running costs.

Summary.

The statement below describes the terms of reference as they had become modified by the boundaries discussed above.

TO CUT THE TIME FROM THE ORIGINAL CONCEPT OF A NEW PRODUCT UNTIL THAT PRODUCT IS IN VOLUME PRODUCTION, WITH NO FURTHER DEVELOPMENT LABORATORY INTERVENTION NECESSARY TO MAINTAIN PRODUCTION, FOR NEW, LARGE ELECTRONIC SYSTEMS AND IN PARTICULAR PUBLIC TELEPHONE SWITCHING EQUIPMENT.

3.3. Study of the Formal Structural Organisation of Telephone Switching Group.

During the early days of the project, a study was undertaken to establish the organisation structure of the Sponsoring Company and how that organisation is used during the transfer of production of new products. It was undertaken by examining Company documents and by interviewing each departmental manager.

As the structure is described in Appendix A, the discussion here will be restricted to some comments about the structure.

Conclusions of the Study.

(a) Product transfer takes place through the functional organisation of the Division. That is, the departments carry out their normal functions even though a major new product may be being transferred from Development. Usually, no special department or project management function is set up to cope with an individual product.

(b) The organisation employed is similar to the first of five models for technology transfer - "Product Specification and Diagrams" (Figure 3-1) described by Gerstenfeld <21>.

In Gerstenfeld's model, the outcome of R&D is a complete set of product "specifications" and drawings, and these are transferred to Manufacturing. Communications between R&D and Manufacturing are limited and mainly take place through interpretation of these drawings and because of incomplete specifications, omissions and clerical errors.

As a modification to this model, Gerstenfeld suggests that an additional stage of a paperwork review

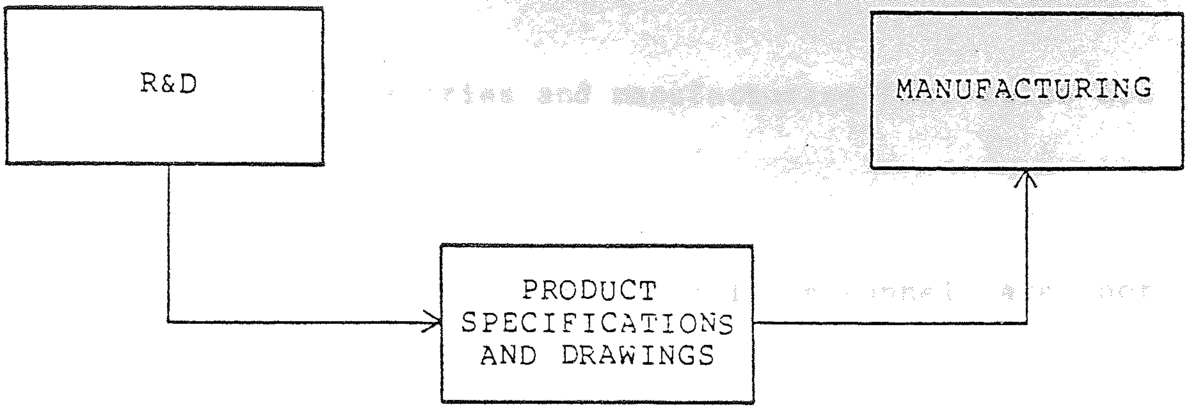


Figure 3-1. Technology Transfer : Product Specification and Drawings (From Gerstenfeld).

may take place before Manufacturing accepts responsibility for the product. He suggests, however, that this is likely to be unsatisfactory as it adds cost, there is not time for significant changes, and designers are reluctant to make significant changes.

Gerstenfeld suggests that the model should only be used when:

- The new product is relatively simple,
- The production process is flexible,
- The transfer time is not critical,
- There is little need to reduce production costs,

- R&D laboratories and manufacturing facilities are separate,

- Skilled multi-functional personnel are not available,

- R&D and Manufacturing operate separately.

Even though this model describes, in principle, the method of product transfer used by the Sponsoring Company, most of the conditions do not apply to that method. The model is, however, an over simplification of the practical process as a two way flow of information does arise through informal communications which take place between departments during their normal activities.

3.4. New Product Development and Transfer Process.

The present Section describes the process of new product development and transfer to production.

Method Used for Collection of Data.

The details of the process which is described below were collected by interviewing Development and other relevant Managers. This was augmented by information extracted from development plans for the two System X

Processor subsystems. During the early stages of the project, when collecting this information, no formal process charts for the development process had been produced. However, by the time of writing this thesis, some stages had been documented for Quality Assurance purposes.

The Development Process.

The detailed development process for 'units of equipment' is presented in Appendix B.

The Transfer Process.

Investigation identified three separate development - transfer processes which take place within the Sponsoring Organisation. These are shown in Figure 3-2 and discussed below:

Development and Transfer of Standards.

The first stage in the development of a new product would usually be the development of its standards. The result is a definition of the rules describing how the product will be physically constructed (Equipment Practice), what standard piece parts and components will be used in its construction and its documentation structure and

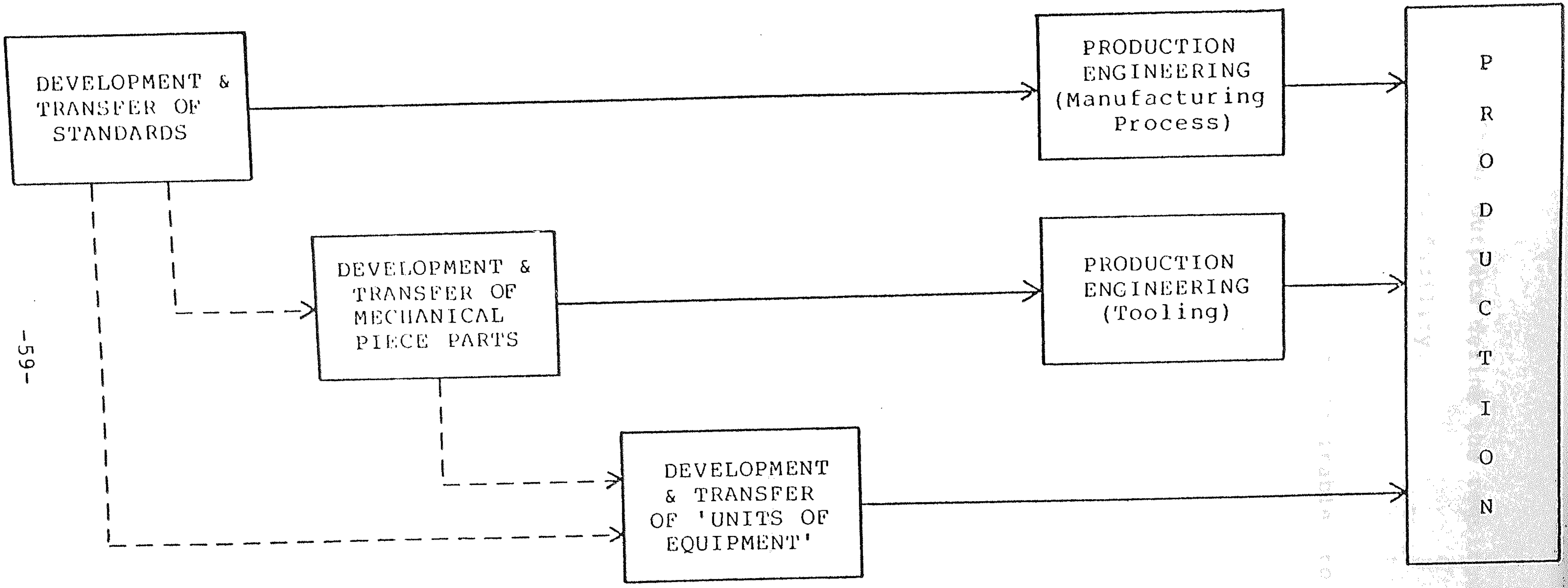


Figure 3-2. The Three New Product Development and Transfer Processes.

form. These outputs define the requirements for the manufacturing facility.

Although it is desirable to define the standards to be used in the design of a new product before its design starts, this is not always possible. Often, the development of the standards takes place in parallel with the product design.

As a number of different products may be designed round the same standards, this development and transfer process may take place only once for all of them.

Development & Transfer of Mechanical Piece Parts.

The outcome of this process is the detailed design of the mechanical piece parts needed for construction of the product. This may be regarded as being part of the previous process. However, it is carried out by a largely separate set of engineers. It leads to the specific activity of tool design and as a result tends to have its own special problems. Like the previous process, this one may take place only once for a number of products.

Development & Transfer of 'Units of Equipment'.

'Units of equipment' are defined as assembled modules of equipment which are designed and constructed using the standards, rules and components defined during the previous two stages. They are usually the physical realisation of electrical circuits and are related to the working of the product. In terms of the equipment structure described in Chapter 2, the 'units of equipment' are the Slide-in-units and Wired Shelf Groups. Software modules may also be covered by the expression "units of equipment", and its development by this process.

If the development and transfer of standards and piece parts has already taken place, then the manufacturing process could have been designed and developed by the time this third stage takes place. There should, therefore, be little or no production engineering associated with it.

On recognising the three separate development and transfer processes, it was decided that this project should concentrate on the third, the development and transfer of 'units of equipment'. The reasons for this decision are given below:

(a) The transfer of 'units of equipment' may take place many times for each transfer of standards or mechanical piece parts. One product may involve the transfer of several hundred 'units' and several products may use the same standards and mechanical piece parts. Savings in the duration of the 'units' development and transfer are, therefore, likely to result, on average, in greater savings to product lead times.

This effect could, however, be less significant if the time for the development of standards and piece parts is very great or on the critical path for development of the complete product. This is not usually significant in the telecommunications industry because changes to standards tend to be evoluntionary rather than catastrophic and development of a new product can often take place around old standards.

(b) In the Sponsoring Company, manufacturing facilities tend to be process orientated rather than product orientated. This is a consequence of the evolutionary nature of product 'Equipment Practices' and means that the transfer of the standards and setting up of the production processes are comparatively simple.

(c) The Company had already initiated a project to investigate the development and transfer to production of mechanical piece parts.

In conclusion, therefore, a project boundary was erected to concentrate the studies on the development and transfer of 'units of equipment'

3.5. Production, Dissemination and Use of 'Formal New Product Information'.

The present Section is concerned with the production, dissemination and use of 'formal information', that is information which is contained within the prescribed formal documentation structure of the new product. It describes when and how the documentation is produced, processed, distributed and used, but not its structure.

A model of these activities, which shows not only the sequence of the operations but also their durations, is exhibited in Appendix C. It takes the form of a PERT network modified to show documents produced from, and input into, each activity.

After a description of the method used to collect the data from which the model is constructed, the discussion following concentrates on some important conclusions which can be drawn from the model. It then describes how these pointed to the specific topic on which the remainder of this thesis concentrates.

Method Used for the Collection of Data.

The basic PERT network which forms the core of the model shown in Appendix C was taken from the Design and Development Plan for the LPU Subsystem of System X (See Section 4.2). This was modified and augmented by information found in Company policy statements and directives and by information supplied by department managers. The stage durations are also taken from the LPU plan.

Limitations of the Model.

The model, exhibited in Appendix C, is a simplification of the project plan for the LPU subsystem of System X, but is typical of other telephone switching equipment development projects. Although the model is a general representation of the development project, it has certain limitations:

(a) It was not possible to make a realistic comparison between achieved durations and planned ones because not all the figures for the former could be derived from the progress and monitoring reports. "b" below also contributed to this difficulty.

(b) The model covers all levels within the equipment structure, and this leads to difficulties in defining the durations of the stages and their completion points.

(c) Feedback from documentation users is not shown. This may occur as a result of problems encountered and usually takes the form of 'Engineering Information Queries'. It is assumed, in the model, to be absorbed within the durations of the other activities shown.

(d) During the course of the LPU project the plan changed quite extensively. This reflected a change to the specification and action required to minimise the effects of project slippage.

Durations.

The planned duration of each activity is shown in the model. It was assumed that three rework cycles would be required. This gives a total development lead time, from the Product Definition through to stable volume production, of 133 weeks (about 3 years). This also represents the actual total time fairly accurately.

Of this total time of 133 weeks only 10 Weeks are required for the transfer process - distribution and printing of production information and preparation of the production database and insertion aid programs. (This does not include time for ordering components and piece parts). It was, therefore, decided that this project should concentrate on the development process which takes the remaining 123 weeks duration.

Of the development time of 123 weeks, 65 weeks (about 50%) is taken up with activities associated with rework. As a result, it was decided that this project would concentrate on attempting to shorten the lead time by reducing the duration of the rework stages.

Although it was felt likely that a significant improvement could be made in this area, it was recognised that the rework activities could not be totally removed (maximum saving of 65 weeks). This is because:

- 'Commissioning' would still be required, although its duration would be less with fewer Design Faults and versions of each unit,

- It was highly improbable that the need for design changes, and hence rework, could be completely eliminated.

3.6. Summary of Chapter.

It is shown below how this project evolved as a result of the studies described in this Chapter.

STAGE/STUDY: Terms of Reference.

CHANGE TO PROJECT:

Product: The project would concentrate on large electronic systems.

Hardware/ Software/ Systems Design: The project would include all three.

Definition of Lead Time: To be defined as the time from the conception of the idea for a new product through to when the product is in volume production with no further support required from the development laboratories.

Size of Development Project: The result of the project should be applicable to large, new product development projects.

Size of Cut Required to the Lead Time: This should be as large as possible, but 20% would be satisfactory.

Cost of Cutting the Lead Time: There would be no restriction on this, but all costs would have to be justified according to the usual criteria of the Sponsoring Company.

STAGE/STUDY: New Product Development and Transfer Process.

CHANGE TO PROJECT:

The project would concentrate on the Development & Transfer to production of 'Units of Equipment'.

STAGE/STUDY: 'Formal Information' production, dissemination and use.

CHANGE TO PROJECT:

The project would concentrate on the Development stage of the new products. It would aim to achieve savings by reducing the amount of time spent on rework activities.

CHAPTER 4

CAUSES OF REWORK

4.1. Chapter Preview.

The present chapter describes a number of studies which were carried out to investigate the cause of redesign.

The Chapter opens with a description of two development projects chosen for the studies, and an explanation of why they were chosen (Section 4.2).

It then proceeds by discussing the studies:

- Rates of Production of Changes for the projects (Section 4.3),

- Rates of Production of Changes for Individual Slide-in-units (Section 4.4),

- Immediate Causes of Design Changes (Section 4.5),

- Causes of Design Faults (Section 4.6),

- Effects of Design Engineer Experience on Number of Changes (Section 4.7).

N.B: Most of the data collected during the studies originated from 'Change Notes'. These, along with the procedures which govern their use, are described in Appendix D.

4.2. Development Projects Used in The Studies and Investigations.

The two projects, which were used in the investigations of the causes and effects of design changes, were concerned with the development of processor subsystems for the System X family of telephone exchanges. The two processors are known as the "LPU" and "SPU". A comparison of their respective sizes is shown in Table 4-1.

Although there were a number of similarities between the two products (see below) and their development projects, the SPU was about a year in advance of the LPU. They were largely developed by separate teams but under a common manager. In the later stages of the projects the two teams were amalgamated into one processor development team. At this stage most of the development work was concerned with incorporating changes into the designs and with production of design

ENTITY	NUMBER IN LPU	NUMBER IN SPU	RATIO OF NUMBER IN LPU TO NUMBER IN SPU
Hardware Major Functional Entity	10	5	2
Wired Shelf Group	11	6	1.8
Slide-in-unit	72	37	1.9
Software Process	29	14	2.1

Table 4-1. Comparison of the Sizes of the LPU & SPU.

documentation. About 30 engineers were involved at any one time with the development of the two products.

Reasons for Choosing the LPU and SPU Projects for Investigating Design Changes.

Listed below are the reasons why the LPU and SPU development projects were chosen for investigating the causes and effects of design changes.

- Each of the projects was at a suitable stage. At the time the investigations started, all the SPU Slide-in-units had been designed, as had most of those of the LPU. Of the LPU units which had been designed a number had not yet reached production or been commissioned. The generation of 'Change Notes' was at a peak for the LPU (as seen in retrospect) but was still running at a significant level for the SPU.

- The two projects were similar in the following respects:

  They were both concerned with the development of Processors.

  Both projects involved the development of hardware, which was mainly digital, and software.

  They both used the same standards (design rules, documentation structure, project control procedures, standard components) and they even had some common Slide-in-units.

The two projects had a common manager which led to a close liaison and interchange of ideas and people.

- The two projects were at significantly different stages of progress. This meant that looking at the two projects could be considered similar to looking at one project at two different times.

- The technology used in the products was thought to be typical of future products, so any conclusions resulting from the studies would be likely to apply in the future.

- The projects were a convenient size - both small enough to record in detail and large enough to produce statistically significant information.

- The products were not dependent on other subsystems for their development to progress. They could be considered as independent systems, but with the definitions of the interface with other System X subsystems included in the original specifications (customer requirements).

Disadvantages of Using LPU & SPU Projects in the Studies.

There were three disadvantages against using the LPU and SPU development projects.

1. Their similarities restricted the variety of products and development techniques which were studied.

2. The development of the System X exchanges and many of its subsystems was dependent on the development of one or other of these two processors. This resulted in management being reluctant to sanction experiments involving new procedures or techniques which, if unsuccessful, could increase the development time.

3. The processors did not suffer from problems associated with the integration of subsystems designed by different teams in different organisations.

4.3. Rate of Production of Changes for LPU and SPU Projects.

The current Section discusses the rate of production of design changes for the LPU and SPU development projects and gives an account of how the information was collected. The comments and explanations given here about the observed phenomena are limited to the comparatively obvious, but a deeper discussion is given in Chapter 7.

Method.

The date, 'Change Note Number', 'Part Number' and whether the change applied to the LPU or SPU were recorded for each 'Change Note' produced for the LPU and SPU over the whole period that the 'Change Note' system was operational, up to the end of December 1979, (a period of three years). This information was stored on a computer file along with information required for other analysis.

Sources of Error.

Possible sources of error are listed below:

1. Clerical errors in recording the information from the 'Change Notes' and in updating the computer file.

This was minimised by printing the file in 'Change Note Number' sequence and looking for missing 'Change Note Numbers' and out-of-sequence dates.

The programs which updated the computer file also performed some checks:

- They verified that the key of "Change Note Number/Part Number" was unique for each 'Change Note' recorded.

- They ensured that the date was realistic.

2. Some 'Change Notes' were not dated. These were recorded with the date of the 'Change Note' with the preceding identification number. The percentage of these was small (about 1%) and effects negligible.

3. Errors made in writing 'Change Notes'.

The Part Number would be validated when the 'Change Note' was used to create a Modification Action Package or in a rework.

Errors in the dates were either obvious (and corrected before analysis) or of negligible effect.

Pattern of Rate of 'Change Note' Production.

The number of 'Change Notes' produced per calendar month for the LPU and SPU are shown in Figure 4-1 and Figure 4-2, respectively. These figures also show the distribution of the 'Change Notes' between the following categories:

Subsystems

Slide-in-units

Wired Shelf Groups, and

Other assemblies.

Figure 4-3 gives a comparison between the pattern of changes produced for the LPU and SPU. The SPU graph is modified in two ways to give greater meaning to the comparison. Firstly, the "y" scale is adjusted to take into account the difference in size between the two processors. The SPU is approximately half the size of the LPU (See Table 4-1). Secondly, the timescale of the SPU graph has been displaced by one year, to take into account the SPU project running a year ahead of that of the LPU.

Observations.

(a) The rate of 'Change Note' production peaks about the same time relative to the start of both projects. Figure 4-4 shows the number of LPU Slide-in-units for which the design was completed per calendar month. The peak production rate of LPU 'Change Notes' occurred about 4 months after the peak of the design completion rate. This is about the time required for manufacturing the first model and for commissioning to

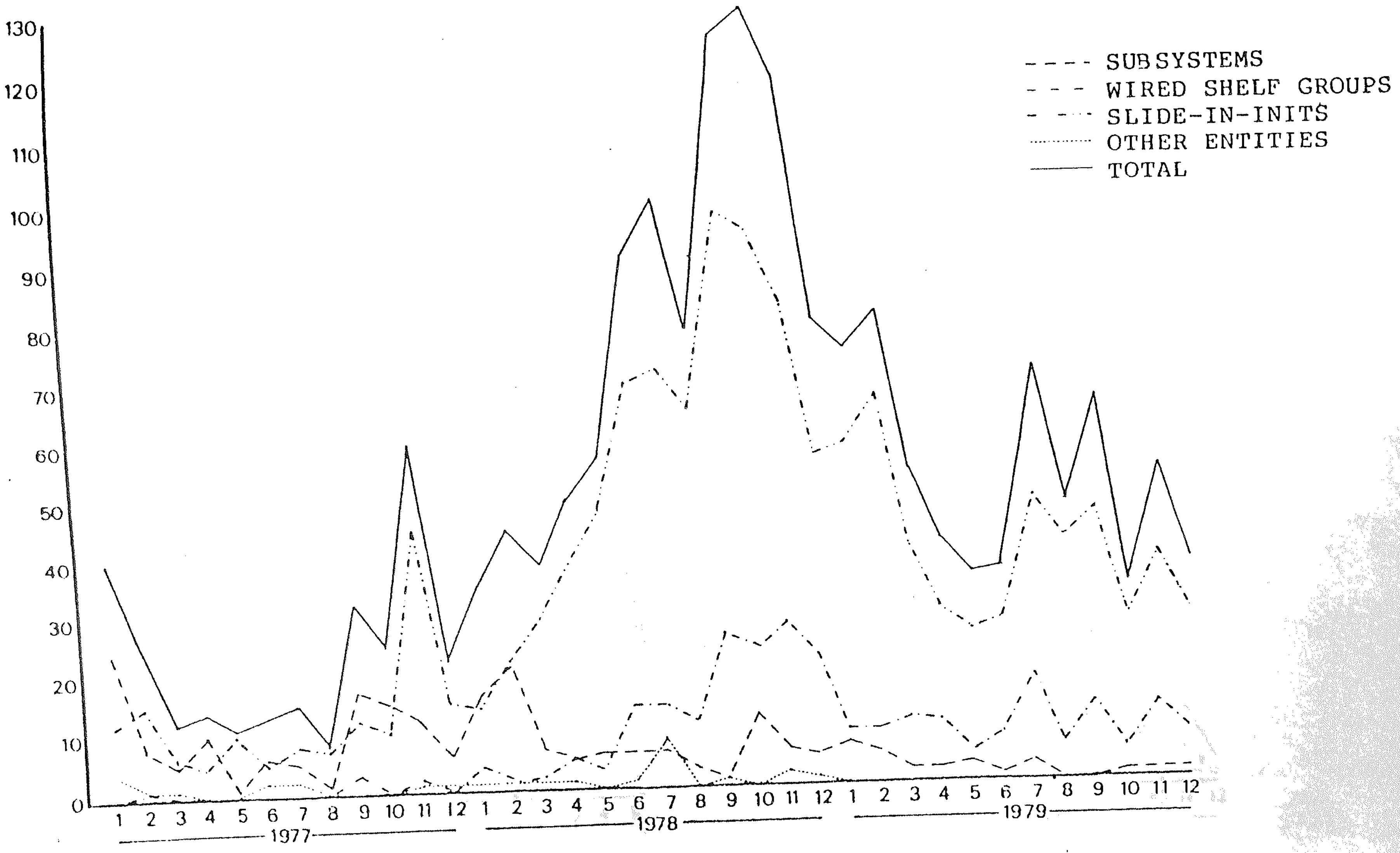


Figure 4-1. Number of 'Change Notes' Produced per Calendar Month for the LPU.

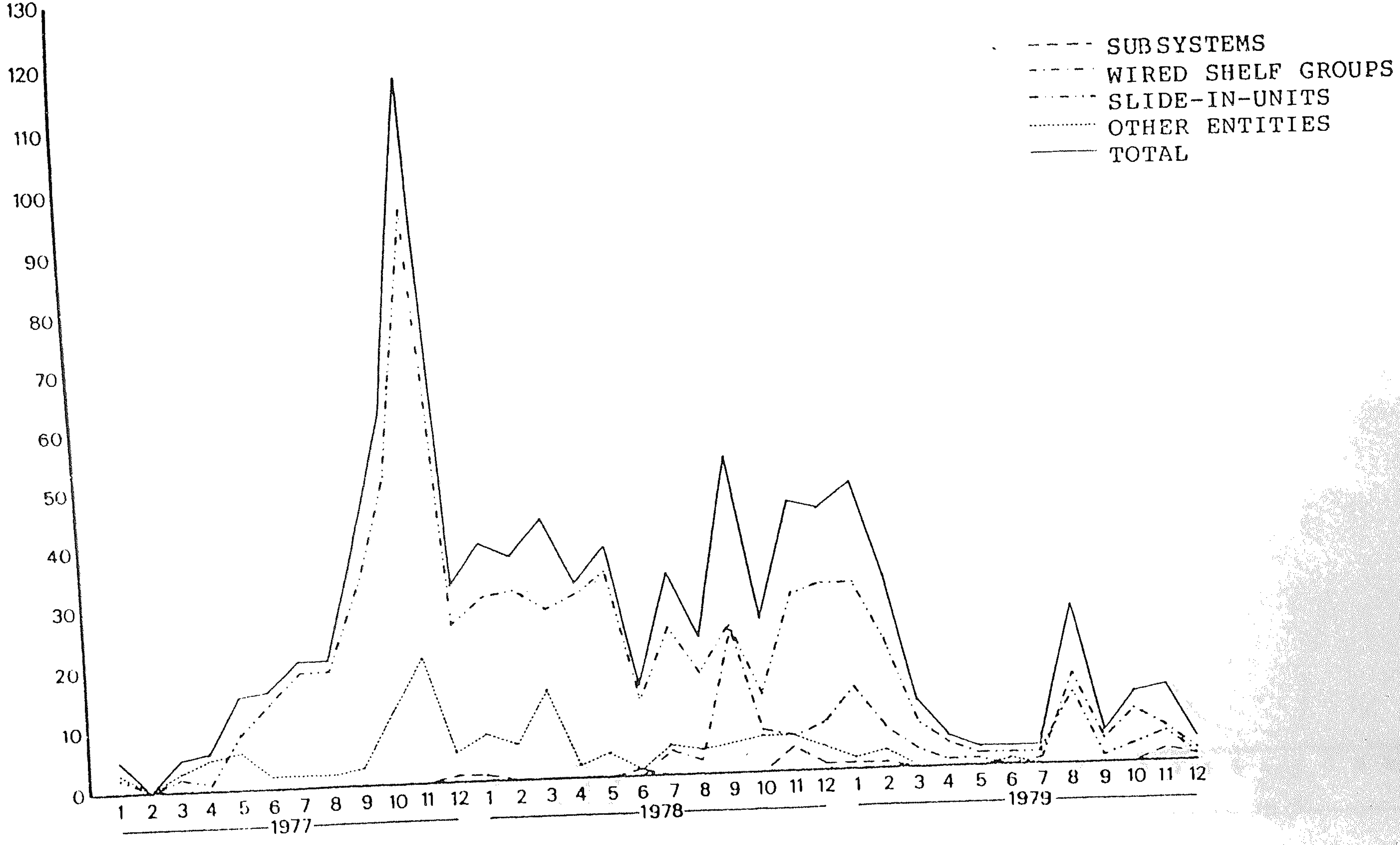


Figure 4-2. Number of Change Notes produced per Calendar Month for the SPU.

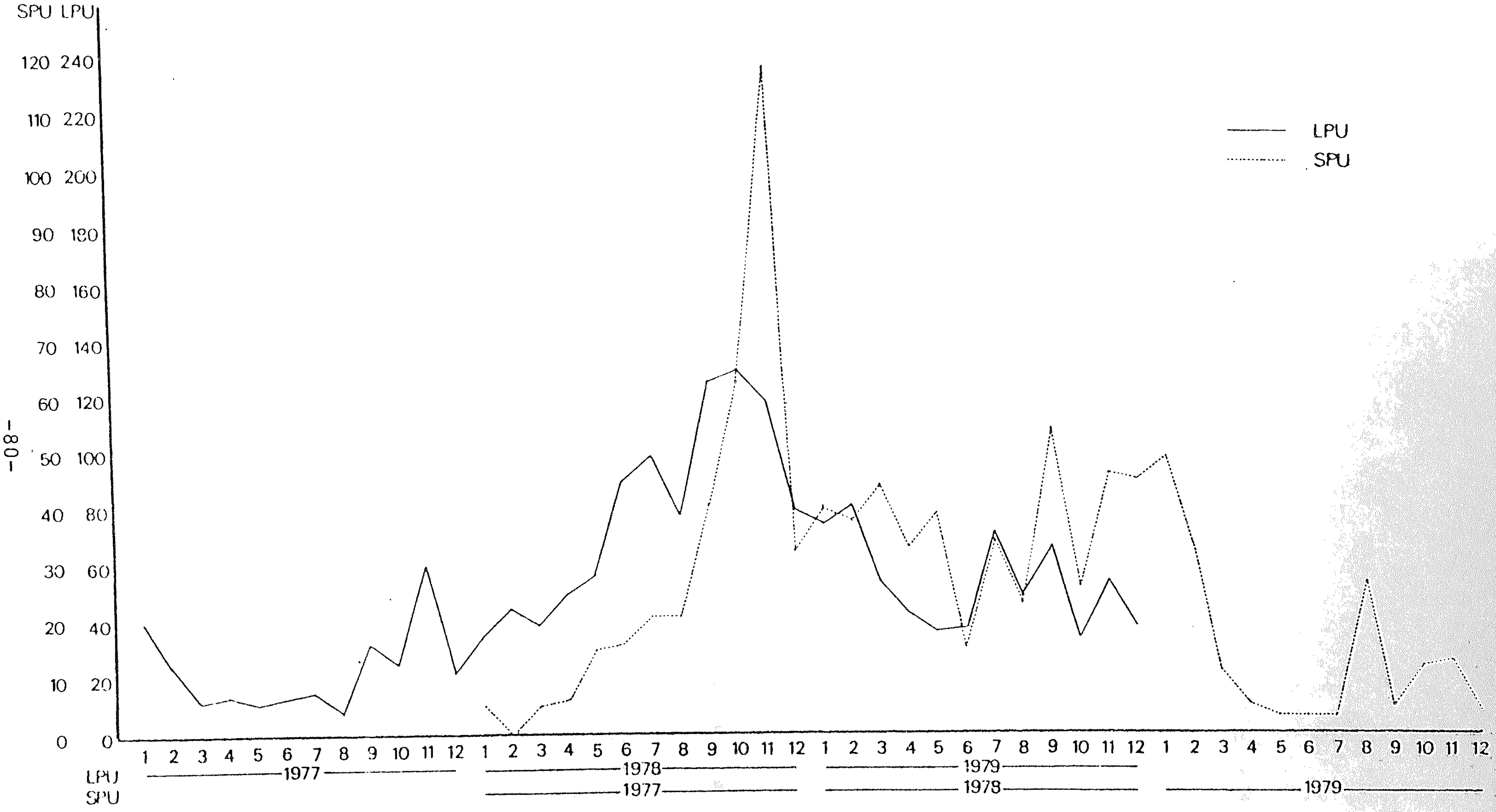


Figure 4-3. Comparison Between the Patterns of 'Change Notes' Produced for the LPU and SPU.

commence. It represents the obvious faults found at 'switch-on'.

(b) The 'Change Note' production rates for the LPU and SPU projects appear to have cyclic patterns, each with a one year period and a peak occurring about October.

(c) The 'Change Note' production rate for the LPU shows a peak for the period of July to November 1979. It has been alleged by the LPU project management that this was due to the transfer to the project commissioning team of an extremely efficient and proficient Engineer.

(d) The production rate for 'Change Notes' referring to subsystems of the LPU shows a decline as the project progresses. This was due to changes being generated as a result of faults found when the Slide-in-units were being specified and designed, rather than during commissioning.

The production rate of subsystem 'Change Notes' of the SPU was significantly lower than for the LPU, even taking into account the different sizes of the products.

(e) The patterns of the production rate of 'Change Notes' for Slide-in-units for the two projects are similar in shape. However, the main peak for the SPU is sharper than for the LPU.

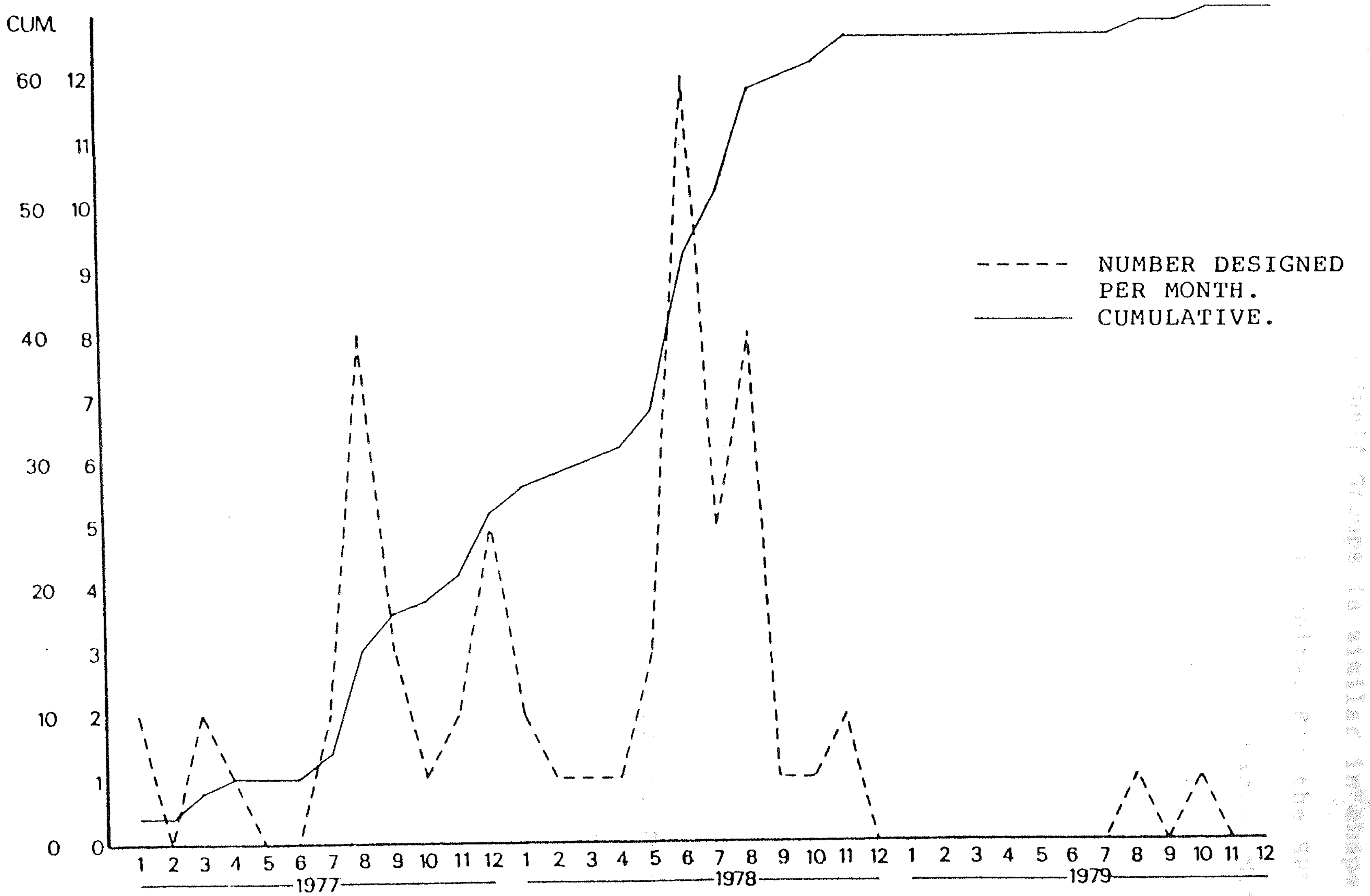


Figure 4-4. The Number of LPU Slide-in-units for which the Design was completed Per Calendar Month.

(f) The pattern of the 'Change Note' production rate for LPU Wired Shelf Groups is similar in shape to the pattern for the LPU Slide-in-units. For the SPU the production rate for Wired Shelf Group 'Change Notes' reached a peak 18 months after the peak for the SPU Slide-in-units. This was because the first SPU models were built to an established 'Equipment Practice' which had no Wired Shelf Groups. The Slide-in-units plugged directly into Racks, the changes for which were classified under the category, "Other Assemblies".

(g) The production rate for 'Change Notes' referring to "Other Assemblies" for the LPU is very small throughout the period under scrutiny. For the SPU the rate is significant, particularly early on. This is explained by the use of the established 'Equipment Practice' for early models. See "(f)", above.

(h) The total number of 'Change Notes' and their percentage breakdown into the four categories, Subsystem, Slide-in-units, Wired Shelf Group and "Other Assemblies", are shown in Table 4-2.

Note:

The preceding discussion and the patterns shown in Figures 4-1 to 4-3 have been concerned with quantities of 'Change Notes'.

	NUMBERS OF CHANGE NOTES (PERCENTAGE OF TOTAL)	
	LPU	SPU
SUB SYSTEM	237 (11.5)	13 (1.4)
SLIDE-IN-UNIT	1463 (71.1)	686 (72.3)
WIRED SHELF GROUP	317 (15.4)	114 (12.0)
OTHER	40 (1.9)	136 (14.3)
TOTAL	2057	949

Table 4-2. Total Number of 'Change Notes' Produced for the LPU and SPU.

Some 'Change Notes' refer to more than one design change but some others merely contain comments which involve no design change.

It will be shown in Section 4.6 that there is, on average, approximately one Change per 'Change Note'. The preceding discussion, therefore, applies as well to the pattern of Change generation as to the pattern of 'Change Note' production.

4.4. Rates of Production of Changes for Individual LPU Slide-in-units.

This analysis is included for three reasons:

1. There was a greater number of Slide-in-unit types than there was of any other category of assemblies. Also, the majority of 'Change Notes' were written for them.

2. The analysis would contribute to an understanding of the development process.

3. Later in this Chapter it is necessary to compare the 'Change Note' production levels for different Slide-in-units, and the analysis given here enables this to take place meaningfully. It would have been misleading to compare the total number of 'Change Notes' for each Slide-in-unit as they would have been commissioned to differing extents.

NOTE: SPU Slide-in-units were not included in the analysis because the results would have been confused by the change of Equipment Practice on the SPU.

In order to compare the level of changes for individual Slide-in-units and for averaging purposes, the 'Design Complete' Date was chosen as the 'origin' of the "development time" scale. For each Slide-in-unit the number of 'Change Notes' produced in constant length periods before and after this date were counted.

Method.

'Change Notes' Information.

Information about the 'Change Note' was extracted from the Computer File described in Section 4.3.

'Design Complete'.

'Design Complete' for each Slide-in-unit was defined as the date when the Printed Circuit Board artwork was produced. Although further documentation would be produced after this date it was chosen for the following reasons:

- It was a date which was well documented in terms of precision and reliability.

- It was readily checkable, being recorded on two independent documents.

- It represented the date of the first opportunity that a Slide-in-unit could be built.

Calculations.

A computer program assigned each 'Change Note' to the appropriate period for the respective Slide-in-unit. It also calculated the mean number of changes produced in each period about 'Design Complete' for all LPU Slide-in-units (excluding those specified in this Section).

Number of Slide-in-units Used to Calculate Each Period Mean.

The mean number of 'Change Notes' produced for each period was calculated independently.

The total number of Slide-in-units (except those excluded from the analysis) was used to calculate the average 'Change Note' Production for periods before 'Design Complete'. This arguably gives results which are too low for those periods because the 'Change Note' system was not operative for the earliest periods of the first units designed.

Slide-in-units Excluded from the Analysis.

