DESIGN FUNCTION DEPLOYMENT
A CONCURRENT ENGINEERING DESIGN SYSTEM

BY

NOSAYABA FRANCIS OSA EVBUOMWAN
BEng MSc DIC CEng MIStructE

THESIS

SUBMITTED FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

CITY UNIVERSITY

IN

COMPUTER AIDED ENGINEERING DESIGN
AND
MECHANICAL ENGINEERING

DEPARTMENT OF MECHANICAL ENGINEERING AND AERONAUTICS
CITY UNIVERSITY
NORTHAMPTON SQUARE
LONDON EC1V 0HB

JULY 1994

CONTENTS

LIST OF ILLUSTRATIONS AND FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I wish to express my heartfelt thanks and appreciation to my supervisor and advisor, Professor Alan Jebb for giving the vision to this project. His wise counsels, insightful comments and leadership were greatly invaluable in directing me throughout the period of the research. I am also greatful for the moral support he gave me as well as being readily accessible whenever he was needed, during the course of my research.

My sincere thanks also go to my colleague, Dr Sangarappillai Sivaloganathan who contributed in no little measure to the success of this project. As the Senior Research Assistant in the Engineering Design Centre, he provided necessary leadership and guidance throughout the period of this research. He was an immensely valuable 'intellectual sparring partner'.

I am also greatful to my other colleagues in the Engineering Design Centre who provided a pleasant environment to work in, as members of a research team. Ms Hilda Iu was also helpful in preparing some of my figures.

My wife and children have been very understanding and supportive throughout the research programme. My deep appreciation and gratitude go to them and to Bola Adesanya for occupying the children.

Finally, I would also like to express my thanks to the Science and Engineering Research Council (SERC), now Engineering and Physical Sciences Research Council (EPSRC), UK, for supporting me throughout the period of the research.

DEDICATION

To my wife Olufunmilayo Olutoyin Anne

and

my children Uyiosa and Demilade

To my teachers in the early years who had confidence in me and gave me confidence.

and

To the Ancient of Days, El Shaddai and My Ebenezer

The Lord God Almighty

Declaration

I grant powers to the University librarian to allow this thesis to be copied in whole or in part without further reference to me. This permission covers only single copies made for study purposes, subject to normal conditions of acknowledgements.

Nosayaba Francis Osa Evbuomwan

ABSTRACT

The current state of activities in the design and manufacturing industry is marked by the various CAD/CAM/CAE systems which exist as islands of automation, and are used by engineers and designers in a non-integrated and ill-structured way. Thus the design problem is examined from separate and different perspectives, rather than as a whole. The goal of this research, is to develop a comprehensive, integrated and generic design system, that will ensure the realisation of concurrent engineering in practice. To this end, Design Function Deployment (DFD) has been developed.

DFD enables the capture of customers' requirements, the establishment of design specifications and constraints in a solution neutral form, the generation of conceptual designs (architectures), the development of detailed designs (layouts), the selection of materials and associated manufacturing processes and the development of suitable production plans. The generated design solutions are optimised against a composite set of multi-criteria (attributes) in a concurrent manner for key factors such as performance, robustness and cost as well as other life cycle issues (manufacture, assembly, serviceability, reliability, environment, etc) in order to choose the most satisfying design.

DFD provides a recipe of design methods to support the designer or design team at any stage of the design process. The optimisation process involves the use of these supporting design tools (methods) encapsulated within it. DFD also provides an integrated product modelling environment which integrates both textual and geometric design information, and enables the capture of other design information related to design intent, rationale and history. The research that led to the evolution and development of DFD involved (a) a detailed investigation and research on Quality Function Deployment, QFD, a technique well suited for capturing and translating customer requirements into design specifications, (b) an extensive review of design philosophies, models, methods and systems and (c) an extensive investigation into concurrent engineering.

The findings of this research has led to the development of the structure of the DFD system, which incorporates (1) a prescriptive design model, (2) a suite of design methods and (3) supporting knowledge/rulebases and databases, which are used for the generation of the design solution space and the optimal selection of the most satisfying design for subsequent implementation.

CHAPTER 1

INTRODUCTION

1.1 GENERAL INTRODUCTION

The design and product development process has been one of the activities that man has performed over the centuries, in varying degrees and levels of details, technology and sophistication. The recent past up till now, has witnessed rapid advancements in science, engineering and technology as well as the advent of computers and the evolution of software engineering. The consequence of these developments on engineering design and product development, has also been the emergence of tools and techniques that can be regarded as computer aided design (CAD), computer aided manufacture (CAM), computer aided engineering (CAE) and computer integrated manufacture (CIM) systems. These systems within the design and manufacturing industry exist as clusters of islands of automation, and are employed by engineers and designers in an ill-structured way in performing necessary design and engineering functions during product development.

Besides these individual clusters of design tools, the product development process as practiced by many companies apart from being sequential in nature, is carried out in a compartmentalised fashion by the various functional groups, that is, marketing, sales, product planning, design, analysis, manufacture, testing, etc. This compartmentalisation of design activities commonly regarded as the 'over the wall' approach to product development, further encouraged the segregation of the cluster of tools highlighted above.

The consequences of the compartmentalisation of design activities, in addition to the engineering tools existing as islands of automation, are the several difficulties and limitations that most companies have to contend with. Key amongst such limitations and difficulties are : (a) the rigid sequence of design decisions that characterise the design process, (b) the manual labour cost of defining, recording and communicating design details, (c) the lack of consideration of downstream issues such as production planning, maintenance, producibility, supportability, etc, early in the design process, resulting in costly design changes, (d) the fragmentation of design data and lack of consistency in design representations, (e) the loss of design information about the product and/or design intent, as the design progresses, and (f) lack of cost information and cost estimation tools.

There is also currently an increasing awareness worldwide that engineering design is undoubtedly the wealth creating function in every human society. This is in line with the view put forward by the Duke of Edinburgh [1] who said that : " *The wealth of a nation, and all that implies, depends in fact upon the efficient organization of its resources both natural and industrial as well as human. In this organization the engineer bears the chief responsibility.* ". In his viewpoint as the chairman of the Design Management Committee (DMC) of the Science and Engineering Research Council (SERC), Professor Paul Braiden wrote in the engineering design newsletter [2] that " Engineering design, closely coupled to an efficient, responsive manufacturing system, producing reliably and repeatedly what the customer wants at an acceptable price, lies at the heart of every buoyant modern economy ", thus reinforcing the earlier viewpoint.

In the past, most manufacturers did not consider issues relating to customers' perception and acceptance of their products. This was partly due to the fact that competition was not rife, and customers had no choice but to accept whatever they got. In today's world however, customers have become more sophisticated; demanding more variety of products as well as favouring 'environmentally friendly' products, this being an offshoot of the increased range of available choices and fierce market competition. Customers are now prepared to pay more for products as long as they contain high technological content, exhibit high quality and reliable performance as well as being aesthetically pleasing and ergonomically acceptable.

A further consequence of increased competition in the market place is the increasing pressure on manufacturers to release their products early, i.e. on 'Time to market'. The term 'Time to market' can be generally defined as **the elapsed time between product definition and product availability.** In addititon, the current wave and emergence of new and improved products, extensions and expansions of product lines, revisions and enhancements of products, thus creates additional pressures on manufacturers to keep a steady stream or flow of new products into the market place [3]. A major influencing factor in achieving 'Time to market', is the design process, which has significant impacts on the downstream functions of product development. Faulty, hurried or inadequate product definition and design, usually results in design changes in the form of engineering change orders (ECO's), which are always costly in both time and money, irrespective of whatever sector of industry. Furthermore, the later these design changes are implemented in the development cycle, the more costly the implementation.

Over the last 5 to 10 years, as a result of increased competition in both the local and international market, several companies in the manufacturing and allied industries, have started to place due emphasis on the design function as the prime mover for the development of quality products at competitive costs and early time to market. Ensuring quality in products can only come about if the design activity is done right the first time.

The need to address the issues discussed so far, led to the emergence of the concurrent engineering paradigm.

Concurrent engineering, (also known as simultaneous engineering, forward engineering, parallel engineering, life cycle engineering, and so on), has received world wide acclaim as a progressive approach to product development. The consequence of this popularity is made evident by the many definitions that have been given to it.

The most popular of these definitions is that given by Winner et al [4] which states that, "Concurrent Engineering is a systematic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause the developers, from the outset, to consider all elements of the product life cycle from conception through disposal, including quality, cost, schedule, and user requirements". A variant of the above definition which is also more encompassing is that given by Cleetus [5], which states that "Concurrent Engineering is a systematic approach to integrated and concurrent development of a product and its related processes, that emphasizes response to customer expectations and embodies team values of cooperation, trust and sharing in such a manner that decision making proceeds with large intervals of parallel working by all life-cycle perspectives early in the process, synchronised by comparatively brief exchanges to produce consensus". An examination of the definitions on concurrent engineering, highlight certain issues that are pervasive in all of them and which can be considered as the main goals of concurrent engineering. These include the need to :

(1) improve and maintain the quality of design and the resulting product.
(2) reduce product development lead time.
(3) reduce product development costs.
(4) integrate the design of a product, manufacturing and production processes.
(5) resolve and manage conflicts and tradeoffs in the early stages of design.
(6) breakdown barriers (walls) between product development functional groups.
(7) encourage the integration and use of all company resources.
(8) respond proactively to customers and their needs.
(9) parallel the design process.

The consideration of the above goals within the context of realising the benefits of concurrent engineering, makes imperative the need to establish procedures and processes and to develop resources, tools, techniques and systems to support them. In this regard, several viewpoints and issues have been promulgated as constituting the basis for the implementation and realisation of concurrent engineering. These are :

(1) Establishing a focus on the client, customer or user of a product
(2) Integration of the organisation
(3) Use of design and product development teams
(4) Employee involvement and participative management
(5) Competitive benchmarking
(6) Focussing all activities on quality, cost and delivery
(7) Adopting a concurrent (parallel) product development process
(8) Integrating design of product with manufacture and support processes
(9) Establishing strategic relationships with suppliers
(10) Integrating computer aided tools and techniques
(11) Use of project management techniques
(12) Consideration of new materials and technologies
(13) Synchronisation of design information and
(14) Use of computer hardware and software.

Other viewpoints include those that support computer integrated manufacture, the advancement of geometric and solid modelling tools, the development of communication protocols to support collaborative design processes and engineering software to support concurrent engineering. In examining the concurrent engineering goals highlighted earlier as well as the implementation issues enumerated above, it is evident that the operative term that runs throughout, within the context of design and product development, is **INTEGRATION**. In seeking to realise the benefits of good design, efforts are being made in industry to integrate all design activities in a coherent manner. For design to ensure quality, it has to be done in a planned and controlled way.

The development and adoption of a systematic methodology for design [6] and the whole product development process is hence needed to :

(1) integrate CAD/CAM/CAE systems which exist as islands of automation.
(2) integrate and support various functional groups and disciplines involved in the product development process, enable the breaking down of walls between them as well as overcome the difficulties associated with compartmentalisation.
(3) increase the efficiency and effectiveness of the design and product development team.
(4) support the synchronisation of design information.
(5) support the wealth creating function of design.
(6) enable the design and product development team to respond proactively to customers in todays' market climate.
(7) enable the early release of products to the market place.
(8) ensure quality in the design process as well as quality of the resulting product.

(9) enable the achievement of the goals and implementation issues of concurrent engineering.

To address this, extensive research was carried out into Quality Function Deployment, Design Models, Methods and Systems and Concurrent Engineering. This led to the emergence and development of Design Function Deployment (DFD) as a comprehensive design system that would (i) ensure the integration of the quality function into design, (ii) integrate the various functional groups participating in design and product development, (iii) integrate the clusters of individual and groups of CAD/CAM/CAE tools and techniques and design related knowledge and activities and (iv) provide a platform for the realisation of Concurrent Engineering.

The DFD system also integrates :

(1) all the design knowledge (related to both the process and product) generated during the design process. In relation to the product, the integration involves both textual and geometric information throughout the stages of design.
(2) the different stages or phases of design right from product planning and conceptual design to detailed design. In DFD, this integration is over five distinct stages starting from customer requirements capture to production planning.
(3) all necessary design methods, tools and techniques (representing the various life cycle aspects of the evolving product) to support the designer/design team during the design and product development process.
(4) the various key players in the product development process, that is, customer, marketing, sales, design, analysis, manufacture, production, testing, service, maintenance, etc.
(5) the design of the evolving product and the associated manufacturing process(es) and other related life cycle aspects. This concept of integrating the design and manufacturing activities is akin to the concept of the shared central database aimed at integrating the CAD features of geometric modelling, analysis, testing and draughting with the CAM functions such as numerical control, process planning, robotics and factory management, as highlighted by Besant and Lui in their book [7].

1.2 AIMS AND OBJECTIVES OF RESEARCH

The principal goal of this work is to develop a comprehensive design system, that enables a detailed and comprehensive examination of the design problem (requirements), the translation of the requirements into solution neutral design specifications and constraints, the generation of alternative workable design solutions and the optimal choice of the design that robustly satisfies all the life cycle aspects and constraints bounding the design. Thus incorporating a design methodology and model, based on the "Design It Right" philosophy. The system is developed as a concurrent engineering design system providing a recipe of design methods, tools and techniques to assist engineers and designers to proactively respond to new and modern demands of the design and manufacturing industry, and to develop products/ processes and systems that satisfy the increasing sophisticated demands and requirements of customers and users. The system is also developed to provide a common integrated product modelling environment that captures both textual and geometric design information (data). The system will also be developed to support concurrency of the design process as well as the concurrent design of the evolving product. The resulting design system will accomodate the various aspects of the design life cycle, starting from the establishment of customer and other requirements down to the selection of materials and associated manufacturing processes and production planning, as well as all forms of design information generated during the design process. Such information includes data about the design history, design intent and the rationale for the design decisions. A key desideratum of the design system will also be the need to support the design of original/innovative, adaptive and variant designs.

1.3 RESEARCH SIGNIFICANCE

Several researchers and designers have done some work in the development of design methodologies and models. A majority of such design approaches, have been associated not only with the traditional and sequential approach to design, but have been limited to particular engineering design domains. Only a few have developed methodologies that could become generic in nature.

There is however presently a lack of truly generic design systems which can accomodate new and emerging technologies in the modern design and manufacturing environment. Current Computer Aided Engineering (CAE) software generally focus on the geometric aspects of the design of an artefact and do not accomodate all other information such as textual data generated during the design process, as well as all the life cycle issues of the design. There is hence a need for a computer aided design system, that addresses these needs and which can help to ensure a more integrated approach to the product development process.

The subject of design and quality is a crucial aspect of product development that needs to be addressed. Quality on the whole can be regarded as a measure of value or satisfaction in the perceptual framework of the customer. It represents an ability to satisfy stated and perhaps unstated specifications but also takes into account the total characteristics of the product or process, including costs, aesthetics, reliability, safety, maintainability and general "fitness for purpose". The focus here is : **QUALITY FUNCTION**, which is the area of responsibility or activities needed in a company or enterprise, through which 'fitness for use/purpose or customer-required quality is achieved [8]. In examining the issue of quality in design, two interacting perspectives need to be considered, that is, (a) the quality of the design process and (b) the quality of the resulting product. The quality of the product is to a large extent dependent on the quality of the design process. Several factors also influence the quality of the design process as well as the output of the process. These include: (a) the designer or design team, (b) the available tools and methods employed, (c) the existing design knowledge, (d) the management of the design process, (e) the active environment in which the process takes place, (f) the nature of the design process adopted, the procedures and the techniques employed and (g) the quality of the list of requirements as well as the resulting specifications [9]. Hales [10], from his research work, has also established that "the effectiveness and the efficiency of the engineering design process are strongly influenced by the way the process is managed".

The consideration of the above, in addition to the increasing emphasis in industry on designing quality into a product as well as the need to get it right first time, therefore necessitates a design system that will not only ensure the realisation of concurrent engineering, but also take account of the factors highlighted above, by providing a platform for ensuring quality of the input and output of the product development process, as well as that of the transformation activities of the design process. To meet these requirements, Design Function Deployment (DFD) has been developed to enable the integration of the quality function into the design and product development process.

1.4 SCOPE OF RESEARCH

This project involved an extensive research on several design models, methods and methodologies proposed by various researchers over the last three decades, as well as Quality Function Deployment (QFD) and Concurrent Engineering.

The second stage then involved the development of the concepts of Design Function Deployment (DFD) in line with designing for quality and the provision of a platform for concurrent engineering.

Thirdly the DFD concepts were then developed into relevant, applicable and useful formats for use by engineers and designers in a manual form. This was in the form of a user manual. This then led to the final aspect of the project, which involved the development of DFD into a suitable form for computer software implementation.

1.5 OUTLINE OF THESIS

This thesis will be discussed under eight chapters, each having their own subsections. This chapter, which is the introduction of the thesis, gives a general overview of what the project entails, why it was embarked on and how the research was carried out. Chapter 2 consists of the literature review of the associated research areas which influenced the work. Chapter 3 discusses the evolution of Design Function Deployment and Chapter 4 sets the scene for the requirements for a concurrent engineering design system. Chapter 5 which represents the kernel of the project, discusses Design

Function Deployment (DFD), its various facets, capabilities, model, implementation structure and design methods within it. Chapter 6 discusses how DFD ensures the realization of concurrent engineering and chapter 7 discusses the critical issues relating to the conceptual framework for the software implementation of Design Function Deployment, DFD, as a concurrent engineering design system. Finally chapter 8 discusses the conclusion of the project and highlights important areas for further research.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

This chapter discusses in detail, previous and relevant research work carried out by other researchers in the area of Quality Function Deployment, Design Philosophies, Models, Methods and Systems as well as Concurrent Engineering. This involves a detailed investigation and review of these subject areas and the work done to date. This chapter provides the background for a large proportion of the research work that led to the evolution of Design Function Deployment (DFD).

The research mission of the Engineering Design Centre at City University, is to expand the discipline of Quality Function Deployment into a full design for quality methodology by the incorporation of quality into design and its maintenance throughout product life. This mission was seen as not only the integration of quality into design, but also involving very important issues such as designing a product 'right first time' and getting it early to the market at a reduced production cost and selling price. Quality Function Deployment, Concurrent Engineering and Design Models, Methods and Systems were investigated and explored in seeking to accomplish the research mission. In the ensuing discussion, each of the research areas are discussed under separate sections.