The following two categories of Slide-in-units were excluded from the analysis:

1. Slide-in-units designed for the SPU by the SPU design team, but which were also used on the LPU. There were five such units.

2. Slide-in-units which had not been designed by the end of 1979 and, therefore, for which design completion dates could not be given. There were two such units.

Means Calculated.

When the means were first calculated, the peak average production rate of 'Change Notes' was shown to occur about 12 months after 'Design Complete'. It was alleged (by those developing the LPU) that this time lapse could be attributed to the first Slide-in-units to be designed (designed before April 1978). After the design had been completed, they had remained untested and uncommissioned for a considerable length of time.

To test this hypothesis three separate means were calculated for each period:

- A mean for Slide-in-units designed before April 1978,
- A mean for Slide-in-units designed after March 1978,

- A mean for all LPU Slide-in-units.

Results.

The graphs presented in Figures 4-5, 4-6, and 4-7, show the mean number of 'Change Notes' produced in 1, 2 and 3 month periods about 'Design Complete', respectively, for the LPU Slide-in-units.

Figure 4-8 shows a graph of the cumulative mean number of 'Change Notes' produced for LPU Slide-in-units.

Figure 4-9 shows the number of Slide-in-units used for calculating the mean number of 'Change Notes' produced for each period.

Observations.

1. Figures 4-5, 4-6, 4-7 and 4-8 show 'Change Notes' generated for Slide-in-units, before design has been completed for those units. These 'Change Notes' mainly describe changes to the Slide-in-unit specifications, or are due to changes to design standards.

2. The graphs showing the average number of 'Change Notes' produced in 1 month periods (Figure 4-5) exhibit a considerable level of "noise" on the curves. This is

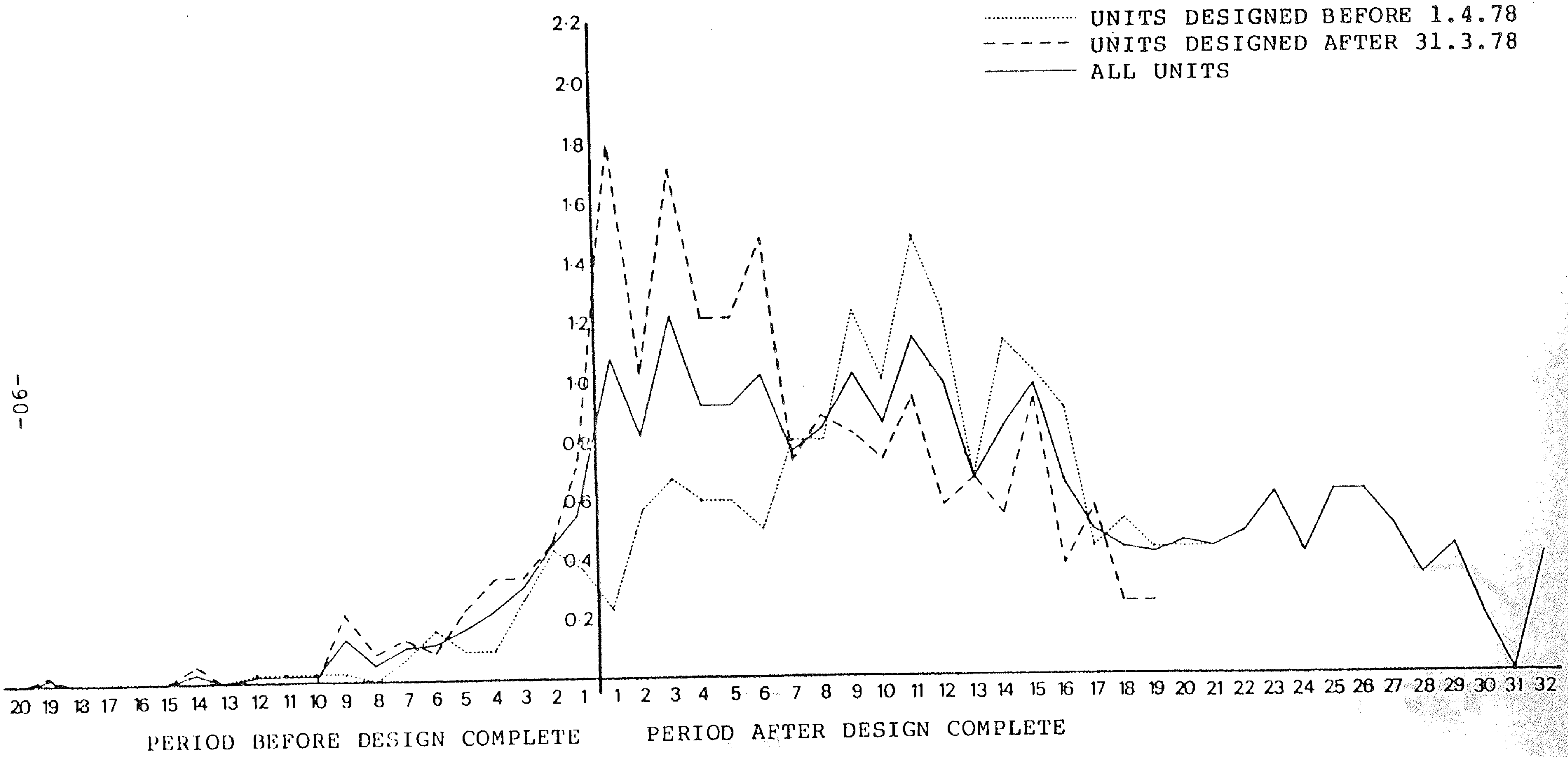


Figure 4-5. Mean Number of 'Change Notes' Produced for LPU Slide-in-units in One Month Periods about 'Design Complete'.

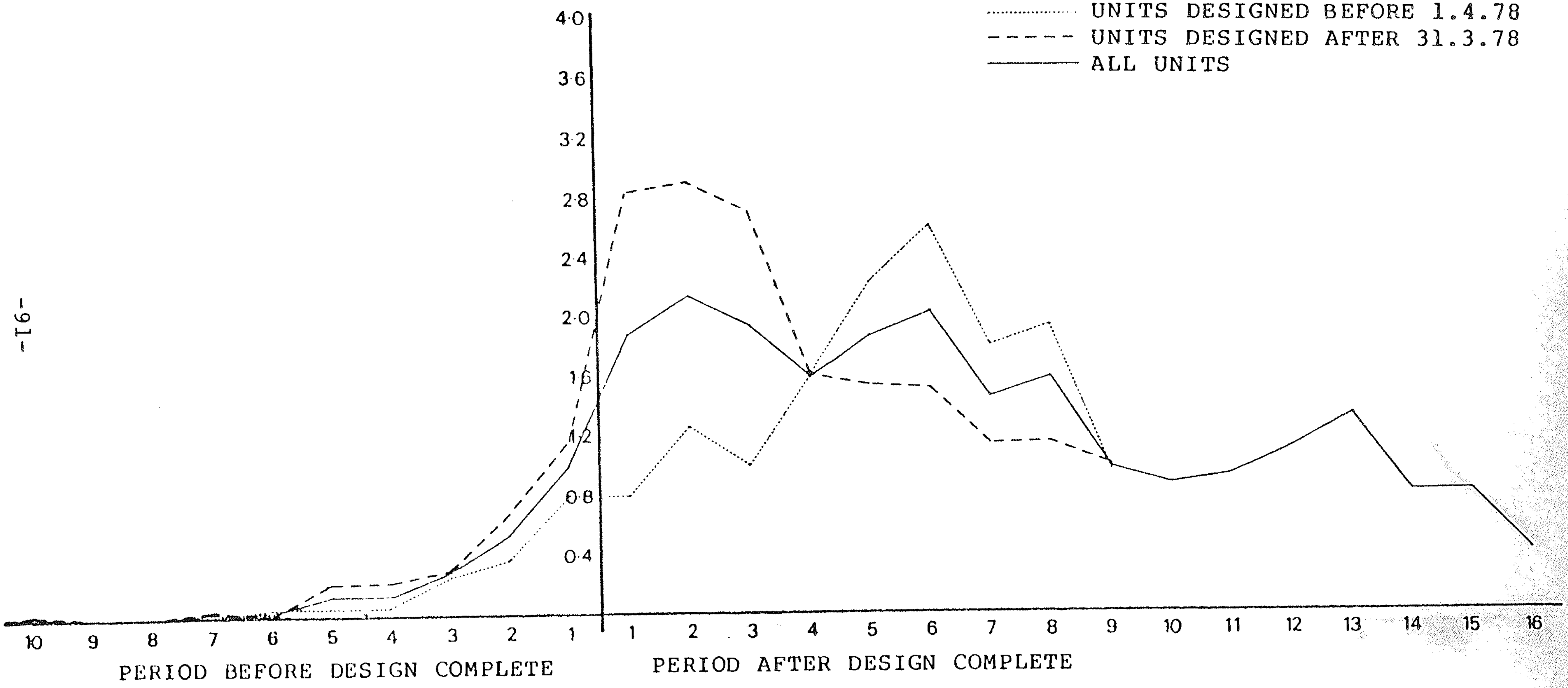


Figure 4-6. Mean Number of 'Change Notes' Produced for LPU Slide-in-units in Two Month Periods about 'Design Complete'.

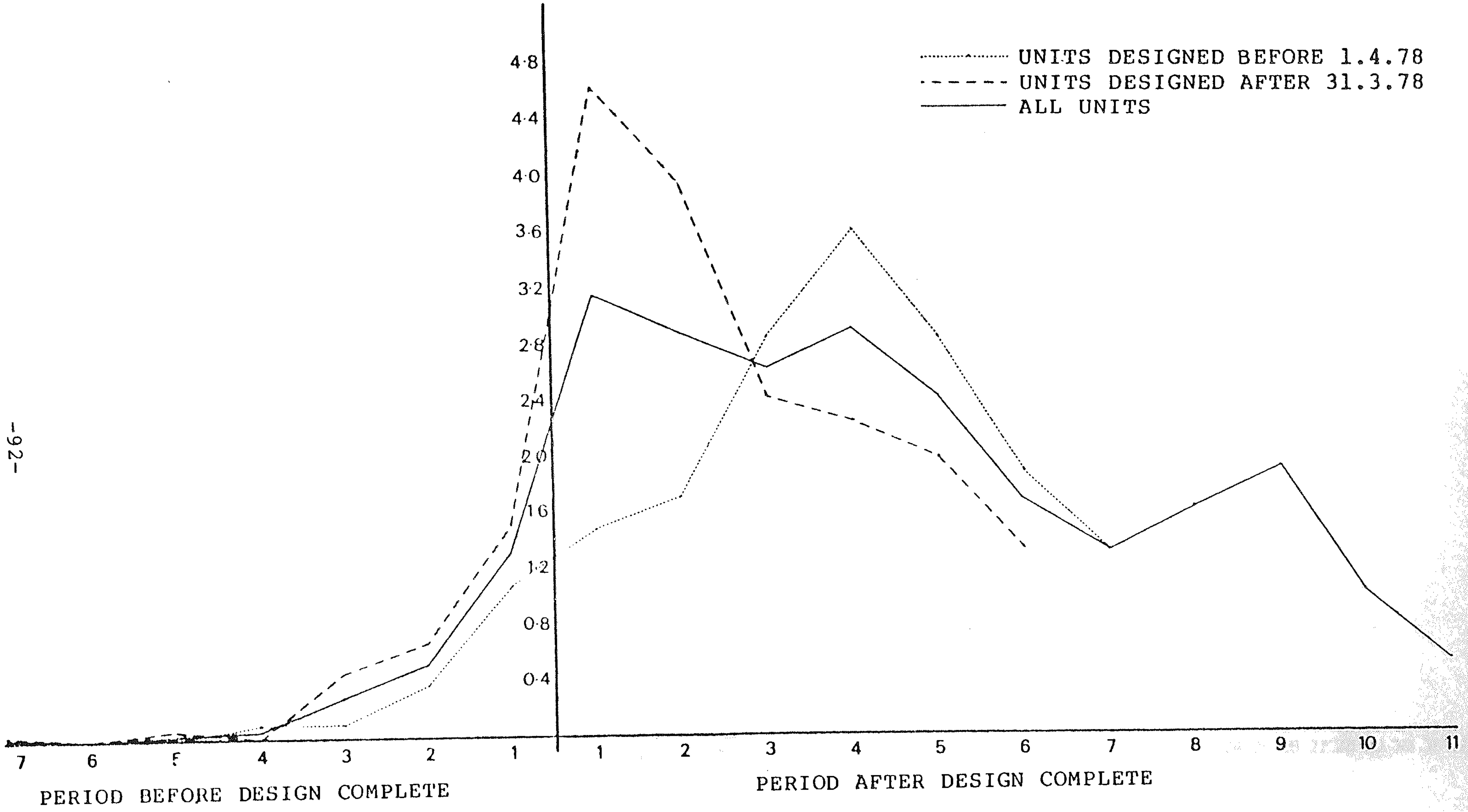


Figure 4-7. Mean Number of 'Change Notes' Produced for LPU Slide-in-units in Three Month Periods about 'Design Complete'.

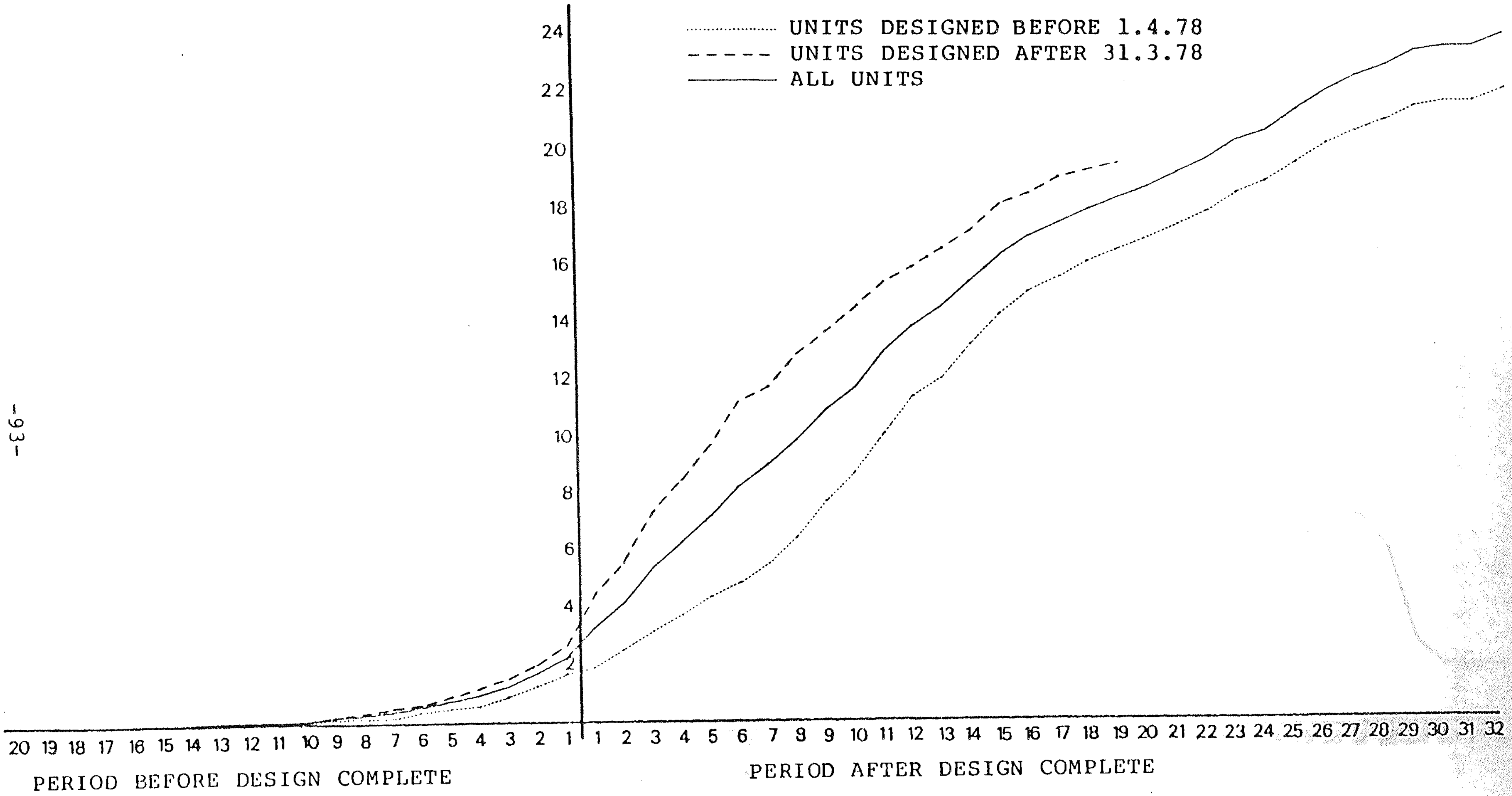


Figure 4-8. Cumulative Mean Number of 'Change Notes' Produced for LPU Slide-in-units.

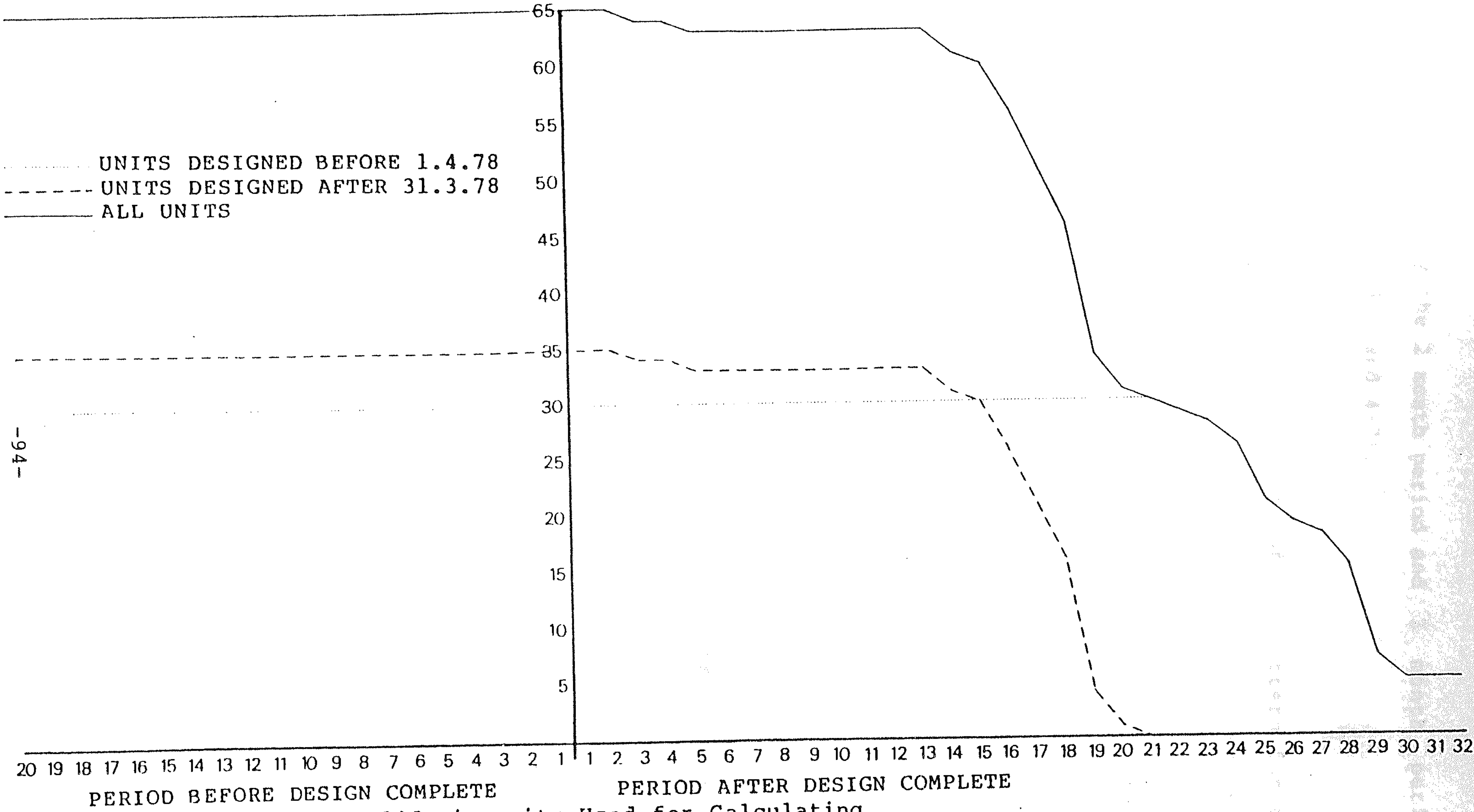


Figure 4-9. Numbers of Slide-in-units Used for Calculating the Means of the Number of 'Change Notes' Produced in Periods about 'Design Complete'.

smoothed out by the 2 month period and 3 month period graphs (Figures 4-6 and 4-7).

3. The main characteristic of the pattern for all the units is the presence of three peak production periods:

The first peak, which appears about 3 months after design completion, is due to units for which 'Design Complete' was after March 1978. It represents when commissioning started for these Slide-in-units, and a large number of comparatively obvious faults were found.

The second peak (11 months after design completion) is due to Slide-in-units for which 'Design Complete' occurred before April 1978. This is also due to the first full commissioning of these units.

The third peak is also due to Slide-in-units which were designed before April 1978. Examination of the unit types which contributed to this peak showed that the 'Change Notes' responsible for the peak were generated round about November 1979. This corresponds with the second peak shown in Figure 4-1, which was attributed to the introduction of a new proficient Commissioning Engineer.

4. Figure 4-8 presents the cumulative production of 'Change Notes'.

Although it shows a slowing down in the production of 'Change Notes', it is difficult to draw any conclusions about whether there will be any difference between the total mean number of 'Change Notes' which would eventually be written for Slide-in-units of the two groups.

4.5. Immediate Causes of Design Changes.

The 'immediate cause' of a design change (referred to in Appendix G as the "Cl Cause") is that which indicates to the development engineer that the change is necessary. It is not usually the ultimate cause. Since one approach to reducing the development time would have been to reduce the number of design changes, it was necessary to understand their immediate causes. To do this a study, which is described below, was carried out.

Method.

For a period of three months, all 'Change Notes' for the SPU and LPU hardware were examined to determine their immediate causes. As the SPU project was about a year ahead of the LPU, it enabled the development process to be sampled at two separate stages. The number of 'Change Notes' examined were as follows:

LPU 'Change Notes':	288
SPU 'Change Notes':	93

The immediate cause of each change was determined by consultation with the appropriate project Team Leader. The causes were then divided into a number of categories (See Table 4-3). Some of these categories had been anticipated before the study, but others were created during it to cover causes not previously encountered.

The list of categories was not shown to the Team Leaders, to avoid bias, but as the study progressed they began to use a fairly clear set of expressions to describe the causes. This probably occurred because of probing and questioning by the researcher.

The reason for determining the causes in this manner was that the person who did the interpretation had to have an overall view of the product system and subsystems, but also had to understand the detail design. Only the Team Leaders had this perspective. They had designed the subsystems, written the specifications for the Slide-in-units and had designed the Wired Shelf Groups.

Design Engineers were occasionally consulted to clarify certain points.

Disadvantage of the Method Used.

The potential inaccuracies which might have occurred using this method arise because of difficulties in determining some of the causes, and because of the fuzziness between the different categories. This meant that some of the judgements which had to be made were largely subjective and, therefore, prone to bias. The Team Leader would be inclined to protect himself.

To overcome this difficulty, two other techniques offered themselves; the use of Assessment Panels and the use of Delphi Techniques. Both these methods were, however, impractical because of the problems of finding people with suitable detailed and over-view knowledge of the products and of the difficulties in persuading them to give up their "valuable time" to participate.

Results.

By the end of the study, 49 categories of immediate causes of changes had been used. Some of these were mutually exclusive, but some augmented others. They are listed in Table 4-3.

MUTUALLY EXCLUSIVE CAUSES

Slide-in-unit specification fault.
Slide-in-unit specification incomplete.
Slide-in-unit specification ambigious.
Slide-in-unit specification change.
Subsystem specification fault.
Subsystem specification incomplete.
Subsystem specification ambigious.
Subsystem specification change.
System design fault.
Subsystem design fault.
Logic design fault.
Wiring design fault.
Circuit design fault.
Hardware realisation fault.
Technology change.
Standards.
Faults in rework.
Logic design review.
Commissioning fault.
Results of rework.
Clerical error.
Documentation control Error.
Information processing system fault.

CAUSES AUGMENTING OTHERS

Hardware change to match software.
New requirement.
Facility enhancement.
Change to allow commissioning.
Low level designed before high level.
Tracking error (In CAD).
Equipment Practice
Error generated in CAD by Engineer.
New component available.
Standards change.
New standard introduced.
New component available.
Standards change.
New standard introduced.
Standards lacking.
Fault in MAP.
Failure mode & effects analysis.
Logic simplification.
Micro-program coding
Data fault.
Incorrect change from commissioning
Change to change note.
Fault in writing wiring information.
Fault in writing change note.
Punch operator error.
Digitising error.
Incorrect reading of document.
Change note missed in rework.
Change note generated during rework.
Fault involving CAD.

Table 4-3. Categories of Causes of Design Changes.

Each of the mutually exclusive categories is discussed below:

Design Faults: These are defined as faults which occur within the design activity, and which cause the equipment to behave in a manner different from the way it was specified to behave. They appear at different levels within the equipment structure and are sub-categorised into the following types:

- System Design Faults,

- Subsystem Design Faults,

- Logic Design Faults, which occurred at Slide-in-unit level during the logic design,

- Hardware Realisation Faults, which arise during the design transformation from a circuit design to a physical design (printed circuit board layout and tracking),

- Wiring Design Faults, which are faults made in drawing up Wired Shelf Group wiring connections.

Specification Faults: These were divided into four types, each of which may occur at any of three levels within the equipment structure. The four types were:

- Errors, where the specification was wrong,
- Changes, where the specification was changed to accommodate some other change,
- Incomplete, where the specification did not give sufficient information for the design to be carried out,
- Ambiguous, where the specification could be interpreted in more than one way.

Clerical Errors: These were faults which could be attributed to simple manual copying errors.

Faults in Rework: These arose because of errors made in incorporating other changes into the design during a rework.

Standards: These changes were caused either because the standards were altered, or because a standard did not exist and insufficient guidance had been given to the Design Engineer.

Logic Design Reviews: These occurred because of investigations carried out into the logic design.

Documentation Control Errors: These errors were caused by incompatible documents being used during the process.

Information Processing System Faults: This category of faults represents malfunctions which occur within computer information processing systems.

Technology Changes: These resulted in design changes which were made to utilise new, better technology or components, or because previously used technology or components had become unavailable.

Commissioning Faults: These were faults to changes which had been specified by the Commissioning Engineers.

Results of Rework: These occurred because of changes made to other parts of the product.

The percentage breakdown of the changes into these mutually exclusive categories is exhibited in Table 4-4.

Comments on Results Shown in Table 4-4.

The main conclusions and discussion about the findings will be left until Chapter 7. However, some brief comments are made here:

CAUSE OF CHANGE	LPU		SPU	
	NUMBER OF CHANGES	PERCENTAGE OF TOTAL	NUMBER OF CHANGES	PERCENTAGE OF TOTAL
Design Faults[1]	133	49.8	37	48.7
Specification Faults[2]	70	26.2	8	10.5
Clerical Errors	35	13.1	19	25.0
Faults in Rework	8	3.0	5	6.6
Standards	7	2.6	2	2.6
Logic Design Reviews	4	1.5	2	2.6
Documentation Control Errors	4	1.5	1	1.3
Information Processing System Faults	4	1.5	-	-
Technology Changes	1	0.4	-	-
Commissioning Faults	1	0.4	-	-
Changes Resulting from Rework	-	-	2	2.6
[1]The 'Design Faults' were made up of:				
System Design	1	0.4	-	-
Subsystem Design	2	0.7	-	-
Logic Design	90	33.7	20	26.3
Hardware Realisation	22	8.2	9	11.8
Wiring Design	16	6.0	7	9.2
Circuit Design	2	0.7	1	1.3
[2]The Specification Faults were made up of:				
Subsystem Specification Faults	27	10.1	4	5.3
Made up of:				
Errors	13	4.9	1	1.3
Changed Specification	9	3.4	3	3.9
Incomplete Specification	5	1.9	-	-
Ambigious Specification	-	-	-	-
Slide-in-unit Specification Faults	43	16.1	4	5.3
Made up of:				
Errors	26	9.7	1	1.3
Changed Specification	11	4.1	3	3.9
Incomplete Specification	3	0.1	-	-
Ambigious Specification	3	1.1	-	-

Table 4-4. Summary of Immediate Causes of Changes for LPU and SPU Projects.

I. The most significant causes of design changes are 'Design Faults'. For both the LPU project and the SPU project about half the changes were caused by 'Design Faults'.

II. The most significant type of 'Design Faults' were 'Logic Design Faults', which accounted for about a third of all changes for the LPU and slightly less (26%) for the SPU.

III. For the SPU, after 'Design Faults', the next most significant cause of changes was 'Clerical Errors' and this accounted for 25% of changes. For the LPU it accounted for 3%. The difference was due to the SPU being more advanced, and at a stage where more documentation was being produced.

IV. 'Specification Faults' were more significant for the LPU (26.2%) than for the SPU (10.6%). Again, the difference is explained by the lead of the SPU project over the LPU project, and the 'Specification Faults' having been previously eliminated for the SPU.

V. Of the 'Specification Faults', more occurred at Slide-in-unit level than at subsystem level (none at system level), but no comment can be made about the relative effects of those faults.

VI. The other categories accounted for a small percentage of the total number.

Number of Changes per 'Change Note'.

The analysis and results discussed in this Section were based on numbers of 'Change Notes'. Not all 'Change Notes', however, contained one change; some merely recorded comments but some others contained more than one change. In order for the results discussed in Section 4.3 to be meaningful, it is necessary to know the relationship between the number of changes and the number of 'Change Notes'. It was possible to extract figures for this while undertaking the study to establish the immediate causes of design changes.

For LPU: 288 'Change Notes' resulted in 267 changes (i.e. 0.93 Changes per 'Change Note').

For SPU: 93 'Change Notes' resulted in 76 changes (i.e. 0.82 changes per 'Change Note)'.

A greater proportion of SPU 'Change Notes' referred to earlier changes than did those of the LPU.

4.6. Causes of 'Design Faults'.

The previous Section showed that the most significant cause of design changes was 'Design Faults', and, in particular, 'Logic Design Faults' at Slide-in-unit level.

Table 4-5 lists a number of factors which could be responsible for causing the level of 'Design Faults', and, in particular, 'Logic Design Faults' which were encountered.

It would have been desirable if at this stage of the project the significance of each of these in causing the 'Design Faults' could have been measured. For the reasons which are given below this was not possible:

- The number of factors was large and it was not known which were the most important,

- Many of the measurements would have been concerned with psychological or social factors which are extremely hard, if not impossible, to measure accurately.

Although it was not possible to measure the significance of each of the factors, an opinion study was performed with the co-operation of the development

Human Factors:

Selection of Engineers,
Training given to development staff,
Low Motivation,
Inappropriate Working Conditions,
Management Structure,
Information/Human Interface.

Procedures:

Inadequate System Specification,
Method of Logic Design,
Methods of Design Checking,
Documentation Production,
Engineer following his own work through,
Commissioning Team Separate from Design Team,
Method of Building Prototype.

Design Aids and Facilities:

CAD too difficult to use,
The use of aids requiring too much training,
Access to Computer Facilities,
Lack of Logic Design Aids.

The Product:

Structure of the Product,
Size of the Product Building Blocks,
Type of Technology Used,
Equipment Standards.

General:

Social Environment,
Unionisation,
Job Demarcation.

Table 4-5. Possible Causes of Design Faults.

staff to make an assessment of their importance. This is described in Chapter 6.

One study was, however, performed to see if one of the most readily measurable factors was significant in causing changes. This was the "experience of the Design Engineer". The study is described in Section 4.7. An attempt was also made to establish the effect of Slide-in-unit complexity on the number of changes. It was not possible to find a measure of complexity which gave a significant correlation. This may have been because:

- The relationship between complexity and number of changes was non-linear,

- The complexity was not an important cause of design changes, <u>or</u>

- The correct measure of complexity was not made. (The following factors were used:

  Number of Signal Nodes,
  Number of Active Devices
  Number of Passive Devices
  Number of Integrated Circuit Packs
  Number of Integrated Circuit Pins.)

4.7. Effects of Design Engineer Experience on Number of Changes.

A simple Linear Regression technique was used to see if a linear relationship existed between the number of changes generated for each LPU Slide-in-unit and the design experience of the respective logic Design Engineer.

"Experience" was defined as the number of months that the Engineer had spent designing logic. The information was collected in the first instance from work records. This, however, only yielded the experience figures of Engineers who had joined the Company as inexperienced graduates within the previous two years. Information about the remainder of the Engineers was gleaned from the Manager responsible for the LPU project. In a few cases, the Engineers themselves were consulted.

In order to overcome the effects of the Slide-in-units being designed at different times, and of having had different amounts of commissioning done on them, the number of changes was counted for each one for a fixed period of 9 months after 'Design Complete'.

Four groups of Slide-in-units were excluded for the study:

- Those for which there was more than 1 Designer (Number: 9),

- Those which were copies or very similar to other Slide-in-units (Number: 10),

- Those for which the designers were not known (Number: 3),

- Those which were designed under the SPU project (Number: 5),

- Those designed less than 9 months before the study was carried out (Number: 3).

Results of Analysis.

A plot of the number of 'Change Notes' produced in the 9 months after 'Design Complete' for each Slide-in-unit, against the experience of the respective designer, is shown in Figure 4-10.

There is no significant correlation between these factors (Correlation Coefficient = 0.02)

Possible Explanations for Low Correlation.

- The experience range may not have been wide enough. Only Engineers with experience of up to about

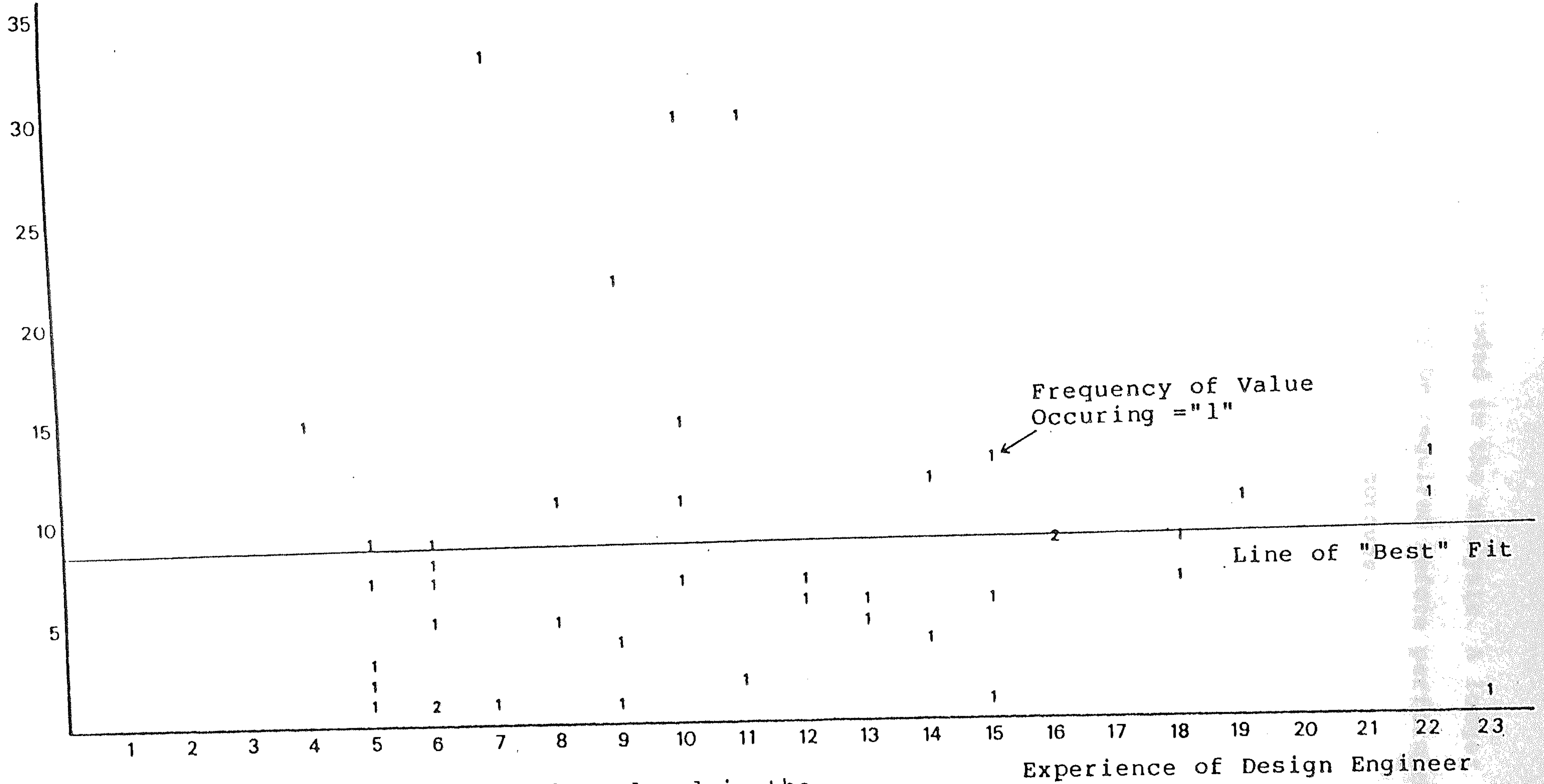


Figure 4-10. Number of 'Change Notes' Produced in the 9 Months after 'Design Complete' for Slide-in-units, against the Experience of their Respective Designers.

two years were included in the analysis. A longer length of time designing may be required before performance, in terms of quality of design, improves.

- The ability to undertake logic design may not be affected by experience and practice. It may be a function of aptitude.

- The effects of experience may have been swamped by other factors.

CHAPTER 5

CAUSES OF DESIGN FAULTS

5.1. Chapter Preview.

Chapter 4 showed that the most significant cause of design changes, in the case of the two System X processors, was Design Faults. The purpose of this Chapter is to identify the reason for this.

It does this by describing an opinion survey (questionnaire) which was carried out and by presenting its findings.

After discussing the reasons for using the opinion survey (Section 5.2), and the objectives of the study (Section 5.3), a description is given of each practical stage of it:

- Design of the questionnaire (Section 5.4),
- Selection of the respondents (Section 5.5),
- Pilot Study (Section 5.6),
- Main field work (Section 5.7),
- Processing of replies (Section 5.8).

The results are then presented (Section 5.9), a critique of the study given (Section 5.10) and the Chapter concludes with a Summary (Section 5.11).

5.2. Reasons For Performing An Opinion Survey.

The reasons for using an Opinion Survey to identify causes of Design Faults were:

- The number of possible causes was extremely large (see Table 4-5) and it would not have been possible to research them all in detail. Also , because nothing was known about the relative importance of each one, it would not have been possible to concentrate on just the most significant,

- Most of the suggested causes would have been extremely difficult to investigate scientifically.

It was, therefore, decided to question those who were involved with design within the Sponsoring Organisation, in order to establish whether a concensus of opinion could be reached as to the most important causes of Design Faults.

5.3. Objectives of the Opinion Survey.

The main objective of the opinion survey was to identify the causes of Design Faults and the relative importance of each one. There were, however, two further objectives:

- To find out if the causes of Design Faults were different for different projects,
- To find out if there was any variation between the causes of Design Faults for hardware, software and system design.

5.4. Design of the Questionnaire.

The present Section lists and discusses the principles adopted whilst designing the questionnaire. The details of the questions are left to the Sections on the pilot study (Section 5.6) and main field work (Section 5.7).