2.2 A REVIEW OF QUALITY FUNCTION DEPLOYMENT

2.2.1 Historical Background of Quality Function Deployment

Quality Function Deployment (QFD) is considered to have originated in 1972 at Mitsubishi's Kobe shipyard site, in Japan, although it appears to have apparently grown out of value analysis/value engineering [11], which was first developed about 40 years ago. Besides value analysis/value engineering, QFD has been found to show some similarities to other methodologies such as 'Structured Product Analysis' (SPA) developed by Vinson [12], Problem Solving and Decision Analysis techniques practiced by Kepner and Tregoe [13], and the Unified Program Planning technique, a precursor to Interactive Structural Modelling developed by Warfield et al [14].

Following the activities in the Kobe shipyard, Toyota and its suppliers then developed QFD in numerous ways, and employed it in resolving their car rust problems. Various other Japanese companies and their suppliers, including manufacturers of consumer electronics, home appliances, integrated circuits, construction equipments, etc, also used Quality Function Deployment successfully. By the early to mid 1980's QFD had grown in popularity, leading to its introduction to the USA by Professor Don Clausing of MIT and Dr Lawrence Sullivan of the American Suppliers Institute, USA. Since then several major multinationals have taken QFD on board, the key players being Xerox, General Motors, Ford, Digital Equipment, Hewlett-Packard and AT&T. In its use, QFD was founded on the belief that products should be designed to reflect customers' desires and tastes, and its was used as a kind of conceptual map which provided the means for interfunctional planning and communications between marketing, design and manufacturing departments. The resulting QFD chart also meant different things to different departments. To engineers, it was a way of summarising basic design data in usable form. To marketing executives, it represented the customer's voice and to general managers, it was used to discover strategic opportunities [15].

QFD has been defined by various researchers from various viewpoints. Some of thekey definitions are reported below.

Akao [16, 50] defines QFD as " ... a method for developing a design quality aimed at satisfying the customer and then translating the customer's demands into design targets and major quality assurance points", while King [17] describes QFD, "as a process, focussed on improving the efficiency of the initial product design. It provides a systematic environment for designing product (service) based on customer wishes (demands) and involving all members of the producer (supplier) organisation". Hales [18] on the other hand defines QFD as "... a method for systematically focussing all organisations in your business unit towards satisfying the requirements on your product which are most important to the customer". A more elaborate definition was that by Mazur [19]. This states that "QFD is a system and procedure to identify, communicate, and prioritise customer requirements so that an organisation can optimise its products and services to exceed customer expectations. Identification is achieved through the voice of the customer analysis, communication is achieved through a series of linked matrices, and prioritisation is derived from the customer, competitors, and the vision of the company. Optimisation activities are then focussed on those areas that mean the most to the customer, beat competition, and are in line with the vision of the organisation". Another definition given to QFD by the American Supplier Institute (ASI) [43] states that "QFD is a system for translating customer requirements into appropriate company requirements at each stage from research and product development to engineering and manufacturing to marketing/sales and distribution"

Without any doubt, examining these definitions reveal that the focus of QFD is on how to satisfy customer demands. Other associated aspects include improving efficiency of designers, improving teamwork and optimising business and product development practices.

QFD emphasizes co-operation, convergent consensual decision making, and systematic linkage of engineering activities [20]. Benefits that have been attributed to QFD include, low manufacturing costs, shorter development time, easier entry into production development of product that appeal to customers and better and improved product quality. QFD also encourages multiple criteria decision making and recognises that product and process development is not a one dimensional exercise, but many trade-offs need to be performed before arriving at an optimal solution [21].

2.2.2 Basic Quality Function Deployment

The previous section gave the introduction and historical background to Quality Function Deployment. This section however, will focus on the basic processes and constituents of QFD. Quality Function Deployment refers to the deployment of quality consciousness throughout the entire functions of company areas such as design, manufacturing, service, etc. [16]. It has also been considered as a tool for the implementation of the strategy of Total Quality Management, during product development [22]. Within the context of the construction industry, QFD is being proposed as a methodology that can be used during the early phases of a project to create more accurate decisions, focus project budgets, define project quality, and to respond to customer's needs. It is also seen as a cross-functional tool that assists technically oriented people, such as architects and engineers, to understand customer requirements sufficiently to develop priorities for these requirements that are customer-oriented and technically correct [23]. The basic planning tool used in Quality Function Deployment is the product planning chart, which is popularly referred to as the "House of Quality", and it is conceptually shown in Figure 2.1. This chart is a visualizing tool, which contains 9 rooms or blocks that represent different planning activities. These activities are described below.

Block 1 - Identification and Establishment of Customer Requirements

The activity here involves thorough identification of the customer requirements, also referred to as the voice of the customer, needs of the customer or customer demands [6], and are called the 'WHATS' in the matrix. They are obtained through market research involving personal interviews and/or focus groups (6 to 8 people involved in discussions). These requirements are usually represented in the customers language and does include **basic needs** (what they just assume the product will do), **articulated needs** (what they say they want the product to do), and **excitement needs** (which, if they were fulfilled, would delight and pleasantly surprise customers). These are also classified by King [17] into expected or demanded quality and attractive or exciting quality. Having elicited these requirements, they are then structured into a hierarchical format for easy management, in the form of primary, secondary and tertiary requirements. This also ensures that during the prioritisation process, the customer requirements can be compared on the same basis and level of generality.

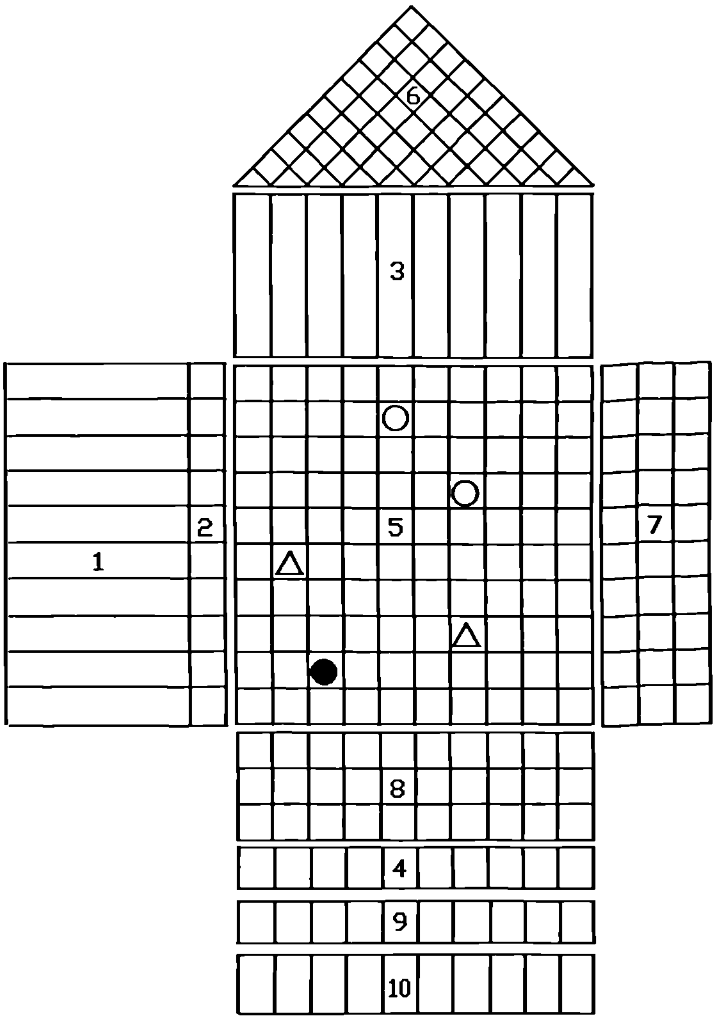


Figure 2.1 The Basic Quality Function Deployment Chart

A popular technique for this process is known as the Kawakita Jiro (KJ) diagram where the customer requirements are organized into affinity groups [24]. The primary requirements represent the strategic top level requirements and set the strategic direction for the product. The secondary requirements which are the elaboration of the primary requirements, represent the tactical requirements. They indicate more specifically what can be done to fulfill the corresponding strategic or primary requirements. The tertiary requirements are the sub-divisions of secondary requirements, and they indicate more specifically how the secondary requirements can be fulfilled [25].

Block 2 - Prioritization of Customer Requirements - The 'Quality Plan'

In any QFD exercise many tertiary customer requirements are derived. Some of them however, will be more important than others. It is hence necessary to prioritize them with respect to one another. This will help the design team to balance the cost of fulfilling a requirement with the benefit to the customer, as well as helping to focus on the most important requirements. The prioritization process usually involves direct market research with customers and results in what is known as the 'Quality Plan'. Quality Plan is the process of weighting the tertiary customer requirements, taking account of the customers' rating, sales advantage and quality target improvement factors. The process of constructing the 'Quality Plan' involves firstly, rating each tertiary customer requirement based on the degree of importance to the customer (usually on a scale of 1 - 5), the degree of improvement needed (i.e. the ratio of the targeted rating by the customers to the current rating of a company's product from competitive benchmarking) and sales advantage (where the values 1.5, 1.2 and 1.0 represent very important, important and not important respectively). These weights should be applied with caution as they change the relative importance of the requirements quite considerably. The absolute importance rating of each teritiary customer requirement is the product of the ratings obtained from the above three contributing factors. Then the relative ratings of the teritiary customer requirements are obtained by normalising the absolute values on a scale of 1 to 9.

Block 3 - Establishment of Design Requirements - The Quality Elements

Design requirements, also referred to as substitute quality characteristics, corporate expectations for the product or engineering characteristics, are known as the 'HOWS', and then represent the design attributes (voice of the engineer) that fulfill the customer requirements. They are usually translated from the customer requirements. These design requirements must be actionable and measurable, in order to ensure competitive advantage and that the design team is working to the same requirement. The final development of the design requirements, is attributetd to two sources, that is, deployment from the customer requirements and then amplification from functional requirements (product expectations characteristics). Once the design requirements have been compiled, they are also categorised into primary, secondary and tertiary design requirements, in a similar way as done for customer requirements using techniques like team voting and the KJ method [24].

Block 4 - Setting of Target Values for Design Requirements

To indicate the measurability of the design requirements, each of them is assigned physical measurements known as target values, which have the least variability as possible. The target values are the levels of performance required for each of the design requirements to assure the level of expectation of the customer. They must be measurable and not based on capability but rather on the level of customer expectation [26]. Associated with the target values are the directions of improvements. For some of the design requirements, exceeding the target values will result in improvements, for others achieving a lower value than the target value gives an indication of improvement, while in the case of some achieving the target value is the most desired condition.

Block 5 - Development of the Relationship Matrix [24]

This involves determining which design requirement influences which customer requirements and by how much. Furthermore, to overcome the traditionally poor translation of the customer requirements into design requirements, the design team assesses the fidelity of the design requirements to the customer requirements, with the help of the relationship matrix. The strength of the positive response of a design requirement to customer requirements (strong, medium, weak, or negligible) is expressed by different symbols or numbers. When numbers are used, the numbers 9, 3 and 1 are used to represent strong, medium and weak relationships respectively. In filling out the relationship matrix, it is important that there is a consensus amongst members of the design team for each relation that is represented in the cell. If any rows are left empty, the team has to develop design requirements to satisfy the customer requirements. On the other hand, if any empty columns appear, it is either the market research did not address all features of the product or those particular design requirements are unnecessary or redundant.

Block 6 - Development of the Correlation Matrix

This block which represents the 'roof' of the 'House of Quality', is used to consider the possible interactions (which could be interferences or synergies) between the derived design requirements, in order to identify areas of inherent conflicts between the design requirements early in the design process. Premature tradeoffs should however be avoided. The use of the matrix helps to avoid making engineering changes and a substantial amount of rework. The correlation matrix helps to provide insight into needed communications between design teams that comprise the product development team, as well as helping to optimize the entire product, avoiding suboptimization.

Block 7 - Competitive Benchmarking with Customer requirements

Benchmarking is the process of comparing one's product or design and associated performance levels against those of other companies or competitors in order to gain and acquire new insights and information about ones relative position to them, and to identify key attributes and opportunities that will guide future improvements and enhancements. The activity in this block involves competitive benchmarking of existing products that address the same market segment to the product being developed. This process is carried out because of the fact that customer perceptions of how well a company's product and competitive products fulfill customer needs are useful for guiding product design [25]. The process involves using a graphical display of how well each customer requirement is fulfilled by both a company's own product and competitive products, and to identify any gaps between the best product and the company's product, and hence identify targets, goals and areas for competitive advantage.

Block 8 - Competitive Benchmarking with Design Requirements

Here both competitive products and a company's own product are benchmarked, against the design requirements, that is, the 'Hows', translated from the customer requirements. The aim here is also to determine targets for the design requirements which satisfy the customer requirements and hence exceed competitors design.

Block 9 - Assessment of Technical Difficulty

This block contains numeric values usually on a 1 - 5 scale, used to indicate the level of technical difficulty involved in the achievement of any particular design requirement. The difficulty may be related to design, manufacturing, suppliers or competitive pressures, but its determination should be tailored to the problem under consideration [26]. Depending on the level of difficulty indicated for the design requirements, necessary actions can then be taken to resolve the impending difficulties, either by carrying out research experiments and tests, utilising new materials and technologies or seeking expert opinions.

Block 10 - Evaluation of Importance of Design Requirements

This block contains the evaluations of design requirements that are derived from the relative significance of customer requirements to achieve the determined target values of design requirements. Based on the strength of the relationship between customer requirements and design requirements (scale of 1, 3 and 9), the importance of the design requirements is the sum of multiplying the relationship by the importance rating of the customer requirements. This resulting value represents the absolute importance of the design requirement. The absolute values for all the design requirements are then normalised on a scale of 1 to 9, to obtain the relative importance values. The importance rating is useful for prioritising efforts and for making design trade-offs.

The above activities described for each block of the 'House of Quality', applies to the Product Planning stage, and they are aimed at determining the necessary product specifications. The overall Quality Function Deployment process however involves, four main stages of Product Planning, Design (Parts Deployment), Process Planning and Production Operations Planning. In the subsequent stages beyond Product planning, a similar procedure as described above, is adopted. In the Design (Parts Deployment) stage, the critical (most highly rated)design requirements, that is, the 'Hows' from the product planning stage, are selected for further study and become the 'Whats' in the stage 2 chart. They are then deployed into parts characteristics which now represent the 'Hows' at this stage. The deployment process here involves establishing the functions of each part following the preparation of a bill of parts and determining the critical parts characteristics necessary to achieve the critical design requirements from the product planning stage. In the Process planning stage, the critical parts characteristics are also selected and become the 'Whats' and are deployed into process planning parameters, that is the 'Hows'. The depolyment process involves determining potential materials, manufacturing processes and shape, after which a master flow diagram is constructed, and from which the critical process planning parameters are determined. In the fourth and final production operations planning stage, the critical process planning parameters from the previous stage are selected and then deployed into production operations parameters. This involves determining the critical control, production maintenance, mistake proofing, education and training requirements, etc, to accomplish the production operations parameters. This deployment through each of the four stages of QFD is shown in Figure 2.2.

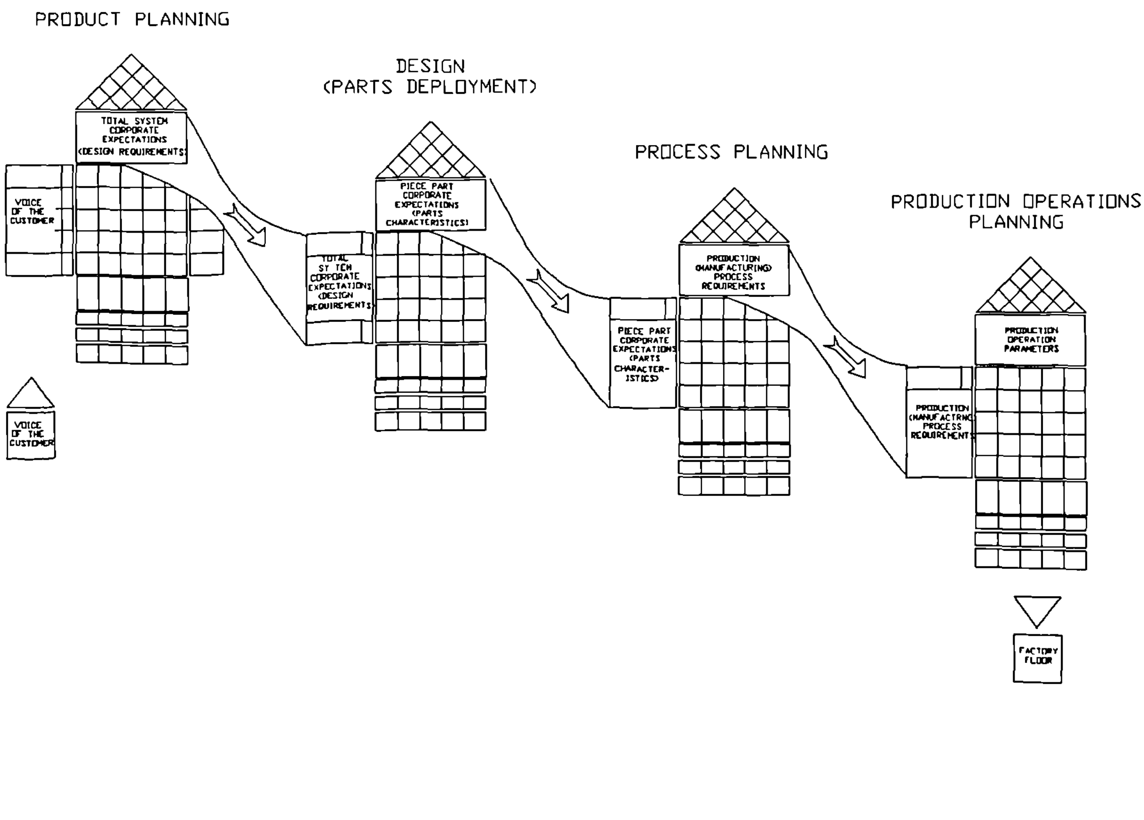


Figure 2.2 The Deployment Through the Four Stages of QFD

The basic form of QFD has been shown to exhibit the following main advantages [22]:

(i) It helps to improve communication throughout the various departments and functions within a company, as well as helping them to focus on their collective tasks as they participate in the product development process. Thus it's effectiveness downstream from stage 2 to 4, diminishes.
(ii) It provides a structured way of designing in quality into the product or system, and ensures that the product development process is guided by the need to satisfy the customers' requirements
(iii) The house of quality charts can be used as databases to build and store a lot of design knowledge during the product development process.

On the other hand certain limitations have also been observed in the use of QFD, and these include :

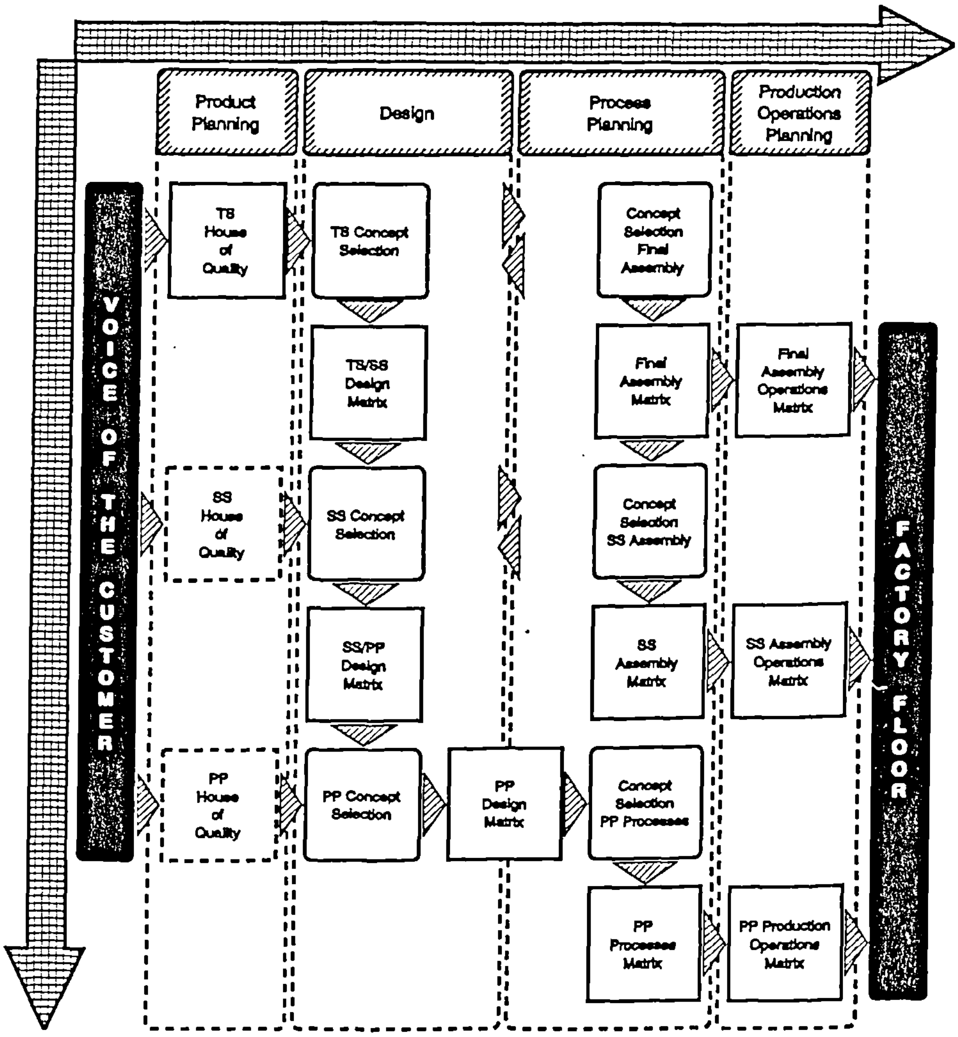
(i) In translating to the second, third and fourth stages involving the determination of piece parts parameters, process parameters and production operations respectively, QFD does not provide a systematic approach, and it is done implicitly by engineering knowledge. Thus it's effectiveness downstream from stages 2 to 4 diminishes.
(ii) For complex products, it is impossible to derive piece part design parameters, while only using product specifications as input information, as it does not account for multiple product levels of system, subsystem and components or parts.
(iii) Basic Quality Function Deployment is only suited for the development of simple and conceptually static products. For products that are conceptually dynamic, which then require the consideration of different product concepts, there is no provision within basic QFD to handle them.

The above limitations led to the development of further enhancements to the basic QFD, and this is described in the proceeding section

2.2.3 Enhanced Quality Function Deployment

The need to overcome the limitations and drawbacks of basic QFD, led to further enhancements being introduced into basic QFD, as described by Clausing [20, 24]. These enhancements were introduced to enable basic QFD to handle (i) complex products as well as those which were conceptually dynamic, by using Pugh's concept selection method [27], (ii) the deployment through multiple levels (system, subsystem, and components) of products or systems. Other enhancements were associated with : the analysis of the context information surrounding the product and structuring of the requirements.

The deployment through multiple levels of the product or system, involves a break down of the design phase of basic QFD into Total System design, Subsystem design and Piece Part design. This was considered to enable the development of complex products that would otherwise have been difficult using basic QFD. Since both the process and production operations planning stages are influenced by the product's structure, both stages are also broken down into subphases as shown in Figure 2.3.



Key: TS - Total System, SS - Subsystem, PP - Piece Part

Figure 2.3 Enhanced Quality Function Deployment Process

The charts used during the above sub-phases are essentially the same as those of basic QFD. The only difference is the shift from the one-dimensional, straight forward deployment in the direction of the factory floor, to a two dimensional deployment forward to the factory floor and downward from the total system level to the piece part level. The procedure followed in enhanced QFD enables product development teams to commence with process planning before the design phase is completely finished. This leads to a considerable speed-up of product development and therefore contributes to the implementation of concurrent engineering [28].

The introduction of Pugh's concept selection into basic QFD, was aimed at extending the applicability of QFD to the development of conceptually dynamic products. The Pugh's concept selection method uses a visual chart as a selection tool. In this chart is recorded the available concepts as headings of the columns and the selection criteria as headings of the rows - Figure 2.4. These available concepts are usually represented in the form of sketches for ease of visualisation. The selection criteria are derived from the system specifications (process specifications) that are provided by the corrseponding QFD matrix (Figure 2.3). At the beginning of the selection process, an initial datum is usually chosen. This datum might be a dominant product or process design which already exists and relates to the selection area under consideration. Each of the proposed concepts is then compared with this datum with regards to its ability to satisfy the selection criteria. The relative abilities of these concepts with respect to the datum, are usually expressed by simple symbols like " + " meaning superior, "-" meaning inferior and "S" meaning same, which are inserted into cells of the chart. A score pattern in terms of " + ", "-", and "S" is obtained by adding up the number of symbols. The selection procedure is repeated several times while using the best concept of the preceding run as a new datum. The hope of using Pugh's technique is that at the end of the process, a superior design will emerge. Within enhanced Quality Function Deployment, concept selection supports both the forward and downward deployment [22].

The use of Pugh's concept selection technique, presupposes that, the concepts to be compared are already known. This is the case when there are existing known design concepts or solutions, such as those available in design texts, manufacturers catalogues, etc. Such existing design concepts are usually related to either small scale products, components or parts. In a situation where new concepts have to be created or generated, Pugh's technique would not be applicable. Other design concept generation techniques such as the Morphological analysis, brainstorming, synectics, etc, would have to be employed . The analysis of contextual information from a world wide economy is necessary when a new product development program is commenced upon. Two techniques that have been found especially useful are: parametric analysis and matrix analysis (reverse concept selection). Parametric analysis involves plotting two characteristics against one another, for groups of related products under consideration. The plot is then used to observe the performance of the products and then eventual decision on the most viable ones. The matrix analysis technique involves essentially the application of the Pugh concept selection method to existing products. This evaluation is done for existing products to select the competitive benchmark products that are used in the subsequent benchmarking activities in the 'House of Quality'. This reverse concept selection activity also provides insights into the strengths of existing designs, which is helpful for subsequent concept development and selection for new products. Both parametric and matrix analysis are advocated to be done early in the product development process. For static concepts, contextual analysis should be completed before the concept development and selection for new products begin. For dynamic concepts, the contextual analysis and new concept development and selection will be iterative. They will guide each other and converge to compatibility as they are mutually developed.

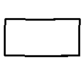
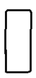
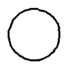
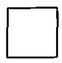
Concepts / Criteria						
A	+	-	+	-	+	DATUM
B	-	S	+	+	S	
C	+	-	+	+	-	
D	S	S	+	-	-	
E	-	+	-	S	S	
S-	2	2	1	2	2	
SS	1	2	-	1	2	
S+	2	1	4	2	1	
Ranking	2	3	1	2	3	2

Figure 2.4 Pugh's Concept Selection Process

In structuring requirements, emphasis was being laid on using a generic structure to identify and compile the needs. Such generic structure it is believed will help to improve completeness and speed in preparing requirements. This generic structure of requirements was based around Pugh's concept of 'Product design Specification' [29]. In analysing these requirements, it is then possible to distinguish between requirements that are imposed by the role or function of the product in carrying out a process, and those requirements which represent intrinsic characteristics or are associated with a company's business strategy, product strategy or the product development plan. As with contextual analysis, the specifications should be completed at the beginning of the development process for static products. For dynamic products, some iteration will usually be necessary to achieve convergence of the concept and the specifications [20].

The benefits attributed to Enhanced Quality Function Deployment in comparison with Basic Quality Function Deployment are : (i) Support of the development of complex products, (ii) Support of the development of conceptually dynamic products and (iii) Reduction in product development time. Inspite of the above advantages, Enhanced QFD was also found to have the following limitations [22]: (i) The systematic determination of design requirements for subsystems and piece parts, is not provided, (ii) The deployment decision as to which design requirements for a Total system are important for the design of a particular subsystem, and therefore to be deployed down to the corresponding QFD matrix is not supported systematically, (iii) The consideration of events that lead to a deviation from the ideal performance of a product is not provided.

2.2.4 Extended Enhanced Quality Function Deployment

To overcome the above limitations of Enhanced Quality Function Deployment, Functional Analysis was integrated into it. The aim of this was to be able to analyse design requirements in more detail. Design requirements, also referred to as corporate expectations are described as measurable terms that determine the ideal performance of a product. However, the ideal performance of a product can be seen as a function, that is performed under constraining conditions [30]. As a result, it can be considered that design requirements describe the conditions under which a product has to perform a function in engineering language, that is in measurable terms. Extended Enhanced Quality Function Deployment hence provides a systematic determination of design requirements for product elements based on the analysis of conditions under which the function of these elements are to be peformed.

In providing a detailed description of functions, within Extended Enhanced QFD, the Amplification Sub-Tree (AST) [22, 31, 32], was used. Function amplification refers to the extended description of the verb-noun statement of the function which is similar to the "Function in the Broad Sense", as introduced by Akiyama [30]. The AST is used to distinguish between characteristics of functions (amplifications) that describe the role of the product in the process of the function, and intrinsic characters, such as mass, volume, expected life, cost, etc. The description of the process distinguishes between the modification of natural processes and man-made processes. The description of man-made processes includes a "Change in the State of Universe" with regard to the geometric, energetic and information configuration of the objective of the function (Figure 2.5). Based on customer requirements to the product, the description of the usage environment of the product, governmental regulations for products, sales information, and manufacturing information, the product development team creates the AST for the function of the product, and then use appropriate measurable terms to guide the product development process.

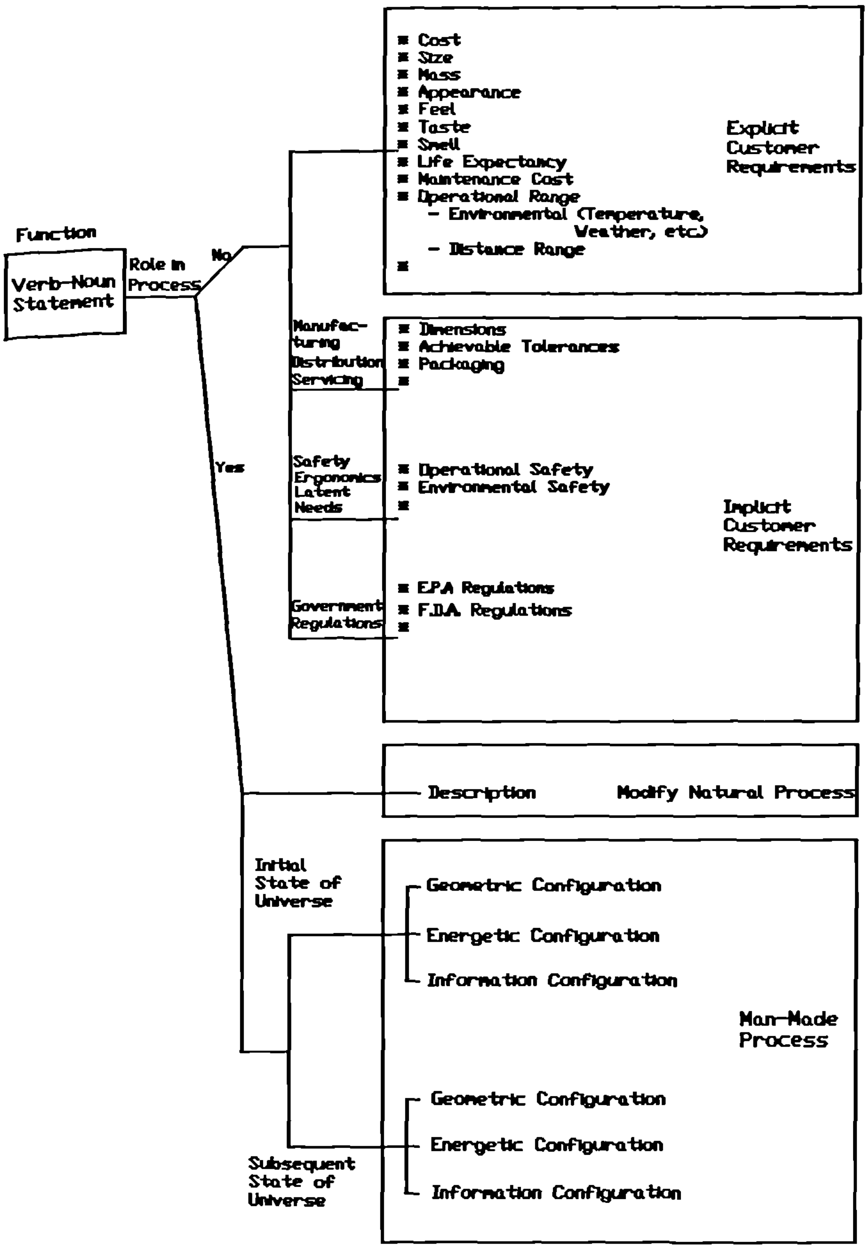


Figure 2.5 The Amplification Sub Tree

During the product planning stage, the activities of Extended Enhanced QFD, are similar to those of the Basic QFD. In Extended Enhanced QFD however, the Total system design requirements (corporate expectations) are determined by iterating between customer requirements for the product to be developed and the function amplifications listed in the above AST (Figure 2.5). This helps the product development team to ensure that all constraints that can guide the product development process are contained within the 'House of Quality'. Furthermore, the product development team is able to decide which customer requirements are to be entered into the Total system House of Quality and those to be deployed directly down to subsystems and piece parts [22].

The next stage, that is, the Design of the Total System in Extended Enhanced QFD, proceeds in a similar manner as the Product Planning stage, except for some few exceptions, which correspond to the procedure of "Deployment through-the levels" (Total System, Subsystems and Piece Parts). The procedure adopted is as follows :

(1) Subfunctions of the product's Top function are first determined without regard to any design concept. This is done in order to gain a clearer understanding of the structure of the concept that is to be selected during Total System Design.
(2) In the case of dynamic Total System concepts, a Total System concept for the product has to be selected. Here only concepts that have been proven as feasible and robust are evaluated.
(3) After selecting a concept, the functions of the subsystems of the Total System are compared to the top function and the subfunctions which were determined before concept selection. This function level check serves to identify the position of a function in the function tree. The relationship among the functions is also analysed.
(4) A function tree is then developed which represents the relationships among all functions revealed by the Extended Enhanced QFD. The function tree is a valuable communication tool during design which documents the interactions among functions. In creating the function tree, the Product Structure Element (PSE) Function Assignment Matrix is introduced and used to gain a clear overview of the assignment of functions to Product Structure Elements. The Product Structure Element (PSE) Function Assignment Matrix is developed simultaneously to Extended Enhanced QFD during Design and contains all functions of the PSE in the rows and related hardware components in the columns. Using the matrix, each function is then assigned to a PSE, as shown in Figure 2.6.

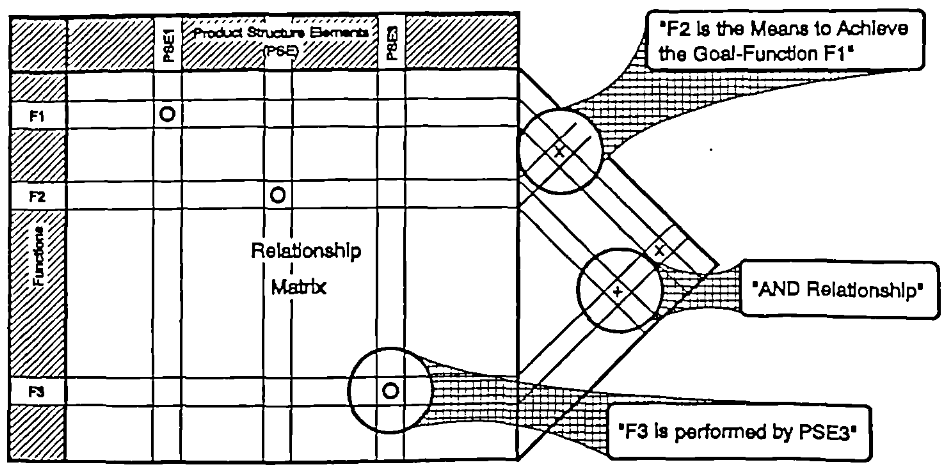


Figure 2.6 The PSE Function Assignment Matrix

(5) In the final step, the Amplification Sub-Tree of the subfunctions are created. These subfunctions are then assigned to related design requirements (corporate expectations) for the Total System, based on the fact that design requirements for a higher level system, which are related to amplifications of functions performed by a subsystem, provide information that is required for the design of the subsystem.

For the Subsystem and Piece Part Design, similar approaches to the steps above, are followed with some modifications. In the case of Subsystem design, interactions between subsystems are usually determined. The subsystem that has the highest interaction with other subsystems, then has the highest priority during design. The design of all the subsystems then proceed concurrently. In the case of Piece Part design, the process is split into two parts, that is, Part Formation and Part Finishing. This is done in order to focus the Extended Enhanced QFD activities for design on particularly critical Piece Part design decisions. For Piece Part Formation, Extended Enhanced QFD is applied to the most important Piece Parts while for Piece Part Finishing, Extended Enhanced QFD is only applied to the design of highly critical part features. The procedures followed in the application of Extended Enhanced QFD to Product Planning, and Product Design of Total System, Subsystem and Piece Part is shown in Figure 2.7.