Possible Types of Questions.

Questions can be categorised into two types: 'open', where the respondent is left to give an answer in his own words, and 'closed', where answers are given in the question and the respondent has to choose the one he feels is most appropriate.

'Closed' questions have the following advantages:

- They are easier and quicker to answer,
- They are easier to code and to analyse the replies,

They do, however, also have the following disadvantages:

- They are harder to compose to avoid bias and to include all the possible answers,
- They do not allow the freedom or spontaneity of 'open' questions.

In the context of the study described here, the most important of these arguments was that the answers should be able to be easily coded and analysed. It was, therefore, decided that the main question ("What are the causes of Design Faults?") should be of the 'closed' type.

There are a number of different types of 'closed' questions, some of which are:

- Checklists,
- Ratings,
- Rankings,
- Inventories,
- Grids.

Oppenheim <22> lists the advantages and disadvantages of each of these.

Problems Associated with the Wording of Questions.

The following problems had to be considered when devising the wording of the questions:

- 'Leading questions',
- 'Loaded words',
- 'Prestige Bias',
- Embarrassing questions,
- Ambiguity.

Each of these is again described by Oppenheim <23>.

Accuracy and Validity of the Answers.

Establishing the validity of an individual's replies can be achieved by asking check questions and searching for consistency within the answers given. For the purpose of this study, it was possible to ask alternative questions which looked at the subject from different perspectives.

Chosen Format of the Questionnaire.

It was decided that the questionnaire would include three questions:

- The first asked what are the causes of Design Faults. It was a 'closed' question and involved 'rating' the given answers on a scale from 0 (does not cause Design Faults) to 5 (very important cause of Design Faults).

- The second inquired about the responsibility for reducing the number of Design Faults and was aimed at providing, by inference, a check on the first question. It was a 'closed' question.

- The third asked what could be done that is not already done by each of the three groups, Design Engineer, Supervisors and more senior management to reduce the number of Design Faults. The question was of the 'open' type and had two purposes:

  1. To provide a validity check for the first question,

2. To increase the richness of the information provided by the respondents.

5.5. Selection of Respondents.

The following criteria were used for selecting the respondents:

- They had to be concerned with design, whether of hardware, software or systems. This included the Design Engineers, their Supervisors and their Managers (within the laboratories).

- The products with which they were involved had to be restricted to new electronic products. These included:

  - System X,

  - One private venture development,

  - A Circuit Techniques design group.

- The respondents had had to be working for the Sponsoring Company for one year or more, so they would have the necessary experience and knowledge.

These criteria led to a sample of 139. Their distribution across rank, project and hardware, software and systems design is shown in Figure 5-1.

5.6. Pilot Study.

The objectives of the pilot study were:

- To ensure that all the relevant answers were included in the 'closed' questions,

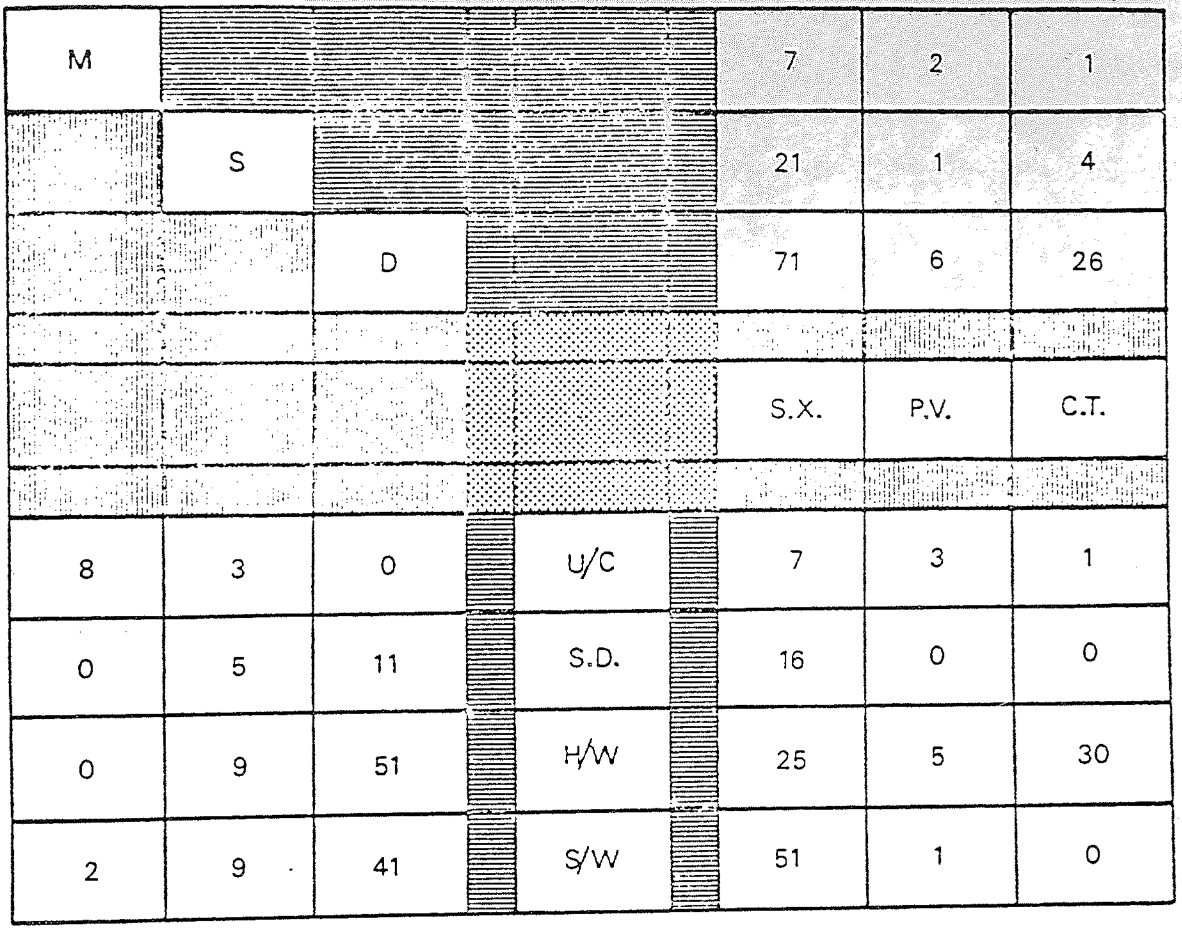
- To test the wording of the questions to eliminate leading questions, loaded words, prestige bias, and embarrassing questions,

- To test reactions to the questions,

- To ensure the ability to code the replies.

The pilot study was carried out on a random sample of nine people selected to represent a cross section of the complete sample. Their distribution across rank, project and hardware, software and systems design is shown in Figure 5-2.

The questioning was conducted through a structured interview with the questions being asked exactly as worded on the questionnaire.



M				7	2	1
	S			21	1	4
		D		71	6	26
				S.X.	P.V.	C.T.
8	3	0	U/C	7	3	1
0	5	11	S.D.	16	0	0
0	9	51	H/W	25	5	30
2	9	41	S/W	51	1	0

RANKS
M=Management
S=Supervisors
D=Design Engineers

EQUIPMENT RESPONSIBILITY
S.D.=System Design
H/W=Hardware
S/W=Software
U/C=Unclassified

PROJECT
S.X.=System X
P.V.=Private Venture
C.T.=Circuit Techniques

Figure 5-1. Distribution of Total Sample of Development staff Selected for Answering the Questionnaire.

Results of the Pilot Study.

It is not necessary for the actual answers, given in response to the questions, to be quoted here as only

M				1	0	0
	S			3	0	0
		D		4	0	1
				S.X.	P.V.	C.T.
1	0	0	U/C	1	0	0
0	1	1	S.D.	2	0	0
0	1	2	H/W	2	0	1
0	1	2	S/W	3	0	0

RANKS
M=Management
S=Supervisors
D=Design Engineers

EQUIPMENT RESPONSIBILITY
S.D.=System Design
H/W=Hardware
S/W=Software
U/C=Unclassified

PROJECT
S.X.=System X
P.V.=Private Venture
C.T.=Circuit Techniques

Figure 5-2. Distribution of Pilot Study Respondents.

their relevance to the objectives of the pilot study is important.

During the course of the pilot study, the following changes were made:

- After the first five interviews, the questionnaire was changed to include a further five choices in Question I. The wording of some of the choices was also slightly altered.

- At the end of the pilot study, an additional five choices were included in Question I, and again changes were made to the wording of some of the choices.

- During the pilot study, Question II contained a section asking for the reasons for the answers given. These provided no useful information and were omitted in the final version.

- The layout of Question II was altered for the final version to highlight the difference between each of the parts.

- The pilot study questionnaire contained a question which was removed for the final version because it provided no useful information. It was, "What measures do you now take to reduce the level of Design Faults made by you or your staff?".

The final questionnaire is exhibited in Appendix E.

5.7. Main Field Work.

The questionnaires were sent individually to each respondent at work with an accompaning letter which explained the purpose of the questionnaire, gave instructions for its return and included a guarantee of anonymity. Two days were given for answering and returning the questionnaires. The distribution, collection and chasing up of replies not received at the end of the two days were performed by two secretaries working for the development department.

Between the time that the pilot study was performed and the main field work carried out, the development department suffered an industrial dispute involving some of its staff. This had the effect of reducing the number of the potential respondents by twenty-six with some projects being hit worse than others. As it appeared that it would continue for a considerable time, and emotions would remain high for sometime after, it was decided not to postpone the exercise. It was, however, recognised that the accuracy of the study might be jeopardized by:

1. The smaller number of respondents,

2. Bias because Union members were largely excluded,

3. Emotional bias because of the dispute.

In addition to the reduction in potential respondents caused by the dispute, there was a further loss due to absence.

The final distribution of respondents is shown in Figure 5-3.

5.8. Processing of Replies.

The present Section is divided into four parts; the first concerned with information relating to each respondent, the other three with the replies to the three questions.

Information about Each Respondent.

The following information, pertinent to each respondent, was coded and put on a computer file:

- A unique number (used in place of the respondent's name in order to preserve anonymity),
- His rank (Design Engineer, Supervisor or Manager),

M				6	2	1
	S			13	1	3
		D		33	5	15
				S.X.	P.V.	C.T.
7	2	0	U/C	5	3	1
0	4	3	S.D.	7	0	0
0	5	31	H/W	13	5	18
2	6	19	S/W	27	0	0

RANKS

M=Management
S=Supervisors
D=Design Engineers

EQUIPMENT RESPONSIBILITY

S.D.=System Design
H/W=Hardware
S/W=Software
U/C=Unclassified

PROJECT

S.X.=System X
P.V.=Private Venture
C.T.=Circuit Techniques

Figure 5-3. Final Distribution of Respondents.

- Whether responsible for hardware, software, systems design or not specific,

- The project on which employed,

- Length of service with the Sponsoring Company.

Answers to Question I.

The ratings of the answers included in the questions were coded and put on a computer file. Statistical information about them was calculated using computer programs.

Other answers given under "Other, please specify" were processed manually, being sorted into groups of similar replies.

Answers to Question II.

Like the answers to Question I, these were coded for processing by computer.

Answers to Question III.

These were divided into groups of similar answers and recorded on a computer file for sorting and listing.

5.9. Results.

This Section exhibits the results of the questionnaire obtained from the main field work. Their overall validity is discussed in the Critique of the

Study (Section 5.10) but some comments are made here about the validity of individual answers.

The replies to each of the three questions are considered in turn.

Question I - Replies of All Repondents.

The means of the ratings given by all respondents to the answer choices of Question I are shown in Table 5-1, in the sequence of their given importance. They are also exhibited, diagrammatically, with their Standard Deviations in Figure 5-4. Figure 5-5 shows the results plotted according to the ranked order of the mean ratings and the ranked sizes of their respective Standard Deviations.

This latter figure allows some conclusions to be drawn about the relative validity of the results. If all the answers were rated with equal certainty, then ranking them in the sequence of their mean values would give the same result as ranking them in the sequence of their Standard Deviations: they would all lie on the diagonal line running from top left to bottom right. In practice, however, none of them lie on that diagonal. It can be argued that the confidence in the results for the answers to the left of the line should be higher than for the answers to the right of it, and that the former group of answers would be most valid. Accordingly, the

	ANSWER	MEAN RATING
10	The level of completeness of Specifications	3.9
19	The amount of experience of the Design Engineers	3.5
20	The methods used for design checking	3.4
1	The training given to the Design Engineers	3.1
6	The complexity of the Product Documentation Scheme	2.8
8	Motivation of the Design Engineers	2.8
9	The formats of Specifications	2.8
24	The level of communications between GEC, BPO and the other PF's	2.7
12	The insufficient use of Design Reviews	2.6
23	The level of communications between the design teams within the Development Department	2.6
5	The complexity of the Product Building Blocks	2.4
15	Working Conditions	2.2
11	The Methods used for Design	2.2
22	The level of communications within the design teams	2.2
4	The size of the Product Building Blocks (e.g.SIU's, S/W Processes)	1.9
7	The Managerial Organisation in the Development Department	1.9
16	The methods of selecting Engineers during recruiting	1.8
3	The accessibility of Design Standards	1.7
21	A lack of design team identity	1.6
2	Product Design Standards	1.5
13	The Computer Development Aids which are available	1.5
18	The general attitude to work in modern Britain	1.3
17	The technology employed in the product	1.3
14	The ease (or otherwise) of access to computer facilities	1.2

The figures down the Left-Hand side refer to identity numbers of the answer choices given in the Questionnaire (See Appendix E).

Table 5-1. Mean Ratings Given by all Respondents to Each Answer of Question I.

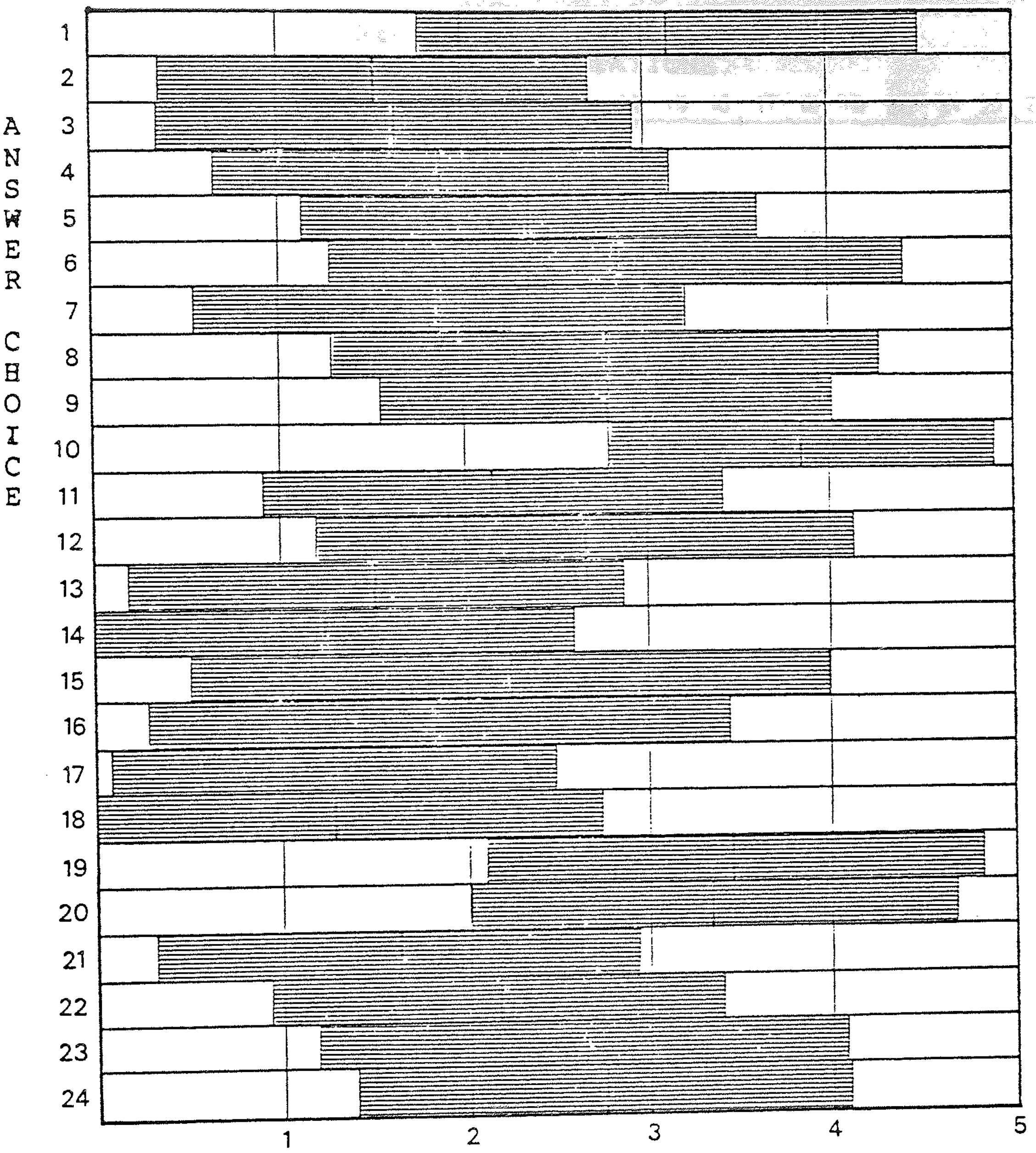
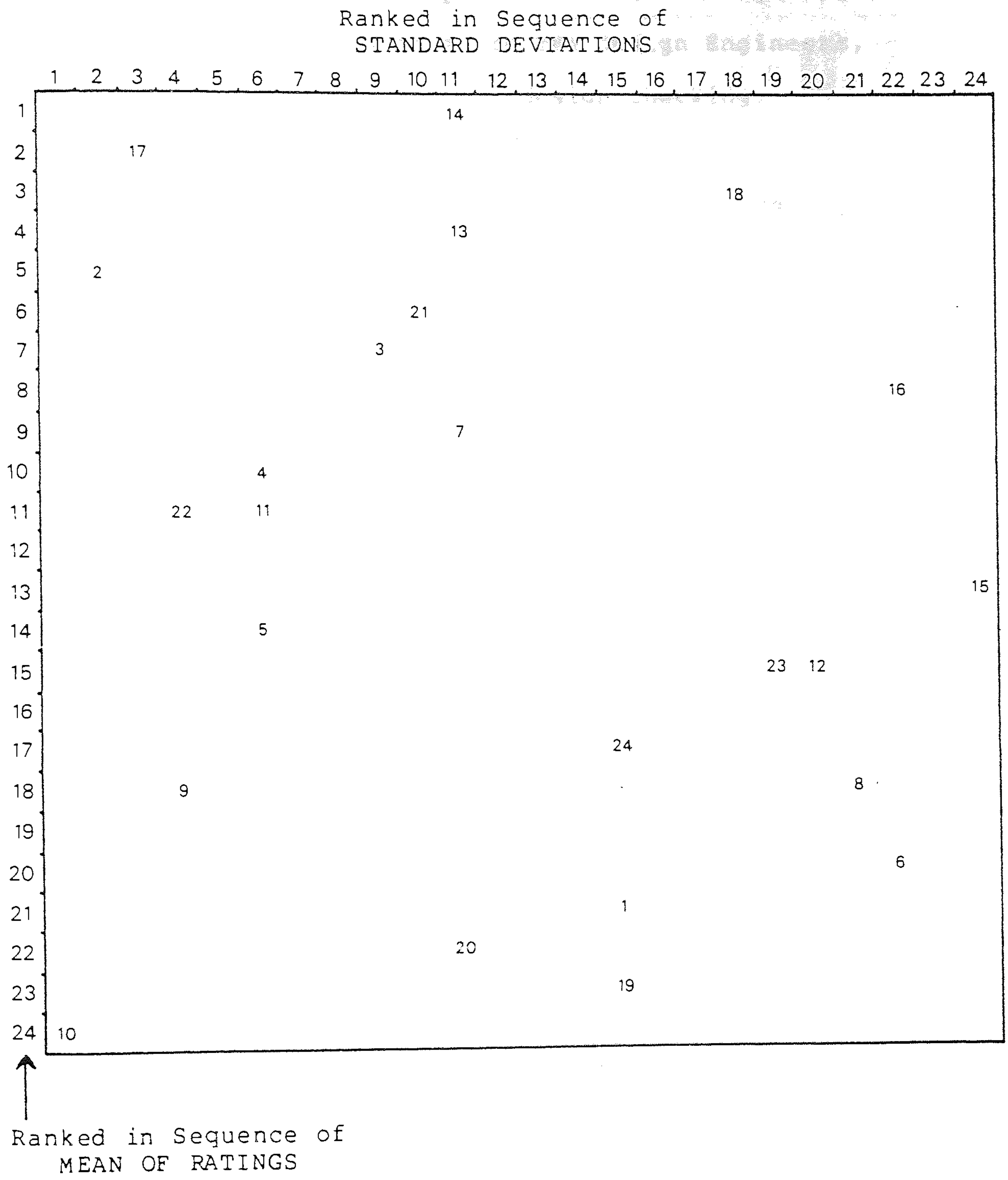


Figure 5-4. Mean Ratings and Standard Deviations Given by All Respondents to Each Answer of Question I.

validity of the results for the four causes of Design Faults, which were rated as being most important, all appear to have a relatively high validity. These were:

- The level of completeness of Specifications,



The figures within the matrix refer to the identity number of the answer choice given in the Questionnaire.

Figure 5-5. Answer Choices of Question I Ranked in Mean of Rating and Standard Deviation Sequence.

- The amount of experience of the Design Engineers,
- The training given to the Design Engineers,
- The methods used for design checking.

Question I - Answers Grouped Under Common Headings.

The answers included in Question I were categorised into six groups of like types.

The answers included in each group are defined in Table 5-2. The mean ratings given to each group, along with their Standard Deviations, are shown in Figure 5-6.

The mean ratings for the following pairs of groups are 'probably' significantly different:

-"Human Topics" AND "The Product"
(95% Confidence)
-"Environment" AND "Development Aids"
(99% Confidence)

Each pair of non-adjacent groups were at least 'probably' significantly different.

Question I - Answers Other Than Those Specified in the Question.

The following causes of Design Faults were given under "Other (Please Specify)":

	GROUPS					
	DOCUMENTATION	DEVELOPMENT METHODS	HUMAN TOPICS	THE PRODUCT	ENVIRONMENT	DEVELOPMENT AIDS
1. The training given to the Design Engineers			●			
2. Product Design Standards	●			●		
3. The accessibility of Design Standards	●	●		●		
4. The size of the Product Building Blocks (e.g.SIU's, S/W Processes)				●		
5. The complexity of the Product Building Blocks				●		
6. The complexity of the Product Documentation Scheme	●	●		●		
7. The Managerial Organisation in the Development Department			●		●	
8. Motivation of the Design Engineers			●			
9. The Formats of Specifications	●	●				
10. The level of completeness of Specifications	●	●				
11. The Methods used for Design		●				
12. The insufficient use of Design Reviews		●				
13. The Computer Development Aids which are available		●				●
14. The ease (or otherwise) of access to computer facilities		●				●
15. Working conditions					●	
16. The Method of Selecting Engineers during recruiting			●			
17. The technology employed in the Product				●		
18. The general attitude to work in modern Britain			●		●	
19. The amount of experience of the Design Engineers			●			
20. The methods used for Design checking		●				
21. A lack of design team identity			●			
22. The level of communications within the design teams		●	●			
23. The level of communications between the design teams within the Development Department		●	●			
24. The level of communications between GEC, BPO and other PF's		●	●			

Table 5-2. Grouping of Question I Answer Choices.

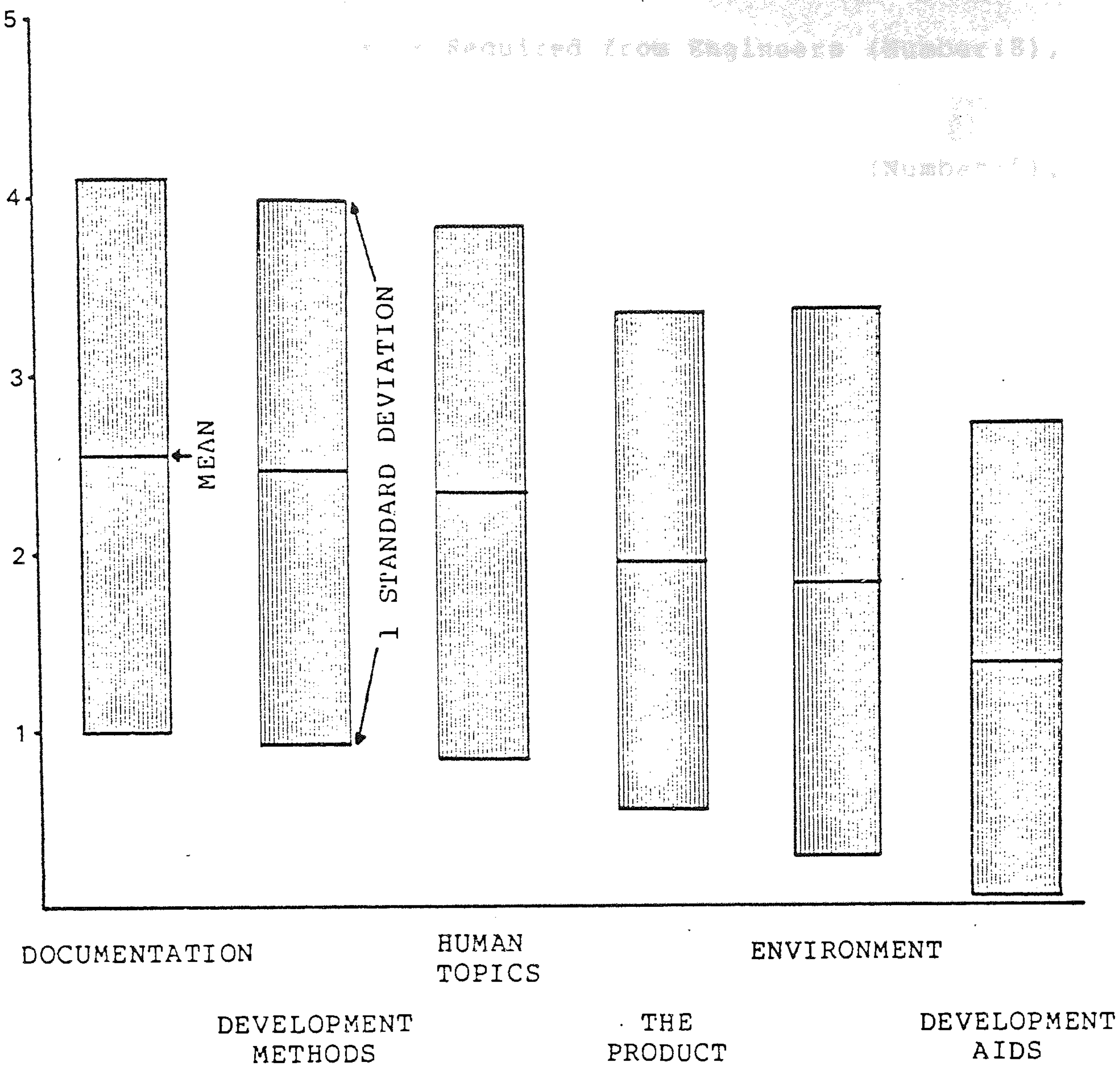


Figure 5-6. Mean Ratings of Question I Answers Grouped by Like Types.

- High Staff Turnover (Number:15),

- Insufficient Time Allowed (Number:12),

- Faulty or Changing Specifications (Number:9),

- Scope of Work Required from Engineers (Number:8),

- Design Team Organisation (Number:5),

- Lack of Training (Number:4),

- Industrial Relations Atmosphere (Number:3),

- And six other causes, each of which was recorded by only one or two respondents. Of those listed above, the following were covered, if only indirectly, by choices included in the Question:

- High Staff turnover (covered by "The amount of experience of the Design Engineers"),

- Faulty or Changing Specifications,

- Lack of Training.

The following, therefore, added to the choices given:

- Insuffícent Time Allowed,

- Scope of Work Required from Engineers,

- Design Team Organisation,

- Industrial Relations Atmosphere.

The most "popular" of these, however, was only mentioned by twelve respondents (15% of the total number).

The fact that only three people mentioned "Industrial Relations Atmosphere" probably indicates that the results were not greatly biased by the industrial dispute which was going on at the time the main field work was performed.

Question I - Differences Between Projects: System X/ Private Venture.

Table 5-3 shows the answer choices for which a significant difference was recorded between the responses given by members of the System X project and the Private Venture project.

Both these results could have been predicted because:

- System X was more complex and required greater attention to specifying than the Private Venture project,

ANSWER CHOICE	SYSTEM X MEAN RATING	PRIVATE VENTURE MEAN RATING	CONFIDENCE OF DIFFERENCE
10. The level of completeness of Specifications	4.1	3.0	>95%
24. The level of communication between GEC, BPO and the other Participating Firms	2.9	1.8	>95%

Table 5-3. Significant Differences between Responses of System X and the Private Venture Project Given for Question I.

- System X was a collaborative project between GEC, BPO and other participating firms whereas the Private Venture project was undertaken solely by GEC.

Question I - Differences Between Projects: System X/ Circuits Techniques Group.

Table 5-4 shows the answer choices for which a significant difference was recorded between the responses given by members of the System X project and the Circuits Techniques Group. Each of the differences is discussed below:

- Size of the Product Building Blocks: A difference might be expected as the Circuits Techniques Group deals mostly with simpler entities than the System X Design Engineers.

- The formats of Specifications: This might also be expected because System X was more complex and required specifying with greater accuracy.

- The level of completeness of Specifications: See above.

- The amount of experience of the Design Engineers: This is explained by a higher staff turnover within the System X project.

- The methods used for design checking: Design checking for the Circuits Techniques Group can be achieved more easily by building the circuit than for the System X development team.

ANSWER CHOICE	SYSTEM X MEAN RATING	CIRCUITS MEAN RATING	CONFIDENCE OF DIFFERENCE
4. The size of the Product Building Blocks	2.1	1.3	>95%
9. The formats of Specifications	3.0	2.2	>95%
10. The level of completeness of Specifications	4.1	3.4	>95%
19. The amount of experience of the Design Engineers	3.69	2.95	>95%
20. The methods used for design checking	3.51	2.74	>95%

Table 5-4. Significant Difference between Responses of System X and the Circuit Techniques Group Given for Question I.

Question I - Differences Between Ranks.

Table 5-5 shows the Question I answer choices for which significant differences were recorded between respondents of different rank. Each of the differences is discussed below:

ANSWER CHOICE	RANK	MEAN OF RATING	RANK	MEAN OF RATING	CONFIDENCE OF DIFFERENCE
12. The insufficient use of Design Reviews	Design Engineers	2.4	Others	3.1	>95%
16. The method of selecting Engineers During Recruiting	Design Engineers	1.5	Others	2.4	>95%
17. The amount of Experience of the Design Engineers	Managers	4.6	Supervisors	3.5	>95%
	Managers	4.6	Design Engs.	3.3	>99%

Table 5-5. Significant Differences between Responses given for Question I by Staff of Different Ranks.

- The method of selecting Engineers during recruiting: This difference may be caused by prestige bias,

- The amount of experience of Design Engineers: Like the previous difference, this one may also be caused by prestige bias,

- The insufficient use of design reviews: This may reflect concern of the design engineers that design reviews put them on "trial".

Question I - Differences for Hardware, Software and System Design.

Table 5-6 shows the Question I answer choices for which significant differences were recorded between respondents associated with hardware, software and system design. These results are discussed below:

- The complexity of the Product Documentation Scheme: The fact that this was given more importance by hardware designers than by software designers was probably due to the need to produce production documentation and, in particular, 'Modification Action Packages' for hardware but not software.

ANSWER CHOICE	TYPE OF DESIGN	MEAN OF RATING	TYPE OF DESIGN	MEAN OF RATING	CONFIDENCE OF DIFFERENCE
4. The size of the Product Building Blocks	Hardware	1.6	Software	2.2	95%
5. The complexity of the Product Building Blocks	Hardware	2.0	Software	2.8	> 95%
6. The complexity of the Product Documentation Scheme	Hardware	3.1	Software	2.2	> 95%
7. The Managerial Organisation in the Development Laboratories	Hardware	2.3	Software	1.5	> 95%
8. Motivation of the Design Engineers	Hardware Software	3.2 2.1	Software System Design	2.1 3.4	> 95% > 95%
10. The level of completeness of Specifications	Hardware	3.5	Software	4.3	> 95%
16. The method of Selecting Engineers during recruiting	Hardware Software	1.7 1.1	System Design System Design	3.1 3.1	> 95% 99%
18. The general attitude to work in modern Britain	Hardware Software	1.0 1.0	System Design System Design	2.1 2.1	> 95% > 95%
24. The level of communications between GEC, BPO and other Participating Firms	Hardware	2.4	Software	3.1	> 95%

Table 5-6. Significant Differences Between Responses of Staff Associated with Hardware, Software & Systems Design.

- <u>The Method of selecting engineers during recruiting:</u> The system designers may have attributed more importance to this factor than did the hardware and software designers because of prestige bias. They were, on average, of greater length of service and more senior than the hardware and software designers.

There are no obvious reasons for the other differences.

Question II.

Table 5-7 exhibits the percentage of respondents who chose each of the Question II answer choices. These results are discussed below:

- Supervisors are thought to be the group most able to reduce the number of Design Faults. This shows consistency with the high ranking given to "Specifications" in Question I as specifications are usually written by Supervisors. It may, however, be due to Supervisors being the one group with whom the other two groups have greatest contact and interaction,

- Design Engineers got a higher percentage in response to the question, "Who would have the greatest potential for reducing the number of Design Faults if only they were given the requisite power, responsibility or facilities" than to the

	REPLIES (PERCENTAGES)				
	Design Engineer	Supervisors	Management	BPO	Other
1. In your opinion, whom does the company hold as being responsible for reducing the level of Design Faults?	24	64	12	0	0
2. Who, in your opinion, should have the greatest responsibility for reducing the level of Design Faults?	21	55	24	0	0
3. Who, in your opinion, currently has the greatest power to reduce the level of Design Faults?	20	44	28	3	3
4. Who, in your opinion, would have the greatest potential for reducing the number of Design Faults, if only they were given the requisite power, responsibility or facilities?	44	45	10	0	0

Table 5-7. Percentages of Respondents who Chose Each of the Question II Answer Choices.

other questions. This may be the result of prestige bias since the majority of respondents were Design Engineers,

- A comparatively small number of the respondents thought that Management could reduce the number of Design Faults. This was inconsistent with the high ratings given to the following Question I answer choices:

  - The experience of Design Engineers,

  - Methods used for checking designs,

  - Training given to Design Engineers, and

  - The complexity of the product documentation scheme.

Question III.

Table 5-8 shows the responses given to Question III, grouped under common headings. The less obvious titles are expanded below:

- "Discipline": This included comments such as "Work Harder", "Be more concientious", "More Self-motivation", etc.,

GROUP	DESIGN ENGINEER		SUPERVISORS		MANAGEMENT	
REPLIES IN ORDER OF NUMBER OF TIMES USED	Discipline	(35)	Design Management	(37)	Pay & Related Topics	(22)
	Design Checking	(14)	Design Checking	(26)	Motivation	(20)
	Method of Working	(14)	Communications	(26)	Communications	(20)
	Specifications	(13)	Specifications	(12)	Recruiting, Training & Career Opportunities	(19)
	Communication	(11)	Discipline	(10)	Planning	(17)
	Non-Parochial View of Work	(9)	Team Organisation	(7)	Project Management	(13)
	Scope of Work/ Responsibility	(6)	Training	(6)	Time Scales	(12)
			Scope of Work	(5)	Facilities	(10)
					Specifications	(9)
					Working Conditions	(7)
TOTAL NUMBER OF RESPONSES	115		137		151	

The numbers in brackets represent the number of times each topic was mentioned. Topics mentioned less than five times are not shown.

Table 5-8. Responses for Question III.

- "Non-parochial view of work": This included replies which expressed the view that Engineers should have an appreciation of wider aspects of their work than their immediate tasks,

- "Scope of Work/ Responsibility": This covered replies concerned with the activities which have to be performed by an engineer in the normal course of his work,

- "Planning": This included all topics related to the planning of development projects,

- "Project Management": This covered all aspects of development project management.

It can be seen from the titles of the categories that they are not mutually exclusive and that some responses would be covered by more than one of them.

The responses given for each of the groups, Design Engineers, Supervisors and Management are discussed below:

- Design Engineers:

The category of responses which generated the greatest response was "Discipline". This gave no support to the results from Question I. However, the next two most popular categories, "Design Checking" and "Method of Working", were consistent with the high ranking given to the "Development Methods" answers in Question I.

- Supervisors:

The importance attributed to "Design Management" and "Design Checking" is consistent with the high ratings given to the "Development Methods" group of Question I answers. Similarly, the importance given to "Specifications" in Question III agrees with the high rating given for that topic in Question I.

- Management:

The importance attributed to "Pay and Related Topics" is evidence of bias caused by the industrial dispute.

The large number of answers covered by each of the topics "Motivation", "Communications" and "Recruiting, Training and Career Opportunities" are consistent with the high ratings given to "Human Topics" for Question I. The results for "Planning" and "Project Management" tend to confirm the high ratings given to "Development Methods" answers of Question I.

5.10. Critique of Study.

One of the main difficulties with using a questionnaire to establish facts which cannot be precisely measured is that they seek opinions about those facts. While these opinions may be sincerely held, they may, nevertheless, be incorrect, subject to bias and influenced by circumstances prevailing at the time of the study. It is, therefore, necessary to try to establish the validity of the results either by employing other techniques or by examining the consistency of the replies.

As was described in Section 5.4, this study employed the technique of using check questions to test for consistency of replies and, therefore, hopefully, to establish their validity

Although the replies given to Questions II and III largely supported the results from Question I (see the previous Section), there was some evidence for the presence of prestige bias from Question II and that the industrial dispute may have influenced the results (Question III).

Further evidence of consistency of the results of the questionnaire came from grouping the Question I responses into those given by:

- Engineers working on different projects,

- Hardware, Software and Systems Design Engineers,

- Designers, Supervisors and Managers.

The last two of these, and in particular the second, however, also exhibited evidence of prestige bias.

While the results of the questionnaire tended to support each other, one major inconsistency remains. This concerns the high rating given to 'incomplete specifications' as a cause of Design Faults. The inconsistency arises through the definition of a Design Fault, namely, "a fault which occurs within the design activity, and which causes the equipment to behave in a manner different from the way it was specified to behave" (see Section 4.5). This definition precludes 'incomplete specifications' from being a cause of Design Faults.

It is difficult to know precisely why this response was given such a high rating, but the following are offered as possible reasons;

(a) Misunderstanding of the question - even though the above definition of a Design Fault was given in the covering letter of the questionnaire,

(b) Prestige bias resulting in transfer of blame to someone or something other than self,

(c) Other difficulties with specifications (such as understandability) being interpreted as incompleteness.

(d) In the early days of the LPU and SPU projects, Slide-in-unit design sometimes started before the specifications had been formally written. It is unclear whether incompleteness was being interpreted as "Not written in time" or "When written, not complete".

The other high rated answers were all theoretically valid and not obviously suspect.

There is one further point which must be made about the results. Although they gave an indication of the relative importance of the various factors causing Design Faults, they did not produce absolute figures for the number of Design Faults actually caused by each one. These could only be established by other experiments and studies.

5.11. Summary of Chapter.

In order to investigate the reasons for Design Faults within the Sponsoring Organisation, the development staff were systematically asked, through a questionnaire, for their opinions of the causes.

Related causes, as shown by the results of the study, were grouped under common headings which are listed below in descending order of their relative importance:

- Documentation,
- Development Methods,
- Human Topics,
- The Product,
- The Environment,
- Development Aids.