2.2.5 Further Extensions to Quality Function Deployment

Some other enhancements and extensions have in the recent past, been added to Quality Function Deployment and they are discussed in the proceeding sections.

Requirements and Failure Causes Analysis (RFCA)

Sontow and Clausing [22] report on some work carried out at MIT, in seeking to iprove on Enhanced Quality Function Deployment (EQFD). The work resulted in the development of Requirements Failure Causes Analysis (RFCA) [9]. This involved the integration of Failure Mode and Effect Analysis (FMEA) into Extended Enhanced Quality Function Deployment. The goal of this work was to provide : (a) Systematic determination of the product's ideal performance guided by customer requirements, (b) Systematic determination and avoidance of events that cause deviation from the products ideal performance as early as possible during product development, (c) Systematic analysis of the relationship between causes and effects of undesired events and (d) Evaluation of design requirements (corporate expectations) and undesired events in order to focus on critical aspects during quality planning.

Requirements and Failure Causes Analysis (RFCA), in a similar way to Enhanced QFD, consists of four phases. Phase 1 represents the Product Planning phase. Here, the customer requirements are systematically determined while also including an evaluation of the requirements with respect to the customer's perspective. The customer requirements are then structured with regard to the system level, and then finally translated into design requirements (corporate expectations) for the product based on the analysis of the product functions.

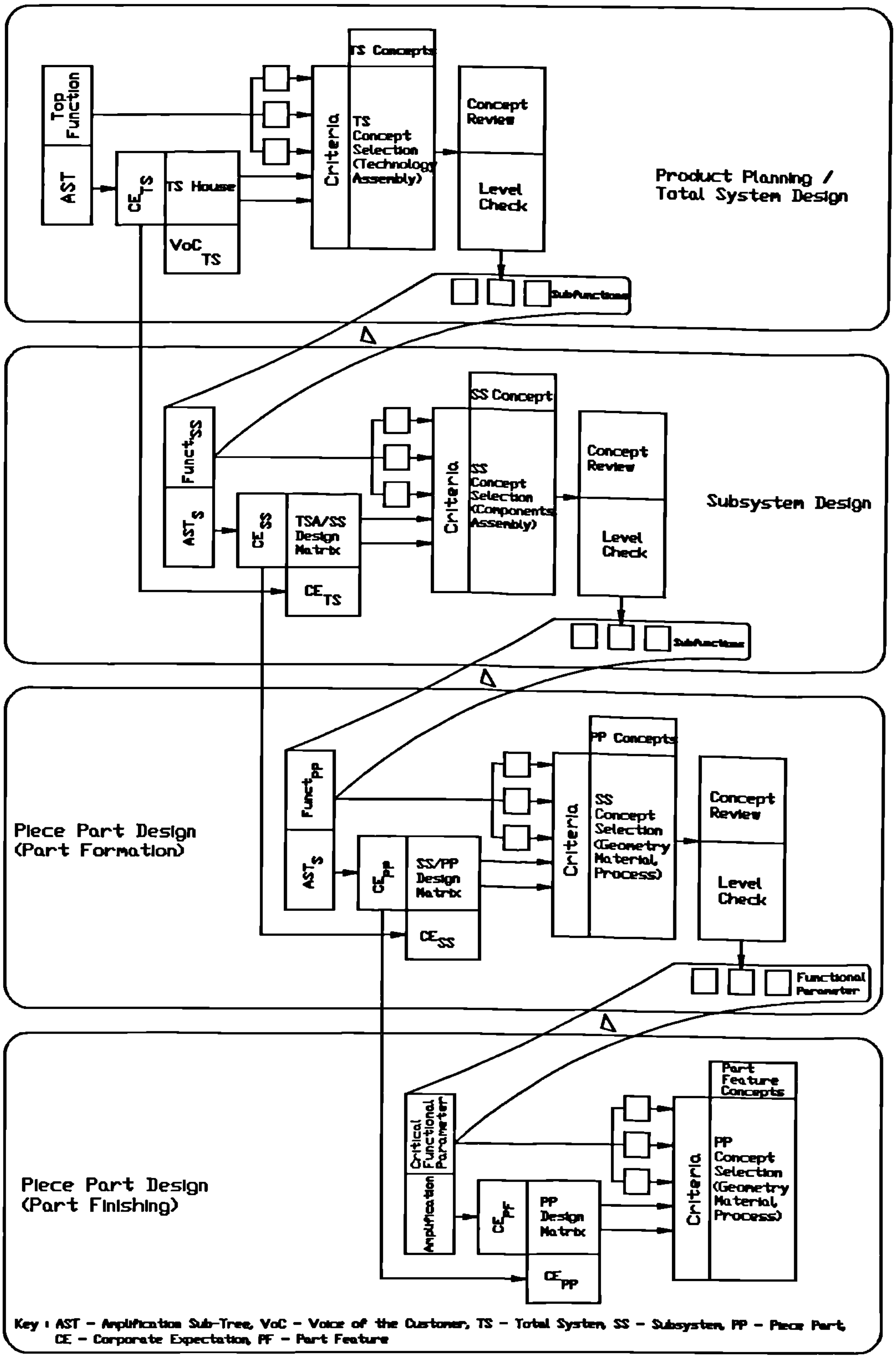


Figure 2.7 Extended Enhanced QFD - Deployment Through the Levels

Phase 2, that is, the Design phase is used to select design concepts for components of the products based on the component functions, design requirements for the components, and causes and effects of failures. The procedure involves first the determination of design requirements for subsystems and piece parts based on the component functions. Then the causes and effects of failure modes are determined, as well as the relationship among failures and design requirements. Based on this information, optimal design concepts for the components are then selected. In order to guide the planning of the design and process activities, the design requirements and causes of failure are also systematically evaluated. In phase 3, corporate expectations for assembly and production processes are first determined taking into account the design requirements from phase 2. Then 'noise factors' that cause failures or production variations are considered systematically. Again the planning activities during process planning are guided by the systematic evaluation of corporate expectations for the processes and 'noise factors' that affect them. Using the above information, optimal assembly and production processes are then selected. Phase 4, involving operations planning for critical assembly and production processes is conducted concurrently with process planning. Operations that are required in order to assemble and manufacture the product are derived from the output of Phase 3, assembly and production processes. In addition, quality assurance measurements are determined with regards to 'noise factors' that affect the performance of assembly and production processes.

Quality Function Deployment for Production

The enhancements here relate primarily to improvements proposed for Phases 3 and 4 of Basic Quality Function Deployment [33]. In Phase 3, based on the information obtained from the planning results, that is, critical part characteristics, target values for part characteristics, materials specifications and approximate production volume, the proposed procedure commences with the establishment of a process planning chart for each part that has critical characteristics. The establishment of this chart also includes the selection of machines, the definition of process steps and the identification of critical process steps. When the process planning chart per part is established, the planning procedure continues with the review of other similar processes, with the purpose of utilizing experience from past production processes. Using this acquired experience, the number of necessary experiments can be reduced. In examining past production processes, process capability indices for the critical process steps can be determined and target values of the process parameters that enable the achievement of a particular process capability can be identified. The long term goal of this second step, is to create a process database for future use. The last step of Phase 3 involves the establishment of a process planning chart for each critical process step. In the process planning chart per process step, the proposed QFD procedure uses FMEA to identify the parameters and noises that influence the capability of a critical process step. Finally, Taguchi methods and/or the process database can be utilized to set the target values for the process parameters.

In the case of Phase 4, the output results of Phase 3 constitute the starting point. Using the information from Phase 3, Phase 4 commences with the establishment of the Production Operations Planning Chart per machine. This chart contains the critical process steps performed at one machine. The first activity in this step is the determination of important production operations that are necessary to control a process parameter. The matrix in the chart is then used to define the appropriate tasks of a production operation or to describe the device (a control document or a control instrument) that has to be applied. The second step in this phase involves the establishment of the Process Control Sheet per machine. The Process Control Sheet per machine, also enables source inspection of the production process in order to minimize variation.This is to assure that process control sheets will be created at least for the production of the critical part characteristics. The activities in this step commences with the determination of the necessary information for a process control sheet. Such information can be obtained from a variety of sources such as the charts of Phases 2 and 3, and they include data for piece parts characteristics, piece part process planning as well as those from the process database. Finally the data for production operations are then entered into the Process Control Sheets.

Quality Function Deployment and Interpretive Structural Modelling

The application of Interpretive Structural Modelling (ISM) to Quality Function Deployment [20], was motivated by the following reasons :

(1) It was considered that there was considerable synergy between QFD and the structural modelling techniques. The basic formats for QFD and Unified Program Planning (UPP), a precursor to Interpretive Structural Modelling (ISM), were considered to be very similar [4]. The objective of QFD and structural modelling techniques is the same, that is, to improve the problem solving process by enhancing the understanding of the components of a problem and its often complex inter-relationships. Structural Modelling techniques, are attempts to improve the ability of designers to surround and fully understand complex problems.
(2) Structural Modelling techniques may help to provide internal consistency to the tree hierarchies created during the process of design. Engineers commonly develop functional trees for the product being designed. The establishment of requirements in QFD also involves hierarchical structuring in the form of trees.
(3) It may be possible to enhance QFD with the powerful visual representations that are the outcome of structural modelling. It is believed that structural models, resulting from the application of ISM could greatly enhance the representation of the 'roof' or 'attic' of the QFD chart (House of Quality) and enable design teams to use the information within to its full potential.

(4) There is promise that the matrix partitioning techniques embedded in ISM could increase the efficiency of constructing QFD matrices.

Interpretive Structural Modelling (ISM) was developed at Battelle Memorial Institute in the early 1970s by John Warfield et al, to aid in the solution of complex problems by providing a formal methodology for structuring the problems [34]. The development of ISM was an attempt to formalise the process and create a generic methodology for structuring complex issues. The mathematical bases for ISM are primarily graph theory, set theory, matrix theory and mathematical logic. The overall ISM methodology consists of the following phases : Preparation, Brainstorming, Voting, Model Construction, and Model Interpretation and Verification. In the Preparation phase, the group develops the relationship to be examined for the given problem. The Brainstorming phase is used to elicit issues that are perceived by the group to be relevant to the problem. In the Voting phase, these issues are narrowed down to the most important ones by anonymous vote. The resulting elements are then used to develop the structural model. Model construction is the heart of the process, and it is the phase in which elements are structured in the context of the relationship. Once the elements and the contextual relationship to be considered are identified, a pairwsie comparison is conducted among the elements with respect to the contextual relationship. The relationships are binary in nature and the pairwise relationships can be represented in an n x n binary matrix, where n is the number of elements. The other activities in this phase proceeds with ISM algorithms using the transitivity property to infer the relationships between the elements during a partitioning process. The partitioning process is iterated until all the non-zero cells in the matrix are filled. The resulting matrix is called the reachability matrix. In the next step the reachability matrix is rearranged into a canonical form so that a structural model is extracted from it using a series of partitions. In the final phase, that is the Model Interpretation and Verification phase, the initial structural model created is then interpreted and refined by the group until a satisfactory model representative of the problem is obtained. The work by Pandey [32] explored three potential application areas within QFD, that is, the correlation matrix, the relationship matrix and functional trees. Of the three, the most promising area of application of ISM in QFD, was the correlation matrix. It was observed that the use of structural modelling for the correlation matrix, could increase the utility of the information contained within and facilitate the use of the information generated during QFD for design. The information in the matrix was considered to have the potential to be used for purposes such as design task partitioning and as a guide for inter-organizational communication. Useful application of ISM to the relationship matrix and functional trees, still needs to be further explored.

2.2.6 Future Directions for QFD Research

Other attempts have also been made to expand the usefulness of Quality Function Deployment by integrating other techniques. A particular case is that reported by ter Haar et al [35], which has made some efforts to integrate Quality Function Deployment and the Design Structure Matrix, a design-process-modelling technique. Locascio and Thurston [36] in their paper report on the use of multi-attribute design optimisation within Quality Function Deployment. The aim was to attempt to improve the ill-structured approach to defining target values for the design requirements (engineering characteristics) using multi-attribute utility and optimisation theory. The methodology adopted involves the following steps : (a) Define design attributes from customer attributes, (b) Replace relative importance with multiattribute utility analysis, (c) Define design decision variables (variables the designer can directly control to effect changes in the customer defined attributes), (d) Define constraints functions (symbolic values in both the relationship matrix and the roof matrix), (e) Transform symbolic constraints to quantitative functional form, (f) Determine design variable and attribute bounds and (g) Structure the design model and solve optimisation problem using an appropriate algorithm. Another interesting development is that reported in the work of Masud and Dean [37] about the use of fuzzy sets in Quality Function Deployment. The philosophy of this work is based on the fact that the input variables in the QFD charts, can be treated as linguistic variables, as they are usually based on human judgement, perception and cognition and are in some cases ambiguous. The purpose of the research was then to investigate how the QFD analysis can be performed when the input variables are treated as linguistic variables, with their values expressed as fuzzy numbers. In this work, the results obtained for the rankings of the design requirements when the traditional QFD ranking process was employed showed some differences from those obtained when the fuzzy approach was used. Although the work showed that the use of the fuzzy QFD was a viable one, there is however the need for further work to be done in order to establish any general pattern. The links between QFD and Value Engineering still merits further work, as reflected in the paper by Snodgrass [49].

2.2.7 Applications of Quality Function Deployment

Dean [38] in his paper, describes efforts made to extend QFD to large scale systems. This work made attempts to link QFD to the system engineering process, the concurrent engineering process, the robust design process, and the costing process. The aim was to generate a tightly linked project management process of high dimensionality which flushes out issues early to provide a high quality, low cost, and hence competitive product. In an application to a proposed Lunar Rover, a need arose to track demands from specific customers, the result being an up front matrix to the basic QFD matrix, of type of customers versus customer demands. These customers had conflicting desires, and they were then valued with respect to the need for the project to satisfy their desires. This quantified the customer political power. Since the customer desires also conflicted with the goals and needs of the project, there was a need to value each customer demand with respect to the goals of the project. The final value for a customer demand then became the value to the project of the customer demand times the dot product of the value of the customer and the correlation between the customer and the customer demand.

Maddux et al [39], have also reported on the use of QFD as a strategic planning tool. This was based on their experience at the Production Engineering Division (PED) of the U. S. Army Missile Command (MICOM), where QFD was employed in the formulation of a strategy to successfully implement and manage a program called Production Engineering (PE) Tools. The intent of this project, was to either locate or develop software or other tools related to the production function, evaluate its effectiveness and utility to other groups, then promote the transfer of this technology throughout the Department of Defense (DoD). In this project, the QFD exercise began with the formation of a QFD team, after which the requirements from three groups of customers were elicited. The team then generated quality characteristics (design requirements - HOWS) that were necessary to successfully meet the customers' needs. From this exercise they were able to identify the critical customers' requirements as well as the design requirements, on which adequate focus was required. The team was also able to develop a more coherent strategy for implementing the PE Tools program as well as steps for its improvement.

Maduri [40], also report on the use of QFD in the development of an Off-Highway Dump Truck. The work demonstrated how QFD was used in establishing the specification parameters as well as the identification of the interactions between the design parameters at the specification stage and generation of some of the specification indices. In using the QFD technique, efforts were made to proceed through the four deployment stages, albeit in a very limited way. The research work by Hochman and O' Connell [41] demonstrate how customer environmental concerns can be integrated into the conventional QFD process. The example used in this case, involved the design of a portable telephone. The emphasis in this paper was the need to recognise that designing for the environment, is part of the larger system, and that marrying the QFD process with environmental design, can be a significant step towards achieving competitive environmental advantage in a product. In reporting on the application of QFD to the design of a Lithium battery, Haldeleib et al [42] discuss the experience and lessons learned in proceeding through the four stages of QFD. The paper showed how customer requirements were elicited, their translation into product measures (design requirements) and how finally the importance ratings of the product measures were determined. It then also illustrated the deployment process through the four stages, using the critical parameters determined at each stage. The benefits gained in going through the four stage process, as well as future plans for further implementation were also discussed. The ASI QFD awareness document [43], also reports some applications of QFD. The first is the one by Davison and Jones [44] discusses application of QFD and Taguchi methods in seeking to improve the quality of their printed circuit boards. This exercise helped them to get closer to their customers as well as being able to reduce their scrap costs. In another article, Nichols [45] reports on the benefits Digital Equipment had derived in applying QFD to several products within the company. Such benefits included : better systematic processes, more accurate source of requirements, optimised functionality, team integration, focussing on target values and breaking down traditional attitudes. De Vera et al [46] have also employed QFD successfully in the design of a blend door actuator, while Sullivan [47] has also taken some steps towards employing QFD for policy management. An interesting application of QFD is that of the Yaesu Book Centre, the largest retail book store in Japan. Their aim was to ensure that they provided books that customers wanted, quickly and pleasantly [48].

In another project reported by Hauser [25], it was shown how Quality Function Deployment was employed to fight back competition in the development of a new generation of Spirometry systems. This system is an important diagnostic tool used in hospitals and by general practitioners. In this project, the customer requirements were elicited through a combination of focus groups and telephone interviews. Customers included pulmonologists, allergists, nurses, distributors, and sales representatives. Throughout the development process, the QFD 'House of Quality' provided an organisational history and a framework for making decisions and also enhanced and focussed the design process on the customer. Through the investment of time early in the process, the company was able to avoid costly redesign and rework.

Some other applications of QFD by various companies and institutions also reported by Hauser [25] include : (i) A manufacturer of consumer stationery products, who used QFD to identify the important customer needs, which helped the company to identify key design attributes, (ii) A manufacturer of construction tools, who used the 'House of Quality' to identify key customer needs that were important to a target segment and which distinguished new products, (iii) A financial institution who used QFD to identify eight important customer needs that were not being addressed effectively by current communications.

2.2.8 Summary

The essence of this review was to examine in detail what Quality Function Deployment meant, its concepts, the current research status, its applicability and future directions for research. In this regard, discussions in this section have focussed on the historical development of QFD, the enhancements made to it, as well as case studies and application areas. The QFD technique can be seen to be quite effective for handling customer requirements and developing specifications. It is also useful for developing existing products, but not as useful for developing new and innovative designs. It's effectiveness in the later stages, that is, manufacturing process and production planning stages, from research carried out seems to reduce in comparison with the early stage of product planning [33], and hence requires closer attention. The key aspects that would be useful in the development of Design Function Deployment, have also been identified. Details of these will be discussed in chapter 3. QFD although has been echoed as a concurrent engineering technique, does not on it's own support the various life cycle and dowmstream issues that need to be considered at the design stage. There is still a wide scope for further developments in the QFD technique, some of which have been identified in section 2.2.6

2.3 A SURVEY OF DESIGN PHILOSOPHIES, MODELS AND METHODS

2.3.1 Introduction

The design activity, although has been performed since prehistoric times, did not however have any structure or organisation to it. It was only just after the middle of this century, that efforts began in attempting to give some formalism to the way design was done. What is design ?, Why is it done ? and How is it or can it be done ?. These questions have been the subject of discussions at various conferences on engineering design and design methodology. In these conferences which were held in the UK [51-55], Europe [56-60] and the USA [61-68], a number of ideas were put forward on design methodology. These ideas were mostly associated with design models, philosophies and methods or techniques as well as applications, and they represented several schools of thought on design and design methodologies. More recently, other researchers have started to also report on computer based design systems.