Of the individual reasons, the following four (in descending order of their importance) were rated by the development staff as being most responsible for causing Design Faults:

- The level of completeness of specifications,

- The amount of experience of the Design Engineers,

- The methods used for design checking,

- The training given to Design Engineers.

Although there were differences in the results for different projects; for hardware, software and system design; and according to replies given by staff of different ranks; these variations tended to validate rather than invalidate the main conclusions.

There was, however, one major discrepancy in the results. This is associated with the high rating given to "the level of completeness of specifications", which is inconsistent with the definition of a Design Fault.

There was also some evidence of prestige bias and of influence of the industrial dispute taking place at the time of the study.

The results only indicated the relative importance of each of the factors in causing Design Faults. It did not produce absolute figures for the number of faults caused by each one. These could only be produced by separate studies.

CHAPTER 6

EFFECTS OF DESIGN CHANGES

6.1. Chapter Preview.

It was argued in Chapter 3 that design changes significantly contribute to the development time of new products. The hypothesis was put that to reduce the time taken to implement changes would reduce the development time. This was based on examination of the development plan for the LPU subsystem of System X rather than on the actual performance of the development team.

The purpose of this Chapter is to present an assessment of the real effects of design changes as measured in practice. It concentrates on two objectives:

1. To measure the direct effect of design changes on the total duration of the LPU development project (to indicate potential savings which could be made), and

2. To assess the cost of the design changes (to indicate the maximum expenditure which could be made to reduce the rework time and still give a cost advantage).

In order to meet these objectives, calculations of average effects would suffice. It was, however, recognised that there would be a number of factors causing variations in the effects of the changes. These are listed in Section 6.2. Two fundamental approaches of measuring the effects on the total project duration are presented in Section 6.3, and the method used described in Section 6.4. In Section 6.5, the effort required to implement each design change is calculated. The accuracy of the result obtained is discussed in Section 6.6. Calculations of the effects of changes on the total development duration and of the cost of changes are presented in Sections 6.7 and 6.8, respectively. The findings are summarized in Section 6.9.

In the first paragraph of this Chapter, reference was made to an LPU development plan. Whilst this plan took the form of a PERT network which was updated as progress was made, it is worth noting here that it was of no use in helping to establish the effects of individual design changes. The main reason for this is that it only reflected the project down to Subsystem level and did not show the importance of activities associated with individual Slide-in-units. It, therefore, could not show how a change would affect any particular path. It also did not show whether slippage was due to changes or to activities simply taking longer than planned.

It is also worth noting here that before this project was undertaken, the Sponsoring Company operated no formal process to monitor effort or time spent specifically on processing design changes. The raw data required for the calculations referred to in this Chapter were, therefore, obtained from a number of sources. These included:

- Estimates based on the personal experience of the development engineers,
- Averages for fixed time periods, and
- Figures recorded specifically for this project.

Details of these are shown in Appendix F which is referred to again later.

6.2. Causes of Variation of Effects of Design Changes on Project Duration.

In Section 6.7, the average effect of each design change on the development project duration is calculated. It was recognised that in practice there would be a number of factors which would cause variation in the individual effects. These are listed below:

a. The extent of the change; that is the number of modifications made to the product design,

b. The level of the product's hierarchical structure at which the fault occurred,

c. The cause of the fault,

d. The total number of changes being generated at any time,

e. Different methods of implementing changes,

f. Whether or not the change affects the critical path of the development project - and this is complicated by the critical path changing,

g. The stage of development at which the change occurs,

h. The availability of resources to implement the change.

Each of these will be dealt with at the most appropriate place in the Chapter, but it is worth noting here that factors "a" through to "e" affect the accuracy of the results and "f" through to "h" the fundamental way the calculations are performed.

6.3. Two Fundamental Methods of Measuring the Effects of Design Changes on the Development Duration.

In principle, two fundamental approaches were available.

The first of these involved extracting the total project slippage due to all design changes from progress reports. Then, using the number of changes (already measured), the effect of each one could be calculated. While this method would have had the advantage of automatically taking into account critical paths, the information needed to establish the slippage due to changes was not available. The method could not, therefore, be used.

The second approach which was adopted was to measure the actual time spent processing individual changes and then to deduce the effect of all changes on the total development duration.

In addition to being the only practical approach, this method had the following advantages and disadvantages over the first:

Advantages:

- It will be seen that this method necessitated calculating the effort required to implement each change. This was also required for calculating the cost of each change,

- It will also be seen that the method enabled the relationship between the resource of development engineers and the effects of changes on the development time to be taken into account. This proved possible because this resource was critically constrained,

- The variations in the effects of changes caused by most of the factors listed in Section 6.2 could be established, at least in principle.

Disadvantages:

- The calculations were greatly complicated because it was not possible to identify a critical path,

- The required data were only available for Slide-in-units. It will be shown, however, that overall this was not a great disadvantage.

6.4. Method Used to Establish the Effect of Each Change.

In a simple world where every change affects the critical path and is treated independently of the others, one would only need to measure the elapsed time from finding a fault to the change being completed. This would then represent the slippage in the project completion due to the change.

In practice, the true situation is far more complicated:

- Not all the changes affect the critical path,

- Changes are not implemented independently of each other, but are "saved up" for processing,

- Changes are implemented in different ways (for some, a Modification Action Package only is produced, and for others a complete rework is undertaken in addition to the production of the Modification Action Package).

Thus, generally, there is no direct relationship between the time to implement fully the change and the delay the change causes to the completion of the whole project.

The only measure which can be made of the consequences of a change, and which is consistent for all the changes, is the effort, in terms of man-hours, which is required to implement the change. How this can be related to the resulting delay in project completion is nevertheless still a complicated issue and will be discussed in Section 6.6.

The processing which takes place to implement a 'Change Note' is described in Appendix F. This Appendix also indicates how the effort required to implement each stage of processing was established for this study and, where possible, shows the raw data. It was necessary to relate the results for each 'Change Note' to the effort required to process each change. The different results arise from two mechanisms:

- The non-unity number of changes per 'Change Note', and

- 'Change Notes' generated for faults made during the implementation of other changes ('Rework Faults').

Before embarking upon the calculations, it is necessary to define the equipment entities for which they were carried out. Ideally, calculations would have been done for each type of equipment entity. Unfortunately, the raw data were only available for Slide-in-units to which the study, therefore, had to be restricted. The disadvantages of doing this were not as great as might at first be suspected because there were considerably more Slide-in-units than any other entity.

Also, the majority of changes were for Slide-in-units (see Table 4-2).

6.5. Calculation of Effort Required to Process Each Change.

Before presenting the results of the calculations described in Appendix F, an explanation is given of how the results were converted from being applicable to 'Change Notes' to being applicable to changes.

Changes per 'Change Note'.

In Chapter 4, it was shown that there were 0.93 changes per 'Change Note'. It was, therefore, necessary to divide the results given by the calculation described in Appendix F (except for Stage "1") by this factor (0.93) to produce the effort required to process each change. Stage "1" ("Find fault, find remedy and write 'Change Note'") was recorded for each change and so does not need conversion.

Rework Faults.

The study to find the immediate causes of design changes gave the result that 3% of changes (for the LPU) were due to rework faults. These were faults which occurred because of errors made during reworks; i.e., implementing other changes. They were not faults of the original design, and the effort required to process them should be regarded as part of the effort of processing the changes due to faults of the original design. To take this into account, the figures produced from the calculations described in Appendix F were divided by 0.97.

Results.

The average or mean numbers of man-hours spent on each stage of processing a design change (as calculated in Appendix F) are listed in Table 6-1.

Totalling these figures yields the result that:

THE TOTAL NUMBER OF MAN-HOURS EXPENDED PROCESSING EACH CHANGE WAS 66.4.

It should be noted that the individual figures shown in Table 6-1 take into account the proportion of changes to which each stage of processing is applied.

6.6. Accuracy of Result.

The result shown in the previous Section is just an average effort required to process a design change. A question which obviously arises is, "How reliable is

ACTIVITY	NUMBER OF MAN-HOURS
1. Find fault, find remedy, and write 'Change Note'	3.3
2. Change existing Slide-in-units	8.5
3. File 'Change Note'	0
4. Produce Modification Action Package	16.5
5. Process Documentation (Modification Action Package)	1.1
6. Process Change	0.6
7. Authorise Change (BPO)	Not Applicable
8. Process Change Documentation (Production)	0
9. Change 'Units' (Production)	21.0
10. Rework Design	14.2
11. Process Documentation (Rework)	0.6
12. Process New Issue	0.3
13. Authorise Change (BPO)	Not Applicable
14. Process New Documentation (Production)	0
15. Test Reworked 'Unit'	0.3

Table 6-1. Effort Expended in Processing Design Changes.

it?". In other words, "What deviation can be expected for different changes?".

Each of the first five factors giving rise to variation, as listed in Section 6.2, will be considered in turn. It should be noted, however, that a definitive standard deviation will not be calculated. This is because the necessary raw data were not available. Simple averages had to be used in the calculations for all stages shown in Table 6-1 except stages "1" (Find fault, find remedy, and write 'Change Note') and "10" (Rework design).

Variation with Extent of the Change.

It can, with some confidence, be predicted that the greater the extent (the number of changes to the product design) of the change, the longer it will take to process and the more effort will be required to implement it. Looking at the process stages shown in Table 6-1, the following would be expected to exhibit this effect:

(a) "1" Find fault, find remedy and write 'Change Note',

(b) "2" Change existing Slide-in-units,

(c) "4" Produce Modification Action Package,

(d) "9" Change 'Units' (Production),

(e) "10" Rework Design.

In practice, however, several 'Change Notes' were included in each Modification Action Package and Rework. This meant that it was impossible to establish the variations for "c", "d" and "e" above. "b" was calculated from average values and so the variation could not be calculated for it.

It is nevertheless possible to show a variation for "a" above. This is presented graphically in Figure 6-1. This shows a linear line of best fit which indicates a positive correlation between the extent of the change and the effort required to find the fault, find the remedy and write the 'Change Note'.

NOTE: The number of circuit modifications was used as a measure of the extent of the change. Alternatives, such as the number of component changes or number of Slide-in-units affected, could have been just as valid but there were insufficient data available for these to establish any relationship.

Variation with Level of the Product's Hierarchy.

This factor is clearly related to the previous one. If a fault occurs in the design of a Slide-in-unit, then only that one Slide-in-unit will need to be changed. If, however, the fault occurred at subsystem level, then it

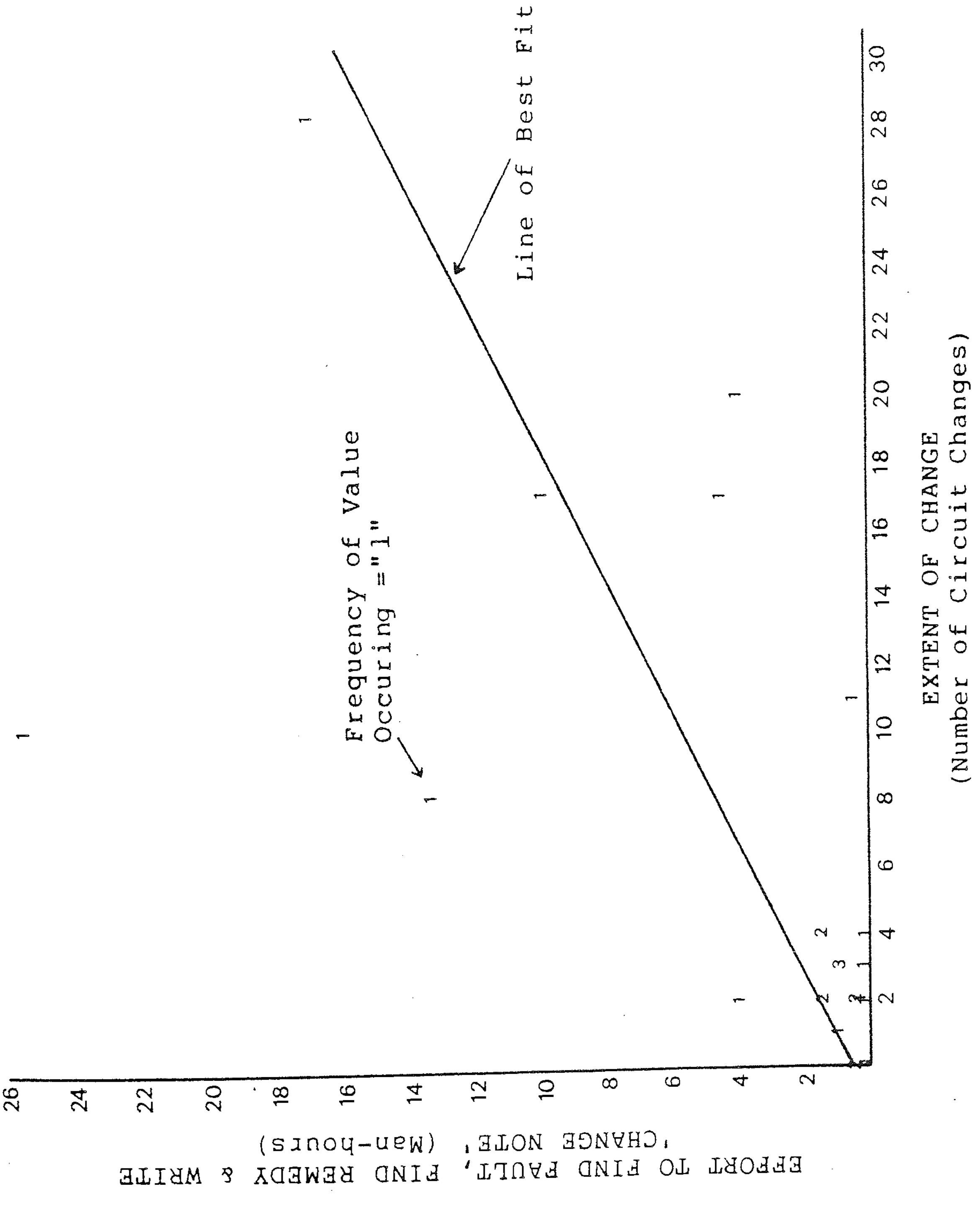


Figure 6-1. Relationship between Extent of Change and Effort reqired to Find Fault, Find Remedy and Write 'Change Note'.

could affect many Slide-in-units and other entities. It would be expected, then, that the higher up the hierarchy the fault occurred, the more effort would be required to correct it. There was, however, insufficient information to prove this.

Variation with Cause of Fault.

For the same reasons as given under "variation with extent of change", it was only possible to establish the variation for stage "1" of the change processing (Find fault, find remedy and write 'Change note').

Table 6-2 shows the figures for stage "1" for the following causes of change:

- Logic Design Faults,
- Hardware Realisation Faults,
- Slide-in-unit Specification Faults.

There were insufficient figures for the other causes to make it worthwhile to include them. The Table shows that Specification Faults take significantly more effort to detect and remedy in commissioning than do Design Faults - at least for Slide-in-units.

CAUSE OF FAULT	EFFORT TO FIND & REMEDY		NUMBER IN SAMPLE
	MEAN	STD. DEVIATION	
Logic Design Faults	1.1	1.1	15
Hardware Realisation Faults	1.9	2.4	4
Slide-in-unit Specification Faults	12.4	8.7	5

Table 6-2. Relationship Between Effort Required to Find Fault, Find Remedy and Write 'Change Note' and Cause of Change.

It is worth noting the Slide-in-unit Specification Faults result in significantly more extensive changes than do Design Faults. This is illustrated in Table 6-3.

CAUSE OF FAULT	EXTENT OF CHANGE		NUMBER IN SAMPLE
	MEAN	STD.DEVIATION	
Logic Design Faults	5.0	4.8	53
Hardware Realisation Faults	1.6	1.8	12
Slide-in-unit Specification Faults	13.2	13.0	23

Table 6-3. Relationship Between Extent of Change and Cause of Change for Slide-in-units.

Variation with Rate of Change Production.

The rate of production of changes can affect the effort required to process each change in two ways:

- Some of the process stages may require greater effort for a higher production rate,
- Some of the stages may require less effort for a higher production rate.

The former effect will be exhibited by stage "1" of Table 6-1 (Find fault, find remedy and write 'Change Note'). The larger the number of faults there are present during commissioning the more they will interact and the more difficult they will be to isolate and remedy. Unfortunately, no evidence was available to prove this. A higher production rate of changes will also add to the complexities of managing and controlling the project.

There are some stages for which each change would require a smaller amount of effort as the rate of production of changes increases. These are notably those associated with the processing of Modification Action Packages, and Reworks. With a higher production rate of changes, more 'Change Notes' are included in each Modification Action Package and Rework. The effort required to implement these, however, will not increase pro rata. The nett effect will be a reduced effort per change required for these stages. Again, data were not available to quantify the effect.

Variation with Different Ways of Implementing the Change.

This factor arises because not all changes follow the same process. According to the standard process, each change would be:

(a) Included in a Rework to produce new "clean" documentation, and

(b) Included in a Modification Action Package to correct existing faulty 'Units'.

In practice, however, the Rework may not be done and production modifications tolerated; that is, the equipment is manufactured to an old issue of documentation and modified following production. This obviously affects the effort required to implement the change. No data were available to quantify the effect.

Summary of Variations.

Whilst the five factors discussed above affect the effort required to implement changes, it was not possible to quantify their effects. Indeed, in some cases it was not even possible to establish whether they result in a nett increase or decrease of effort per change.

6.7. Effects of Changes on Total Development Duration.

In Section 6.5, we established the average effort required to implement a design change. In theory some changes may affect the project's critical path but others would not. This picture would be further complicated by the critical path changing with time. It will be argued later that in practice for the LPU project the resource of development engineers was constraining and critical and all changes affect the project duration. The question we need to answer, then, is, "How do we relate what we know about implementing design changes (the effort required to perform each stage of processing) to the delay that they cause to the total project?".

We shall do this in two ways, making different assumptions for each. However, before we take this course, let us consider the development project as consisting of three hypothetical phases and that these phases are distinct and distinguishable. (We shall argue what happens if they are not later). To avoid confusion with terminology used elsewhere in this Thesis, the phases will not be named but just referred to as Phases I, II and III. The characteristics of each are:

PHASE I

* This is essentially a design phase and preceeds the building of models. Documentation is being produced but will not have been released and so is not subject to rigorous change control procedures,

* The effort and time to implement a change will be small since only documentation has to be altered, and even that will not invoke a change control procedure. Changes may not be recorded as having been made,

* The number of changes will be relatively very small,

* Involvement in change implementation will be restricted to design engineers,

* The critical resource will be the design engineers.

PHASE II

* In this phase, models are produced and commissioned. In addition to the documentation (which is now formally controlled), there also exists physical hardware,

* The effort to implement a change will be large since not only will the documentation need to be altered, but so will the hardware. The associated time for implementing a change will be relatively long, but since there will not be the absolute need to produce perfectly 'clean' documentation, some short cuts will be able to be made. All changes will need to be fully and formally recorded,

* The number of changes will be very large,

* Involvement in change implementation will now include documentation processing departments, commissioning, model shops and production, in addition to design engineers. Production, however, will tend to treat its involvement as being on a "special" basis. Its normal procedures may not be adhered to. There is also flexibility as to whether a particular job is performed in a production department or a model shop,

* The critical resource will be development engineers - at least it was for the Sponsoring Company at the time the project was performed. Note that "development engineers" includes both design and commissioning engineers who were, at least in principle, interchangeable.

PHASE III:

* In this phase, the product has been developed and tested, but it is necessary to clean-up the documentation for final hand over to production. Each change generated at this time must not only be implemented for existing equipment (which could be numerous), but will also require a rework to be carried out to produce the "clean" documentation. Production and, therefore, also the project may be held up until this has happened,

* The effort required to implement each change will be large, as will the time taken. The reasons for this are similar as for Phase II, but a rework will need to be done for each change, and production procedures will need to be fully followed,

* The number of changes will be relatively small as most of the faults should already have been remedied,

* Implementation of the change will involve the same groups as Phase II,

* There is no obvious critical resource, but, as will be seen later, the time required to implement a change will be dominated by the time taken to follow the production procedures.

The three phases above have been described as being distinguishably separate, and although in practice they merge into each other, we shall continue to think of them in this way when establishing the effect of each change on the total development time.

Changes generated during Phase I will not be considered further. This is because, by definition of Phase I:

- The quantity is insignificant,

- The time and effort required to implement each one is very small,

- Most of the changes are not formally recorded.

Their total effect will, therefore, be insignificant.

Changes Generated During Phase II.

It has been indicated above that the critical resource required for implementing design changes during this phase is the development engineers. Before this fact can be used, a relationship must be found with the critical path.

For the LPU there were about 20 development engineers who were responsible for:

- Designing the 72 Slide-in-units,

- Commissioning the Slide-in-units, Subsystems and System, and

- Implementing changes to the 72 Slide-in-units.

Most of the engineers could be employed to perform any of these tasks for any of the Slide-in-units.

Although it was obviously necessary to set priorities to get the work completed in an orderly fashion, no one Slide-in-unit could be singled out as being on the critical path during this phase. What was important was the combined resource of all the engineers. This being the case, and assuming there were enough changes to keep all the engineers busy (which there were), then the delay caused by each change would be the number of man-hours of engineers' effort required to implement it, divided by the number of engineers available.

The development time is then affected by the following stages of processing as shown in Table 6-1:

"1" Find fault, find remedy and write 'Change Note'	3.3 Man-hours
"3" File 'Change Note'	0 Man-hours
"4" Produce Modification Action Package	16.5 Man-hours
"10" Rework design	14.2 Man-Hours
"15" Test reworked 'unit'	0.3 Man-hours

The total effort required for these activities is 34.3 man-hours. Given that there were 20 engineers available (excluding supervisors), then the effect of each change on the development time was 1.7 hours.

If all the hardware changes for the LPU (about 1900) occurred during this Phase II, and changes to all equipment entities had the same effect as changes to Slide-in-units, then their total contribution to the development time was 3230 hours or 86 working weeks.

An examination of what happens in Phase III will show a different result.

Changes Generated During Phase III.

The characteristic of Phase III is that "clean" documentation and unmodified equipment need to be produced. This means that each change will result in a rework and that the project will be delayed until the new-issue Slide-in-units are produced. It can be assumed that the critical factor affecting the overall development time is the elapsed time taken to find the fault, find the remedy and write the 'Change Note', perform the rework and to manufacture new Slide-in-units. The components which affect the overall development time are then:

A. The time taken to find the fault, find the remedy and write the 'Change Note' (Stage "1" in Table 6-1) (3.3 hours)

B. The time taken to rework the design (Stage "10" in Table 6-1) (180 hours),

C. The elapsed time required to process the documentation, update the production computer files and manufacture the new Slide-in-units. The total duration of these activities is laid down in a Company manual as 29 weeks (1087.5 hours). It should be noted that this time is an elapsed time and includes 'waiting time'.

The effect of each change on the overall lead time is then calculated by:

Adding the durations of "A", "B" and "C".

This yields a figure of 1270.8 hours for the delay to the whole project for each change.

Overall Conclusion.

The result for Phase III is orders of magnitude greater than the result for Phase II and it is obviously meaningless to use the result for Phase III to calculate the effect for all changes to the LPU. The figure is so large because of the waiting time included in the 29 weeks for processing the documentation, updating the production computer files and manufacturing the new Slide-in-units. The number of changes during Phase III would, by definition, be very small. Indeed, the calculation of the figure above implies only one change is being processed at any one time.

The assumptions made for our theoretical Phase II, however, are far more consistant with the project as it was in practice up to the time of writing. It can be assumed that the result derived will be fairly accurate in practice, i.e.:

* Each change delayed the project by 1.7 hours, and

* The total slippage due to all changes was 86 working weeks.

It must be noted here that the figures shown represent average values based on a large number of changes. Should a change be critical to the project as a whole, then although in theory it would cause a greater delay, ways would be found to circumvent the official procedures. Short cuts would be taken to enable progress to continue.

It only remains to add that in practice the phases are not distinctly separate as we had assumed but merge into one another. The discussion and results show, however, that the later a change is produced, the greater will be the effect on the total project duration.

6.8. Direct Effects of Design Changes on the Cost of Development.

This Section describes the method used to establish the direct effects of design changes on the development cost of the LPU and presents the results obtained.

Method.

The direct effect of a design change on the development cost was calculated from three components:

1. The cost of the effort required to accomplish each stage of the change implementation. The effort for each stage was available from the calculations described in previous Sections. The cost figures were provided by the Sponsoring Company's accountants on the basis of a total cost per man-hour for each department. This included a direct labour cost and overheads and were valid for the financial year 1979/80.

2. The cost of materials for physically modifying Slide-in-units as a result of Modification Action Packages. This information was provided by the accountants.

3. The cost of scrapping old 'issues' of Slide-in-units which were replaced following a rework. To

establish a figure for this, the value of Slide-in-units of the LPU and SPU scrapped for this reason over a period of three years, was taken and divided by the number of changes included in reworks during that period.

Results.

For reasons of confidentiality, the cost of each stage of processing will not be shown. The overall cost per Slide-in-unit change was calculated to be £543. This was made up of £478 for labour, materials and overheads and £65 for scrapped Slide-in-units.

The total cost to the LPU project for all its changes was about £1m.

Validity of the Results.

Some fundamental assumptions were made in producing the figures above. These are discussed below along with other aspects of the validity of the results.

- The total cost of LPU changes was calculated assuming that changes to entities other than Slide-in-units on average cost the same as those for Slide-in-units. Although, as discussed in terms of effort required to implement a change, there is a likelihood of a difference occurring, the vast majority of changes were for Slide-in-units. Any falsity in the assumption will, therefore, not significantly invalidate the overall result.

- It was assumed that changes not originating from commissioning make the same contribution as those which do. No evidence was available to suggest that this assumption was wrong.

- The overhead figures, included in the cost of effort for each department, were rather crude since they were averaged across a wide range of products.

- The results shown do not include indirect effects of changes on the development cost. These include increased planning and increased management costs.

6.9. Summary of Chapter.

The effects of changes on the development duration and cost for the LPU were synthesised from the contributions of each activity required to implement the changes. The following results were derived:

- The overall development time of the LPU was increased by 1.7 hours for each change produced (a total of 86 working weeks for all LPU changes),

- The overall development cost of the LPU was increased by £543 for each change produced (a total of about £1m for all LPU changes).

These figures only represented average values and, although it was shown that variations would occur, it was not possible to quantify them. It was shown, however, that the later in the project that a change is generated, the greater will be the effects of that change on the project duration.

For the LPU, the critical path was shown not to be a major factor in determining the project delay due to a change (See Section 6.7). This was due to the large number of tasks required to be completed by comparatively few people. Also, any change which did become critical to progress proceeding could always be expedited by short cutting the implementation process.

NOTE.

It was not possible, within the timescale of this project, to measure the effects of design changes on other factors which they would be expected to affect:

- Indirect Costs, due to lost sales and a shortened 'pay-back' period,

- Tying-up of Resources which could be used for other development work,

- Degradation of Product Performance,

- Poorer Quality of Design,

- More Difficult Documentation Control,

- More Difficult Planning and Project Control.

CHAPTER 7

REWORK AND ITS RELATIONSHIP TO THE DEVELOPMENT PROCESS

7.1. Chapter Preview.

In the present Chapter, published models of new product innovation, design and development are used to build up a new theoretical model which highlights the importance of iterations in the development of large hierarchically structured products. The causes of the iterations, their mechanics and their consequences are predicted from it. The validity of the model is then assessed by comparing the findings of the practical investigative research described in preceding chapters with the predictions.

7.2. New Product Innovation.

New product innovation is defined as:

"The process of taking an idea or concept and producing and marketing a new or improved product".

Twiss <24> describes innovation as a conversion process from scientific knowledge and materials to a product. Although organisations can adopt a number of different strategies for innovating - 'offensive', 'defensive', 'imitative', 'dependent', 'traditional' or 'opportunist' - the stages through which they go exhibit the same general pattern.

These stages are shown in a model of cash flow in a typical course of innovation, by Montgomerie <25>. They are reproduced in Figure 7-1, and each is discussed below.

Basic Research.

Basic Research is performed to establish the laws of nature or to find the properties of materials. It is often concerned with deriving theoretical mathematical models and is not usually orientated towards producing a particular product.

Applied Research.

Applied Research is performed to take the theoretical findings from the Basic Research to a form in which they can be applied to produce something useful.

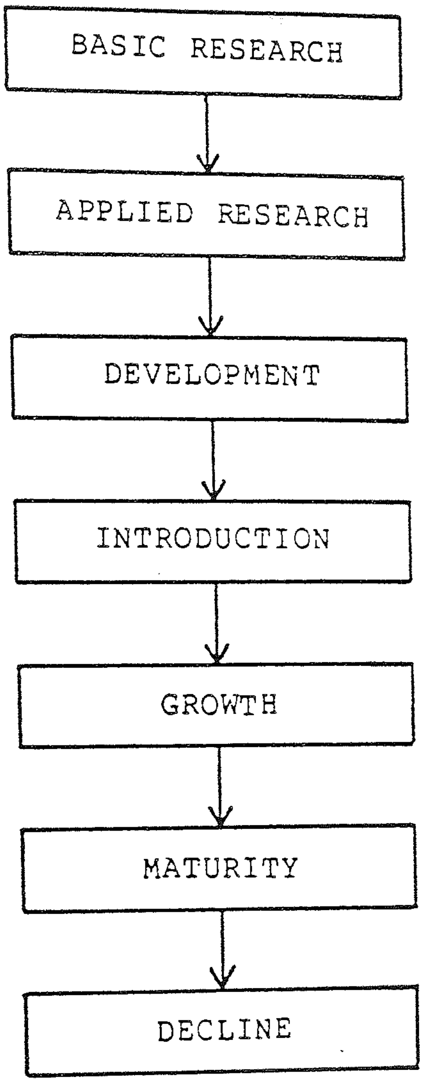


Figure 7-1. The Innovation Process: Montgomerie.

Development.

Development is the stage in which the new product or process is designed, tested and evolved. It is discussed further in Section 7.3.

Growth, Maturity and Decline.

These refer to the state of the product in relation to the market, and will not be considered any further here.

Pannenborg put forward a similar model of the innovation process <26>, and this is reproduced in Figure 7-2.

This model shows the major communication lines that must be operative to check continuously whether the goal pursued will satisfy the end user.

A third model in this general area (shown in Figure 7-3) is used by Rubenstein and Douds to illustrate the Research-to-Marketing linkage of Research and Development <27>. They treat these as a series of linked functions, each one of which would be performed by its own specialists. The main activities at the interfaces between the stages are communications and decision making. There is a roughly sequential flow of work down the stages.

7.3. Development and Design.

When discussing the concepts of "design" and "development", there is a difficulty which arises from

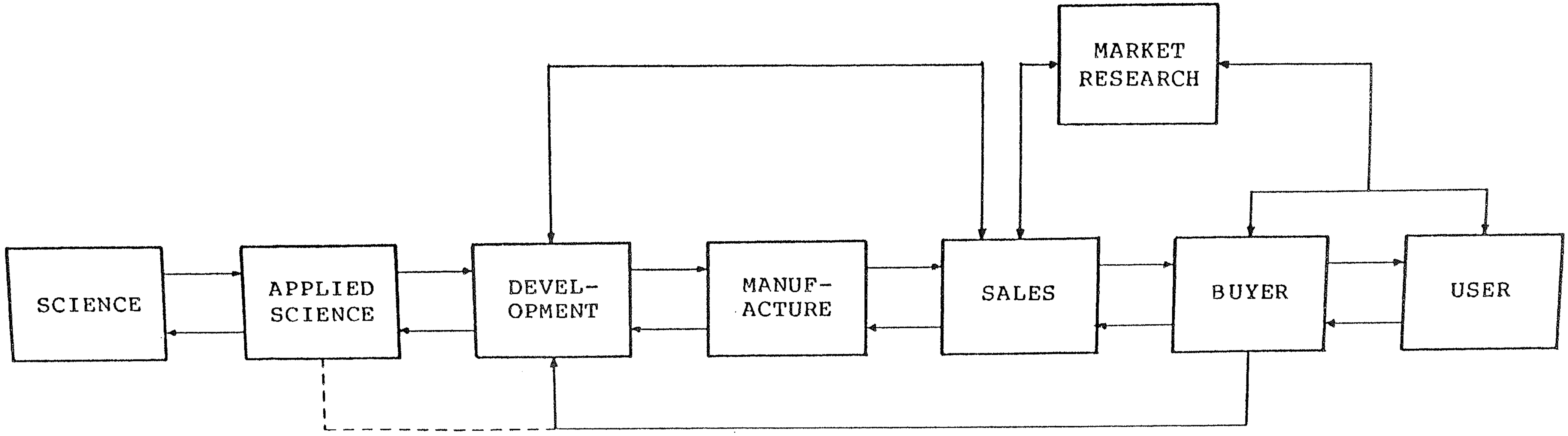


Figure 7-2. Innovative Chain: Pannenborg

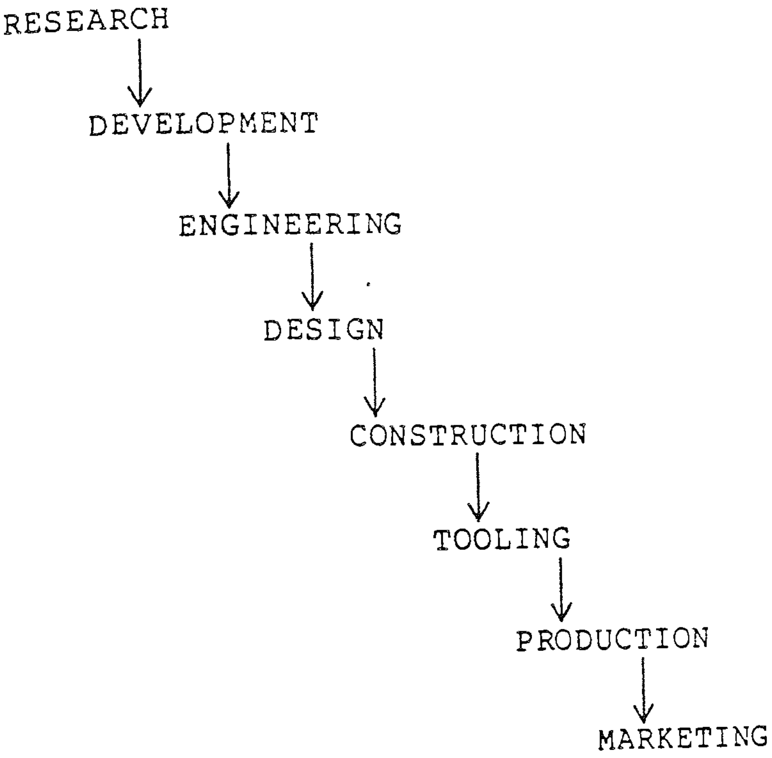


Figure 7-3. Research-to-Marketing Linkage of Research & Development (from Rubenstein and Douds).

differences in the terminologies used by authors writing in this area.

Dixon <28> describes an "Engineering Design Problem" as follows:

> "Devise, subject to certain problem solving constraints, a component, system or process to

accomplish a specified task optimally, subject to certain solution constraints".

One of the key words in this definition is the word "optimally". This implies that either a model - probably mathematical - is used to derive the optimal solution, or an iterative process involving prototypes must take place.

Dixon goes on to produce a model of the "Design Process", which is reproduced in Figure 7-4.

The stages of this model can be divided into four activity categories:

- Defining the objectives,

- The creative problem solving activity,

- Evaluation and optimisation of the solution,

- What is done with the proved and tested design.

In other words, the creative process or producing the design is just one stage in a wider process. The other stages, which must be carried out before the design is used, are:

- To produce a specification for the design, and

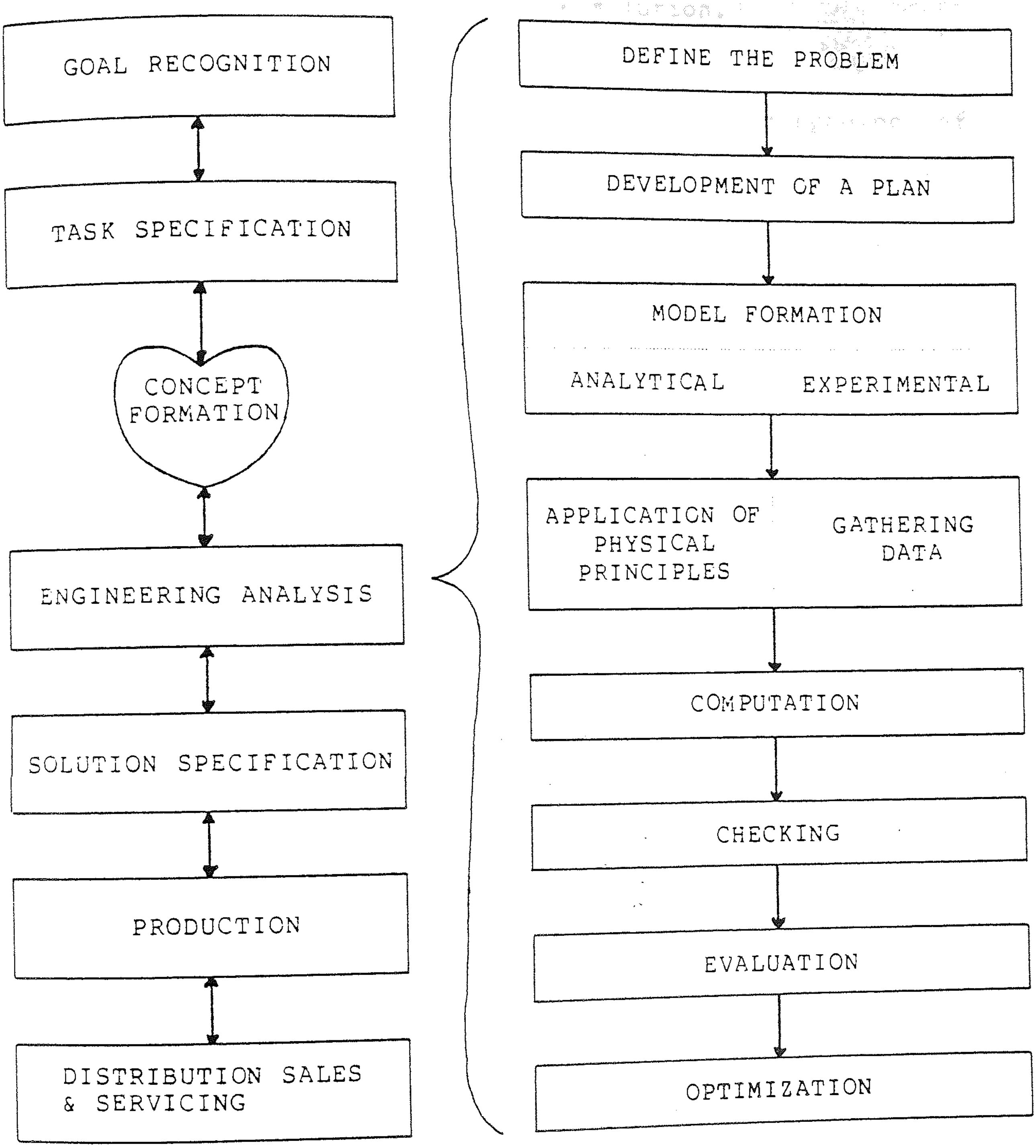


Figure 7-4. Dixon's Model of the Design Process.

- To test and optimise the solution.

Asimow <29> has proposed a model of "Morphology of Design", a term he defines as "A Progression from the abstract to the concrete". This model, however, like that of Dixon's, also embodies activities outside the creative process of producing the design.

In order to clarify subsequent discussion in this thesis, DESIGN will be taken to mean only the process of syntheses or inventiveness which produces the new thing or concept which may be the whole or part of the product.

DEVELOPMENT will be defined as the series of linked activities which result in a new product or process being designed optimally and prepared for commercial exploitation.

"Design", thus, becomes one stage of the process of "Development". This is illustrated in a simplified model of "Development", which is presented in Figure 7-5.

In these terms, Dixon's model of the design process and Asimow's model of Morphology of Design better describe the process of "development", rather than that of design. However, while adequately representing the development of a structurally simple product, they fail

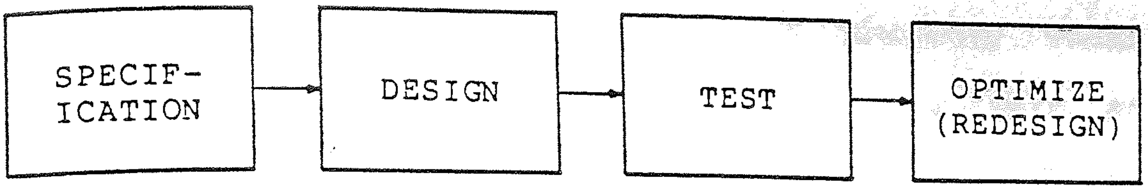


Figure 7-5. Simplified Model of Development.

in two respects to represent that of a large complex system.

The first is that they do not adequately describe the development of a hierarchically structured system. The second is that while implying some iteration to earlier stages within the process, it is not shown in sufficient detail to throw light on its effects on the duration or other characteristics of development. These problems are confronted in Sections 7.4 and 7.5, respectively.

7.4. Large System Development.

For a relatively simple product, the whole development programme can be undertaken by one engineer, or a small team of engineers. If, on the other hand, the product is large and complex, it may be necessary to divide it up into manageable sized blocks, or 'subsystems', so that several teams of engineers can develop different parts of it simultaneously. These subsystems can later be brought together to form an integrated product, or they can be treated as individual modules from which one product, or a family of products, can be constructed.

The advantages, however, of dividing a system into subsystems will only be achieved if certain criteria are met. By far the most significant is that the subsystem interfaces must be so designed that the subsystems are capable of working together. This can only be done if design starts at the highest, system, level and that the specifications for the subsystems and their interfaces are derived from the design of the system.

Design will then take place at each progressively lower level, of which there may be several. This is known as "top-down" design, and the stages of it are shown in Figure 7-6.

Note: Figure 7-6 shows two levels of subsystem, "Subsystem 1" and "Subsystem 2", but there may, in practice, be any number. There will, however, probably be less than five.

The first stage in the process is to create a system specification from the customer requirements or from information generated by market research. The process then passes through a series of design stages,

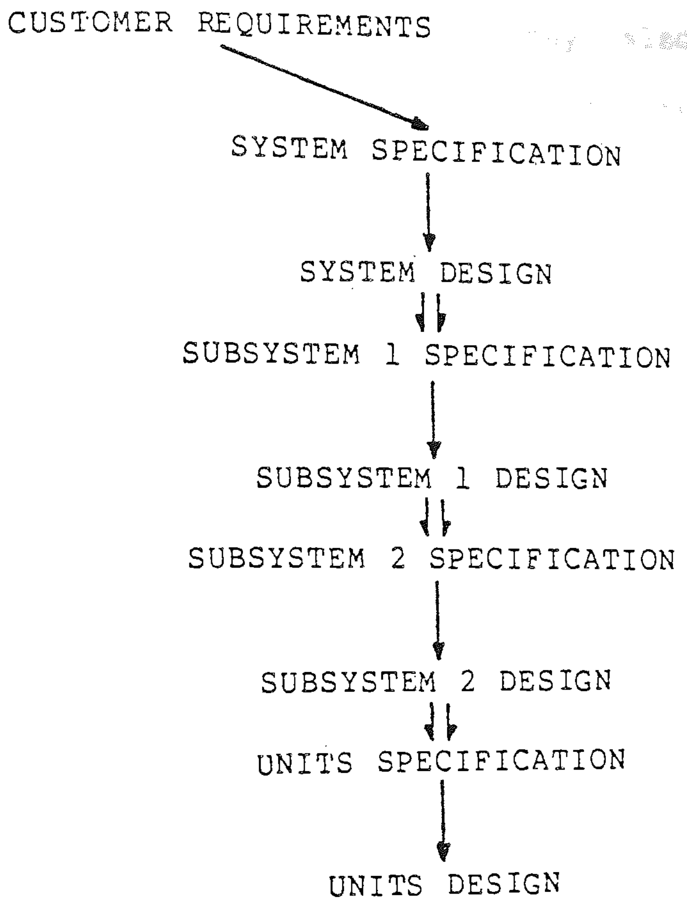


Figure 7-6. The Stages of a Large System Design Process.

each of which results in specifications for entities at the level below. This process continues until the design employs only previously designed and available entities. These may be components, circuits or even subsystems.

While the main activity is associated with top-down design, it is likely that there will be some bottom-up design taking place at the lowest levels. This may be the consequence of technology allowing more complex devices to be used, or it may also result from a tendency for project management to require physical proof of progress. During top-down design, no equipment is physically realised until the lowest levels are reached. In order to "prove" progress, therefore, there is pressure to start designing at the bottom.

The lowest level, shown in Figure 7-6, is "units". In the hardware case, these will be constructed mainly of commercially available standard components.

As well as applying to hardware, the model also represents software design. In this case, the "units" signify program modules.

Two concepts, which are not shown in Figure 7-6, are introduced in a model of top-down design for software by Ulrickson <30> (see Figure 7-7). They arise from a block identified as "Interface and Debug Major System Block", and are:

- Iterations from lower to higher levels in the design, and

- Testing.

The model fails, however, to recognise that there are iterations at all stages in the process. This shortcoming is partly overcome in a model for software

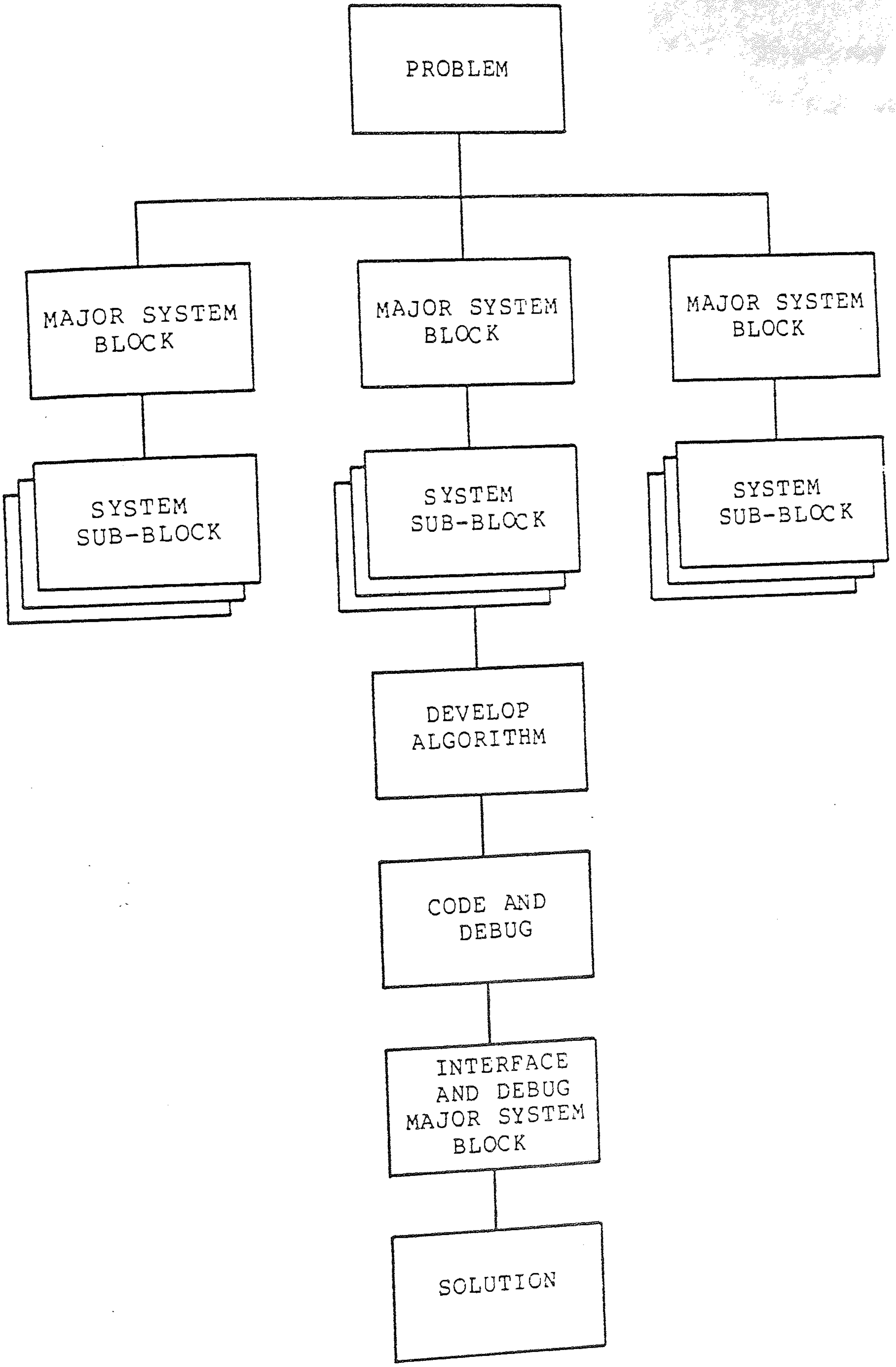


Figure 7-7. Top-Down Process for Software Design: Ulrickson.

development exhibited by Benson <31> and reproduced in Figure 7-8. This uses more general terminology, which could equally as well apply to hardware.

These concepts of iterative loops and testing are added to the model of Figure 7-6 and shown in Figure 7-9. Like Benson's model, this one is also divided into two parts, the first being concerned with the creation of the product design, - the "Design Phase" - and the second with the validation of the design.

7.5. Iterative Loops in the Development Process.

In an ideal world, there would be no need for iterative loops in the development process. Specifications and designs would be correct first time, and when the product was first tested it would work perfectly to the required performance. Unfortunately, there are a number of factors, outside the actual design, which prevent this happening. These are shown in Figure 7-9 as "External Factors". They cause the specifications and designs to be imperfect and necessitate going back and changing work which has previously been carried out.

Iterations from the System Specification to the customer may be necessary because, in design, it may be found that the requirements could not be achieved or

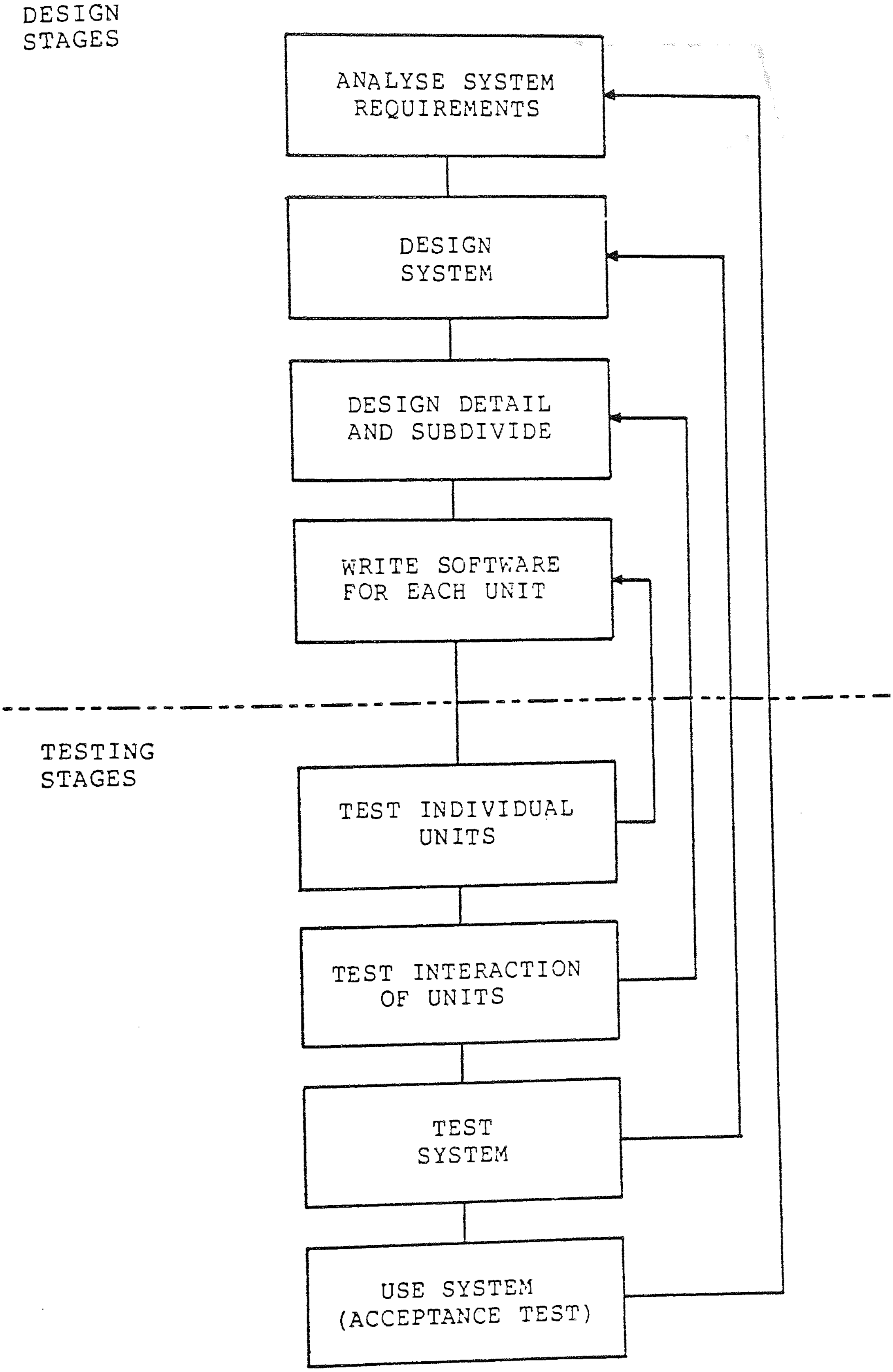


Figure 7-8. Developing a Software Product (Benson).

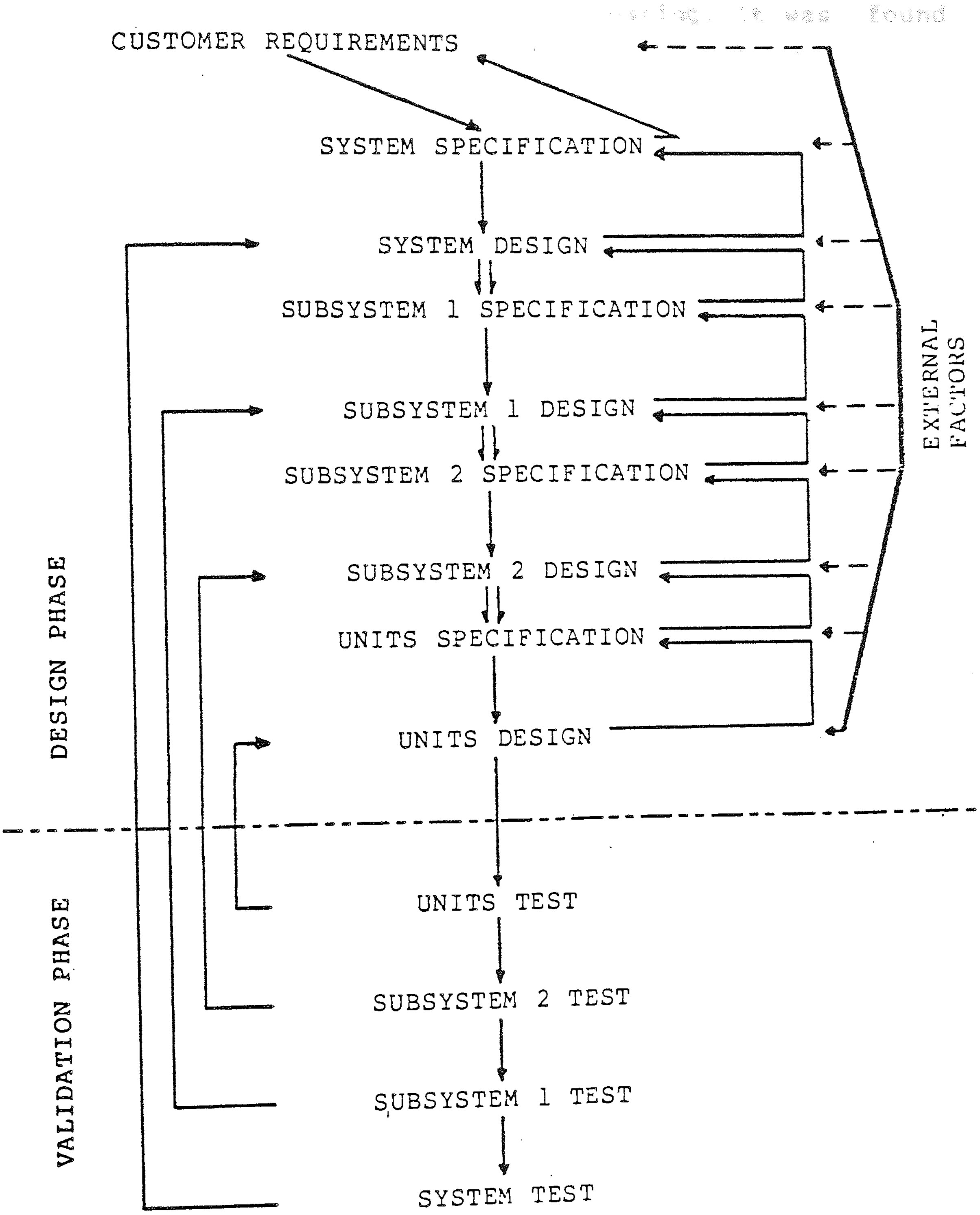


Figure 7-9. The Development Process for a Large System.

were ambigious, or because, after testing, it was found the system did not perform as expected.

During the Design Phase of system development, there may be iterations from each stage as it is being undertaken to the previous one. This may lead to a ripple effect of changes up the hierarchy.

During the Validation Phase, iterations may occur from each stage of test to the corresponding stage of design.

As a consequence of factors acting outside the design, they may also be initiated at any of the stages which have already been completed.

It should be noted that the system has a pyramidal structure. For one system, there will be several subsystems at Subsystem 1 level, more subsystems at Subsystem 2 level and many units. Thus, it can be expected that the incidence of iterative loops will be greater at the lower levels than at the higher levels, although the effects will be greater at the higher levels.

Nature of Iterations.

An iteration is triggered by the identification of a need for a change to a previously designed or specified part of the system. It is, then, the process of making the change.

In the context of the discussion which follows, "iterations" and "design changes" can be treated as being synonomous.

N.B: As in previous Chapters, the term "Design Changes" is taken to mean changes to Specifications and the Documentation as well as to Design.

Causes of Design Changes.

A nomenclature for describing the causes of design changes and their inter-relationship is described in Appendix H.

In Figure 7-9, "External Factors" represents a hierarchy of causes of design changes. That is, it not only represents the immediate (C1) causes but also the less direct (C2, C3, etc.) causes.

Effects of Design Changes.

Outlined below are six effects of design changes. These are primarily predicted from the model of the Development Process but, for completion, some effects not predicted from the model have been included.

(a) Increase of Lead Time.

For every iterative loop in the process, some action must be taken by the development organisation. This may involve redesign, or it may result in no more than an investigation and analysis. Whatever action is required, however, it will take time, and if other factors remain constant this time will contribute to the development and product lead times being increased.

There are three components to the time associated with the iterative loops:

- The time required to find the need for the change,

- The time required to make the change and carry out the redesign,

- The time due to the iteration itself. The fact that it is taking place implies that some effort is being expended. This takes time.

The lead time may also be affected by the need to rebuild models and prototypes.

It is worth noting at this point that the higher up the design structure the iteration goes the greater will be the number of stages of action. It, thus, becomes less attractive to make design changes at the system and higher order subsystem levels than at the lower levels.

This point is noted by Benson <31> who states that the most expensive errors are made early (while system design is taking place) and that these errors cannot be found until system testing takes place.

Deficiencies in the overall product performance may, therefore, be accepted and the System Specification changed when final testing takes place. Alternatively, a new "Mark" or "Release" of the product may be developed. This is often seen as a completely new design and may have new customer requirements included in it.

(b) Increase in Development and Product Costs.

The cost of the development project will be increased directly by the cost of iterations and, indirectly, by the cost of the increased lead time.

The former will be made up of components resulting from:

- Action required to find the need for the change,
- Action required to affect the change,
- Effort to rebuild prototypes, and

- For the production and management of the extra documentation which will be required.

The indirect costs arise through two mechanisms. Firstly, the development time will eat into the product life cycle, and the period available for recovering development costs will be reduced. This is particularly important in the electronics industry where rapid technological changes are taking place and product life cycles are being reduced.

The second mechanism is concerned with the relationship between the market-launch date and the competitive position of the producer. The work of the Boston Consulting Group <32> showed that the first company into the market with a particular product is likely to be the market leader. It is also likely to achieve the lowest unit costs and greater profits.

(c) Documentation Control.

Every time a design change is implemented it must be documented. If the number of changes is high, then it will be difficult and expensive to ensure that documentation is effectively controlled and that compatible sets of documents are maintained.

(d) Quality of Design.

"Revision in design, be they major or minor, always threaten the integrity of the whole design" <33>. Changes may drastically affect the system reliability, or result in a hazardous condition if failure occurs. They may simply result in the design being inferior to that if the flaw resulting in the change had been recognised during the original design or specification.

(e) Motivation of the Design Team.

Engineers prefer to create a new design to having to change and alter other people's designs - or even their own. If there is a need for extensive redesign, motivation and morale will be low and more errors will be made. Members of the design team will leave, expertise will be lost and the psychological group structure will be broken down. Motivation will sink further.

(f) Tying up of Resources.

Resources will be tied up on work which may be avoided, and this will mean that other projects will not be able to go ahead. The overall profitability of the company will be reduced.

7.6. Model Validity.

In this Section, the validity of the model is investigated by comparing some predictions made by the model with practical results obtained from the investigations described in Chapters 3 through to 6. The arguments given are qualitative rather than quantitative.

Model Prediction: That a common model can be used for different development projects.

This is supported by the fact that the change generation patterns for the LPU and SPU developments were similar (Figure 4-3), both in terms of quantities of changes generated and how those changes are distributed across the different entities within the product structure.

Model Prediction: That the change generation patterns will be similar for all entities at a particular level in the product structure hierarchy.

This is supported by Slide-in-units and Wired Shelf Groups showing the similar change generation pattern (Figures 4-1 and 4-2). In terms of the model, these entities are both "Units".

Model Prediction: That design changes cause the development time to be increased.

The first part of this prediction is supported by the analysis in Chapter 6, which shows that, on average, each change to an LPU Slide-in-unit contributed 1.7 hours to the total development time for the LPU.

Model Prediction: That design changes cause the development cost to be increased.

This is also supported by the analysis in Chapter 6.

Model Prediction: That design changes which occur late in the project, at the time of system testing, result in the greatest effects.

This was shown to be true in Chapter 6. However, the effect was caused by practical reasons which had nothing to do with the prediction of the Model.

Conclusion.

While the model predictions discussed above were shown to be true in practice, the model validity was not conclusively proved. However, neither was it put in doubt.

7.7. Summary of Chapter.

From the literature concerned with innovation, design and development, a model, which represents the development process for large hierarchically structured systems, was devised. This model highlighted two aspects of the process:

- Firstly, the hierarchical structure of the product, and

- Secondly, the occurrence of iterative loops which are caused by design changes.

It was further shown that the iterative loops have a number of adverse effects on the potential success of the development project.

Predictions made by the model were compared with practical results which had been obtained by studying the LPU and SPU development project. While this comparison did not conclusively prove the validity of the model, it did not contradict it.

The main conclusion that can be drawn from the model is that three methods can be used to reduce the element of the development time caused by design changes:

1. Reducing the number of changes by reducing the effects of the "external factors" which cause them,

2. Speeding up the execution of the iterative loops themselves, or

3. Finding the need for each change as early as possible in the process, thus reducing the need for a ripple of iterations up and down the system hierarchy.

PART II

CHAPTER 8

HYPOTHESIS OF SOLUTION

8.1. Chapter Preview.

Part I of this thesis was concerned with the problem to be solved by this project; that is, what was causing the lead time of new products to be so long? Part II is concerned with finding a solution to that problem, testing the solution and assessing its effectiveness. The present Chapter is dedicated to the first of these, namely, finding a solution to the problem.

It commences by restating the pertinent aspects of the problem (Section 8.2). It then argues the merits and disadvantages of three basic methods of overcoming these (Section 8.3), and the most suitable is selected. The philosophies behind the approach are discussed (Section 8.4), and a number of techniques to achieve it outlined (Section 8.5). It is left to Chapter 9 to discuss which of these most suitably form the basis of an improved development process.

The Chapter closes with a summary of its main conclusions (Section 8.6).

8.2. Main Features of the Problem.

The main task of the project was to cut the lead time for getting new products into volume production. In order to formulate a solution for this task, a study of the causes of long lead times was carried out and the results presented in Part I of this thesis. The main conclusion of this work and, therefore, the main features of the problem were:

- An important factor influencing the lead time was the time required to process design changes, and

- The most common cause of design changes was Design Faults and, in particular, Logic Design Faults affecting Slide-in-units.

A study was performed to identify the causes of Design Faults (see Chapter 5), and some light was thrown on these. It was not possible, however, to make quantitative measurements of their importance. Also, the most important reason for Design Faults indicated by the study (deficiences in specifications) was contradictory to the definition of a Design Fault and, therefore, of suspect validity.

It was indicated above that the most common cause of design changes was Design Faults. This does not mean that Design Faults had a greater effect on the development time than other causes of change. It was not possible to prove this. It just indicates that there were significantly more Design Faults than other causes of changes.

8.3. Three Approaches for Reducing the Lead Time.

Consideration of the main features of the problem listed above leads to the proposition that to reduce the total effects of Design Faults would offer a promising way of significantly reducing the development time. Three approaches to do this need consideration . They are:

1. To eliminate the causes of Design Faults (C2 - Causes of design changes),

2. To reduce the effects of the iterative loops associated with design changes, i.e. speeding them up, and

3. To detect and remedy faults during the design stage before the design is committed to hardware.

Each is discussed below.

Eliminate the Causes of Design Faults.

The basis of this approach is to remove the 'External Factors' (See Figure 7-9) which cause Design Faults to occur and which prevent the "perfect" design from being achieved every time. This offers, in theory, the maximum savings and is, therefore, the "ideal" solution which should in any event be sought after. There are, however, three major disadvantages associated with the practical aspects of it:

- The method only aims to eliminate the causes of Design Faults. Design changes will still occur as the result of other Cl-causes, such as Specification Faults and Clerical Errors. It, therefore, only aims to affect about 50% of design changes,

- The number of causes of Design Faults is large, and although the research described in Chapter 5 indicated the development staff's view of the most important, it did not give a sufficiently reliable and definitive result to justify concentrating effort in this direction - particularly as the top rated cause was contradictory to the definition of a Design Fault,

- For practical reasons, it would not be possible to eliminate all the causes of Design Faults (even if they had been more clearly identified), and so, again, design changes would still occur in significant numbers.

Reduce the Effects of Design Changes by Speeding Up the Iterations.

This method aims to cut the development time by reducing the effects of design changes through speeding up the iterations shown in Figure 7-9. It would do this without regard for the cause of the change, and therein lies its major advantage. It does, however, have three disadvantages:

- It is the least "ideal" method as it tackles only the symptoms, paying no attention to the causes,

- Although savings may be made by speeding up the iterations, they will still take time, cost money and tie up resources,

- It is unlikely that major savings could be made using this method as the management responsible for the related areas already has as one of its aims the ongoing objective of speeding up the iterations.

Detect and Remedy Faults During the Design Stage.

The basis behind this method is to detect Design Faults during the design phase and to remedy them at that time. It, therefore, eliminates the need for the iterative loops to run from 'test' back to a redesign by not allowing the faults to be physically built into the product or prototype. The approach, while not as ideal as eliminating the causes of Design Faults, is an improvement over simply speeding up the iterations.

Like the previous approach, it should already be management's aim to find ways of minimising the number of Design Faults passed outside the design activity. As the evidence in Chapter 4 shows, however, this had not been done with total success. There are two probable reasons for this:

- The management had not, until this project was initiated, carried out detailed studies into the causes and effects of design changes and, therefore, did not appreciate their significance,

- This approach implies that more time must be spent early in the project to achieve greater savings later on. This works against pressures for the design team to show "progress".

Although this method is primarily aimed at reducing the effects of Design Faults, it was thought that it might be possible to make it more general and to devise techniques to detect other 'C1-causes' of design changes as well. It is helped in this respect by not taking regard of the 'C2-causes' of design changes.

8.4. Philosophies behind the Solution.

When it came to choosing which solution to adopt, the first approach had to be ruled out because of the uncertainties over causes of Design Faults, and the second because of its lack of potential. It was decided, then, that the solution to be tested would be based on the third approach, which was to detect and remedy the Design Faults during the design phase. The aim would be to reduce the number of iterative loops passing through test and back to redesign. In particular, the solution would concentrate on the 'units' level of equipment, as shown in Figure 7-9.

Various techniques which could be employed to achieve it are discussed in the Section following. The basic philosophies behind it are explained below.

The solution which it was proposed to adopt would work through two separate but related philosophies. The first is:

That by spending more time at the early stages of development (detecting and remedying faults in the design), even greater savings would be made later on.

The second, the principle of which is mentioned by Kurkjian <34> is:

That the collective performance of the development organisation can be improved by increasing the amount and speed of feedback about that performance.

It should be noted that this does not imply that the performance of the individuals would not improve; indeed it would. The sum of the individuals performance would be reflected in the overall performance, and it is that with which the method is concerned.

It will become clear how the first of these philosophies could be achieved in practice in the Section following, but it is worth commenting here how the second would be realised. This can be done by considering the LPU development.

At the time that the development process was studied, it was many weeks between designing the logic of Slide-in-units and commissioning them, and even longer before the changes were used in a rework. There was some evidence (although not proved) that this had the following consequences:

- The original designers had sometimes left or been moved by the time the rework was done so that someone else had to take time to make themselves familiar with the circuit, or, more likely, they simply incorporated the changes without regard to their more obscure effects,

- The Design Engineers repeated similar faults on other Slide-in-units they designed during the intervening period between the designing and commissioning of their first one,

- It was not known until well into the development what quality (in terms of faults or their absence) of design would be achieved and so what level of resources would be required to handle the changes,

- It was not known until too late that special measures to reduce the number of Design Faults could have had significantly beneficial results.

All these consequences could have been alleviated if the faults had been found earlier in development (during the design phase) and if they had been monitored. This would probably have also resulted in greater motivation for producing 'correct' designs.

8.5. Possible Techniques for Detecting Design Faults.

The purpose of the present Section is to outline a number of techniques which could be used to detect Design Faults during the design phase of a development project. This is to show that the philosophies discussed in the previous Section could be viable for cutting the development time. It was not the intention that all of the techniques listed would have their effectiveness tested in practice. Some would have been impractical in the circumstances of the Sponsoring Company and some would have required further investigation for which there was not time during this project. The selection of those to be tried is discussed in Chapter 9. The techniques are listed and discussed below (in no significant order).

Logic Simulation.

A tool to do this, "TEGAS" (see Johnson's paper <35>), was available within the Sponsoring Company. It had, for reasons stated in Appendix B, only been used on a limited scale. Two of these reasons, those concerned with the poor quality of the software and writing of the 'macros', had, however, been largely remedied by the time this project had reached the phase of testing the solution. The development staff were confident by this time that the system was in a satisfactory condition to be used in earnest.

Most logic simulation systems, TEGAS included, are used to test complete chunks of circuitry after they have been designed. At the time of writing, however, there was some talk of developing systems which can be used interactively to test the circuit progressively as it is designed. While such a system would be an ideal implementation of the philosophy to be tried, no satisfactory one is yet available.

An outline of the operation and capabilities of TEGAS is presented in Appendix I

Design Reviews.

Essentially, a 'design review' consists of an examination of a design by people other than the original designer, usually concentrating on one or more particular aspects of the design, such as correct functional operation, manufacturability, testability, reliability, etc.. They can take many forms; being formal or informal; carried out by large or small groups; dealing with design principles or design details.

'Design reviews', the value of which is emphasised by Corfield <36>, often form an integral part of the assessment procedures for Government development contracts <37,38>. The NASA approach is detailed by Boss and McGaffin <39>. They are also described by Jacobs <40> who list the following advantages which they give:

- Maturity of the design is accelerated,
- Cost improvements are frequently achieved,
- Delivery dates are frequently improved,
- Customer design approval is sometimes obtained,
- New product lines may be generated,
- Staff capabilities are improved,
- Design changes are reduced,
- They have formed the basis for successful defences in Product Liability cases.

Checking Techniques.

These can be divided into three groups with different objectives:

1. To check the design against design rules and standards. This is usually comparatively simple because the design rules and standards are

documented and can be listed in such a way as to aid checking.

2. To check the functional performance of the design. This is the most difficult objective to satisfy.

3. To check the design reliability and failure modes and effects. It can be done, up to a point, using mathematical algorithms, but often requires specialist expertise.

The main difficulty, which any checking technique has to overcome, is that people are not very good at checking, and in particular at checking their own work. They "see" what they expect to see and if there is a flaw in the logic design they will very often overlook it. The methods below can help to overcome this:

- The check is performed by an engineer other than the one who designed the circuit: This is an improvement over an engineer checking his own work, but if the design is complicated it may take a long time, be frustrating for the checker (leading to low morale) and may not be effective.

- The Use of Checklists: This is simple to implement for checking the design against design rules and standards because they will already be listed in some form. It is far more difficult to implement for checking the functional performance of the design.

When the LPU and SPU design changes were investigated, there was some evidence to indicate that certain types of Design Fault were more common than others. If this were true, then checklists of commonly faulty areas could be generated from an analysis of previously found faults and used to direct checking towards those problem areas. The technique would not result in the detection of all faults, or necessarily the most serious faults, but it could significantly improve the efficiency of checking. The use of checklists is described by Roberts <41>.

- The Use of a Quality Assurance Department as Independent Checkers: While this technique may be satisfactory for checking that design rules and standards are followed, it will be unsatisfactory for checking the functional performance of the design. Its main advantage, however, is that it can be used to ensure that other checking is carried out.

- Build Checks into the Computer Aided Design System: Some checks can be built into the CAD system so that they have to be passed before, say, Printed Circuit Board layout and tracking can be undertaken. These checks can only verify that the design rules and standards have been followed.

The Computer Aided Design system used by the Sponsoring Company did contain some checks, but more could have been added. It was also possible to progress to later stages of design even though the tests had failed or not been carried out.

- Incentive Schemes: The detection of Design Faults may be improved through incentive schemes to improve motivation. These could take the form of payments based on the effectiveness of checking or taking account of the quality of designs in personal assessments.

'Bread Boards'.

This technique consists of constructing a circuit in the quickest possible manner to test the function of the design. The technique has the following disadvantages:

- As the functioning of high speed circuits is dependent on the layout and interconnections of these circuits, the "bread board" can give unreliable results,

- It tends to lead to a considerable amount of time being wasted on unprofitable experimentation.

8.6. Summary of Chapter.

Three possible approaches for reducing the effect on the development time of design changes were:

- To reduce the causes of Design Faults,

- To speed up the iterative loops,

- To detect and correct faults during the design phase.

The last of these offered the best opportunity for a significant improvement. It would work through two separate philosophies:

- By spending more time early in development even greater savings could be made later on,

- By increasing the amount and speed of feedback about the performance of the development organisation, that performance could be improved.

A number of techniques which could be used to realise these philosophies were discussed.

CHAPTER 9

THE TRIAL

9.1. Chapter Preview.

This Chapter describes how the philosophies proposed for reducing the lead time were tested by applying some of the techniques discussed in the previous Chapter. It opens by outlining how the development process could be altered to include the new techniques (Section 9.2). This is followed by a discussion on how the effectiveness of the principles could be tested in the environment of the Sponsoring Company (Section 9.3). It explains that it was necessary to perform the experiment with 'Rework' Slide-in-units with one group being used as a 'Trial' group and another as a 'Control' group. The similarities and dissimilarities of the two are compared in Section 9.4.

The Chapter then sets out the results of the experiment (Section 9.5), gives a critique of it (Section 9.6) and presents a summary of the conclusions which can be drawn from it (Section 9.7).

9.2. Proposed New Process.

Figure 9-1 exhibits the original process for the development of Slide-in-units as it occurred in practice. Although the process is described in Appendix B, it is represented here in a slightly different form to emphasise those activities which are relevant to the arguments put forward in this Chapter.

The proposed new process is represented in Figure 9-2. All the original stages remain except for the Team Leader Design Check which, due to the work loads of the team leaders, had not been performed effectively. The two new stages are discussed below.

Run Logic Simulator.

In the Sponsoring Company, the 'TEGAS' system could be used in either its 'Mean Value Simulation Mode' ('Mode 2') or its 'Worst Case Tolerence Mode'. A study had been performed to assess likely gains to be made by using the latter mode in place of or as well as the former, but this had shown only a limited improvement. The stage would, therefore, only entail the running of the 'TEGAS Mean Value Simulator' and would preferably be performed by an engineer other than the original designer.

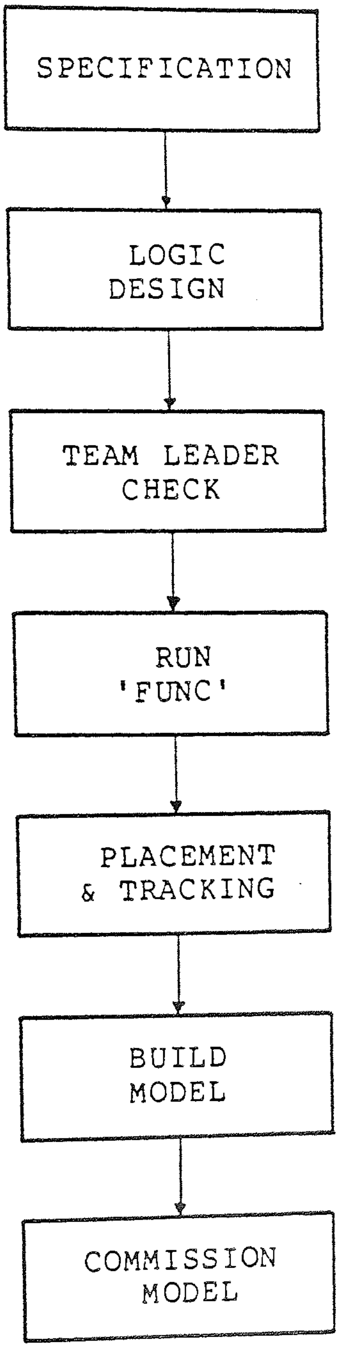


Figure 9-1. Original Process for the Development of Slide-in-units.