The main focus of this section, is to give a detailed elucidation of design philosophies, models, methods and systems, which have been proposed and developed over the years. Discussions will centre on : definitions of design and design methodologies, the nature and features of design problems and the design process, as well as the stages of thought in design and product classification. The nature and control of design goals will also be discussed, including an extensive review of many design models, methods and systems.

2.3.2 Definitions of Design

Several designers, engineers and researchers from observation and experience, have expressed their views on the definition of design or what they consider design to be. Some of such viewpoints are expressed below.

Feilden [69] "Engineering Design is the use of scientific principles, technical information and imagination in the definition of a mechanical structure, machine or system to perform prespecified functions with the maximum economy and efficiency".

Feilden [70] "design ...the highest manifestation of the science and art of mechanical engineering ...The synthesis of many elements including knowledge of production methods and of methods of applying the results of research, market requirements and so on".

Wallace [71] "Engineering Design is the process of converting an idea or market need into the detailed information from which a product or system can be produced"

Hillier et al [72] "...design is essentially a matter of prestructuring problems either by a knowledge of solution types or by a knowledge of the latencies of the instrumental set in relation to solution types". Hence design is seen as a process of 'variety reduction' - Darke [73].

Finkelstein [74] " Design is the creative process which starts from a requirement and defines a contrivance or system and the methods of its realisation or implementation, so as to satisfy the requirement. It is a primary human activity and is central to engineering and the applied arts ".

Luckman [75] "Design is a man's first step towards the mastering of his environment ... The process of design is the translation of information in the form of requirements, constraints, and experience into potential solutions which are considered by the designer to meet required performance characteristics ... some creativity or originality must enter into the process for it to be called design".

Archer [76] " ...design involves a prescription or model, the intention of embodiment as hardware, and the presence of a creative step ".

ElMaraghy et al [77] " Design is the transformation or mapping process from the functional domain to the physical domain which satisfies the stated functional requirements within identified constraints ".

Gasson [78] " Design is ... a highly innovative cross-disciplinary process through which man seeks not simply to satisfy himself, but also seeks to satisfy the needs of others".

Rabins et al [79] " Engineering Design is the creative process that is the essential source of all new products. It involves imagining many different ways to satisfy a need. Sometimes multiple, even conflicting, requirements and constraints must be reconciled"

Lord Caldecote [80] " ...the basic design function... to design a product which will meet the specification, to design it so that it will last and be both reliable and easy to maintain, to design it so that it can be economically manufactured and will be pleasing to the eye".

La Rota et al [81] "Design...a process where a set of intentions and constraints are converted into a detailed set of instructions that physically describe and specify methods for the manufacture of material parts, products, and systems. From an Artificial Intelligence perspective, the design task is characterised as an under-constrained and ill-structured problem solving process which involves search for solutions in a large space of alternatives".

Chandrasekaran [82] "The design problem is specified by a set of functions to be delivered by an artifact, a set of constraints to be satisfied by the artifact during its functioning, and a repertoire of components assumed to be available and a vocabulary of relations between components. The solution to the design problem consists of a complete specification of the set of components and their relations which together describe an instance of the artifact which satisfies the requirements of its functions and constraints".

Suh [83] "Design may be formally defined as the creation of synthesized solutions in the form of products, processes or systems that satisfy perceived needs through the mapping between the Functional Requirements (FRs) in the functional domain and the Design Parameters (DPs) of the physical domain, through the proper selection of DPs that satisfy FRs".

The foregoing definitions of design reflect the various viewpoints of the proponents. In general, certain keywords and phrases can be noted which have strong bearing on design. These include : *needs, requirements, solutions, specifications, creativity, constraints, scientific principles, technical information, functions, mapping, transformation, manufacture, and economic.* The word customer, although absent seemed to be implicitly represented by the words need, requirements or market. Taking account of these keywords, one can then define design as :

"the process of establishing requirements based on human needs, transforming them into performance specifications and functions, which are then mapped and converted (subject to constraints) into design solutions (using creativity, scientific principles and technical knowledge), that can be economically manufactured and produced".

2.3.3 The Nature and Features of the Design Process

The design process for any design model, usually exhibits certain properties and features which represent various associated viewpoints and philosophies, activities and processes that occur during the process. These features as highlighted by several researchers [84-86]are discussed below :

(a) *Design as an Opportunistic Activity*
Design is a process where both, the top-down and bottom-up approaches, are used by the designer in an opportunistic manner. The designer, depending on the domain can adopt any approach or a combination of the approaches, that best suits his/her conditions.

(b) *Design as an Incremental Activity*
Engineering design is also seen as an evolutionary process, and changes are proposed to the current design in order to move to a 'better' design. These changes can be seen as improvements or refinements, and they are usually a percentage of the total amount of information held in the current design.

(c) *Design as an Exploratory Activity*
Smithers et al [87] propose an exploration-based model of design, describing the design process as a knowledge-based exploration task. Design in this case is hence

classified as exploration, rather than search, because knowledge about the space of possible solutions has to be obtained before goals can be well formulated.

(d) *Design as an Investigative Process (Research)*
The first step of the design process is normally an enquiry into the clients needs and expectations, available design techniques, previous similar design solutions, past failures and successes, etc. This stage requires knowledge of research methods and skills in applying these methods.

(e) *Design as a Creative Process (Art)*
In some respects, a tentative solution to a design problem can be invented or copied. However, there is generally no explicit logical procedure which leads from the problem to the solution. In the creative process of design, a solution is created with the help of know-how, ingenuity, good memory, pattern recognition abilities, random search in the solution space, lateral thinking, brainstorming, analogies, etc. This stage of the design is heavily dependent on such concepts as "gift of the designer", "feeling for the subject", etc.

(f) *Design as a Rational Process (Logic-Based)*
The checking and testing of proposed solutions normally involves logical reasoning, mathematical analysis, computer simulation, laboratory experiments and field trials, etc. In this regard, certain activities in design constitute a rational process involving the use of well known scientific and analytic methods, which in many cases are also automated. Such automation activites cover what has come to be known as computer aided design and computer aided engineering.

(g) *Design as a Decision Making Process (Value-Based)*
In the design process, designers usually make a lot of value judgements particularly when there is a need to adopt an alternative course of action or to choose between competing design solutions. Such judgements and evaluation, are usually based on experience and criteria derived from the customer or clients' requirements. The selection of a particular configuration of a layout, or a shape of a product generally involves considerable uncertainties, which might be in the form of conflicts or trade-off scenarios. These uncertainties are resolved by estimating the values which the customers or clients place on the alternative solutions.

(h) *Design as an Iterative Process*
The iterative activity is the most common process in design. After the proposal of a preliminary design, it is then analysed with respect to the constraints and if the design does not adequately satisfy the constraints, it is revised, based on experience and the results of the analysis. This analysis and re-design cycle continues until a satisfactory solution is found.

(i) *Design as an Interactive Process*
Interactive design brings the designer directly into the process by forcing him/her to be an integral part of it. This is necessitated in situations where : (i) the design problem is ill-defined and the designer is still in the process of formulating the problem, (ii) there are insufficient analytical tools developed to enable quantitative analysis and (iii) there is little or no experience available or associated with the design problem. Interactive design utilises time-sharing, that is, the sharing of a computer's time between a number of users, and lends itself to heuristic techniques of design by allowing the designer to quickly manipulate his/her model and learn from previous situations.

The above views on the nature and features of the design process, represent different facets of the overall design process. They are dependent on the engineering or design domain from which the particular viewpoint is expressed as well as the nature, type, variety and complexity of the particular artefact/process or system being designed. Most of the viewpoints are however complementary to each other. A comprehensive design system must therefore be able to support these various facets of design involving :
(i) a top-down and bottom-up approach, (ii) the evolutionary process of design, (iii) the knowledge-based/exploratory aspects of design, (iv) the investigative and search aspects of the design process, (v) the creative process in design, (vi) the logical reasoning process involved in design, (vii) the iterative as well as the interactive process involved in design, (viii) the making of decisions based on value judgements and (ix) the mathematical analysis and computational simulation processes performed during design.

2.3.4 Definitions and Viewpoints on Design Theory and Methodology

The subjects of design theory and design methodology, although amply discussed by researchers, have not been fully explicated. Some definitions have however been given to them by various designers and researchers, and are reported below.

The American Society of Mechanical Engineers (ASME) [88] defines the field of design theory and methodology as " ... an engineering discipline concerned with process understanding and organised procedures for creating, restructuring and optimising artifacts and systems". **Design theory** is taken as a collection of principles that are useful for explaining the design process and provide a foundation for the basic understanding required to propose useful methodologies. Design theory is about design; it explains what design is, or what one is doing when designing. On the other hand, design methodology is a collection of procedures, tools and techniques for designers to use when designing. **Design methodology** is prescriptive as it tells one how to do design, while design theory is descriptive as it tells one what design is.

Rabins et al [79] " ...design theory refers to systematic statements of principles and experientially verified relationships that explain the design process and provide the fundamental understanding necessary to create a useful methodology for design ".

Cross [89] " Design Methodology .. is the study of the principles, practices and procedures of design in a rather broad and general sense. Its central concern is with how designing *is* and *might be* conducted. This concern therefore includes the study of how designers work and think; the establishment of appropriate structures for the design process; the development and application of new design methods, techniques, and procedures; and reflection on the nature and extent of design knowledge and its application to design problems ".

Finkelstein [74] " Design Methodology does not offer a prescription for all or for any particular design problem. It provides a useful framework for the structuring of the design process, the generation of design concepts and for evaluation and decision in design. It ... meets many needs of : the teaching of design, the organisation of design, the provision of aids to the work of designers and the automation of design ". Eder [90], also comments that formal methodologies act as a framework for guiding the designer's thoughts, and that they are no substitute for creative thinking, but can help to ignite the intuitive processes. They also appear to help the designer by allowing the generation of more solutions to simple sub-problems and elimination of unsuitable combinations.

These viewpoints represent the first steps towards defining what might be regarded as design theory and design methodology. The definitions by ASME and Cross are particularly encompassing and are worth noting.

2.3.5 The Nature and Stages of Thought in Design

In the process of design, most designers tend to go through certain stages, referred to here as stages of thought, as they move from an abstract problem to a realizable product. These are the divergent, transformation and convergent stages, and are analogous to the three stages of analysis, synthesis and evaluation of the design model proposed by Jones [51].

(a) **Divergence**

This is the act of extending the boundary of a design situation in order to have a large enough solution search space. The design objectives are unstable and tentative, and the problem boundary is ill-defined. Evaluation of generated ideas is deferred to a later stage. Provision is made for some flexibility in the sponsor's or customers' brief, which might be revised or evolve during divergent search. Preconceived ideas are also got rid of, and a lot of relevant information as possible is gathered. Rational and intuitive actions are involved in this stage. The divergent search approach aims to de-structure the original design brief, while identifying the features of the design situation that will permit a valuable and feasible degree of change. Searching in a divergent manner, also provides as cheaply and as quickly as possible, sufficient new experience to counteract any false assumptions that the design team held at the beginning. Divergent search is most productive in the initial stages of design and usually provides useful ideas for the transformation stage.

(b) **Transformation**
This is the stage of pattern making, high level creativity, flashes of insight, changes of set and inspired guesswork. Here judgements of values and technicalities are combined when making decisions. The objective here is to impose upon the results of the divergent search, pattern that is precise enough to permit convergence to a single design. This is the stage where objectives, design brief and problem boundaries are fixed, critical variables are identified, constraints recognized, problems are split up into sub-problems and sub-goals are modified if necessary.

(c) **Convergence**
The main objective of the convergent stage is to progressively reduce secondary uncertainties as fast as possible as well as ruling out alternatives. The models used to represent the range of alternatives remaining, should become less abstract and more detailed. The end result of this stage, should be the reduction of the range of options to a single chosen design as quickly and as cheaply as can be managed and without the need for unforseen retreats or recursion.

2.3.6 The Variety of Design Problems

A design is strongly influenced by the lifestyle, training and experience of the designer, and the creativity and effort a designer puts into a design varies, depending on the type of design problem - Juster [91]. Design problems which confronts engineers and designers can be classified under the following types [91-94]:

(a) Routine Designs

These are considered to be derived from common prototypes with the same set of variables or features and the structure does not change. Here a design plan exists, with sub-problem decomposition, alternatives and prototypical solutions known in advance.

(b) Redesigns

This involves modifying an existing design to satisfy new requirements, or improve its performance under current requirements. The end result of redesigns may also exhibit some form of creative, innovative or routine design content. Redesigns will be discussed under adaptive designs and variant designs.

Adaptive, Configurative or Transitional designs

This form of designs involves adapting a known system (solution principle remaining the same) to a changed task. They also involve improvements on a basic design by a series of 'detail' refinements.

Variant, Extensional or Parametric designs

This follows an extrapolative or interpolative procedure. The design technique involves using a proven existing design as a basis for generating further geometrically similar designs of differing capacities.

(c) Non-Routine Designs, Original or New designs

These form of designs are also known as original designs and are classified into Innovative and Creative Designs

Innovative Designs

Here new variables or features are introduced, which still bears some resemblance to existing variables or features, and the decomposition of the problem is known, but the subproblems and various alternatives to their solution must be synthesised. In other situations, alternative recombination of the subproblems may yield new designs. It is also considered that solving the same problem in different ways, or different problems in the same way (by analogy), would fall under this class.

Creative Designs

In this case new variables or features are introduced which bear no similarity to variables or features in the previous prototype and the resulting design has very little resemblance to existing designs. For creative designs no design plan is known, a priori, for the problem under consideration.

Sriram et al[93] in the light of the foregoing describes the creative-routine spectrum of design as follows : "*At the creative end of the spectrum, the design process might be nebulous (hazy), spontaneous, chaotic, and imaginative, whereas at the routine end, the design is precise, predetermined, systematic, and mathematical*".

2.3.7 Product Design Classification

The end result of any design process is a product or system. Such products depending on the engineering discipline or domain, vary in one way or the other, for example, buildings and bridges are products of civil/structural design; machine tools, gas turbines and robots are products of mechanical design; electric circuits and microprocessors are products of electrical/electronic design; process plants are products of chemical engineering design; aircrafts and spacecrafts are products of aeronautical designs, and so on. Product variation also arises depending on the market segment, knowledge available, the design process and manufacturing capabilities. In the light of general constraints, products can be classified as either overconstrained or underconstrained, and depending on the customer demands and competition in the market, some products are considered as static or dynamic. These various forms or classifications are discussed below [24, 95].

(1) Static Product Designs

Static products are those whose market share is undiminishing and no changes are being demanded in the product. The design concept is already known from existing products, and hence such products are considered to be conceptually static (also referred to as dominant design). Further development concentrates on the manufacturing process, automation, etc.

(2) Dynamic Product Designs

Dynamic products have a limited life before the next generation supercedes it. Here development is focussed on the product, and the design process involves the development of new, radical and alternative designs. The resulting products are considered to be conceptually dynamic.

In discussing the dynamic-static spectrum of products, Clausing [24] highlights the following types of products sandwiched between the two extremes :

(a) Genesis Product

A genesis product is an original product, and has no previously related product. It usually can lead to the evolution of diverse families of products.

(b) Radical Product

A radical product may at the higher (concept) level of abstraction exhibit similarities to previous designs, but is radically different in the final designs of the subsystems.

(c) New Product

A new product is less than a radical product, and involves the use of a new or an alternative technology in the development of an already existing product.

(d) Clean Sheet (Generational)Product

A clean sheet generational product is usually characterised by a major step forward, and the design process commences from a clean sheet of drawing paper, with little reference or relationship to prior relatives. The genesis, radical, and new products are by definition clean sheet designs.

(e) Market-Segment Entry (New) Product

A market-segment entry (new) product moves a company for the first time into a new market segment, following an earlier new (or radical) product.

(f) Market-Segment Entry (Generational) Product

A market-segment entry (generational) product is a new product in a market segment following after a previous product in the same market segment.

(g) Associated Product

An associated product changes some technologies to provide a different capability, resulting in an entry into a somewhat different market segment. An example is the case of laser printers, which are based on xerographic copiers.

(h) Variant Product

Variant products are relatively small changes on a base product. They include feature enrichment or removal, and cost/performance upgrades.

(i) Customised Product

Customising of products extends over a range from adding brand names to fairly major changes for specific customers. Customised products tend to have a limited role in the general subject of product development.

(3) Overconstrained Product Designs

These products tend to exist in the high technology markets. Here, the design process evolves around analysing alternative proposals until the correct (or most acceptable) solution is found. Overconstrained products are usually subjected to several constraints of function, materials, manufacturing processes, some of which might be conflicting, and the product undergoes several analysis and trade-off situations. In this case, major changes in product concepts rarely occur.

(4) Underconstrained Product Designs (Ideas Centred)

In the case of underconstrained designs, the design activity is centred around bringing products into the market to satisfy market demands. There are usually not very many constraints, and the designer has ample room for innovation. The focus here is usually on the product concept, and materials and techniques are chosen to satisfy the required function and recognisable market style. Most industrial designs fall into this category, and development is on aesthetics, ergonomics and functionality.

(5) Underconstrained Product Designs (Skill Based)

This form of designs focus on the manufacturing aspects of product development. Efforts are usually concentrated on the capabilitites and skills available in the company.

2.3.8 Design Goals

Design goals can be defined as the purpose for design actions and decisions taken in each design step. They guide the choice of what to do at each point during the design process. They are not artifact descriptions, but they prescribe how those descriptions should be manipulated [96]. Design goals represent one or more decision points from a problem solving point of view, and they define some of the dimensions of the design space. Each design goal explicitly specifies the design parameters it is responsible for, and also specify the design parameters on which they depend [97]. Design goals do exhibit one form of interaction or the other in the form of (i) Goal conflicts - involving non simultaneous achievement of two goals, (ii) Goal sharing - achieving a subgoal helps achieve a goal other than its ancestors in the goal tree, and (iii) Goal prerequisites - one goal must be achieved before another goal in a different part of the goal tree.

Types of Design Goals

Typical types of design goals include :(1) **Functionality Goals** - This kind of goal usually represents an unimplemented specification.(2) **Performance Goals** - These goals seek to satisfy requirements on effieciency, cost, reliability, etc, with which the designed product or system satisfies its functional specifications. These goals are often used as criteria for selecting among alternative implementations of a functional specification.(3) **Knowledge Goals** - These form of design goals seek to gather information required to carry out the design, and need to be made explicit. (4) **Design Process Goals** - These goals govern the route taken in arriving at a design, and involves issues related to prudent use of time and cost control in the design process.

Strategies for Controlling Interacting Design Goals [96]

In the control and management of the design process, there is need to explicate strategies for how to handle interacting design goals. For any two design goals, possible relationships that can occur between them are:

(a) **Independence** - These goals do not affect each other, and they can be achieved in any order with the same net result. Hence no special control strategy is needed to order them.(b) **Cooperation** - Achieving one goal makes it easier to achieve the other. These goals should be achieved in whichever order best exploits the relationship among them. Proposed strategies include : (i) achieving prerequisite goals first, (ii) achieving more general goal first and (iii) learn by solving easier goal first - This strategy can be effective if the designer is capable of learning from experience. (c) **Competition** - One goal can be achieved only at the expense of the other. These goals must be integrated according to their relative importance. Proposed strategies include : (i) sacrificing less important goals, when one goal completely dominates the other, (ii) relaxing one of the goals, if both goals are important and (iii) treat as trade-off, if goals are relative preferences and choose a compromis solution to satisfy some overall utility function. (d) **Interference (Interacting)** - One goal must be achieved in a way that takes the other goal into account. These goals can be achieved in a variety of ways, depending on the nature of the interaction. In this case, proposed strategies include : (i) achieving goals sequentially, (ii) deferring commitments - start with decisions imposing fewest constraints, (iii) making critical decisions first - start with most constrained decisions, (iv) merging goals and implementing them as single specifications, (v) using goals as selection criteria, (vi) combining ordering of goals and (vii) using goals to budget. In practice, the above goal control strategies, are used simultaneously and in combination.

2.3.9 Philosophies of Design

There have been various schools of thought expressed by designers and researchers as regards how design is, might be or should be done. This undoubtedly has resulted in a lot of controversies, which is still on-going. Three schools of thought within the British design community were expressed by Broadbent in the book Design: Science: Method [98]. The first group believed that the design process should be chaotic and creative, the second group believed that design should be organized and disciplined, while the third group argued that no design process should be imposed on a designer [99].

Support for the first view point is usually based on the argument that the design function is an art, and hence cannot be taught, which seems to imply that designers are born and not made. Archer [76], in support of the second viewpoint comments that "Systematic methods come into their own, under one or more of three conditions: when the consequences of being wrong are grave; when the probability of being wrong is high (e.g. due to lack of prior experience); and/or when the number of interacting variables is so great that the break-even point of man-hour cost versus machine-hour cost is passed ".

Cross[100] reporting on Lawson's work [101] which compared the ways in which designers and scientists solved the same problem, states that "The scientists tended to use a strategy of systematically exploring the problem, in order to look for underlying rules which would enable them to generate the correct or optimum solution. In contrast, the designers tended to suggest a variety of possible solutions until they found one that was good or satisfactory. The evidence from the experiments suggested that scientists *problem - solve by analysis*, whereas designers *problem - solve by synthesis*. Scientists use 'problem - focussed' strategies and designers use 'solution - focussed' strategies".

Yoshikawa [102] in his paper on Design Philosophy, also discusses design from some philosophical viewpoints attributed to various designers who belong to the Semantics, Syntactics and Past Experience (Historicism) schools of thought. These viewpoints constitute the platform for most of the controversies in the design community. Some of them however complement each other, while others are completely contradictory.

Semantics School

This school of thought is attributed to Rodenacker [103]. The central dogma of this school is that any machine, as an object of design, is something that transforms three forms of inputs, viz : substance, energy and information into three outputs respective to each input, but having different states from the inputs. The differences between the inputs and outputs are called functionality. The initial requirements are usually given in terms of the functionality, which has to be analyzed into a logical structure, which gives connections between sub-functionalities. On decomposing the initial functionality into finer sub-functionalities, these resulting sub-functionalities are substituted with particular physical phenomena that realize the transformations respectively.

Syntax School

This school is associated with the effort made to give some formalism to the design process, and attention is paid to the procedural aspects of the design activity rather than on the design object itself. Here attempts are made to abstract the dynamical or temporary aspects from the design neglecting the static aspects of design as emphasized in the semantics school. The process of abstraction is considered as the premise for improving the universality of design models belonging to this school, which are usually regarded as prescriptive models. This philosophy which emphasizes the dynamical (morphological) aspects of design can be combined with the semantics one which emphasizes the static (anatomical) aspects of design to achieve a more sophisticated design methodology.

Past Experience (Historicism) School

Arguments put forward by those belonging to this school of thought are usually that universality which is the target of most design methodologists, is contradictory to practical usefulness, and that the creativity of designers can be hampered and may deteriorate if design methodologies are adopted. In this school, emphasis is placed on the significance of case histories of design, including all necessary knowledge to be learnt for improving design ability. This school of thought is closely associated with the view that the design ability cannot be acquired efficiently in a theoretical manner, but by experience.

The above schools of thought although stand their grounds in their arguments are however relevant in one way or the other with respect to design. In today's world, it is increasingly becoming evident that design approaches belonging to the syntax (prescriptive models) school of thought, are more likely to stand the test of time. Wallace [104] in his article points out that, "the engineering design process cannot be carried out efficiently if it is left entirely to chance...", and "... the aim of a systematic approach is to make the design process more visible and comprehensible so that all those providing inputs to the process can appreciate where their contributions fit in." Furthermore, the need to equip and train young engineers as well as support collaborative design teams will necessitate the adoption of a structured and systematic approach to design.

2.3.10 Design Models

Design models are the representations of philosophies or strategies proposed to show how design is and may be done. Often, they are drawn as flow diagrams, showing the iterative nature of the design process by a feedback link.

In the past, design models which arose from various philosophical viewpoints, have tended to belong to two main classes, namely prescriptive and descriptive models. The prescriptive models are associated with the syntactics school of thought, and tend to look at the design process from a global perspective, covering the procedural steps (i.e. suggesting the best way to do). The descriptive models on the other hand are concerned with designers actions and activities during the design process (i.e. what is involved in designing and/or how it is done). More recently, another group of models known here as computational models have started to emanate. These computational models, place emphasis on the use of numerical and qualitative computational techniques, artificial intelligence techniques, combined with modern computing technologies. Some of these models although discussed under one of the above classes, however share some characteristics of the other classes.

2.3.11 Prescriptive Models on the Design Process

These models in general tend to prescribe how the design process ought to proceed and in some cases appearing to suggest how best to carry out the design process. They also attempt to encourage designers to adopt improved ways of working. They usually offer a more algorithmic and systematic procedure to follow, and are often regarded as providing a particular methodology.

A good number of these models emphasize the need to perform more analytical work, prior to the generation of solution concepts [100]. Models put forward by proponents of prescriptive models are discussed below.

<u>Model by J.C. Jones</u>

This model by Jones [51] is principally made up of three stages viz: Analysis, Synthesis and Evaluation.

At the **Analysis stage**, the first activity, involves producing a random list of factors which are related to the problem to be solved and/or to its solution. Such factors could include the basic functions to be fulfilled, the constraints of the production organization, the customer, the user, the environment and others like maintenance, storage life, installation etc. These factors are then classified into workable categories and subcategories as the case may be, for easy handling. With the knowledge that factors affecting design do interact, matrix charts are then systematically used to investigate all possible interactions as well as elucidating the pattern of relationships between the factors, with the aid of diagrams. The next step involves rewriting all the design requirements into performance specifications which have no reference to shape, materials and design, in a bid to separate the design problem from its solution. The above performance specifications then have to be agreed upon by all persons concerned with the design, before proceeding to the Synthesis stage.

In the **Synthesis stage**, the first activity is that of creative thinking involving the use of techniques like brainstorming, to generate ideas and solutions to the performance specifications. One or more partial solutions are then proposed for each performance specification. Each partial solution is considered completely independently of any other. Limits are then established for each partial solution within a range of dimensions, shape and variations in material properties, that will satisfy any performance specifications. The next step then involves the combinations of compatible partial solutions into combined solutions with the least departure from the performance specifications. In planning for production, it becomes necessary to select one or more of the combined solutions. The last activity in the Synthesis stage, involves what is known as solution plotting. This is a means of making clear the relationships between solutions. Two kinds of solutions are usually considered viz: (i) Trends - i.e. relationships between previous solutions and (ii) New Solutions - i.e. relationships between all possible alternative solutions.

The last stage of this model, is the **Evaluation stage** which involves mainly two activities. These are (i) methods of evaluation and (ii) evaluation for operation, manufacture and sales. Under methods of evaluation, Jones advocates the use of evaluation methods to detect errors at the stage when they can be most cheaply corrected. Such methods include evaluation by performance specifications and evaluation by use of precise judgements, judgements extended to include wider experiences of many persons and separation of logical decision and judgements. In evaluating for operation, manufacture and sales, Jones suggests the provision of in house facilities in the form of pre-operation, pre-production and pre-sales units (teams) to carry out evaluation on designs before the most viable ones are decided upon for production. Figure 2.8 illustrates this model.

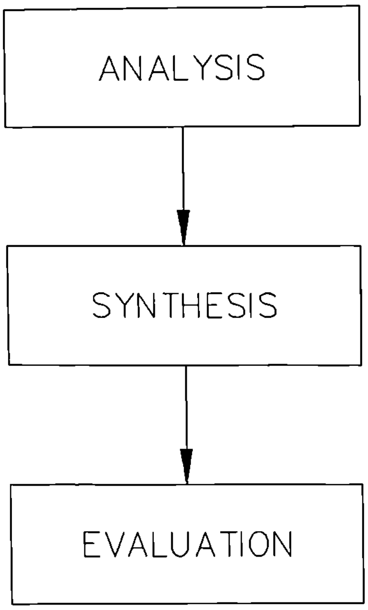


Figure 2.8 The Design Model by Jones

This model emphasizes the need to establish specifications in a solution neutral from as well as investigating interactions between design factors. The synthesis stage does exhibit a bottom-up approach in developing the overall design. The idea of evaluating the designs by the pre-operation, pre-production and pre-sales team, is a late occurrence in this model. These teams in a modern manufacturing industry, should be involved right from the start of the design process. In this model, they should be involved at the analysis stage.

Model by Asimow

In representing the design activity, Asimow [105] shows the morphology of design in three phases which bears on the solution of the design project, while the design process which deals with the solution of sub-ordinate problems, is represented as a sequence of operations as every step of the morphology is proceeded. Figure 2.9(a) illustrates this model.The three phases of design represented in the morphology are Feasibility study phase, Preliminary design phase and Detailed design phase.

Feasibility Study Phase

In the feasibility study phase, the first activity involves the establishment of the original need of the project. The next step then involves the exploration of the design problem and the identification of the design parameters, constraints and major criteria. In the next step, plausible solutions are generated for the problem. The last three steps in this phase, involves the evaluation of the proposed solutions and the selection of potentially viable ones, based on their physical realizability, economic worthwhileness and financial feasibility. Figure 2.9(b) shows the feasibility study phase.

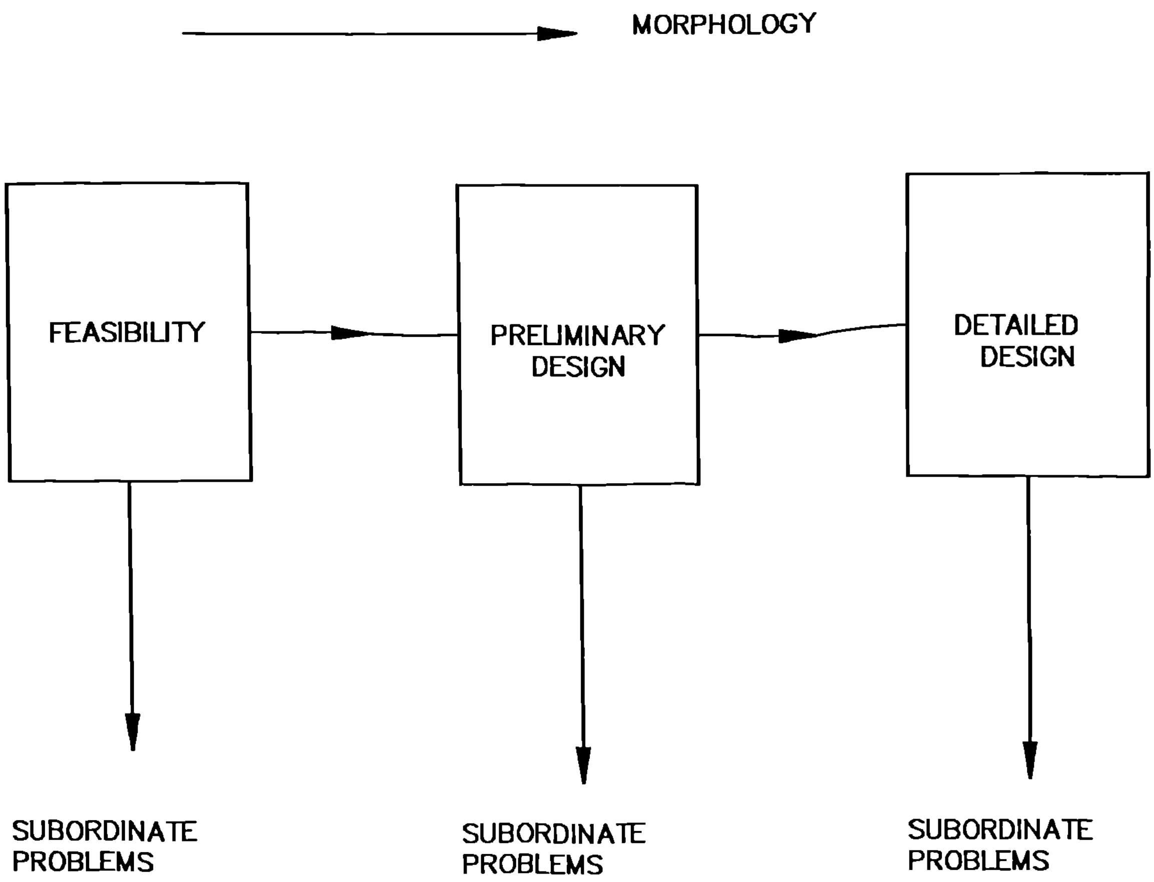


Figure 2.9(a) The Design Phases by Asimow

Preliminary Design Phase

In the preliminary design phase, the first step is to select the best design concept from amongst the viable solutions obtained from the feasibility phase. Mathematical models are then prepared for each of the solutions of the physical object to be materialized. The third step involves the performance of Sensitivity analysis to establish to first approximation, the fineness of the range within which the major design parameters of the system must be controlled. Compatibility analysis which is the next step, involves the investigation of the tolerances in the characteristics of major components and critical materials that will be required to insure mutual compatibility and proper fit into the system. Step 5 is the Stability analysis which is performed to examine the extent to which pertubations of environmental or internal forces will affect the stability of the system. Stability analysis is done inter-alia to : (i) ensure that the system as a whole is not inherently unstable and (ii) determine the regions which are inherently unstable in the design parameter space to insure their avoidance.

PHASE 1 FEASIBILITY STUDY

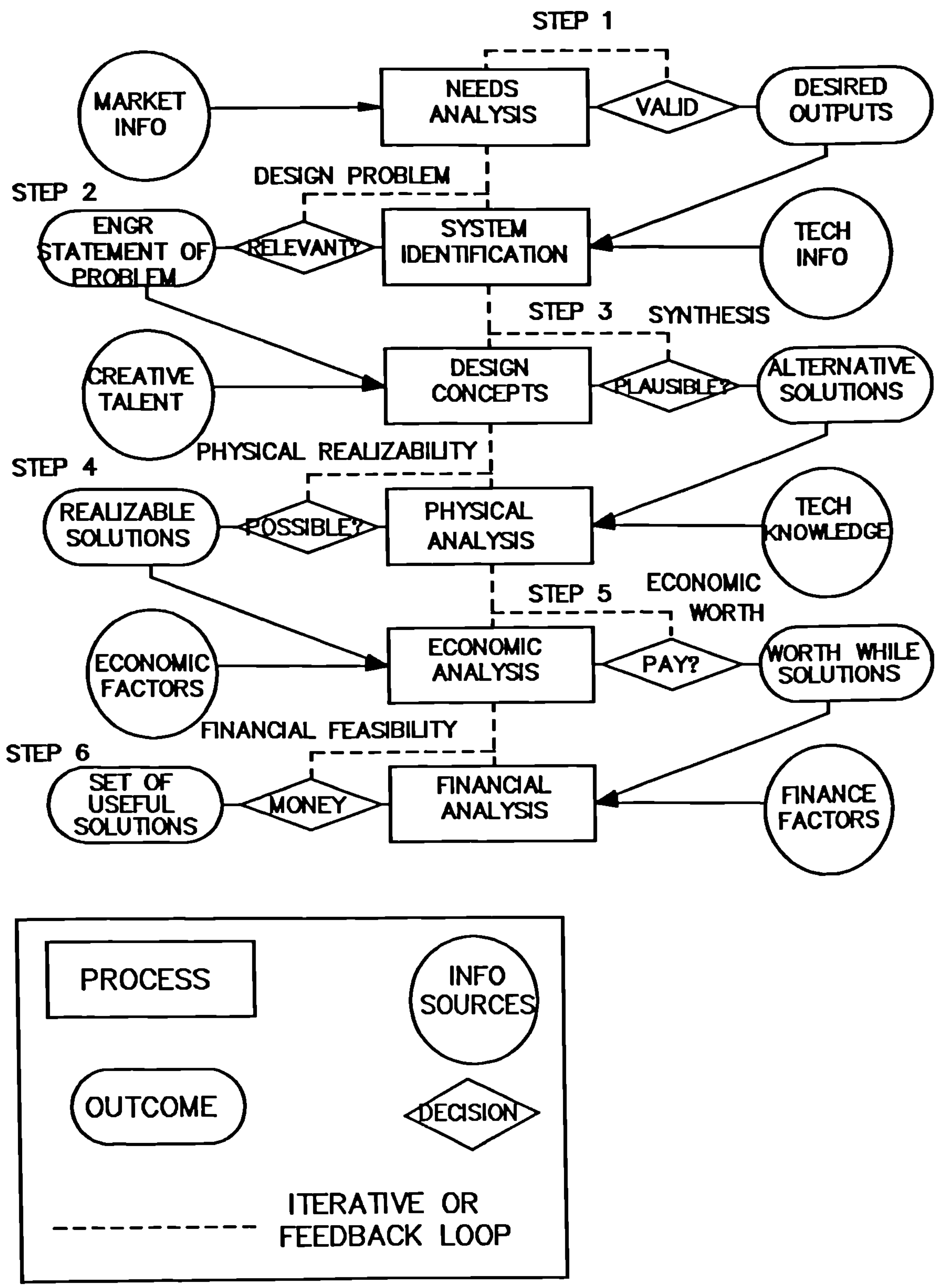


Figure 2.9(b) The Design Model by Asimow - Feasibility Design

In the sixth step, the chosen concept goes through an optimization process. When there are several competing concepts, formal optimization is performed to discriminate and find the best solution from amongst the alternative solutions. The seventh step involves the evaluation of the system against future performance with respect to socio-economic conditions and the rate of wear of the quality of the product. In the eighth step, an attempt is made to predict how the system would behave under various kinds of conditions it may operate in. In the ninth step, the system is subjected to experimental design, and the critical parts of the design concept are also subjected to the hard realities of physical test. The final step involves the process of simplifying the design concept before it is submitted as the proper solution for further development in the detailed design phase. Figure 2.9(c) illustrates the steps in the preliminary-design phase. The third, fourth and fifth steps in this phase, if considered in today's terms, are somewhat synonymous to Taguchi's system, parameter and tolerance design.