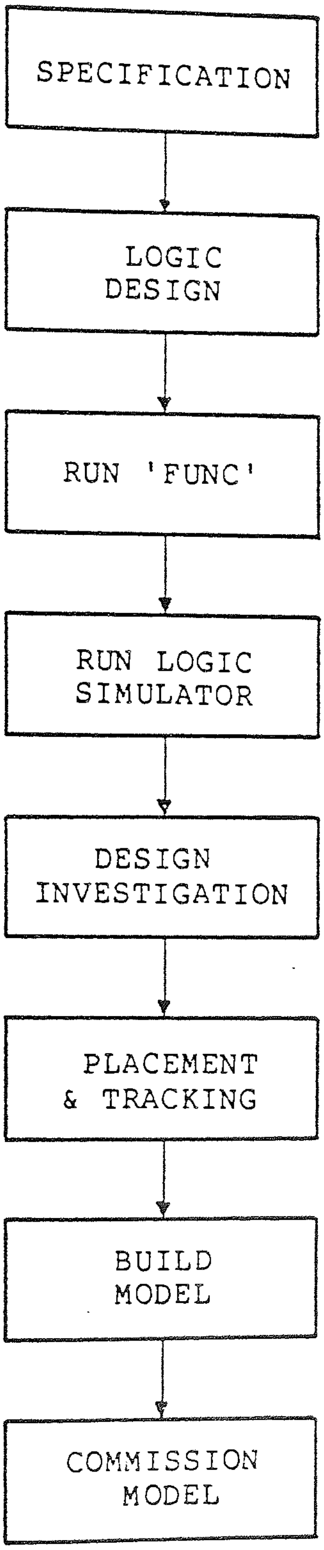


Figure 9-2. Proposed New Process for the Development of Slide-in-units.

Design Investigation.

This stage would in effect be a design review involving design checking by engineers with as wide a range of experience as possible and would include:

- A manual check of the design's functional performace, preferably performed by an engineer other than the original designer and preferably guided by a checklist,

- Reliability, Failure Mode, Manufacturability and Testability investigations,

- A Quality Assurance investigation to ensure the design rules were upheld.

There is one further feature of the new process. That is that the 'placement and tracking' functions would not be allowed to be carried out until 'FUNC' had been run with no faults detected and the new stages completed satisfactorily. The go-ahead would require authorisation from the relevant Team Leader and the Design Quality Assurance Department.

Note: 'Bread Boards' were judged to be unsuitable for the technology employed in the product. Another technique mentioned in Chapter 8, 'Incentive Schemes' forms part of the much wider subject of motivation and will not be included here.

9.3. Implementation of the Trial.

The present Section takes the proposed new development process and discusses the options for testing its effectiveness. Consideration is given to the various practicalities and the circumstances in the Sponsoring Company at the time when the trial was expected to take place.

In setting up the trial, three issues had to be decided:

- How the effectiveness of the new process would be assessed,
- On what Slide-in-units the experiment would be performed,
- Precisely what new process would be tested.

Each of these is discussed below.

Method of Assessment of the Effectiveness of the New Process.

As the objective of the project was to improve new product lead times, the effectiveness of the new process could be satisfactorily assessed by comparing its performance directly with the performance of the original process. In practice, there were three ways this could be done:

1. Two similar and comparable groups of units would be employed. One would be exposed to the new process and the other would go through the original process. A comparison could then be made of the resulting development times for the two groups. The method, however, suffered from the difficulty of ensuring the two groups were comparable.

2. The same sample of 'units' would be put through the two processes by two groups of engineers and the respective development times compared. With this method there would be the difficulty of ensuring that the design teams were of comparable ability. It would also suffer from the overriding problem of being expensive on resources. The validity of the results could also be harmed by:

   - Communications taking place between the two development teams, and

- The two designs for each 'unit' being different for reasons other than the differences in the processes.

3. 'Units' already designed and commissioned would be designed again, starting from the original specification, utilising the new process and new staff. The disadvantages of this method would be:

   - That resources would be wasted,

   - The designs might not be the same, thus invalidating the comparisons,

   - Faults produced earlier might be known by the team designing the units for the second time.

A variation on the last two methods would be to have the logic design stage performed only once but have the subsequent stages exposed to the two processes in parallel. Whilst this would overcome the problem of having different designs, there would still be other difficulties; the most important being that resources would still be tied up, not producing progress for development.

For the reason of the availability of resources, the first method, using two different sets of 'units' for the trial and control groups, had to be employed.

Selection of 'Units' for the Trial.

Ideally, the trial would have been performed on 'units' being designed for the first time from their specifications. In practice, however, there were no such 'units' planned to be designed within the timescale of this project. It was, therefore, necessary to use 'units' being 'reworked' and to treat the 'rework' as though it were an original design stage.

Whilst this was the only practical way the philosophies could be tested, it led to great difficulties in interpreting the results and relating them to new design.

The Trial Process.

Just as it was not possible to use new designs to test the proposed process, so it was not possible to include all the activities of the process described in Section 9.2. The following had to be omitted:

- Reliability, failure mode, manufacturability and testability investigations: because the necessary expertise was not available,

- The design function check by an engineer other than the one who had reworked the circuit: because of pressure on resources.

It was, nevertheless, possible to include:

- Run 'FUNC',

- TEGAS mode 2, Mean Value Simulation Mode, but carried out by the rework engineer,

- QA check of adherence to the design rules.

The new trial process is presented in Figure 9-3, which also shows the activities introduced to produce test results.

9.4. Selection and Comparison of the Two Groups of Slide-in-units.

The two groups of Slide-in-units were initially selected on the following basis:

- The 'units' were planned to be reworked at the time of the trial,

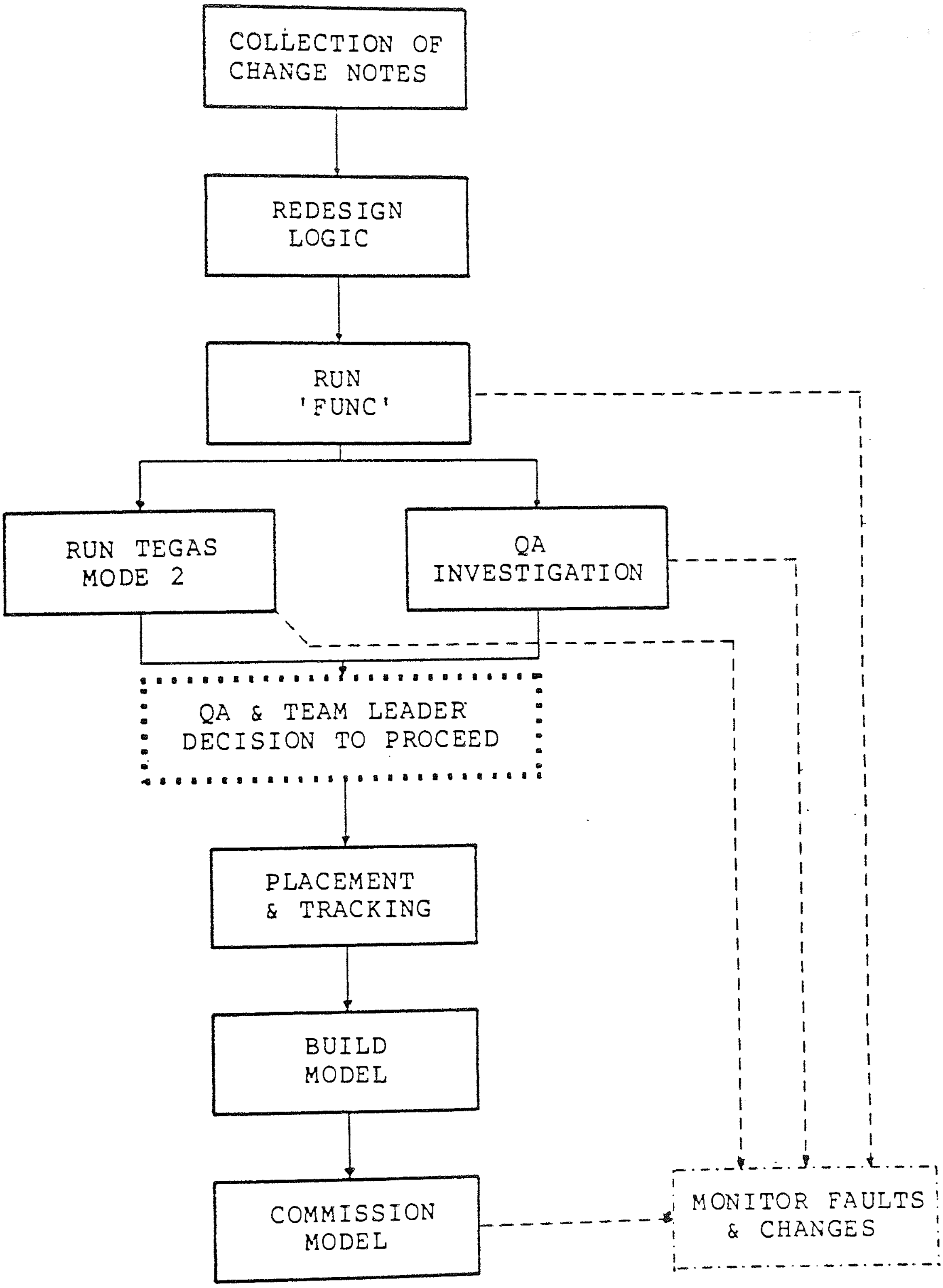


Figure 9-3. The Trial Process.

- The priorities of their reworks - Slide-in-units for which the reworks were urgently required could not be included in the 'Trial' group,

- The suitability of the Slide-in-units for the new techniques - Analogue and microprocessor circuits could not be included because of the inability of TEGAS to simulate them.

These gave four 'Trial' 'units' and five 'Control' 'units'.

Table 9-1 compares the following factors for the two groups:

- The 'complexity' of the circuits as measured by the number of logic devices,

- The number of previous reworks,

- The number of 'Change Notes' included in the 'trial' rework,

Note: The Slide-in-unit identification numbers are used for identification purposes only.

	'UNIT' NUMBER	COMPLEXITY	NUMBER OF PREVIOUS REWORKS	NUMBER OF 'CHANGE NOTES' IN REWORK
TRIAL GROUP	T1	108	0	22
	T2	160	1	16
	T3	154	1	28
	T4	139	0	46
	MEAN	140	0.5	28
CONTROL GROUP	C1	112	1	4
	C2	104	0	14
	C3	103	0	6
	C4	106	0	5
	C5	145	0	24
	MEAN	114	0.2	10.6

Table 9-1. Comparison Between the Slide-in-units of the 'Trial' and 'Control' Groups.

9.5. Results of The Experiment.

To assess the effectiveness of the proposed new process, it was necessary to measure the reduction, which results from it, in the number of Design Faults which get passed to manufacturing, and in the effort required to correct them. This had to be compared with the extra effort expended in looking for faults.

The reduction in the number of faults not detected during the rework and checking stages was assesed in two ways:

1. By comparing the number of faults found in, and subsequent to manufacture (in commissioning), for the two groups of Slide-in-units, and

2. By measuring the number of faults found in the additional checking stages as a percentage of all faults produced by the rework for the trial group, and assuming that the resulting percentage would hold good for all units and for new designs.

The results of applying these two methods are presented below.

Comparison of Faults Found Subsequent to Rework.

The number of faults found subsequent to the rework (represented by the number of 'Change Notes' produced) for each Slide-in-unit in the Trial and Control groups is shown in Table 9-2.

This shows that the 'Control' Slide-in-units had marginally more changes produced for them than did the

	UNIT NUMBER	NUMBER OF 'CHANGE NOTES' PRODUCED SUBSEQUENT TO REWORK
TRIAL GROUP	T1	0
	T2	1
	T3	1
	T4	2
	MEAN	1.0
CONTROL GROUP	C1	1
	C2	0
	C3	0
	C4	3
	C5	2
	MEAN	1.2

Table 9-2. Numbers of 'Change Notes' Produced Subsequent to Rework for 'Trial' and 'Control' Slide-in-units.

'Trial' Slide-in-units. It is, however, difficult to draw any conclusions about savings that would be made against a new design as opposed to a rework from this result.

Faults Found During the Checking Stages.

The numbers of faults found during the checking stages for the 'Trial' Slide-in-units are presented in Table 9-3. This also shows the numbers of design rule contraventions. These were shown because:

UNIT NUMBER	NUMBER OF FAULTS FOUND:		
	DURING 'FUNC'	DURING TEGAS	DURING QA
T1	0 (0)	1 (0)	12 (12)
T2	1 (1)	1 (0)	8 (8)
T3	1 (1)	7 (7)	14 (14)
T4	4 (0)	3 (0)	12 (12)
MEAN	1.5 (0.5)	3.0 (1.8)	11.5 (11.5)

Table 9-3. Numbers of Faults Found During Checking Stages of New Process. (The figures in brackets show the numbers of Design Rule Contraventions.)

(a) They emphasise the large number of faults of this category,

(b) Similar faults were probably widespread, but undetected in the 'Control' group Slide-in-units, and

(c) Most of them would have been the result of the original design and not the rework, and, therefore, could not be included in the assessment of the results.

Tables 9-2 and 9-3 together show that, excluding design rule contraventions, for the 'Trial Group':

- The total number of faults found by the additional checking stages was 5,

- The total number of faults generated by the rework (found in the additional checking stages and commissioning) was 9,

- The percentage of all rework faults found by the additional checking stage was 56%.

Time Required for Checking Stages.

The time required for the new checking stages, running TEGAS and Quality Assurance, was assessed in two ways:

1. By comparing the times taken to complete the reworks for the 'Trial' and 'Control' groups of Slide-in-units. This had the advantage that it took into account how the checking stages affected subsequent stages through mechanisms other than reductions in the number of faults.

2. By measuring the durations of the new checking activities as they were performed.

The results produced by method '1' are presented in Table 9-4, which shows that on average the 'Trial' Slide-in-unit reworks took 5.8 weeks longer than did those of the 'Control' Slide-in-units.

Note: The "Rework Duration" was taken as the time from the start of the rework until the Printed Wiring Board artwork was approved.

The results produced by Method '2' are presented in Table 9-5. The times taken to perform the Quality Assurance checks are not included because they were performed by people other than the design engineers, in

	UNIT NUMBER	REWORK DURATION (WEEKS)
TRIAL GROUP	T1	9
	T2	10
	T3	20
	T4	13
	MEAN	13.0
CONTROL GROUP	C1	12
	C2	6
	C3	2
	C4	4
	C5	12
	MEAN	7.2

Table 9-4. Durations of Reworks of 'Trial' and 'Control' Slide-in-units.

parallel with the TEGAS simulation, and took about two weeks. They, therefore, did not affect the rework durations. The amount of effort required for them was about 40 man-hours.

The average time taken to perform the TEGAS simulation was 7.5 weeks. This compares with the implied figure of 5.8 weeks for the checking stages produced

UNIT NUMBER	TIME TAKEN FOR TEGAS SIMULATION (WEEKS)
T1	3
T2	8
T3	16
T4	3
MEAN	7.5

Table 9-5. Times Taken to Perform TEGAS Simulation on 'Trial' Slide-in-units.

from the first method. This difference may be accounted for in three ways:

1. Because of dissimilarities in the two groups of Slide-in-units,

2. Because of differences in the abilities or motivation of the Engineers working on the control and trial groups of Slide-in-units, respectively, or

3. Because of the interacting effects between the checking and subsequent stages of the rework.

Since the figures in Table 9-1 tend to suggest that the Slide-in-units in the Trial Group were more complex than those of the 'Control' group, "1" above appears not to be the explanation. No quantitative measurement was made to prove "2" but there was no evidence of it either. It, therefore, must be assumed that "3" offers the explanation for the difference, and the figure of 5.8 weeks is the more accurate. It will be used in subsequent calculations.

Nett Savings.

The nett savings in development effort resulting from the additional checking stages are presented below:

(a) For Reworks:

The mean reduction in the number of faults was 1.2 per Slide-in-unit (see Table 9-3: 3.0 less 1.8 Design Rule Contraventions), which is equivalent to 41 man-hours per Slide-in-unit using the figure of 34.3 man-hours of development effort required to process each change (See Chapter 6).

Cost of achieving this was 218 man-hours.

The calculation above excludes Design Rule Contraventions on one hand and the running of 'FUNC' on the other.

If the Design Rule Contraventions were critical, which for 'development models' they were not, then the reduction in the number of faults would have been greater.

(b) For a New Design:

For the LPU project, there were about 1400 changes to Slide-in-units. Assuming the additional checking stages would have made a 56% saving to these gives a reduction of 784.

With 70 Slide-in-units, the saving per Slide-in-unit would have been 11.2 changes.

Using the figure of 34.3 man-hours of development effort required to process each change yields a saving of 384 man-hours per Slide-in-unit.

The cost of achieving this would have been 218 man-hours.

The Nett saving = 166 man-hours per Slide-in-unit.

These figures again exclude Design Rule Contraventions and the running of 'FUNC', which had been used for Slide-in-units during their initial designs.

(c) For the Whole LPU Project.

Nett Savings per Slide-in-unit = 166 man-hours

Nett Savings for all 70 Slide-in-units = 11620 man-hours (310 man-weeks)

With 20 Engineers working on the project: Total saving to development time = 15.5 weeks

Given a total development time of about 3 years: Percentage reduction in development time = 11.2%

Note: If the Design Rule Contraventions are taken into account, then the additional checking stages would have reduced the number of changes caused by the rework by 93%. Performing a similar calculation of that above indicates a reduction of 39 weeks for the total development time, or 28% of it. This result, however, lacks credibility because there was no evidence that any of the 1400 changes to the LPU Slide-in-units were caused by Design Rule Contraventions.

9.6. Critique of the Experiment.

When the experiment was performed, circumstances dictated that less than perfect methods had to be used.

The purpose of the present Section is to discuss how this affected the experiment and the results obtained. The following topics are covered:

- The validity of using Reworks,
- Comparison between the two groups of Slide-in-units,
- Sample Sizes,
- Validity of the time taken to run TEGAS,
- Faults remaining undetected,
- Effects of the experiment on the performance of the Engineers.

The Validity of Using Reworks.

There would have been two main differences between using reworks and new designs for the trial of the new process.

Firstly, the number of Design Faults resulting from reworks were smaller than for new designs. The new procedures, therefore,appeared to be less effective than if they had been tried on new designs, and the results obtained less accurate.

Secondly, the distribution of types of faults may have been different for rework than new designs. It is unclear how this would affect the results.

Comparison Between the Two Groups of Slide-in-units.

The 'Trial' and 'Control' groups of Slide-in-units are compared in Table 9-1. This showed that differences occurred for the following factors:

- The Complexity of the Slide-in-units,

- The number of 'Change Notes' included in the reworks,

- The number of previous reworks to the trial taking place.

For each of these, the mean figure for the 'Trial' group was greater than that for the 'Control' group. The 'Control' group was only used to establish the extra time taken to perform the additional checking stages. This is likely to have been exaggerated by the differences in the two groups and the total savings in development time under-estimated.

Sample Sizes.

The circumstances at the time the experiment was carried out resulted in the samples of Slide-in-units, which constituted the 'Trial' and 'Control' groups, being small; 6 percent of the LPU Slide-in-units. Also, because reworks had to be used, the numbers of faults and changes were small.

The experiment, however, showed that the new checking stages could give some reduction in the number of faults which pass into manufacturing.

Validity of the Time Taken to Run TEGAS.

Table 9-5 showed that it took, on average, 7.5 weeks to run the TEGAS simulator for each Slide-in-unit. For the following reasons, this figure is probably higher than could have been achieved if the simulator had been used in a larger number of cases:

- The engineers performing the rework had had no previous experience of using TEGAS,

- A number of problems were encountered with the TEGAS software and 'Macros'. These were all quickly corrected but if TEGAS had been used more frequently they would probably already have been eliminated. Table 9-6 shows the extent of these problems.

The cost of achieving the reduction in the number of faults would probably be lower than indicated by the experiment, if logic simulation were used more extensively.

In addition to this, the running of the TEGAS <u>logic</u> simulator would probably result in a reduction in the time required to run the TEGAS <u>fault</u> simulator. The latter times were not included in the rework durations.

UNIT NUMBER	FAULTS IN TEGAS SOFTWARE	FAULTS IN 'MACROS'
T1	1	0
T2	4	5
T3	-	1
T4	-	-

Table 9-6. Problems Encountered When Performing TEGAS Simulation.

Faults Remaining Undetected.

It is possible that at the time of writing this thesis, faults in the 'Trial' group Slide-in-units remained undetected. It is not possible, however, to state quantitatively how this would affect the results.

Effects of the Experiment on the Performance of the Engineers.

When performing an experiment on the working procedures of people, the performance of these people inevitably improves because the experiment is taking place and attention is being paid to them. This is known as the 'Hawthorne Effect' <42>. In order to minimise this effect, the following measures were taken:

- Researcher intervention was kept to a minimum. As far as possible, all activities were performed by the people who would have performed them if the process had been established. Some intervention was, however, required with the 'Trial' group in order to specify the new process and persuade the participants to follow it. Interaction was also needed to collect the results.

- As far as possible, the amount of attention paid to the two groups was balanced. For the reasons given above, this could not be perfectly achieved.

The overall consequence of the Hawthorne Effect was probably to cause the resuls of the experiment to be better than they should have been.

Overall Effect of Inaccuracies.

It is not possible to state precisely what the overall effect of the factors discussed above would be, but there is an approximate balance in the number causing the results to be deflated and the number causing the results to be inflated.

Therefore, even though there may be a question over the accuracy of the results, it can be taken that the hypothesis under test was proved true.

Other Factors Influencing the Total Savings for the Whole LPU Project.

It has been shown that a saving of 11.2% of the LPU development time would have been achieved by the introduction of the additional checking stages. For two reasons, this would be an under estimate of the potential savings:

1. Only the savings to the Slide-in-unit changes were taken into account. The same philosophy would probably have resulted in savings in the development of the other equipment entities.

2. Design Rule Contraventions were not taken into account because there was no evidence that the 1400 LPU Slide-in-unit changes were caused by Design Rule Contraventions. While this was true at the time of writing, it may be assumed that all the Slide-in-units contained faults of this category and that eventually these faults would have to be corrected.

9.7. Summary of Chapter.

An experiment which was designed to test two hypotheses for reducing the development time of new products was performed on a group of Slide-in-units being reworked. The experiment showed that by introducing new checking stages into the development process the following savings would be made:

- For Reworks:

41 man-hours of effort per Slide-in-unit, but at a cost of 218 man-hours (a nett extra effort of 177 man-hours),

- For New Designs:

A nett saving of 166 man-hours of effort per Slide-in-unit,

- For the Whole LPU Project:

310 man-weeks of effort, or 15 weeks (11.2%) of the development time.

For a number of reasons, the experiment had to be carried out in less than ideal conditions and, as a result, the figures above may not be highly accurate. The results do, nevertheless, tend to support the hypotheses under test.

PART III

CHAPTER 10

DISCUSSION OF SOLUTION

10.1. Chapter Preview.

The experiment described in the previous Chapter tested one method for reducing the development time of a new product in a particular set of circumstances. The purpose of the present Chapter is to examine what conclusions can be reached about justifying the results of the experiment for more general applications.

The Chapter opens by discussing the two philosophies behind the solution under test (Section 10.2), and whether the results of the experiment verified these philosophies, or just the techniques used (Section 10.3). It argues that the philosophies were proved, but goes on to discuss whether both were verified, or just one of them (Section 10.4). It then highlights the limitations of the objectives of the experiment (Section 10.5) and how the solution can be justified for more general application than that under test (Section 10.6). Section 10.7 summarises the Chapter.

10.2. The Philosophies Under Test.

In Chapter 8, two philosophies were proposed for reducing the development time of new products. These are restated here for the convenience of the reader.

The first philosophy was:

> "By spending more time at the early stages of development (detecting and remedying faults in the design), even greater savings would be made later on".

The second was:

> "The collective performance of the development organisation can be improved by increasing the amount and speed of feedback about that performance".

While these philosophies were presented as being separate, it was suggested that they were closely related. Just how close becomes apparent on considering the following arguments:

> Spending more time at the early stages of development, detecting Design Faults is in effect increasing the speed of feedback about the design and the performance of the designer.

Conversely, in order to increase the speed of such feedback, it is necessary to detect and find the faults earlier than is currently the case.

Although these arguments show that there is a close relationship between the two philosophies, they are, however, separate. It is certainly possible to devise methods of increasing the quantity of feedback without having to detect the faults any earlier.

10.3. The Experiment - A Test of the Philosophies or the Techniques?

The results of the experiment showed that, within the limitations that will be discussed in Section 10.5, a process including certain checking techniques would probably make a significant difference to the development time of new products.

It can, therefore, be said that the techniques used in the experiment were tested by it. One of these, TEGAS, however, was shown to have been in an imperfect state. Other logic simulation systems or, indeed, other techniques, such as those listed in Chapter 8, might have been able to produce even more beneficial results. Thus, it can be argued that the philosophies, irrespective of the techniques used, were tested along with the techniques.

10.4. Which Philosophy Was Tested?

Although the two philosophies are closely related, it would be useful to know which could be the most powerful so that the most suitable techniques could be incorporated into the development process. It is, therefore, desirable to know whether the results of the experiment were due to one more than the other.

On examining the experimental process, the following conclusions about its pertinence to the two philosophies become apparent:

- The process was a direct conversion of the first philosophy (concerned with spending more time early in the process) into practice.

- The second philosophy (concerned with feedback) was also present. The fact that faults were to be found earlier in the development process meant that feedback about the quality of the design was bound to happen more rapidly. In the experiment, this feedback was only used for correcting the design, not to its maximum by also optimising the design method.

The experiment, therefore, tested a combination of the first philosophy and the second. Since the two philosophies were working together, however, it is not possible to state which, if they could be separated, was proved to be the most effective.

10.5. Limitations of the Objectives of the Experiment.

Although the experiment was designed to test the effectiveness of the two philosophies, it could only do this within various limitations. These are discussed below.

Optimum Amount of Checking and Feedback.

The first limitation of the experiment was that it only tested the philosophies to see whether they worked when implemented in one particular set of techniques. It is likely, however, that the effectiveness of the first philosophy would be dependent on the amount of effort expended in the checking phase. A likely representation of the relationship between the amount of effort used and its effectiveness is represented by the shape of the curve shown in Figure 10-1. This shows that there is an optimum amount of checking above which its effectiveness drops off.

Through the experiment, no attempt was made to try to evaluate what the maximum saving would be or how much effort must be expended to achieve that saving. Indeed,

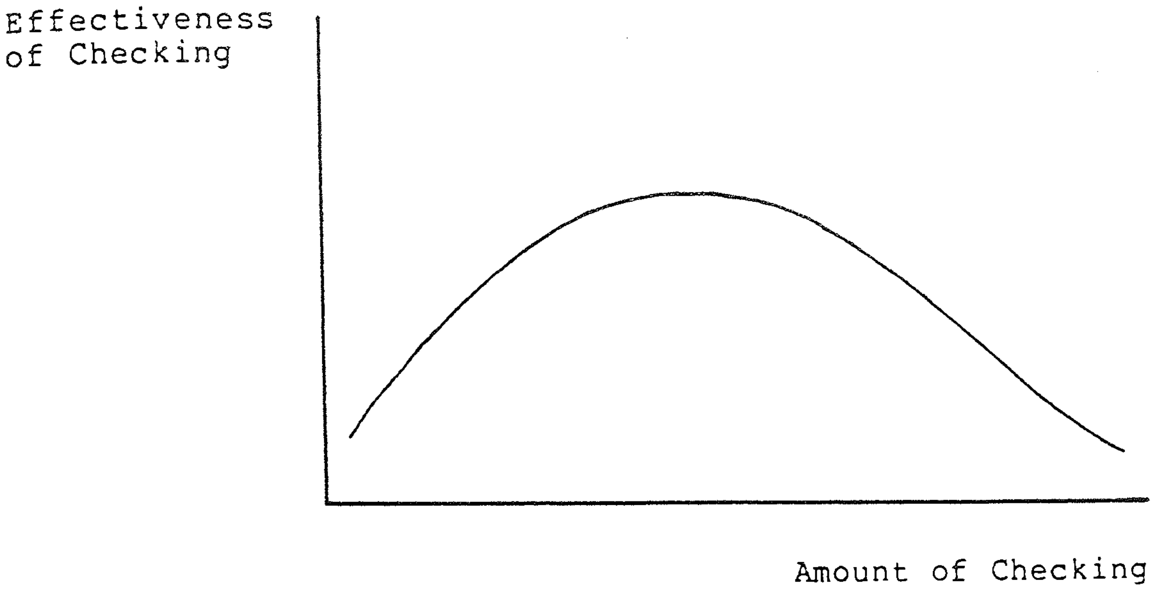


Figure 10-1. Likely Relationship Between the Amount of Checking and the Effectiveness of that Checking (In Terms of e.g: Reduction of Development Time).

it did not even indicate where on that curve relative to the peak the trial process would be represented.

Similar comments can also be made about an optimum amount of feedback.

Product Type and Product Technology.

The experiment was only performed on the development of part of one product type and so could only indicate the values of the philosophies for that case. The limitations of that case were as follows:

- The product type was that of a large electronic system,

- The level in the product system hierarchy was that of the 'Unit' (see Figure 2-1) or the lowest level of assembly above the components,

- Only hardware was involved,

- Only digital (logic) circuits were included.

The effects of removing these limitations of the product type are discussed in the following Section (10.6).

Environment.

The third limitation of the experiment was that it was only carried out in one environment. Apart from the more general characteristics of that environment, such as the organisation of the development laboratories and the motivation of the staff, two factors played an important part in affecting the experiment.

The first was the number of faults being made during design. It is not known how this would compare for other products of a similar complexity or for the same product being designed by another organisation.

Clearly, the effectiveness of the philosophies would increase with the number of faults being made.

The second environmental factor was the level of managerial pressure exerted to get progress and for that progress to be visible in the form of completed designs, even though those designs contained faults.

Measure of Effectiveness of Trial Process.

The effectiveness of the trial process was assessed by simply calculating the reduction in overall development time from the results of the experiment. No effort was made to measure other gains which could result from it. Some of these are listed below:

- Improved quality of design,

- Less documentation,

- Higher morale and motivation of staff,

- Release of resources for other projects.

10.6. Generalisation of Solution.

The purpose of this Section is to examine the validity of applying the conclusions drawn from the results of the experiment to other products.

Since little is known about the number of changes generated for other products, it is only possible to discuss here whether techniques are available for applying the philosophies to those products. The two techniques used in the trial process and the other techniques listed in Section 8.5 are discussed in turn.

Quality Assurance Design Verification.

It can be assumed that this technique would be effective for detecting Design Rule Contraventions, whatever the product, providing the design rules have been developed.

Design Functional Simulation.

In order to assess the usefulness of Design Functional Simulation on other products, the removal of each of the product limitations listed in Section 10.5 is considered in turn (in reverse order to the list in Section 10.5).

Digital (Logic).

The result of removing this restriction implies that the product contains analogue electronic circuits. Simulation systems are available for these - MONICA <43> being one example.

Hardware.

The removal of this restriction would result in the product being of Software. Again, techniques for implementing the philosophies are available and well documented, for example, by J. Aron <44>. He also offers evidence of the effectiveness of the techniques.

Level in Product Hierarchy ('Unit').

Although techniques exist for simulating electrical system designs at a higher level than that of 'Units', these are generally not as satisfactory as those designed for 'Unit' simulation. They also tend to get less satisfactory the higher up the hierarchy for which they are designed. Work is, however, being undertaken at a number of establishments to improve the quality of high level simulators and to design new ones.

Product Type (Electronic).

One of the criteria influencing whether the philosophies would be effective in the development of products, other than electronic ones, is the number of faults which occur during design.

There are some industries, however, where design must be proved before a prototype can be built, either because the product is a 'one-off' (for example, a civil engineering construction) or because of safety reasons (for example, in nuclear engineering). For many such products, analysis and simulation models have been developed, but for the reason of achieving a faultless design before construction starts rather than in order to reduce the development time.

Models are available, therefore, for a wide range of engineering activities and including:

- Models of the properties of building materials,

- Aerodynamic simulation models,

- Models of the properties of fluids,

- Mechanical structure models,

- Chemical process simulation models,
- Factory simulation systems.

In the areas where such systems do exist, the philosophies could in principle be applied as effectively as for electronic products.

Other Techniques.

The other techniques listed in Section 8.5, namely:

- Design Reviews,
- Design check performed by an engineer other than the designer,
- Use of Checklists,
- Check built into Computer Aided Design process, and
- Incentive Schemes,

are not product dependent and in theory could be applied to any product.

The use of 'Bread Boards', however, is restricted to electronic products.

10.7. Summary of Chapter.

Taking the results of the experiment as they stand, it was shown that they prove both the effectiveness of the two proposed philosophies for reducing new product lead times and the techniques used. It was not, however, possible to draw any conclusions about the relative effectiveness of the two philosophies.

The scope of the experiment was also limited in a number of other respects:

- It did not attempt to measure the optimum level of checking and the savings which would result from that,

- It was only performed in one environment of which the main characteristics were a large number of design changes and strong pressure for visible progress to be achieved,

- The measure of effectiveness of the trial process was limited to an assessment of its resulting savings in the development time,

- The experiment involved the development of a limited product range (Digital, Hardware 'Units' of a large electronic system product).

An assessment was made of whether the philosophies could be applied in more general circumstances and to other products.

While it was not possible to prove that their effectiveness would be as great as for the trial, it was shown that the techniques are available for different products. The effectiveness is, however, likely to be dependent on the number of design changes.

CHAPTER 11

EVALUATION OF PROJECT

11.1. Chapter Preview.

The present Chapter sets out to review the project as a whole and to evaluate its findings and conclusions.

It starts in Section 11.2 by discussing the course the project took and how it evolved. The Chapter then lays out the achievements of the project. This is covered by two Sections, 11.3 and 11.4, the first being concerned with its Contribution to Knowledge and the second with its practical benefits for the Sponsoring Company.

Although significant achievements were indisputedly made, there were a number of areas closely associated with the research, which it was either not possible to cover fully, or not possible to cover at all. These are outlined in Section 11.5.

The Chapter closes with the main conclusions (Section 11.6).

11.2. Direction of Research.

Initially, when the project started, it was envisaged that it would concentrate on the interface between development and production. As shown below, it did, in fact, finish up focusing on the development process.

Since the project's initial terms of reference were very general - "to cut the lead time for getting new products into volume production" a number of project boundaries were erected.

These resulted in the project concentrating on the introduction of large scale electronic products from the time of specification to the time that the product is in volume production. In order to maximise the savings, it was also decided to concentrate on the development and transfer of the lowest assembly level of the equipment - 'Units'.

Working within the boundaries, study was carried out into the development and transfer of two telecommunications processor systems. This study showed that, firstly, the development time overwhelmed the time for transfer to production and that, secondly, a significant proportion of the development time was taken up with activities associated with design changes. The most important cause of these design changes was shown to be 'Design Faults' (where the product does not perform as specified) and, in particular, 'Logic Design Faults'.

When searching for published works in this area, it was found that their availability was extremely limited, probably due to its sensitive nature. Using that which was available and findings of the studies, a model of the development process for large hierarchical systems was developed. Its main feature was its representation of the iterative loops.

The work up to that stage of the project was concerned with the problem. It then turned to finding a solution of how to cut the lead times.

Two philosophies were proposed as the answer. The first was based on the hypothesis, "that by spending more time at the early stages of development detecting and remedying faults in the design, even greater savings would be made later on", and the second on the hypothesis, "that the collective performance of the development organisation can be improved by increasing the amount and speed of feedback about that performance".

Using the techniques of logic simulation and Quality Assurance checking, the philosophies were tested in an experiment which showed that an 11.2% saving in time could have been made for the development project.

When, at the beginning of the project its boundaries were erected, it was the intention to remove as many of them as possible later in order to broaden its conclusions. Whilst it was not possible to prove categorically that the philosophies would be as effective for other products and technologies - because of insufficient knowledge about their design change levels - it was shown that the techniques are available to implement them.

The course of the research, outlined above, is in a number of ways typical of IHD projects.

One of the main features of these is that they are concerned with problems 'in the real world'. Often, at their outset, these problems are unclearly defined and a large portion of the project time can be spent seeking the problem. This project was no exception as about two thirds of its duration was spent reaching the point where it could be said that what needed remedying were the effects of design changes and that the main cause of design changes was Design Faults. It should, however, be noted that during the time spent defining the problem the research went off at a number of tangents and followed a number of courses that were later abandoned. These have not been discussed as they are not relevant to the theme of this thesis. They do, however, reflect another characteristic of IHD projects: this is that the direction of research often changes dramatically during the project.

11.3. Achievements of the Project - Contribution to Knowledge.

As indicated in the Chapter Preview, the achievements of the project can be divided into two categories. The first, covered by this Section, is concerned with its contribution to knowledge.

The knowledge contributed by the project falls into four areas, which are listed below:

1. Increased understanding of the development process of large hierarchical systems and, in particular, the part that iterations play within the process.

2. The importance of design and specification changes to the development process and their mechanisms.

3. The cost of design changes, in terms of resources required to detect and correct them, and their effects on the development time.

4. The effectiveness of the two philosophies in reducing the development time of new products.

11.4. Achievements of the Project - Practical Benefits for the Sponsoring Company.

The project produced the following benefits for the Sponsoring Company:

1. It directed attention towards the major problem of design changes and faults and, in particular, their importance in influencing development lead times.

2. It resulted in the production of a philosophy and practical process which is capable of giving significant savings in terms of reduced lead time and reduced development cost.

3. It enabled a dispassionate and largely independent study of the Company's development process and methods for handling new products to be made.

4. A number of recommendations and suggestions were made on subjects not directly concerned with the main theme of the research. They have not been

included in this thesis as they do not form part of its theme.

11.5. Research Not Completed.

The subject of design changes with the development process of new products is too vast to cover completely in a three year PhD project. Although parts of it were included in the research of this project, there are a number of closely related topics which were not covered or which were only covered in part. They are listed below:

- The importance of design changes and faults in the development of products and technologies other than those studied, including software.

- The causes of Design Faults. (This was only researched in part.)

- The effects of design changes on factors other than the development time and cost. This includes motivation of development staff, documentation control and quality of design. It was also not possible to establish what effects design changes, caused by different factors or generated at different times in the development process, have on the lead time and development cost.

- Optimum levels of feedback and effort expended in design verification to produce maximum savings.

- Techniques for implementing the philosophies. (Only two were tried in the experiment. They were Logic Simulation and Design Quality Assurance verification of conformity with design rules.)

In addition to these topics, the research would not be complete without a full scale implementation of the philosophies and long term measurement of their effects.

It is sufficient to say here that further work is required in these areas. Details of that work are given in Chapter 12.

11.6. Conclusions.

The project, which was initiated with very general terms of reference, evolved to concentrate on the specific area of the effects of design changes on the development process of new products. In spite of the sensitivity of this subject and the lack of published information about it, a considerable amount was learnt about the mechanisms of iterations caused by design changes. Also, philosophies, which were shown to be capable of making significant savings in the lead time of new products, were devised and tested.

CHAPTER 12

FUTURE WORK

12.1. Chapter Preview.

There are many aspects of Research and Development and new product innovation which require further study. It is not the intention to list all of them here, but only to outline those directly related to the main theme of this thesis, which is concerned with the effects, causes and mechanics of design changes and iterations in the development process. Five areas for further work, which relate to those areas of incomplete research as listed in Section 11.5, are described:

- Importance of design changes for different product types and technologies (Section 12.2),
- Causes of Design Faults (Section 12.3),
- Effects of design changes on factors other than lead time and development cost (Section 12.4),
- Optimisation of Philosophies (Section 12.5),

- Techniques for implementing the Philosophies (Section 12.6).

The Chapter then describes (Section 12.7) how one of those techniques could be implemented at full scale and how other benefits could be derived from it. A summary of the Chapter is given in Section 12.8.

12.2. Importance of Design Changes for Different Product Types and Technologies.

In order for it to be proved that the solutions proposed in this thesis can be applied effectively for other product types and technologies, the importance of design changes in the development of these products must be established. Any study to do this must overcome the major barrier of the sensitivity of the subject to development organisations. Such research would probably best be performed through an independent institution, such as a university. A guarantee of anonymity for participating organisations would have to be given when results were published. There would, however, be a strong incentive for companies and other bodies to participate in that they would benefit from such work.