Detailed Design Phase

The detailed design phase consists of nine steps. The first step here is to prepare capital budgets and time schedules for the design. The second and third steps involves the overall designs of subsystems and components respectively. The fourth step involves the detailed design of parts. The preparation of assembly drawings for the components and subsystems constitute the fifth step. In the sixth and seventh steps, the prototype is constructed and tested respectively. The eighth step involves the analysis of the difficulties encountered in both steps six and seven as well as prediction of performance under conditions of customer's operations. The final step then involves making minor revisions as convergence is made towards the final design. Figure 2.9(d) shows the detailed design phase. The approach to design in this model, is closely related to a top-down approach, with concept design occurring at the feasibility phase and detailed design occurring at the detailed design phase.

The Design Process

In his book, Asimow [105] expresses the design process in steps of Analysis, Synthesis, Evaluation, Decision, Optimization and Revision. The first step of analysis involves the process of understanding the design problem and making an explicit statement of the goals which the designer wishes to attain. The second step involves the proposal and synthesis of plausible solutions by imagination and creativity. The third step is aimed at judging and validating the design solutions relative to defined goals and if there are alternatives, selecting the most suitable one. The chosen solution is then optimized at the fifth step and eventual revision to improve this chosen solution is performed at the final and sixth step. The important aspect here is that these six steps are repeated at each of the morphological phases.

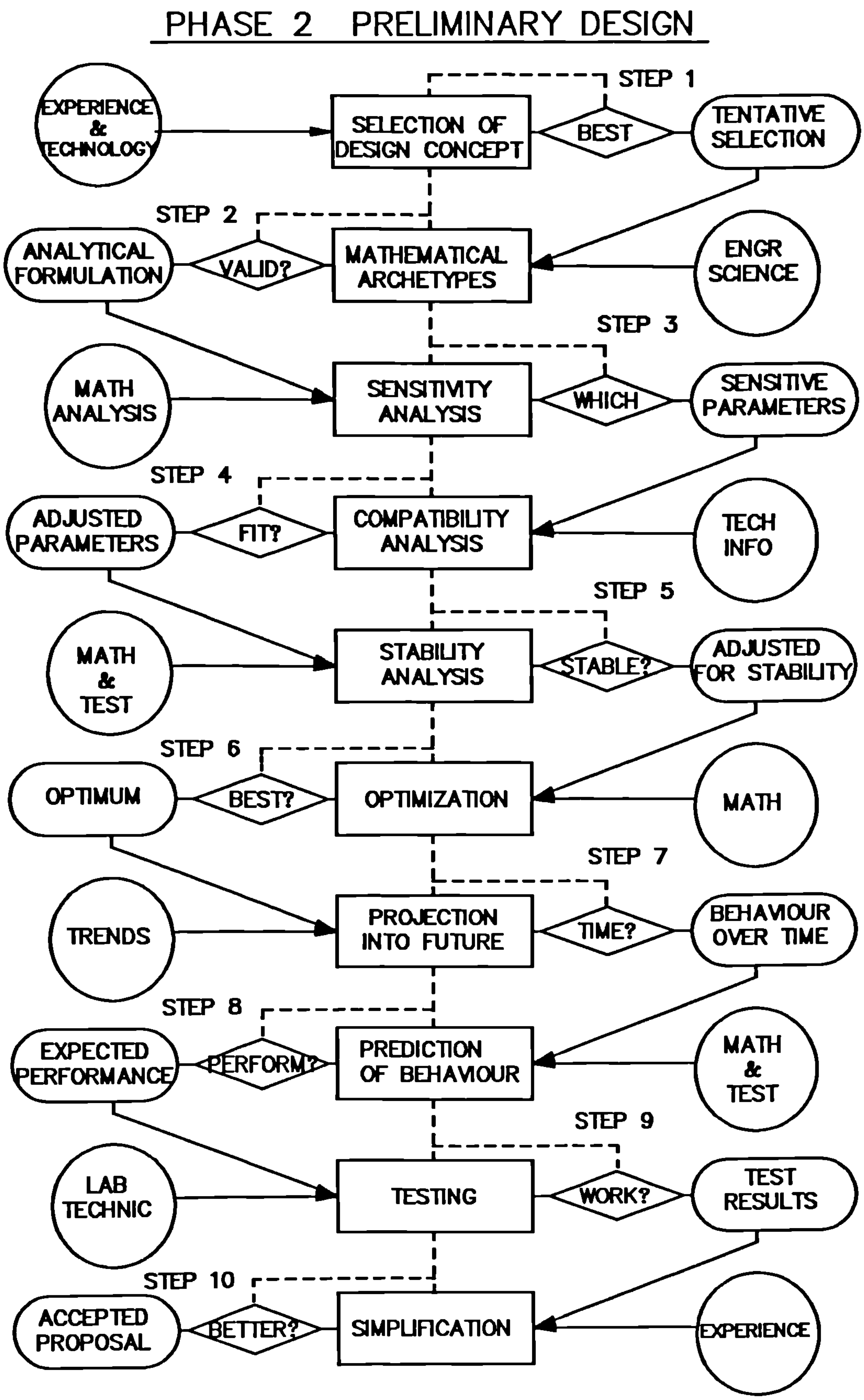


Figure 2.9(c) The Design Model by Asimow - Preliminary Design

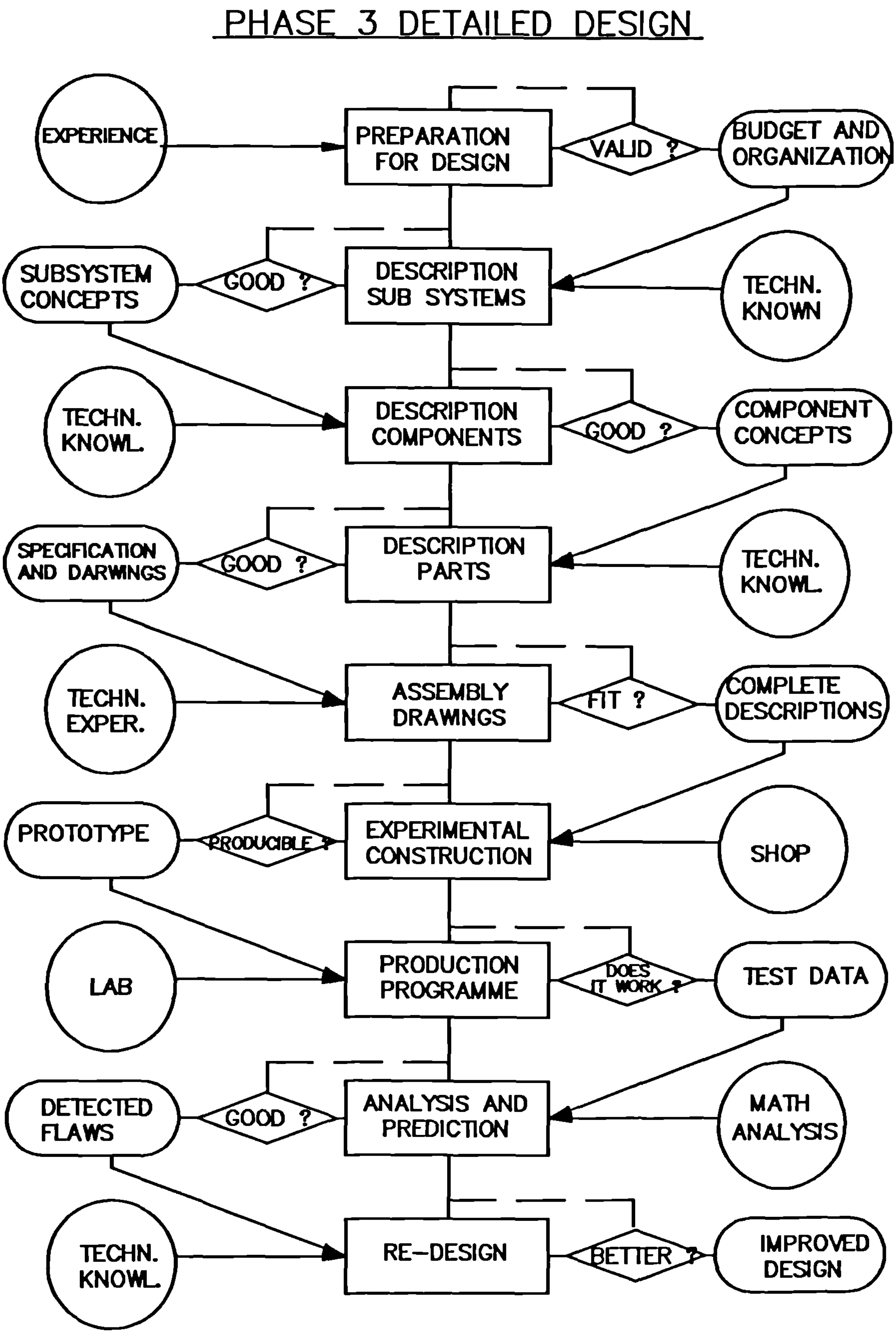


Figure 2.9(d) The Design Model by Asimow - Detailed Design

Model by Pahl & Beitz

Pahl and Beitz [94] represent their model of the design process in four main phases, which are : (1) Clarification of the task, (2) Conceptual Design, (3) Embodiment Design and (4) Detail Design.

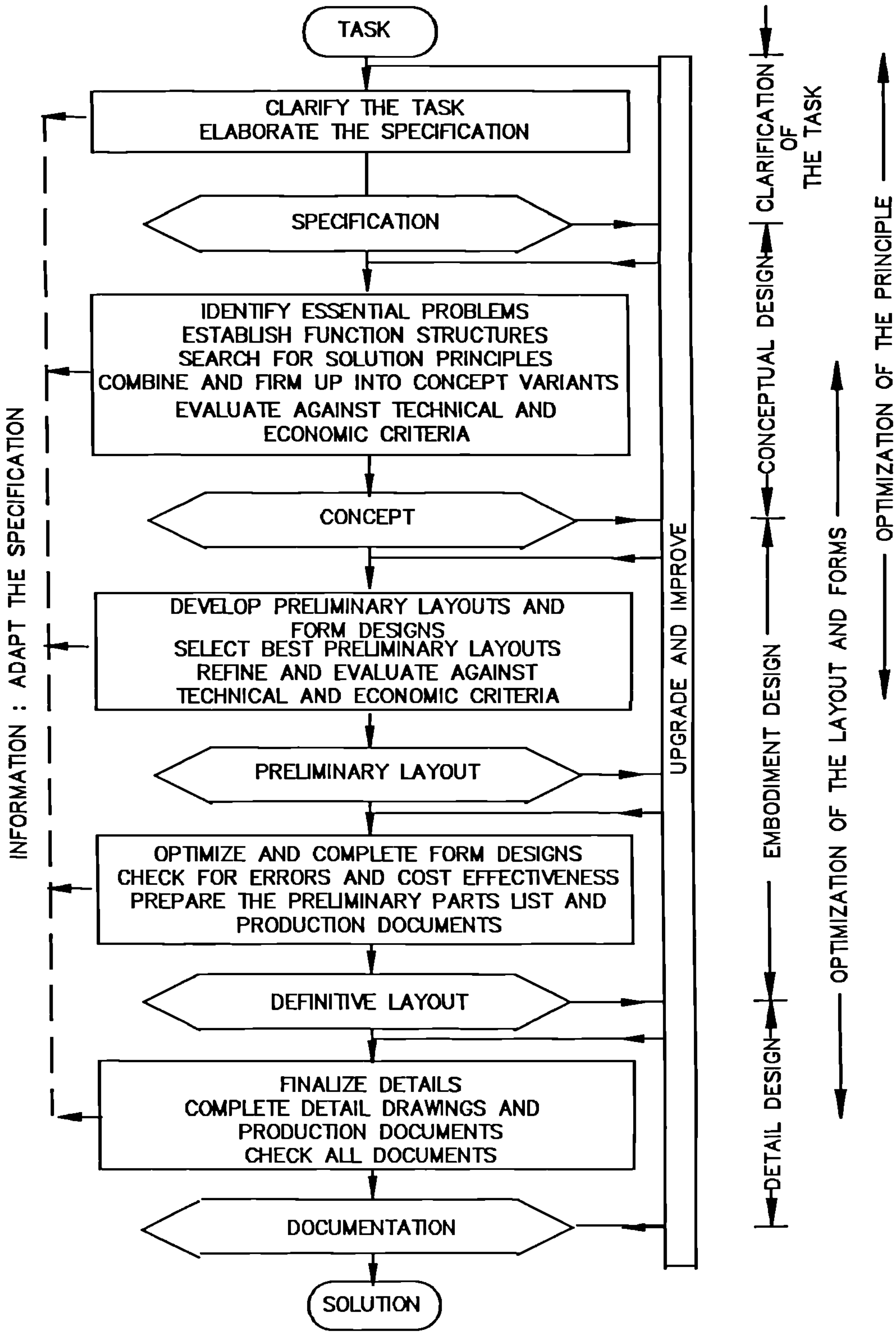


Figure 2.10 The Design Model by Pahl & Beitz

The first phase of clarification of the task involves the collection of information about the requirements in a solution neutral form to be embodied in the solution and also about the constraints. The second phase which is the conceptual design phase involves the establishment of function structures, the search for suitable solution principles and their combination into concept variants. At the embodiment design phase, the designer starting from the concept, determines the layout and forms and develops a technical product or system in accordance with technical and economic considerations. At the last phase of detail design, the arrangement, form, dimensions and surface properties of all the individual parts are finally laid down, the materials specified, the technical and economic feasibility rechecked and all the drawings and other production documents produced. Figure 2.10 illustrates this model.

Model by VDI 2221[100]

This model is produced by Germany's professional engineers body, Verein Deutscher Ingenieure (VDI) in their guidelines VDI 2221 "Systematic Approach to the Design of Technical Systems and Products". The VDI 2221 model expresses the design process in seven stages. The first stage involves the clarification and definition of the design task. The output from the first stage is the design specification, and it is constantly reviewed, updated and used as a reference in all the subsequent stages. The second stage of the process consists of determining the required functions of the design and producing a diagrammatic function structure. In stage three, a search is made for solution principles for all sub-functions, and these are combined in accordance with the overall function structure into a principal solution. This is divided in stage four into realizable modules and a module structure representing the breakdown of the solution into fundamental assemblies. Key modules are developed in stage five into a set of preliminary layouts. These are refined and developed in stage six into definitive layouts, and the final documents are produced in stage seven. Details of the above description of the VDI 2221 design process model is shown in Figure 2.11(a), while Figure 2.11(b) [94] shows the flow of the design process starting from the level of tasks to detail design of components.

Model by Watts

Watts [106] in his paper, represents the design process by an iconic model of a designer or design team in dynamic relationship with an environment containing the total spectrum of scientific and technological knowledge. The design process is described as consisting of three processes of Analysis, Synthesis and Evaluation, as also proposed by Jones[51]. These processes are performed cyclicly from a lower (more abstract) level to a higher (more concrete) level (representing design phases), as represented by the helical path in Figure 2.12. In moving from the abstract level to the concrete one, the designer or design team during the design process, frequently reiterate at one or more levels, and decisions are made along the way as shown on the surface of the cylinder. A state function D of the design is associated with the process path and can be externalized as a set of statements at intersections of the path and the decision line. Various states of the design thus relate to the different levels. The design states (Dm, Dn, etc) give a vertical structure to the process, and proceed through analysis, synthesis of design concepts, evaluation of feasibility, optimization, revision and communication.