12.3. Causes of Design Faults.

Chapter 5 described a study which was carried out to increase knowledge about the causes of Design Faults. Although some general conclusions were reached, it was not possible to arrive at detailed quantitative conclusions. Since major advantages would be achieved by doing just this, it is desirable that further work be performed in this area.

12.4. Effects of Design Changes.

In Part I of this thesis, it was shown how the development time and cost were affected by design changes. It was also suggested that other factors, such as development staff motivation, documentation control and the quality of design could also be influenced.

It would be desirable to know the nature and importance of these effects in order to be able to assess the full implications of design changes and what level of resources should reasonably be applied to reducing their number and their effects.

In Chapter 6, it was shown that the influence of design changes on the development time is dependent on the stage of the project when the change occurs. Knowing more about this relationship would help in deriving a really effective philosophy for development.

12.5. Optimization of Philosophies.

In order to maximise the effectiveness of the proposed philosophies, it would be necessary to establish optimum levels for the feedback and for the amount of effort expended verifying the design. This would require a longer term experiment with different groups of 'units' (or the same group, more than once) being exposed to different amounts of checking.

12.6. Techniques for Implementing the Philosophies.

The experiment described in Chapter 9 only employed two new techniques, not previously used in the Sponsoring Company, to implement the philosophies; namely, logic simulation and Quality Assurance verification of conformity with design rules. There was some indication that the first of these, at least, was not as effective as it might have been. Other techniques, such as those listed in Chapter 8, should be investigated on a trial basis to assess their usefulness.

12.7. Full Scale Implementation.

The present Section proposes a full scale implementation of the two philosophies for reducing the development and, therefore, lead times.

The method is essentially independent of product types, technology, hardware/ software, etc., but will be described as for hardware. It will use the technique of 'Checklists' described in Section 8.5. This consists of analysing all design changes with the purpose of finding common areas of design which are particularly prone to faults. Checklists are then produced to direct checking towards those areas. The technique would take longer to produce gains than some other techniques but, as will be shown, it does have major additional benefits.

The Development Process.

The development process employing the proposed method is presented in Figure 12-1.

The main feature of the process is that immediately after the design stage, design check and verification takes place. The difference between this process and that of the experiment is that this one employs checklists as previously described. The checklists are generated from the dynamic analysis of design changes wherever and whenever they are generated.

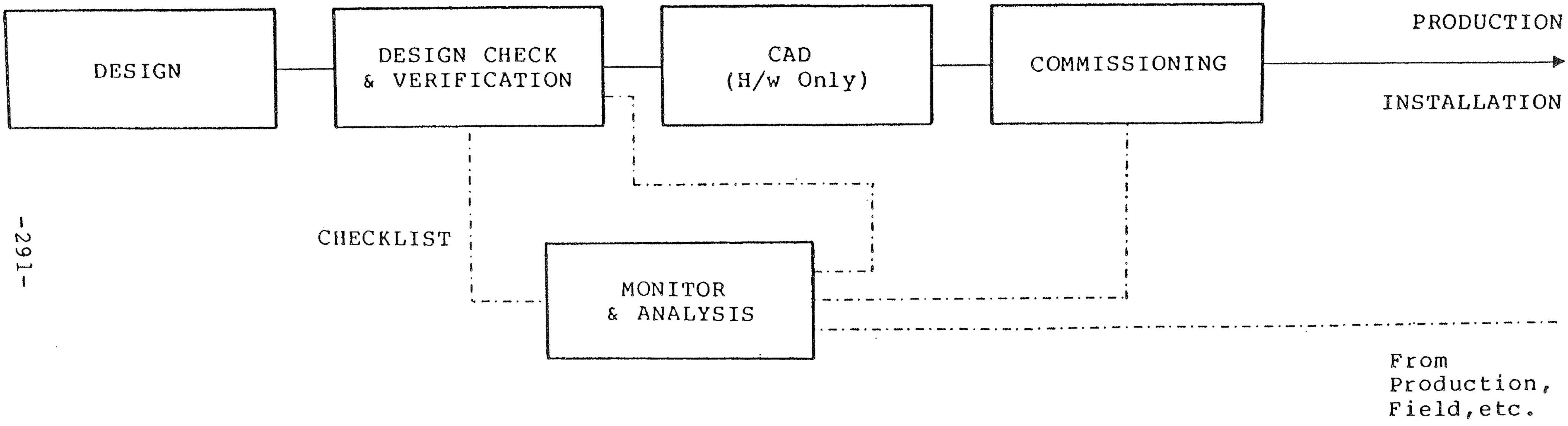


Figure 12-1. Development Process Using 'Checklists' for Design Verification.

It is hypothesised, but not proved, that a Pareto analysis would show that the majority of faults would be associated with very few circuit features. Thus, efficient checking could be directed to those features and significant savings made.

Resources Required.

The resources required would probably be only:

- For Anaylsis: An engineer for a project of about 50 designers, and access to computer facilities for simplifying and accelerating the analysis.

- For checking: About 1 man-week for an entity equivalent in size and complexity to a Slide-in-unit. (This effort would be considerably less than that required for Logic Simulation using TEGAS.)

Other Benefits.

The process described above has the potential for the following additional gains which could be made at little extra cost:

(a) Monitoring of Design Quality

The method would enable project management to detect variation in the quality of design within reasonable timescales. Rapid response to such variations can, therefore, be made, thus optimizing the quality of design.

(b) Training

Following on from (a) above, the method can be used to detect shortfalls in the training given to engineers.

(c) Testing of Other Techniques

The proposed process would enable other techniques which may be included in the stage "Design Check and Verification" to be effectively assessed without a major description of the whole development process.

(d) Design Rules

The analysis may show how changes to the design rules could increase the ease of design or the manufacturability/ testability/ maintainability/ reliability of the product.

(e) Common Design Blocks

The method could highlight blocks of design common to a number of assemblies. These may not otherwise be recognised and may currently be being designed over and over again.

(f) Personnel Assessments and Incentive Schemes

The technique could be used to provide information for Personnel Assessment or Incentive Schemes.

The method can, therefore, produce a number of additional advantages over and above its original objective and it can be applied to any product type or technology.

12.8. Summary of Chapter.

The Chapter outlined areas of research which are closely related to the main theme of this thesis, and which still need to be undertaken. It also described how one technique, 'Checklists' could be implemented and what additional advantages it could give to project management.

APPENDICES

APPENDIX A

TELEPHONE SWITCHING GROUP: DEPARTMENTAL ORGANISATION AND FUNCTIONS

CONTENTS

1. Introduction.

The purpose of this Appendix is to describe, in outline, the formal organisational structure of Telephone Switching Group, with particular emphasis laid on the functions for getting new products into volume production. It does not discuss the effectiveness of the structure but simply records it as it was at the time of writing. It also describes working parties and committees which have been set up to help with the transfer of new products.

For each department documented, an outline is given of its functions and responsibilities, and where the department is large and carries out a number of functions, its internal structure is shown.

2. Overall Structure.

Figure A-1 shows the overall structure of Telephone Switching Group and the departments from which it is constructed.

The Group also makes use of the services of a number of departments central to GEC Telecommunications Ltd., which are not shown. These include the Personnel Department, the Training Department, and the Central Data Preparation Department. Some work is also carried out for it by the GEC Central Research Unit.

3. Engineering Department.

Structure.

The Engineering Department is made up of three functional departments:

1. Development Laboratories,

2. Contract Engineers,

3. Drawing Office.

The functions of these departments are described in Sections 3.1 to 3.3, respectively.

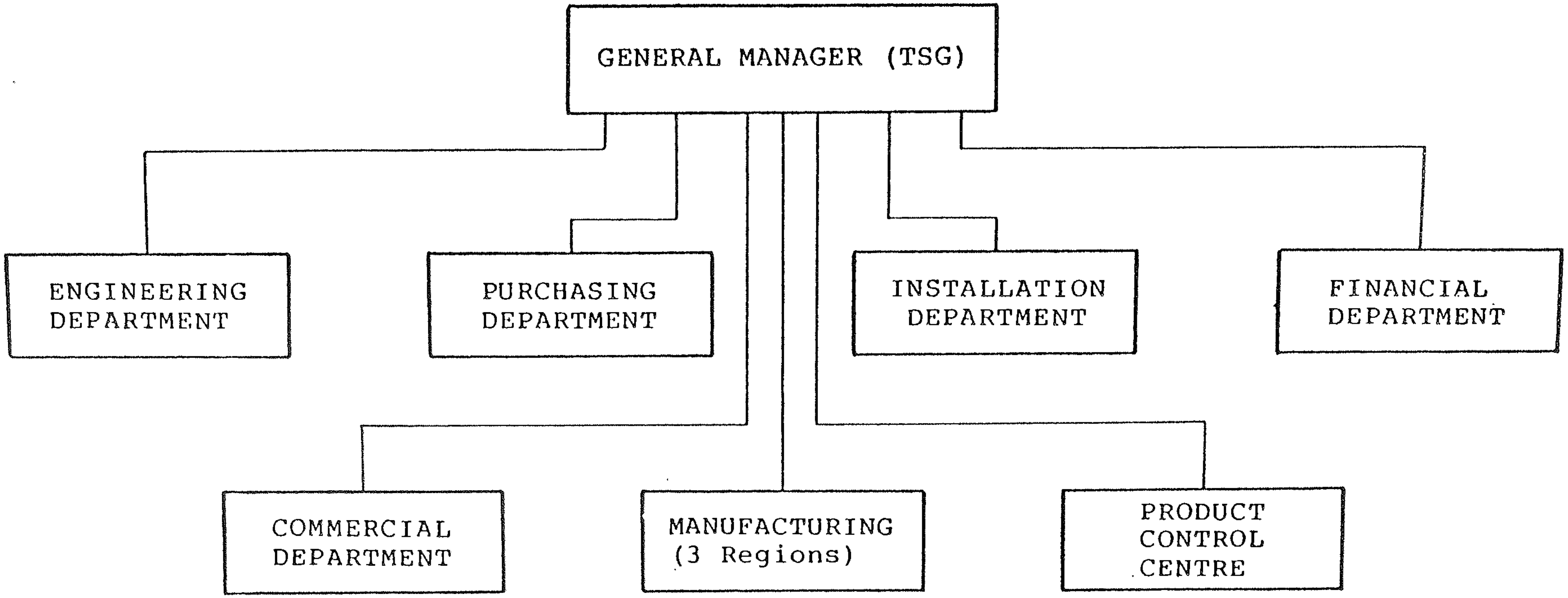


Figure A-1. Departmental Structure of Sponsoring Organisation.

3.1. Development Laboratories.

The development laboratories are split into five sections, each of which is again sub-divided. These latter sub-divisions are only described where the information is relevant to the main theme of this thesis.

The five sections are:

1. Technology & Standards,

2. Private Venture, System Development,

3. System X, System and Subsystem Development,

4. Product Planning & Advanced Development,

5. Development Services.

Technology and Standards.

This section is responsible for developing the "tools" that will be required by the Systems Development Sections during the course of their work. Through one of its sub-sections, Design Quality Assurance, it also ensures that the tools are used correctly.

The section contains the following sub-sections:

(a) Mechanical Techniques Group.

This group is responsible for developing standards pertaining to the mechanical construction of products and for designing mechanical piece parts.

(b) Circuits Techniques Group.

This group is responsible for the development of circuit standards and for designing some circuit building blocks which are used in the construction of systems. It also develops some special circuits, which do not conveniently fall into one system or another, and some special components.

The circuit standards contain rules which must be used by the Systems Engineers, and lists of approved components.

(c) Documentation Standards Group.

The group is responsible for the development of documentation formats and structures for new products.

(d) Engineering Computer Services.

This group is responsible for the development and maintenance of Computer Aided Design (CAD) systems, logic simulation systems, software development facilities (emulators, etc.) and other computer systems used by the Engineering Department. It is also responsible for maintenance and availability of computer equipment used by the Engineering personnel.

(e) Design Quality Assurance.

The Design Quality Assurance Group ensures that all designs meet the laid down standards and that Quality Assurance schemes are operated correctly.

Private Venture System Development.

This section uses the building blocks, standards and computer facilities, developed by the Technology and Standards section, to develop new products, but where the development is funded by the Company. It is also responsible for development of established products when specific new customer requirements are called for, when a new market need has been recognised or when a cost reduction is required.

System X, System and Subsystem Development.

This section, like the Private Venture System Development section, develops systems and subsystems using the building blocks, standards and computer facilities developed by the Technology and Standards Section. The systems and subsystems, however, are all part of System X.

Product Planning & Advanced Development.

This section is responsible for planning and investigating future products not yet under development.

Development Services.

This section provides a variety of services required by the development laboratories. Only those relevant to this thesis are discussed below.

(a) Laboratory Model Workshop and Stores.

The laboratory model workshop builds models of new or modified equipment where the quantity is small and where full production documentation has not been produced. It also carries out modifications to models.

(b) Planning.

This group carries out a liaison function between individual work areas and the Post Office over planning and progress reporting. It also compiles progress reports and collects information for five year planning.

(c) Documentation Flow Control Group.

This group is responsible for:

- Distribution of product documentation to and from areas outside the development laboratories,

- The control and recording of all product documentation produced by the development laboratories,

- Carrying out certain verification checks on product documentation.

3.2. Contract Engineers.

The Contract Engineers are responsible for converting contract requirement information into factory ordering information. They also design the necessary support equipment which is required for an exchange. This includes cabling, support ironwork and power supply.

3.3. Drawing Office.

The drawing office provides a drawing service for the development laboratories and for the Contract Engineers.

4. Installation Department.

The department is responsible for installing and commissioning Telephone Switching Equipment in the field. Apart from operations, the department also includes an Accounts section, a Personnel section and a Technical section, which carries out planning, provides technical services and controls quality.

5. Commercial Department.

The Commercial Department is responsible for all sales from Telephone Switching Group, both to U.K. and foreign customers. It submits tenders, negotiates contracts and reports progress to the customer. It also produces statistical information about sales, produces forecasts for planning purposes and carries out traffic dimensioning for some products.

6. Estimating Department.

The Estimating Department is part of the Telephone Switching Group financial department. It produces base cost estimates for products and estimates for development contract costs.

7. Purchasing Department.

The Purchasing Department is responsible for purchasing all bought-out items required by Telephone Switching Group.

The department also buys for other Divisions and GEC Telecommunications central departments.

8. Manufacturing.

The manufacturing activity of Telephone Switching Group is split into three regions; Midland, Northern and Scotland. Each region is divided into between 2 and 4 production units, each of which concentrates on a few particular product or equipment types. The units typically contain the following departments:

Production,

Production Control,

Quality Control,

Production Engineering,

Accounts.

Some also contain a personnel department.

9. Product Control Centre.

The Product Control Centre(PCC) consists of a group of departments which are responsible for planning, setting standards for, and performance monitoring of, the engineering, manufacturing and installation departments.

9.1. Central Planning & Scheduling Department.

This department incorporates the 3 sections below:

Planning Section - produces forecasts which are relevant to the short term, medium term and long term.

Scheduling Section - is responsible for scheduling work on to each of the Production Units.

Stock Control Group - monitors inventory throughout Telephone Switching Group to ensure that the best use is made of it.

9.2. Product Support Department.

The Product Support Department is a central production engineering department within Telephone Switching Group. It has four main functions:

1. To develop new ways, or to establish the "best" way to manufacture particular equipment and to produce process standards.

2. To give support to manufacturing units making particular products.

3. To develop new work study techniques and to devise a work study policy for Telephone Switching Group.

4. To monitor the progress of new product projects to establish production engineering requirements.

9.3. Special Facilities Department.

The department designs, develops, builds, commissions and maintains production test equipment and special manufacturing machinery. It also writes programs for automatic testers.

9.4. Documentation Control Department.

The department is concerned with four main aspects of documentation.

1. Documentation Control and Procedures,
2. Configuration Management,
3. Assembly Aids - This section is responsible for the production of automatic assembly machine programs,
4. Data Preparation - This section is responsible for the fixed data updating and maintenance of the data bases.

9.5. Central Systems Department.

The department is responsible for the definitions, design and maintenance of business systems used by Telephone Switching Group.

9.6. Installation Support Department.

The department devises methods for the installation and commissioning of new telephone switching products.

9.7. Product Training Department.

This department is responsible for devising two types of training programme for new products; shopfloor operator training and familiarisation, and testing and commissioning technical training.

9.8. Primary Parts.

This department is responsible for the development of Telephone Switching Group Policy on the manufacture of primary parts.

9.9. Quality Assurance Services Department.

The function of the Quality Assurance Services Department is to devise Quality Assurance procedures to be employed within Telephone Switching Group and to ensure that they are being maintained.

10. New Product Working Parties and Committee.

Although in the past there have been a number of working parties set up to consider various aspects of new product introduction, only two exist at the time of writing; System X Technical Problems Solving Meeting and Central Planning and Scheduling Department Project Control Meeting.

Most liaison and information flow takes place on an inter-departmental basis.

There are, however, a number of task groups, working parties, and committees which were set up between the Post Office and the three manufacturers to consider aspects of System X.

System X Technical Problems Solving Meeting.

System X Technical Problems Solving Meetings are held regularly to bring to the attention of all concerned technical problems associated with the transfer of System X to Production, and to find solutions to those problems. It is attended by representatives from the relevant manufacturing units, development laboratories, Product Support Department, Documentation Control Department, the Special Facilities Department and any other department to which it is relevant.

Central Planning & Scheduling Department Project Control Meeting.

The purpose of these meetings is to monitor the manufacturing progress of new products, to identify the causes of delays and to initiate remedial action. It consists of representatives from the Contract Engineers, Development Laboratories, Manufacturing Units, Central Planning and Scheduling Department and any other department to which it is relevant.

APPENDIX B

DEVELOPMENT PROCESS FOR HARDWARE 'UNITS OF EQUIPMENT'

The development process for 'units of equipment' - both Slide-in-units and Wired Shelf Groups - is shown diagrammatically in Figure B-1. Each of the stages is discussed below.

This model does not show the top down systems design which it assumes has already taken place. The first stage shown in Figure B-1 is "Define Unit".

Define Unit.

The team leader responsible for the relevant part of the equipment produces a detailed specification for the Slide-in-units or Wired Shelf Group. The Slide-in-unit specification contains detailed descriptions of the circuit functions, input and output signals and their timing.

A Wired Shelf Group specification is unnecessary as all the information necessary for

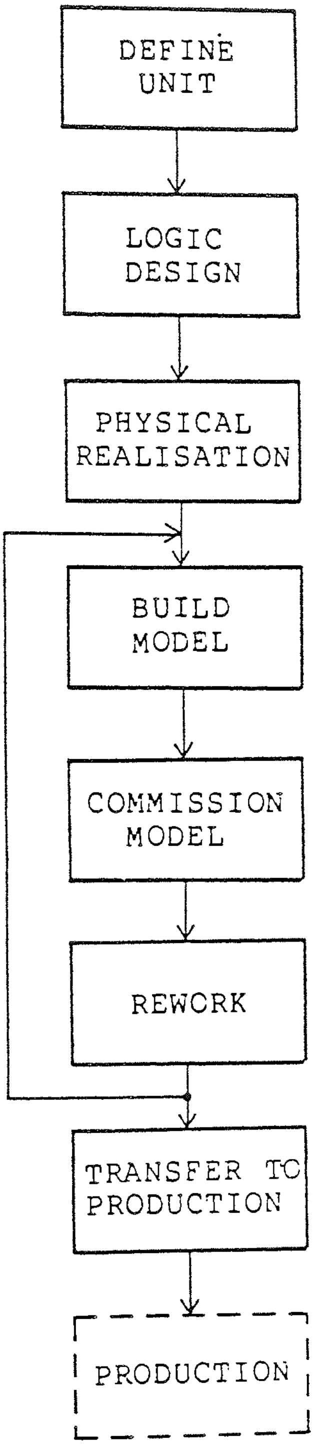


Figure B-1. The Development and Transfer Process for 'Units of Equipment'.

design is contained within the 'Functional Entity' and Slide-in-unit specifications.

Logic Design.

For Slide-in-units, the design is undertaken by a design engineer who reports to the relevant team leader. The process is carried out using manual, ad hoc methods: that is, no formal algorithmic techniques are employed. On completion, the circuit is manually checked by the team leader.

Although this stage is described as "Logic Design", it is also taken to cover the small proportion of boards which contain analogue circuits.

The Wired Shelf Group designs are carried out by the team leaders. They specify the connections which have to be made between the printed circuit board edge connectors.

Physical Realisation.

Physical realisation is the process of converting the circuit into a physical piece of equipment. For a Slide-in-unit, this is a printed circuit board design and is performed by a Computer Aided Design suite of programs operated interactively by the Design Engineer. After the circuit diagram has been digitised into a computer readable form, three activities take place. Firstly, the circuit is checked against some fundamental design rules, such as device fan-outs and outputs connected to outputs. This is performed by a program called 'FUNC". Secondly, placement for the components is established and, thirdly, the tracking is put on. The Computer Aided Design system also generates some manufacturing documentation, such as board artwork, drilling control tapes, layout diagrams and stocklists.

For Wired Shelf Groups, a Computer Aided Design system generates the routing for the connections and produces manufacturing documentation.

Build Model.

A first-off prototype of the equipment is constructed. The printed wiring boards are manufactured on a standard production facility with an expedited service. The quality may, however, be lower than for normal production. Assembly is carried out in the development laboratory model shop or in a Small Quantity Production area in one of the manufacturing departments.

Commission Model.

The design of each Slide-in-unit is not functionally tested in isolation as it may only form part of a functional circuit. Commissioning first takes place then at Functional Entity level and this may involve several Slide-in-units. Testing is firstly undertaken for hardware alone and then for hardware and software together. When a fault is found, the commissioning team work out what modification is required to correct it and record this on a 'Change Note'.

Rework.

The 'Change Notes' generated during model commissioning are collected together for each 'unit of equipment' and the design is modified to take these into account. If the changes are comparatively minor, then a 'Modification Action Package' will be produced. This is used by the manufacturing areas to modify 'units' made to the original documentation. If the changes are more extensive then the 'logic design' and 'physical realisation' stages will be repeated to produce a clean set of documentation. A 'Modification Action Package' may still need to be produced in order that 'units' already made may be updated.

The model described above represents the process as it took place in practice before the trial described in Chapter 9 was undertaken. It had been the intention, however, to simulate the operation of each System X Slide-in-unit circuit after logic design. This was to be performed by a packaged system, TEGAS. For the LPU and SPU this was only used in a small number of cases (28 Slide-in-units), and even with these was not used effectively. The reasons for this were:

- TEGAS had a large number of software faults,

- Models of many of the devices used in System X had not been written,

- TEGAS is difficult to use and appropriate expertise was limited,

- There was considerable pressure to get the first-off Slide-in-units built and the simulation stage was one which could be missed out while still enabling them to be produced.

TEGAS was, however, used to evaluate the effectiveness of production GO/NO GO test programs for a number of Slide-in-units. A suite of programs is also available to convert the TEGAS waveforms into PROTEST language instructions for the automatic testers used by the Sponsoring Company.

An outline of the operation and capabilities of TEGAS is given in Appendix I.

N.B

It should be noted that by the time the trial was conducted, most of the TEGAS faults had been corrected. It was felt that it was in a state where it could be employed satisfactorily.

APPENDIX C

PRODUCTION, DISSEMINATION, DISTRIBUTION AND USE OF FORMAL DOCUMENTATION FOR NEW PRODUCTS

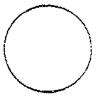
Introduction.

The model exhibited below shows how formal new product documentation is produced, processed, distributed and used relative to the various stages of developing a new product and transferring it into volume production.

This Appendix describes the symbolism used in the model of the process and then goes on to present the model.

Model Symbolism.

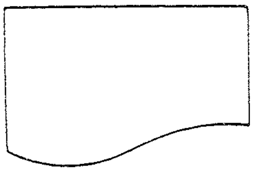
(a) The model is in the form of a conventional network (similar to a PERT) but, in addition to exhibiting activities, it also shows how documents are created and used. The symbols used are shown in Figure C-1.



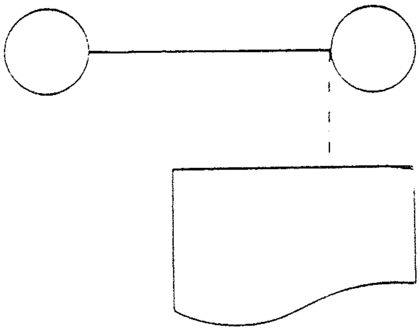
Node: The start and/or end of an operation:

A process which is broken down to show more detail.

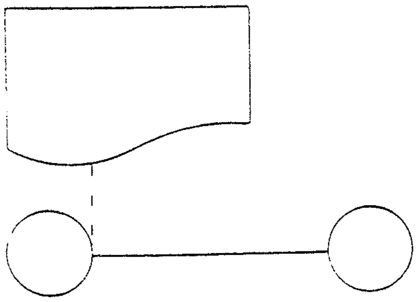
A process which is NOT further broken down.



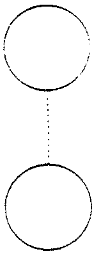
Document or Set of Documents



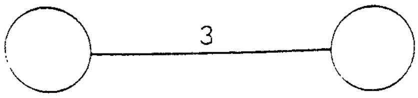
A document leaving or being produced by an operation. This will often occur at the end of the operation. Where a number of documents leave an operation at different times and where for convenience they are shown as a set, they will be shown leaving at the end of the operation.



A document entering or being used during an operation. This will often occur at the start of the operation. Where a number of documents enter an operation at different times and where for convenience they are shown as a set, they will be shown entering at the start of the operation



Two nodes representing the same point in the overall process.



Duration of activity.

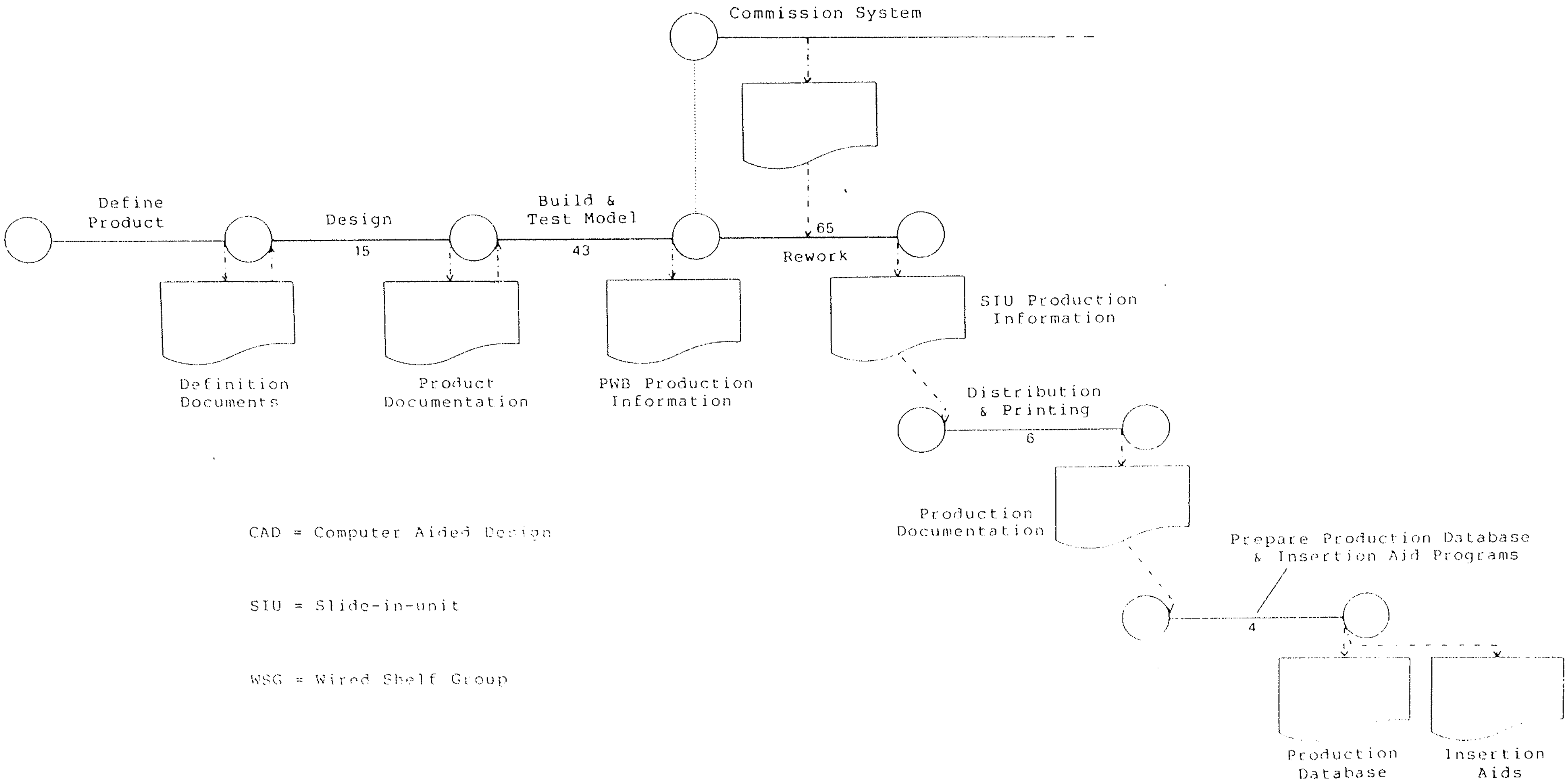
Figure C-1. Symbols Used in the Model of Information Flow.

(b) The model is hierarchical. That is, high-level operations or documents can be broken down to show more detail. This is only done where necessary to reveal facts relevant to the discussion in this thesis.

Model.

The model of the formal new product information production, processing, distribution and uses is shown in following pages.

Model of New Product Information Production, Processing,
Distribution and Use.
I. OVERALL PROCESS.

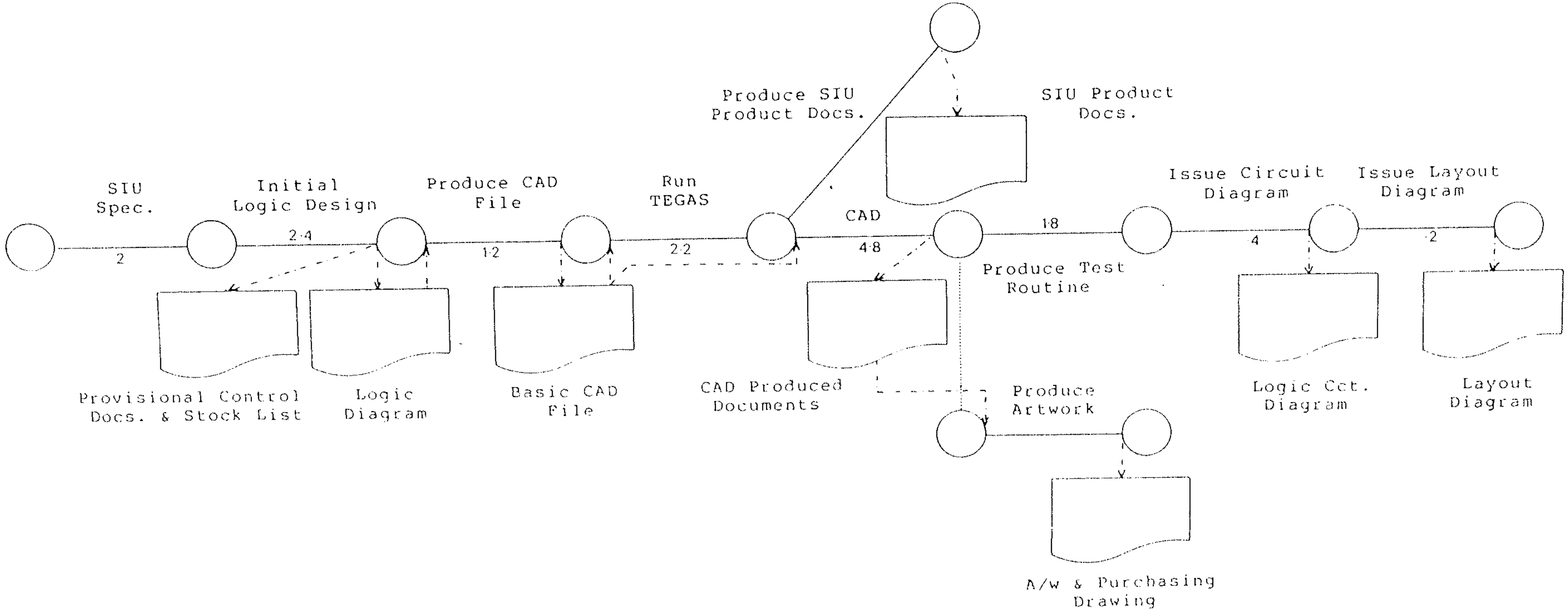


CAD = Computer Aided Design

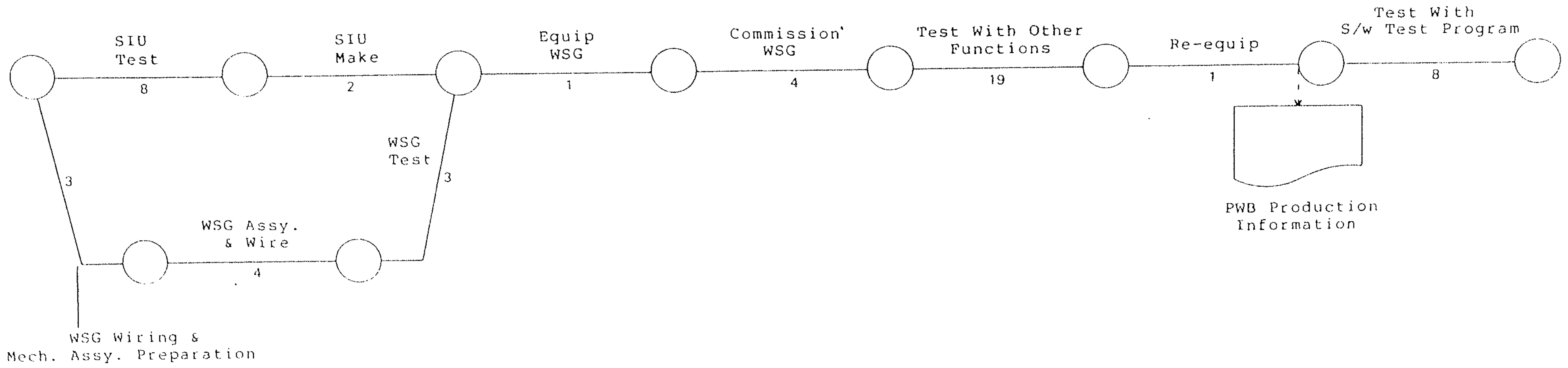
SIU = Slide-in-unit

WSG = Wired Shelf Group

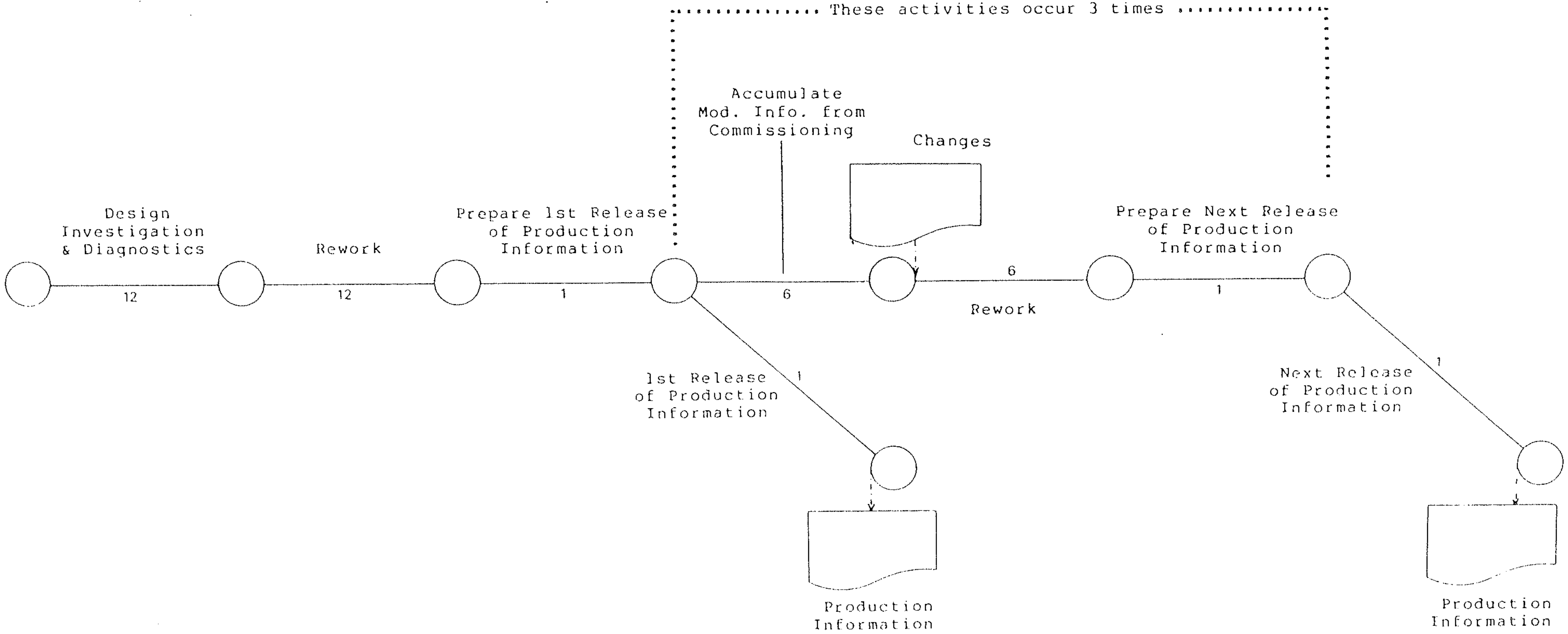
Model of New Product Information Production, Processing, Distribution and Use.
II. DESIGN PHASE.



Model of New Product Information Production, Processing,
Distribution and Use.
III. BUILD AND TEST MODEL.



Model of New Product Information Production, Processing,
Distribution and Use.
IV. REWORK ACTIVITIES



APPENDIX D

'CHANGE NOTES': DESCRIPTION & PROCEDURES

Function of the 'Change Notes'.

The 'Change Notes' provide a means of recording changes made to a product's design, specification or documentation by different groups of people within the development laboratories. The changes may then be brought together for inclusion in a 'rework' or Modification Action Package. (See Appendix B).

Description of 'Change Notes'.

On each 'Change Note' the following information is recorded:

- Part Number of the piece of equipment to which the 'Change Note' applies,
- A unique identification number known as the "Change Note Number",
- A textual description of the change,

- The date of writing the 'Change Note',

- The signature of the originator.

The textual description of the change should consist of two parts. The first, a brief explanation of the reason for the change, is often omitted and, if included, may only be a limited comment. The second is a description of the action required to make the change.

In addition to the information listed above, the time taken to find the fault, to remedy it and to document it, and the corresponding number of man-hours spent by the Commissioning Engineers, were also recorded in order to provide information for this research.

Creation of 'Change Notes'.

'Change Notes' originate from two sources. The first is from model commissioning when a fault is found. The second source is the design team. They produce 'Change Notes' as a result of design investigations, new technology, new standards or queries from production and other areas.

Use of 'Change Notes'.

The 'Change Notes', produced for each product, are collected together and filed. From time to time, depending on the number of modifications, the seriousness of the modifications and the manufacturing programme, the 'Change Notes' for a piece of equipment are collected together and the changes incorporated into a Modification Action Package or a rework.

APPENDIX E

FINAL FORM OF QUESTIONNAIRE

1. From your experience, how do you rate each of the following factors in causing Design Faults made in E.880?
(For each one, please tick the appropriate box).

	Does Not Contribute To Design Faults →					Important Cause Of Design Faults
	0	1	2	3	4	5
1. The training given to the E.880 Design Engineers						
2. Product Design Standards						
3. The accessibility of Design Standards						
4. The size of the Product Building Blocks (e.g. SIU's, S/W Processes)						
5. The complexity of the Product Building Blocks						
6. The complexity of the Product Documentation Scheme						
7. The Managerial Organisation in E.880						
8. Motivation of the E.880 Design Engineers						
9. The Formats of Specifications						
10. The level of completeness of Specifications						
11. The Methods used for Design						
12. The insufficient use of Design Reviews						
13. The computer development aids which are available						
14. The ease (or otherwise) of access to computer facilities						
15. Working Conditions						
16. The Methods of selecting Engineers during recruiting						
17. The technology employed in the product						
18. The general attitude to work in modern Britain						
19. The amount of experience of the E.880 Design Engineers						
20. The methods used for design checking						
21. A lack of design team identity						
22. The level of communications within the design teams (e.g. POPTS, MLE, UPS)						
23. The level of communications between the design teams within E.880						
24. The level of communications between GEC, BPO & the other PF's						
25. Others (Please specify)						
....						
....						

II. Please select one group of people (by ticking the appropriate box) to answer each of the following questions.

	Design Engineers	Supervisors (Team/Group Leaders)	More Senior Management	B.P.O.	Other(Please Specify)
1. In your opinion, whom does the company hold as being most responsible for reducing the level of Design Faults made in E880 ?	☐	☐	☐	☐	☐
2. Who, in your opinion, <u>should</u> have the <u>greatest responsibility</u> for reducing the level of Design Faults made in E880 ?	☐	☐	☐	☐	☐
3. Who, in your opinion, <u>currently</u> has the <u>greatest power</u> to reduce the level of Design Faults made in E880 ?	☐	☐	☐	☐	☐
4. Who, in your opinion, would have the <u>greatest potential</u> for reducing the number of Design Faults made by E880, if only they were given the requisite power, responsibility or facilities.	☐	☐	☐	☐	☐

III. In very few words (key words and phrases only) please indicate what, in order to reduce the occurrence of design faults, can be done, that is not already done, but which is in their power to do, by each of the following groups of people:

DESIGN ENGINEERS	SUPERVISORS (TEAM/GROUP LEADERS)	MORE SENIOR MANAGEMENT

NAME......................................

APPROXIMATE LENGTH OF SERVICE WITH [illegible] .... YRS.

APPENDIX F

THE PROCESSING OF DESIGN CHANGES AND CALCULATION OF EFFORT REQUIRED

A model of the processing of changes was developed and is shown in Figure F-1. This is discussed below, first generally, and then in detail stage by stage. Detailed calculations for the effort spent at each stage are shown.

It should be noted that, instead of Figure F-1 representing the processing of design changes, it represents the processing of 'Change Notes' (See Appendix D). This is taken into account in the calculations, which produce figures for the effort per change (See Section 6.5 for a detailed explanation of the adjustment).

Legend Used in Figure F-1.

The boxes shown represent the activities which are performed on the 'Change Notes'. Each one is inscribed with the department that carries out the activity and a description of the activity. Each box is also given an identification number.

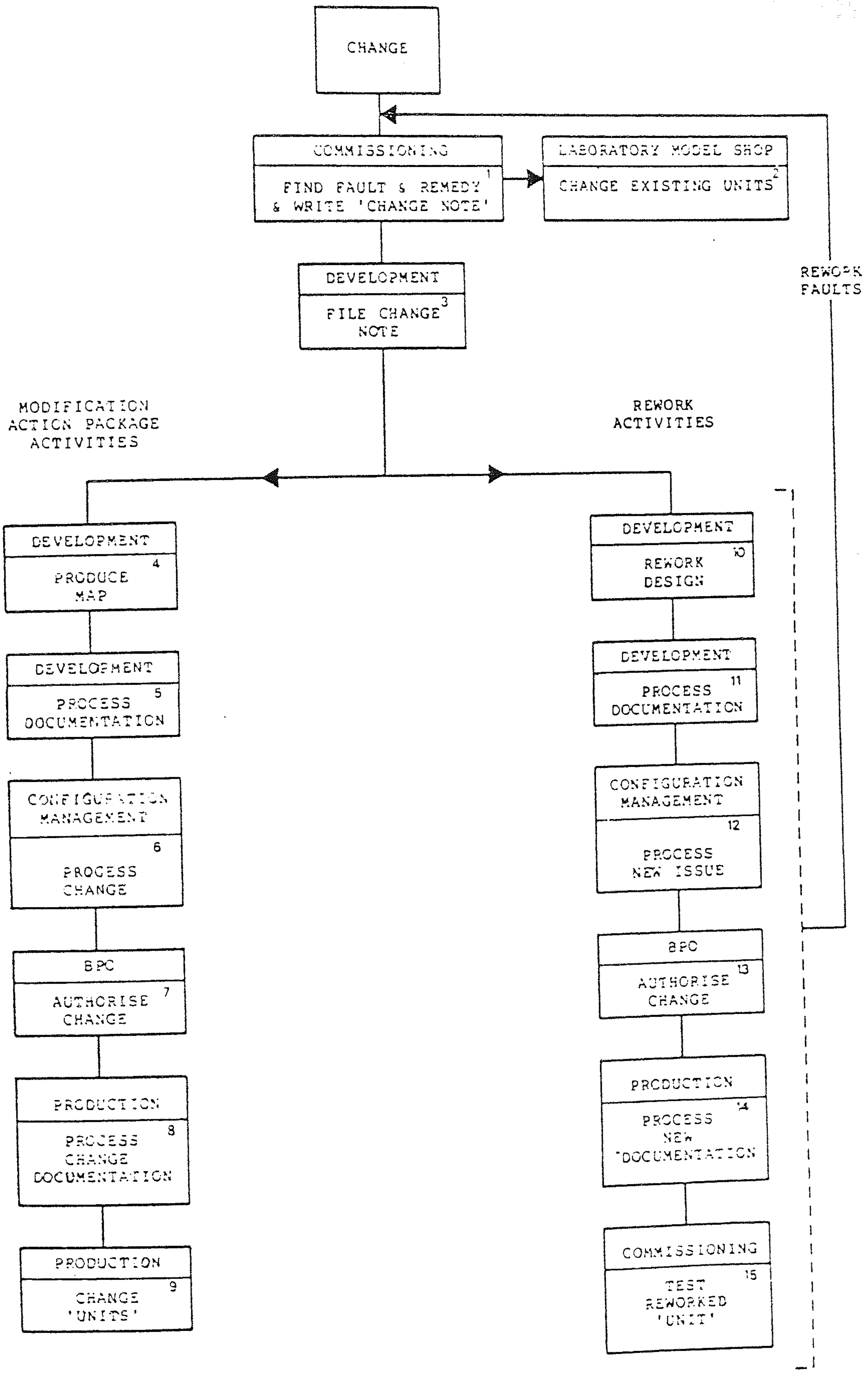


Figure F-1. Processing Design Changes for Slide-in-units.

General Description of the Process Exhibited in Figure F-1.

The activities are conveniently divided into three groups. The first, which consists of activities 1, 2 and 3, takes place for all 'Change Notes' - once only.

The 'Change Notes' then undergo either one or both of two processes. In one (activities 4 through to 9), they are used to produce a Modification Action Package, and in the other (activities 10 through to 14), they are used as base data for a rework. Normally each 'Change Note' will be used in the production of a Modification Action Package and a rework.

As the 'Change Notes' are processed, some mistakes and errors, which would result in additional 'Change Notes', occur. These are shown feeding back with the label, "Rework Faults".

Detailed Descriptions of the Stages shown in Figure F-1.

Each of the stages shown in Figure F-1 is described below along with an account of the methods used to establish its duration.

1. Find Fault, Find Remedy and write 'Change Note' (Commissioning).

This activity is the one which initiates the change. Although it is shown as being undertaken by the Commissioning Section of the development laboratories, some 'Change Notes' were produced by the Design Engineers in the course of design investigations, as the result of new standards or as a result of after-thoughts, etc.. Figures for the Commissioning Section were used in the calculations because:

- While some 'Change Notes' could be shown as definitely originating from Commissioning, it was not possible to say, with conviction, that the others definitely did not originate there,

- Figures were available (see Section 4.2) for the time and number of man-hours required in Commissioning to detect the fault, find a remedy for it and to write out the 'Change Note',

- No figures were available for the time taken to produce 'Change Notes' (and, therefore, changes) initiated outside the Commissioning Section.

As the commissioning figures were used in the calculations, the results were biased towards Design Fault-triggered changes. This, however, was the most common category of change.

The figure for the number of man-hours used to initiate a change was established by calculating the mean of all the individual figures which were available. These numbered 34, 19 of which were for Design Fault triggered changes.

CALCULATION:

Effort per Change = ((Sum of efforts provided by
commissiong Engineers)
DIVIDED BY (Number of Changes))
DIVIDED BY (Adjustment for Rework
Faults)

Sum of Efforts = 108 Man-hours
Number of Changes included = 34
Effort per Change = (108/34)/0.97 Man-hours

<u>Effort per Change = 3.3 Man-hours</u>

Figure F-2 shows an histogram of the distribution of durations for this activity. These are equivalent to efforts since only one engineer worked on each fault.

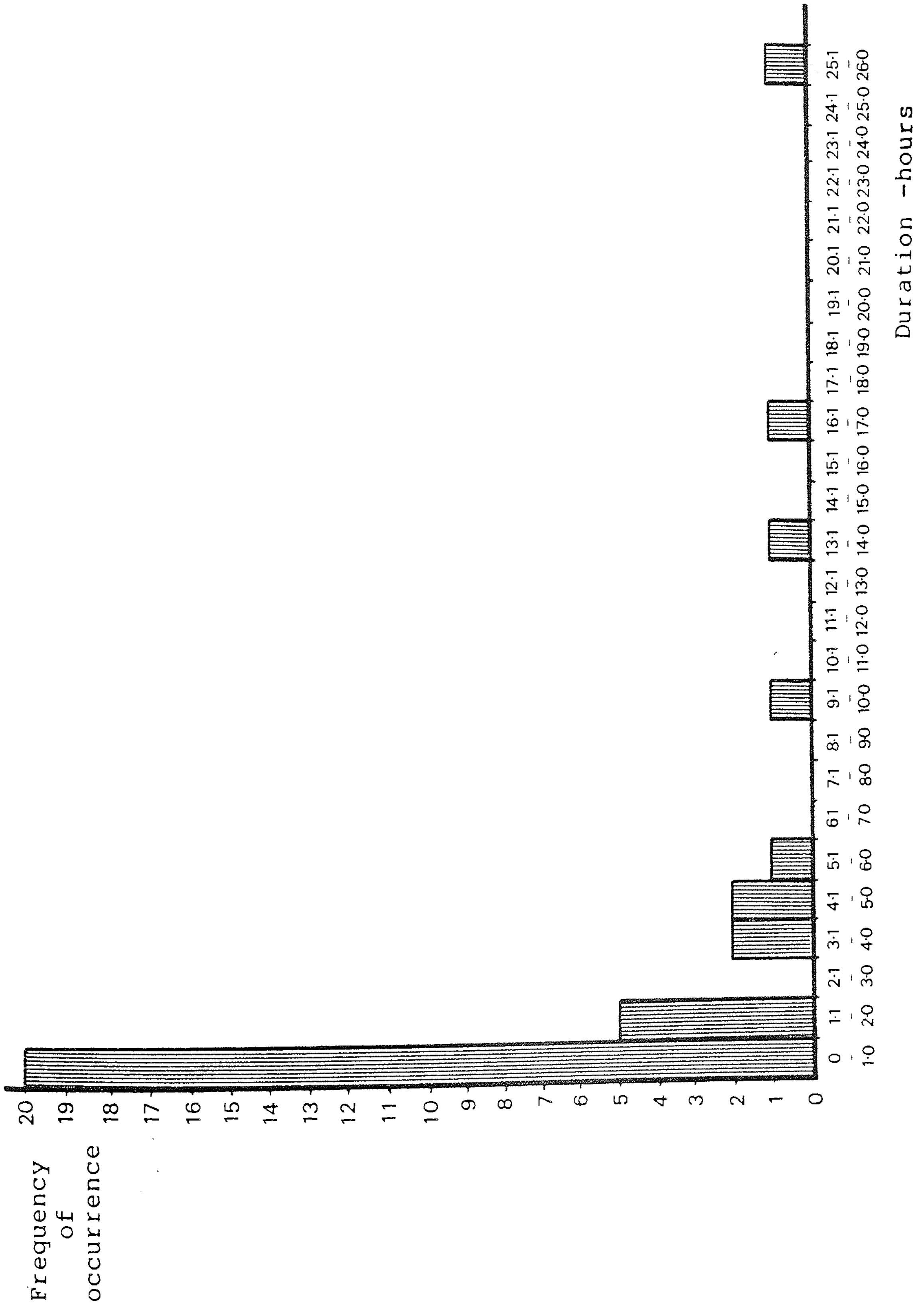


Figure F-2. Histogram of Frequency of Occurrence of Durations to Find Fault, Find Remedy and Write 'Change Note' against Duration.

2. Change Existing Slide-in-units (Development Laboratory Model Shops).

This activity involved modifying, in the development laboratory model shops, Slide-in-units already on the models being commissioned. The time taken per 'Change Note' was calculated from:

- The number of Slide-in-units requiring modification per 'Change Note',
- The number of man-hours used for modification over an 11 week period, and
- The number of units modified over the same period.

The number of Slide-in-units modified per 'Change Note' was calculated from the number of Slide-in-units modified during the 11 week period and the number of 'Change Notes' produced during the same period. Whilst there was not a direct tie between these two factors, this was the only way that the number of Slide-in-units modified per 'Change Note' could be established from the available data. In a state where the production of 'Change Notes' and modification of Slide-in-units are constant, the relationship would give reliable result. The figures used were for the last quarter of 1979. This was a time of comparatively stable 'Change Note' production. The result obtained was probably accurate to at least a first order of magnitude.

CALCUATION:

Effort per Change = (((Number of Man-hours used
in 11 weeks)
DIVIDED BY (Number of 'units'
modified per in 11 weeks))
TIMES (Number of 'units' modified
per 'Change Note'))
DIVIDED BY (Adjustment for effort
per Change)

Number of Man-hours used in 11 weeks = 2640
Number of 'units' modified in 11 weeks = 375
Number of 'units' modified per 'Change Note' =
(Number of 'units modified in 11 weeks)
DIVIDED BY (Number of 'Change Notes'
produced in 11 weeks)
Number of 'Change Notes' produced in 11 weeks = 343
Number of 'units' modified per 'Change Note' = 375/343
= 1.09

Effort per Change = ((2640/375)*1.09)/0.9 Man-hours

<u>Effort per Change = 8.5 Man-hours</u>

3. File 'Change Note' (Development Laboratories).

This activity, which takes place in the development laboratories, comprises the clerical functions of filing two copies of each 'Change Note', recording their receipt and posting one copy to the British Post Office. In practice, the activity takes a negligible time compared with the other activities.

4. Produce Modification Action Package (Development Laboratories).

This represents the drawing up of a Modification Action Package by the Design Engineers. The effort for this activity was calculated from the following information:

- The average length of time to produce a Modification Action Package (based on Team Leaders' estimates),

- The mean number of 'Change Notes' included in each Modification Action Package (extracted from records kept of 'Change Notes' and Modification Action Packages for each Slide-in-unit).

CALCULATION:

Effort per Change = ((Average Effort to produce MAP)
DIVIDED BY (Number of 'Change Notes'
incorporated in each MAP))
DIVIDED BY (Adjustment for Effort
per Change)

Average Effort to produce MAP = 2.5 Man-weeks
(1 man for 2.5 weeks)
= 93.75 Man-hours

Number of 'Change Notes' incorporated into
each MAP = 6.29

Effort per Change =(93.75/6.29)/0.9 Man-hours

Effort per Change = 16.5 Man-Hours

5. Process Documentation (Development Laboratories).

This is a Development Laboratory (Document Flow Control Group) activity in which all the required compatible documents are brought together, checked and distributed.

No measured figures for the duration of the activity were available so it was necessary to use estimates given by those performing the work. In any event the duration was comparatively small so any error would not be of very great overall significance.

CALCULATION:

Effort per Change = ((Average Effort per MAP)
DIVIDED BY (Number of 'Change Notes'
incorporated inot each MAP))
DIVIDED BY (Adjustment for Effort
per Change)

Average Effort per MAP = 6.5 Man-hours

Number of 'Change Notes' incorporated into each MAP
= 6.29

Effort per Change = (6.5/6.29)/0.9

<u>Effort per Change = 1.1 Man-hours</u>

6. <u>Process Change (Configuration Management)</u>.

The Modification Action Package documentation is checked for 'issue' compatability and the computer data base updated.

As with the previous activity, no figures were available for the duration and estimates had to be accepted from those involved with the work. Again, the duration was small.

CALCULATION:

Effort per Change = ((Average Effort per MAP)
DIVIDED BY (Number of 'Change Notes'
incorporated inot each MAP))
DIVIDED BY (Adjustment for Effort
per Change)

Average Effort per MAP = 3.5 Man-hours

Number of 'Change Notes' incorporated into each MAP
= 6.29

Effort per Change = (3.5/6.29)/0.9

Effort per Change = 0.6 Man-hours

7. Authorise Change (BPO).

This activity is shown for completeness. It only takes place for System X and so is not included in the calculations which are designed to represent a "typical" project.

8. Process Change Documentation (Production).

This consists of receipt of the change documentation, by Production, and distributing it to those who will use it on the shopfloor. In practice, the activity is so short that it can be excluded from the calculations.

9. Change Slide-in-units (Production).

This is the process of modifying Slide-in-units according to the Modification Action Package. The modification is performed on units already made and on others to be made in the future.

The duration was calculated from the following information:

- The mean number of 'Change Notes' included in each Modification Action Package,
- The mean number of Slide-in-units modified for each Modification Action Package (obtained from records of "up-issuing" of Slide-in-units),
- The mean number of man-hours spent modifying each Slide-in-unit (calculated from the number of Slide-in-units modified over a 19 week period, and the total number of man-hours spent during that period). Whilst it could not be shown whether the figure for this element was typical or not, the large number of Slide-in-units included (1127) should give a reliable result, particularly if the figure does not change with the stage of development, etc..

CALCULATION:

Effort per Change = (((Average Effort to modify
each 'unit')
TIMES (Number of 'Units' changed per
MAP))
DIVIDED BY (Number of 'Change Notes'
incorporated in each MAP))
DIVIDED BY (Adjustment for Effort
per Change)

Effort to modify each 'unit' = (Effort expended
in 19 weeks)
DIVIDED BY (Number of 'units'
modified in 19 weeks)
= 4256/1127
= 3.8 Man-hours

Number of 'units' changed per MAP = 31.5

Number of 'Change Notes' incorporated into
each MAP = 6.29

Effort per Change =((3.8*31.5)/6.29)/0.9 Man-hours

Effort per Change = 21.0 Man-Hours

10. Rework Design (Development Laboratories).

The redesign activity results in a new set of 'clean' documentation being produced for the Slide-in-unit.

The duration contributed by each 'Change Note' was calculated from the following information:

- The mean time taken for each rework (This was extracted from a rework progress report),

- The mean number of 'Change Notes' incorporated into each rework (extracted from records kept of 'Changes Notes' and reworks produced for each Slide-in-unit).

It was assumed that all changes would eventually be incorporated into a rework.

CALCULATION:

Effort per Change = ((Mean Effort per Rework)
DIVIDED BY (Number of 'Change Notes' incorporated into each rework))
DIVIDED BY (Adjustment for effort per change)

Mean Effort per Rework = 180 Man-hours
Number of changes incorporated into each Rework = 14.02
Effort per Change = (180/14.02)/0.9 Man-hours

<u>Effort per Change = 14.2 Man hours</u>

11. Process Documentation (Development Laboratories).

This activity is similar to that described under "5" above and the same comments apply.

CALCULATION:

Effort per Change = ((Average Effort per Rework)
DIVIDED BY (Number of 'Change Notes' incorporated into each rework))
DIVIDED BY (Adjustment for effort per change)

Average Effort per Rework = 7.0 Man-hours (Engineers' estimate)

Number of changes incorporated into each Rework = 14.02

Effort per Change = (7.0/14.02)/0.9 Man-hours

Effort per Change = 0.6 Man-hours

12. Process New Issue (Configuration Management).

This activity is similar to that outlined in "6" above except that it takes place for "clean" documentation as opposed to Modification Action Packages.

The same comments regarding collection of information about the activity duration apply.

CALCULATION:

Effort per Change = ((Average Effort per Rework)
DIVIDED BY (Number of 'Change Notes'
incorporated into each rework))
DIVIDED BY (Adjustment for effort
per change)

Average Effort per Rework = 3.5 Man-hours
(Engineers' estimate)

Number of changes incorporated into each Rework = 14.02

Effort per Change = (3.5/14.02)/0.9 Man-hours

Effort per Change = 0.3 Man-hours

13. Authorise Change (BPO).

The same comments as for "7" above apply.

14. Process New Documentation (Production).

The same comments as for "8" above apply.

15. Test Rework 'Unit' (Commissioning).

This activity represents the effort required to retest the design of a Slide-in-unit after it has been reworked.

No recorded figures were available for the effort required for this stage. The figures used in the calculation were based on estimates made by the development personnel.

CALCULATION:

Effort per Change = ((Average Effort to commission
Reworked design)
DIVIDED BY (Number of 'Change Notes'
incorporated into each rework))
DIVIDED BY (Adjustment for effort
per change)

Average Effort to commission Reworked design =
4.0 Man-hours
(Engineers' estimate)

Number of changes incorporated into each Rework = 14.02

Effort per Change = (4.0/14.02)/0.9 Man-hours

Effort per Change = 0.3 Man-hours

APPENDIX G

EXAMPLES OF CHANGES

This Appendix shows, as examples, the extent of 25 of the LPU changes. They were chosen at random but do illustrate the diversity of the full range of changes.

Explanation of Abbreviations Used:

SIU	Slide-in-unit.
WSG	Wired Shelf Group.
C-n	The number, "n", of 'cuts' (A 'cut' is the breaking of a circuit connection on a printed circuit board).
S-n	The number, "n", of 'straps' (A 'strap' is the making of a circuit connection on a printed circuit board).
NC-n	The number, "n", of new components required.

RC-n	The number, "n", of components removed.
Dis-n	The number, "n", of wired connections removed (from a Wired shelf Group).
Con-n	The number, "n" of wired connections added (to a Wired Shelf Group).

Examples of Changes.

CAUSE OF CHANGE	EXTENT OF CHANGE	TIME REQUIRED TO FIND FAULT & REMEDY (HOURS)
Slide-in-unit Changes.		
SIU Specification fault	C-1, S-1	0.1
SIU Specification fault	C-3, S-3,	13.5
SIU Specification fault	C-3, S-14, NC-2	4.5
SIU Specification fault	C-20, S-8, NC-4, RC-1	17.0
Logic Design Fault	C-1, S-1	1.5
Logic Design Fault	C-1, S-1	0.3
Logic Design Fault	C-1, S-1	1,5
Logic Design Fault	C-4, S-7	0.5
Logic Design Fault	C-1, S-2	1.0
Logic Design Fault	C-2, S-2	1.5
Logic Design Fault	C-2, S-2	1.5
Logic Design Fault	C-1	1.0
Logic Design Fault	C-2, S-1	0.3

CAUSE OF CHANGE	EXTENT OF CHANGE	TIME REQUIRED TO FIND FAULT & REMEDY (HOURS)
Logic Design Fault	C-1, S-1	0.5
Logic Design Fault	C-1, S-1	1.0
Hardware Realisation Fault	Turn components	0.5
Hardware Realisation Fault	Drawing only	0.5
Hardware Realisation Fault	C-1, S-1	0.5
Documentation Control (Wrong document used)	C-1, S-1	4.0
Change to 'Change Note'	Document change only	0.1
Logic Simplification (Design Review)	C-7, S-13	4.0

Wired Shelf Group Changes.

Subsystem Specification Fault	Dis-6, Con-7	5.0
Wiring Design Fault	Con-5	1.0
Wiring Design Fault	Dis-5, Con-5	0.5

APPENDIX H

NOMENCLATURE USED FOR THE CAUSES OF DESIGN CHANGES

A discussion about "Causes" of any phenomena like "Design Changes" will involve a hierarchy of causes and effects. For example, a Design Change may be immediately caused by a Logic Design Fault. This, however, will not be the ultimate cause of the change because something must must have caused the Design Fault to occur. It may have been due to, say, a distraction resulting from poor working conditions, which may in turn be due to the investment policy or profitability of the company. Thus, a hierarchy of causes is built up. In order to appreciate how immediate a cause is to the final effect, the following convention will be used.

The immediate cause will be known as the "C1 cause". This would be the "Logic Design Fault" in the example above.

The immediate cause of the "immediate" or "C1 cause" will be known as the "C2 cause".

The process continues and is shown diagrammatically in Figure H-1.

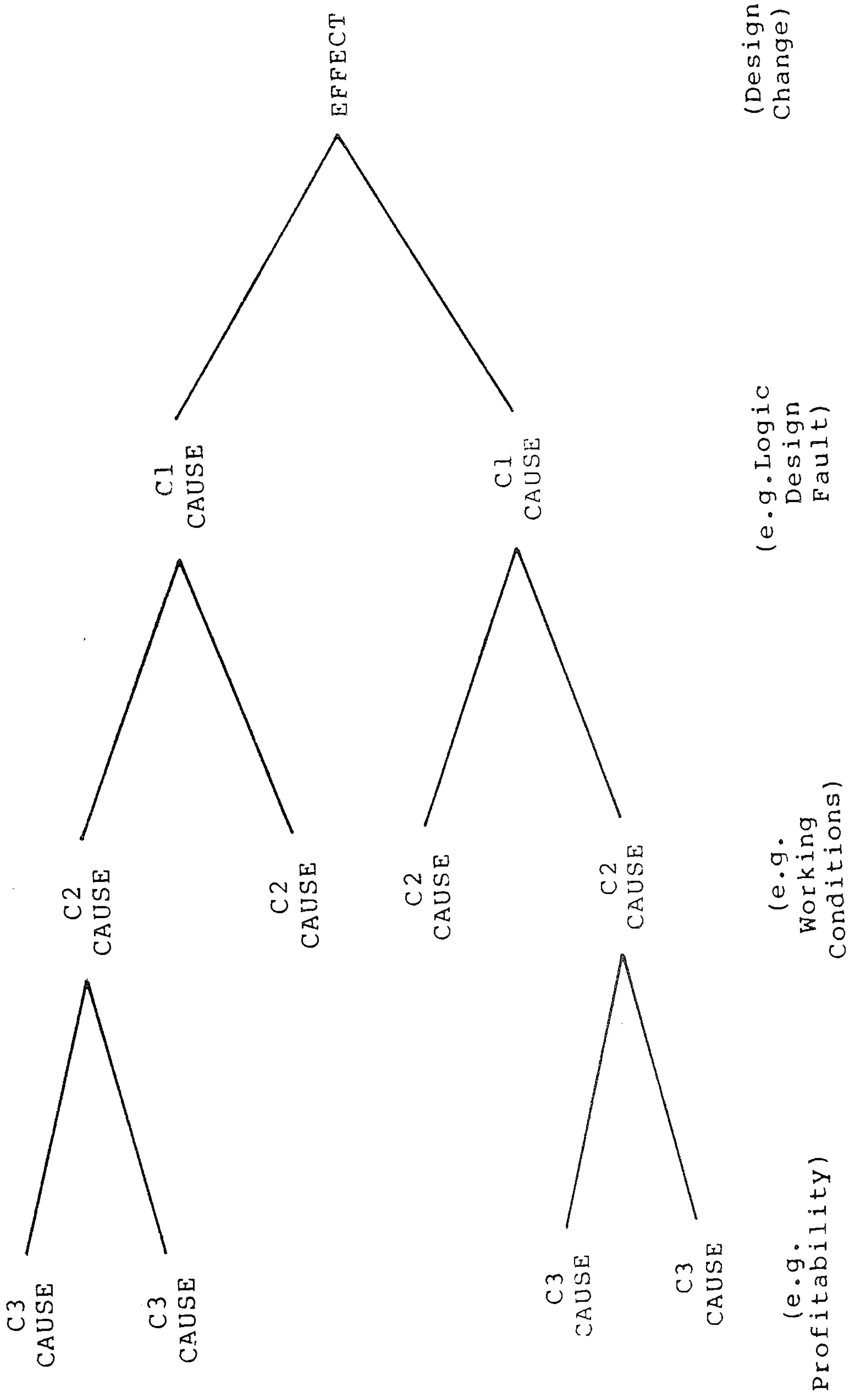


Figure H-1. Convention Used to Describe Causality of Design Changes.

APPENDIX I

OPERATION AND CAPABILITIES OF TEGAS

Since TEGAS played such an important part in the trial described in Chapter 9, an explanation is given here about how it is used and what are its capabilities.

Main Functions.

TEGAS has three modes of operation:

- Mode 2: This simulates the function of the circuit assuming mean value component values,
- Mode 3: This simulates the function of the circuit assuming worse case component values,
- Mode 5: This is used to assess the effectiveness of test programs.

In all cases, it is only the logical operation of the circuit that is simulated. The electrical signals are assumed to be free from spuriously picked-up noise.

Operation.

After the designed circuit is coded into the computer (digitised) for computer aided design of the physical realisation, it is passed to TEGAS for simulation. TEGAS sets up a model of the circuit using the connections defined during the digitisation and models of the individual integrated circuit components (known as "macros"). The design engineer can then proceed with simulation.

He provides, in coded form, a set of input signals. This can be applied to the normal connection points of the circuit or any other internal points. In Modes 2 and 3, TEGAS responds by indicating how different parts of the circuit would respond. Although TEGAS does indicate some circuit problems, it is up to the engineer to verify that the result given by TEGAS is what he expected. For complicated circuits, the engineer can isolate parts of them and exercise those parts independently of the rest.

In Mode 5, the engineer specifies waveforms similar to those which would be provided by a circuit tester. TEGAS simulates the circuit performance and indicates the percentage of nodes that are exercised as a result of those waveforms: in other words, the effectiveness of the test signals. When the engineer is satisfied that the waveforms are adequate, they are used to generate the tester control program. This is done by the computer.

Capabilities of TEGAS.

This Section addresses the question of what types of fault TEGAS can be used to detect.

As indicated in the preceeding description, they fall into two categories:

1. Faults Directly Detectable by TEGAS. These faults are not usually associated with the functional objectives of the circuit, but are usually more closely related to violations of the more obscure design rules. The faults are often indicated by TEGAS without the engineer needing to provide input signals. Typical faults of this type that might be detected are:

   - Circuit race conditions where two signals follow different paths to the same component and the result of which gets there first is unpredictable, and

   - Unpredictable power-up states.

2. Faults which Must be Found by the Engineer. These necessitate the engineer interpreting the

simulated output of the circuit and and comparing this with what he would expect. This category of faults includes circuit function faults.

It must be noted that for the second category of faults, which are probably the most difficult to find by other means, the effectiveness of TEGAS is dependent in the "input" signals provided by the engineer and how he interprets the "output".

NOTE.

Although the faults detectable by TEGAS have been divided into two categories, either type may include Logic Design and other faults. Specification faults will only be picked up by accident.

REFERENCES

1. NORRIS, K.P. "The accuracy of Project Cost and Duration Estimates in Industrial R&D" R&D Management Vol:2 No:1 Oct 1971 pp 25-36.

2. EPTON, S.R. "The Underestimation of Project Duration": An explanation in terms of a Time-Horizon" R&D Management Vol:2 No:3 June 1972 pp 141-142.

3. GREEN, K. "Project Duration & the Time Horizon" R&D Management Vol:3 No:2 Feb 1973 pp 97-98.

4. GARNER, J. & NAIMAN, C.S. "Making Technology Transfer Happen" Research Management Vol:XXI No:3 1978 pp 34-38.

5. KEIDER, S.P. "Why Projects Fail"

Datamation Vol.20 No.12 Dec 1974 pp 53-55

6. KURKJIAN,L.H. "Tools for Effective Project Management of Medium Size R&D Programs"
IEEE Transactions on Engineering Management Vol:EM-10 No:2 1963 pp 70-73

7. RUBENSTEIN,A.H. "A Program of Research on the Research & Development Process"
IEEE Transactions on Engineering Management Vol:EM-11 1964 pp 103-112

8. TWISS,B.C. "Managing Technological Innovation"
Longman 1974

9. GEE,S. "Factors Affecting the Innovation Time Period"
Research Management Vol:XXI No:1 1978

10. MARSHALL,A.W. & MECKLING,W.H.
"Predictability of the Costs, Time & Success of Development" In National Bureau of Economic Research "The Rate and Direction of Inventive

Activity: Economic and Social Factors"
Princeton University Press 1962 pp 461-475

11. GEE,S Op.Cit.

12. PESSEMIER,E.A. "Product Management: Strategy and Organisation"
Wiley/Hamilton Series in Marketing 1977 pp 190-193

13. BOSTON CONSULTING GROUP INC.
"Perspectives on Experience" 1968

14. CONLEY,P. "Experience Curves as a Planning Tool"
IEEE Spectrum June 1970 pp 63-68

15. TWISS,B.C. "Managing Technological Innovation"
Longman 1974.

16. MONTGOMERIE,G.A. "Industrial Innovation and the Interdisciplinary Approach"
IEE Control & Automation Division: Chairman's Address 1976.

17. HARRIS,L.R.F. "Introduction to System X"

International Switching Symposium
Paris, May 1978

18. PRICE,C.D.E. & ALLEN,D.
"System X Information System"
International Switching Symposium
Paris, May 1978

19. KIRTLAND,J.P. & TIPPLER,J.
"System X - Architecture & Technology"
International Switching Symposium
Paris, May 1978

20. POST OFFICE TELECOMMUNICATIONS
"System X, The Complete Approach to Telecommunications"
Brochure 1978

21. GERSTENFELD,A. "Effective Management of Research and Development"
Addison-Wesley 1970

22. OPPENHEIM,A.N "Questionnaire Design and Attitude Measurement"
Heinemann Books on Sociology
Heinemann Educational Books 1966

23. OPPENHEIM,A.N Op.cit. pp 49-80

24. TWISS,B.C. Op.cit.

25. MONTGOMERIE,G.A. Op.cit.

26. PANNENBORG,A.E. "Industrial Innovation"
IEE Proceedings Vol.127 Pt:A No:7
Sept 1980 pp 495 - 497

27. RUBENSTEIN,A.H. & DOUDS,C.F.
"A Program of Research on Coupling
Relations in Research & Development"
IEEE Transactions on Engineering
Management Vol: EM-11 No:4 Nov 1969
pp 137-143

28. DIXON,J.R. "Design Engineering: Inventiveness
Analysis & Decision Making"
McGraw Hill 1966 pp 9-11

29. ASIMOW,M. "Introduction to Design"
Prentice Hall Series in Engineering
Design 1962 p.12.

30. ULRICKSON, R.W. "Solve Software Problems Step by
Step"
Electronic Design Vol.2 18th Jan
1977 pp. 54-58

31. BENSON,K. "Ensuring the Quality of Software"

Telesis Feb 1979 pp 23-27

32. BOSTON CONSULTING GROUP INC.
Op. cit.

33. ASIMOW, M. Op.cit. p 40

34. KURKJIAN, L.H. Op. Cit.

35. JOHNSON, D. "Simulation, Verification and Test Package for Logic Design"
Electronic Engineering Vol 52 No 633
Jan 1980 pp 81-89

36. CORFIELD, K.G. "Product Design", A report carried out for National Economic Development Council
National Economic Development Office
1979

37. MINISTRY OF DEFENCE
"Defence Standard 05-21/1 Quality Control System Requirements for Industry"
Defence Quality Assurance Board

38. BRITISH STANDARDS
"Guide to the Operation and Evaluation of Quality Assurance Systems"
BS5179 1974

39. BOSS,R.E. & McGAFFIN,C.H.
"Elements of Design Review for Space Systems"
NASA, Washington D.C.
Report Number NASA-SP6502 Contract NASW-1128 1967

40. JACOBS,R.M. "Design Review"
Proceedings of the 6th Annual Product Liability Conference, Newark, N.J. 19-22 Aug 1975 IEEE, New York. pp 49-59

41. ROBERTS,J.A. "The Design Error Contribution to Failure"
Microelectronics & Reliability
Vol:14 No.2. April 1975 pp 159-172

42. ROETHLISBERGER,F.J. and DICKSON,W.J.
"Management and the Worker An Account of a Research Program Conducted by the Western Electric Company, Hawthorne Works, Chicago"

Harvard University Press 1939

43. ENSTONE,C.A.G. and COSLING,L.M.
"Operating Manual for MONICA (Modular On-line Interactive Circuit Analysis)
Hirst Research Centre, GEC Ltd. Internal Document 1976

44. ARON, J "System Development"
I.T.S.C Joint IBM and University of Newcastle Seminar on Computing System Design, 1976

GLOSSARY

Change Note	A development department generated note used to record the need for a design, specification or documentation change. See Appendix D.
Commissioning	This term is used for the process of testing models and prototypes to evaluate a new design.
C1 Cause	The immediate cause of an effect. See Appendix G.
C2 Cause	A secondary cause of an effect. See Appendix G.
Design	The process of synthesis or inventiveness which produces the new thing or concept, which may be the whole or part of the product. See Section 7.3.
Design Fault	A fault which occurs within the design activity and which causes the product to behave in a manner different from the way

	it was specified to behave. See Section 4.5.
Development	The series of linked activities which result in a new product or process being designed optimally and prepared for commercial exploitation. See Section 7.3.
Equipped Shelf Group	A physical equipment assembly consisting of one to three shelves equipped with Slide-in-units. Each Equipped Shelf Group usually represents a circuit function.
Formal New Product Information	Information which is contained within the prescribed formal documentation structure of a new product.
FUNC	A program contained within the Computer Aided Design suite of programs, used within the Sponsoring Company, to verify that the design conforms to certain design rules.
Functional Entity	Part of an Electronic System which performs a basic function.

Lead Time	The time from the original concept of a new product until that product is in volume production, with no further development laboratory intervention necessary to maintain production. See Section 3.2.
Informal New Product Information	Information about a new product which is not contained within the prescribed formal documentation scheme. It includes memoranda, verbal communication, etc..
LPU	The large System X processor subsystem.
MAP	See "Modification Action Package".
Model	An advanced prototype built as far as possible to manufacturing standards using a volume production process.
Modification Action Package	A package of documentation which describes how manufacturing should modify a piece of equipment to incorporate a change.

Rack	A mechanical assembly used to support Wired Shelf Groups and other equipment elements. See Section 2.5.
Rework	The activity of incorporating design changes into the design to produce a new "clean" set of documentation.
Slide-in-unit	The lowest assembled level of hardware. It typically consists of components mounted on a printed wiring board which plugs into a shelf or Wired Shelf Group.
Specification	A description of the requirements that a new design should aim to achieve.
SPU	The small System X processor subsystem.
Subsystem	The highest subdivision of a system. They can be either hardware or software and, in the former case, can often function as stand alone systems.
System	A complete product which is capable of performing a function.
TEGAS	A logic simulation system used within the Sponsoring Company.

Telephone Switching Equipment	Equipment used in the construction of telephone exchanges - usually in the public telephone network.
Unit	See "Unit of Equipment".
Unit of Equipment	The lowest level of assembled hardware in a large hierarchically structured product. They are often constructed as Slide-in-units.
Wired Shelf Group	A physical equipment assembly into which Slide-in-units plug. They contain the necessary wiring to inter-connect those Slide-in-units.

The End