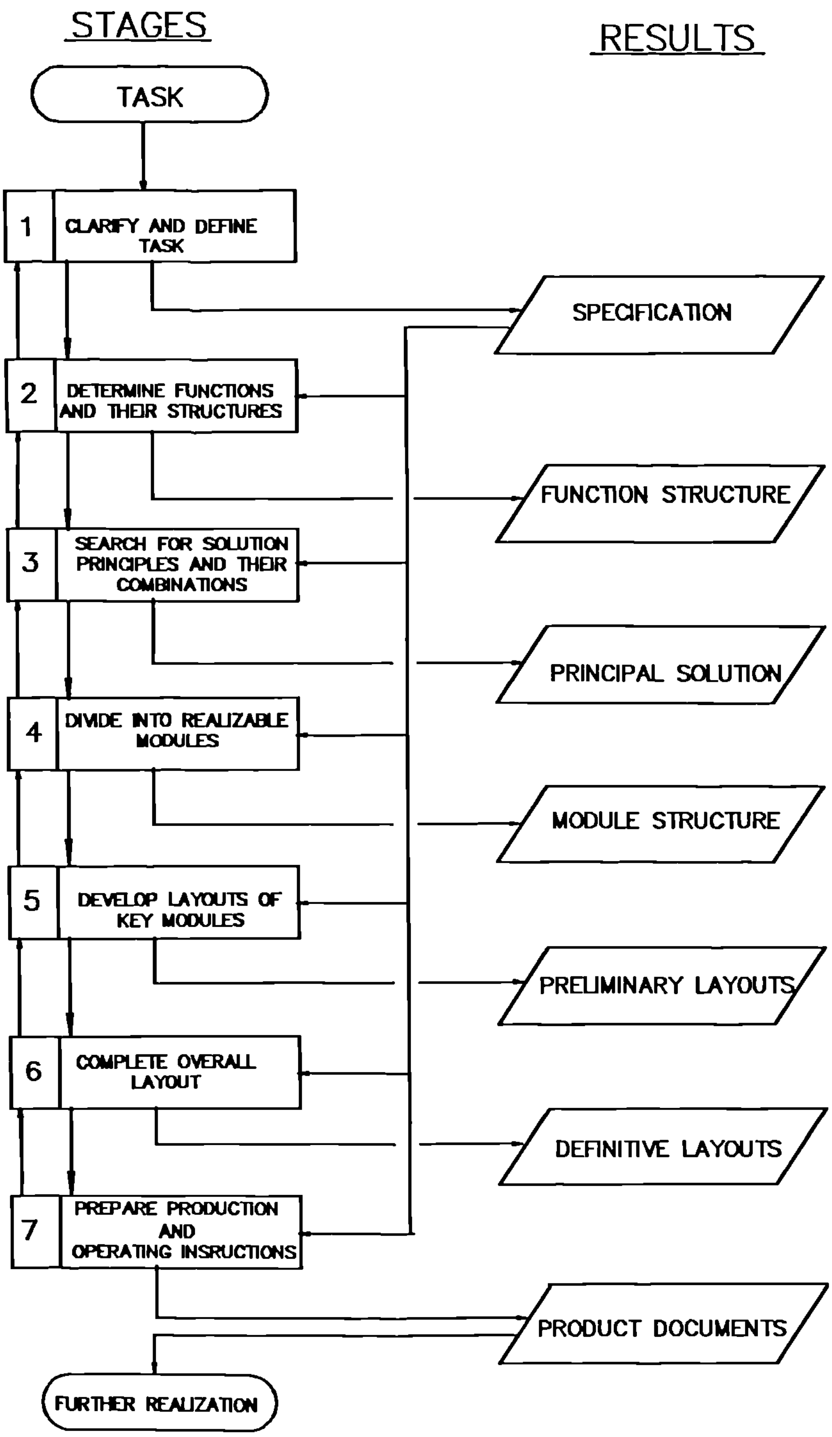


Figure 2.11 The Design Model by VD1 2221

The process can be considered complete when the designer releases into E (a particular environment) a communication P, being a set of prescriptions for the embodiment of the design. The end to which P is a means is an artefact A: this possesses several functional attributes, some of which fulfil the need implied by N; others enhance the profits and reputation of the designer and his company while others could have effects which are far reaching into the socio-economic environment.

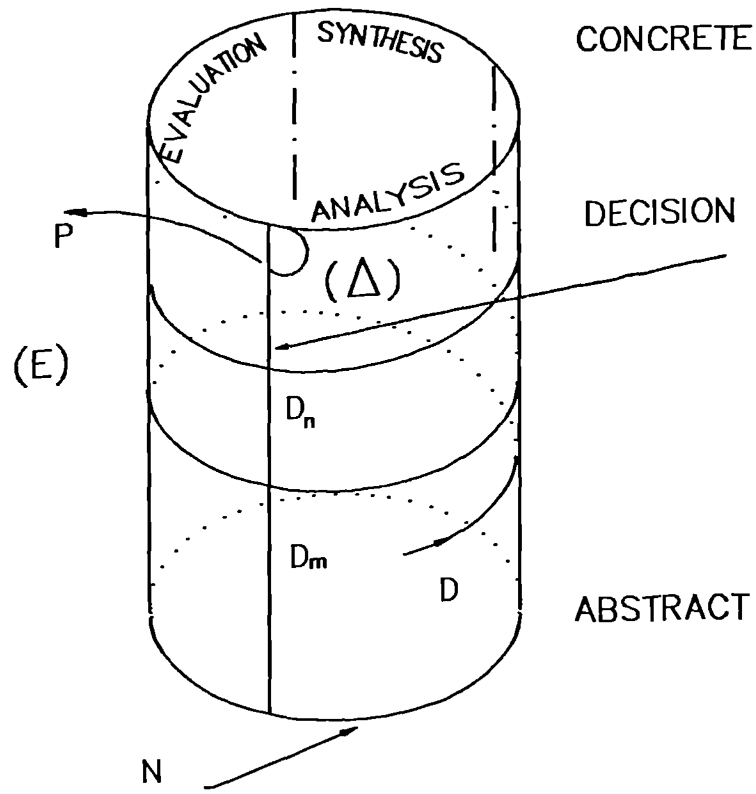


Figure 2.12 The Design Model by Watts

Model by Marples [107]

This model represents an attempt to abstract the process of design, as a result of design case studies carried out. These studies were used to illustrate designing as a sequence of decisions leading from the original statement of the requirements to the specification of the details of the 'hardware' to be manufactured.

The starting point in this model is a statement of the main problem to be solved. This represents the starting node in the 'Marples Tree'. From this node, based on principles along which this problem can be solved, sub-problems are then derived that must be solved before a solution to the main problem is possible. This involves a cyclic process of analysis of the problem, theorizing solutions, delineating these solutions, and modifying them (again involving analysis, theorizing, delineating,etc). This technique can result in novel solutions if it is intelligently used by a lone designer, by a designer in charge of a team of designers or by a small team of designers [90,107]. Figure 2.13 shows a general representation of a typical sequence of the design process. In this figure, the final solution is the sum of the solutions a(21211), a(22211), a(22221) and a(232). Here, a(21211) is the first proposed solution to the first sub-problem of the second proposed solution of the first sub-problem of the second proposed solution to the original problem !. If for example a(2) is preferred to a(1) or a(3), all the sub-problems p(21), p(22), and p(23) must be solved. Similarly, if a(222) is accepted as a solution to p(22), then sub-sub-problems p(2221) and p(2222) must be solved. In the figure, a vertical line denotes a problem, while a slanting line denotes a solution. Any problem or solution can easily be picked out of the tree by means of the series of letters and numbers required to reach it from the origin. This permits easy cross-referencing of notes, layouts, and when necessary statements about the difficulty encountered in solving a particular problem, and availability of solutions [90]. Eder [90] further proposes that all precedent solutions to the main problem such as competitor's models, should also appear on the design tree. This is analogous to competitive assessment in QFD.

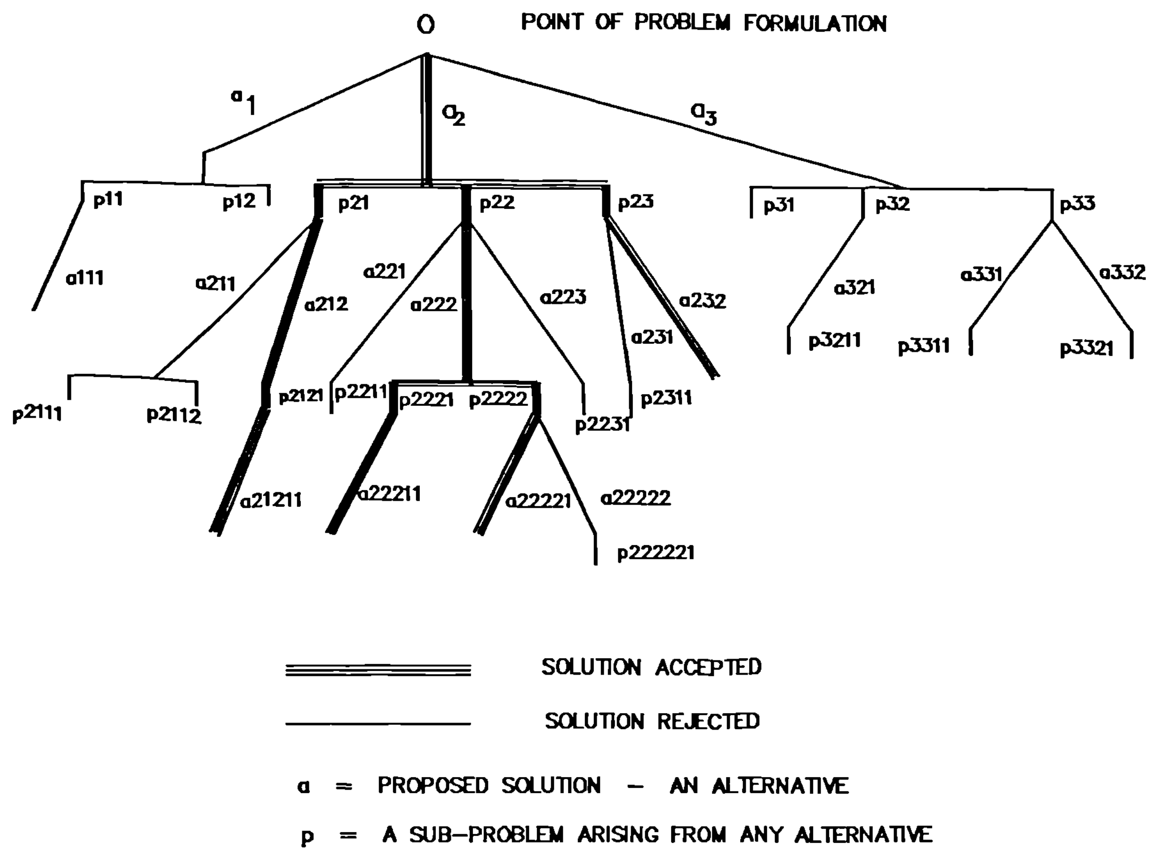


Figure 2.13 The Design Model by Marples

The model is used to discuss the search of possible solutions, the strategies for their examination, and the rules for choosing between them. Each decision made involves the consideration of various proposals ; predictions of the outcomes of each with particular emphasis on the sub-problems raised by it and an evaluation of the outcomes in terms of criteria arising from the natural properties of the materials involved ; engineering values ; prior decisions and the judged tractability of the unassessed portions of the design [107].

The model by Marples involves three principal phases of Synthesis, Evaluation and Decision. At the Synthesis phase, two activities are involved, that is, the search for possible solutions and the examination of proposed solutions. This phase is then followed by the evaluation of the viable solutions, before a final decision is made in choosing a particular solution.

Synthesis Phase - This involves :
(1) The search for possible solutions. Here three main strategies are employed in classifying proposed solutions, while seeking for viable ones. These are : (a) solutions judged not to be feasible, (b) solutions judged feasible but inferior and (c) solutions judged feasible and considered to be the best.
(2) The examination of proposed solutions - At this stage, two strategies can be employed. These are : (a) examination of proposals serially, one after the other until a solution is found which is feasible and (b) examination of proposals in parallel until the best solution can be distinguished.

Evaluation Phase - This involves evaluation of viable solutions against certain criteria that must be applicable to the resulting 'hardware'. These criteria are classified under the following main headings :

Natural Properties
These are usually given, and are closely associated with the properties of the materials involved in the realisation of the product, such as corossion, service temperature and pressure, strength etc.

Engineering Values
They are usually independent of the particular design and are in most cases general to engineering, can be in the forms of standards of acceptability. They include issues such as quality, safety, reliability, costs,manufacturability and so on.

Prior Decisions
This criteria includes high level administrative decisions such as Time, Manpower availability and expense budgets previously decided upon. They can also include acceptable decisions made higher up in the tree which would certainly influence decisions downwards as the design becomes more detailed. These prior decisions can however be altered if it becomes imperative.

The need for geometric ingenuity

This refers to the judged tractability of the unassessed portions of the design. This criteria usually involves an extremely wide unknown region.

Decision Phase

This phase occurs at each stage in the design tree, where decisions have to be made in choosing a particular solution from among a number of alternatives, as well as setting out the sub-problems to be tackled in the next stage. Decisions made usually depend on the level of abstraction of the decision. Early in the design stage, decisions are more strategic and may involve major company policies and hence are made by the Chief Engineer. Further down the tree, when the design gradually becomes more detailed, decisions are made by consensus of the design team involving the engineer responsible for the project. These decisions are made taking into account the evaluation criteria set above.

The above three phases are constantly repeated at each sub-tree level, as the design activity progresses from the top of the tree downwards, with the level of abstraction of the design problem decreasing. At the top, the problem and its solutions are described in relatively abstract terms, while at the bottom more detailed repre-sentions of bits of hardware from particular materials are envisaged.

Model by Archer

Archer [76] defines the nature of design methodology in his model in six stages (steps), viz:

Programming	Establishment of crucial issues and Proposal of course of action
Data Collection	Collection, classification and storing of data
Analysis	Identification of sub-problems; preparation of design specifications; reappraisal of proposed programme and estimation.
Synthesis	Preparation of outline design proposals
Development	Development of prototype design(s); preparation and execution of validation studies.
Communication	Preparation of manufacturing documents.

The above six stages are further classified and grouped into three phases, namely: Analytic, Creative and Executive.

In describing his model, Archer comments that:

" *...the special features of the process of designing is that the analytic phase with which it begins requires objective observation and inductive reasoning, while the creative phase at the heart of it requires involvement, subjective judgement, and deductive reasoning. Once the crucial decisions are made, the design process continues with the execution of working drawings, schedules, etc again in an objective and descriptive mood. The design process is thus a creative sandwich. The bread of objective and systematic analysis may* *be thick or thin, but the creative act is always there in the middle*". Figures 2.14(a) and 2.14(b) show the stages and phases of the design process as well as their interrelationships.

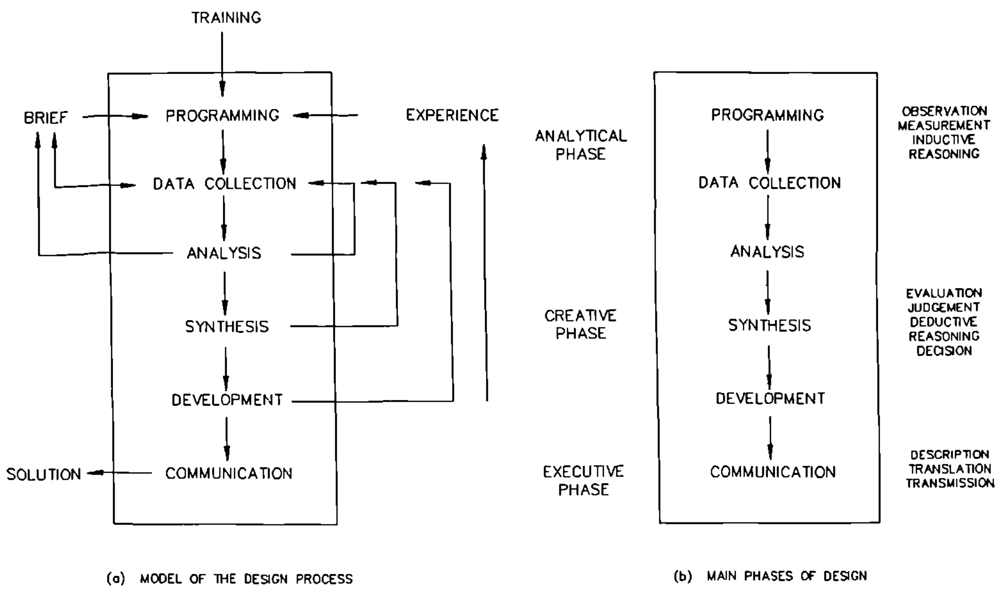


Figure 2.14 The Design Model by Archer

Model by Finkelstein [74]

This model takes account of the fact that the design process consists of a sequence of stages beginning from the perception of need and terminating in a final firm description of a particular design configuration. Each stage is itself a design process involving several iterative sequence of the steps, subprocesses as shown in Figure 2.15.

In this model, the design process begins with information gathering and organisation, delineating the design problem and collecting in an organised manner the necessary information needed for its solution. The principal input to this stage is the requirement specification. The next step then involves the formulation of value criteria, which arise from the requirement and thus forms the basis for the evaluation of the candidate designs. This step is then followed by the generation of a set of proposed candidate solutions. This is the central activity of design and the fundamental process on which design is based. The candidate designs are then analysed to determine their attributes which are relevant to the requirements. The analysis done here may involve calculations, simulation, modelling and so on. The results of the above analysis of the attributes of the candidate designs are then used to evaluate them. This sequence is concluded by the final step where decision is made on a particular candidate design. The chosen design can then be accepted for implementation or as the basis for specification for the next design stage.

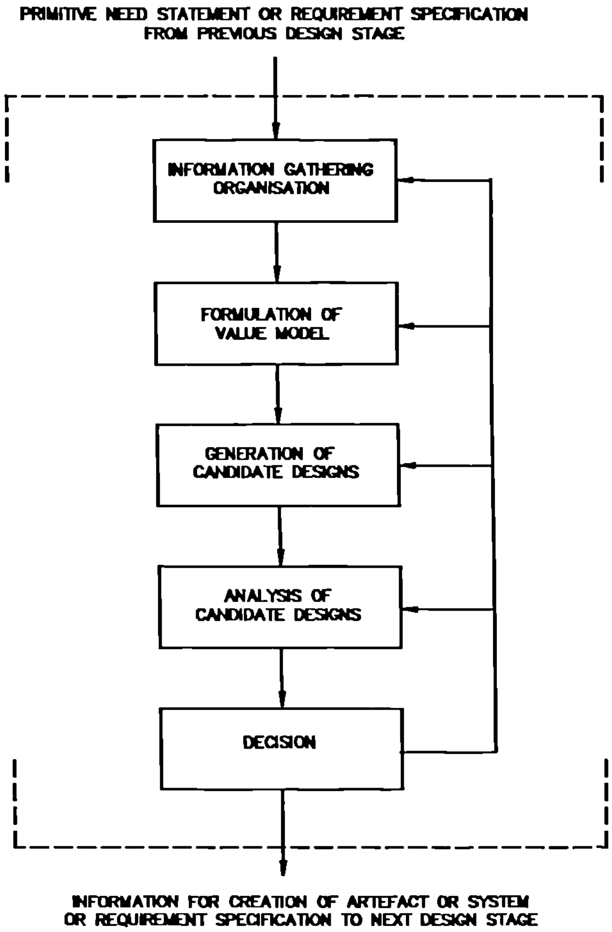


Figure 2.15 The Design Model by Finkelstein

Model by Krick

Krick[108] in his model, describes the design process in five stages of Problem formulation, Problem analysis, Search, Decision and Specification. The first step of problem formulation involves defining clearly in a broad and detail-free manner the design problem to be solved. The next step involves analysing the design problem and arriving at a detailed definition of the problem in terms of specifications, constraints and criteria. In the third step, the search for and generation of alternative solutions is performed through inquiry, invention and research. The decision stage which is the fourth step involves the evaluation, comparison and screening of alternative solutions until the best solution evolves. Finally, the fifth step which is the specification stage is performed. This involves a detailed documentation of the chosen design with engineering drawings, reports and possibly iconic models being the resulting output. Figure 2.16 illustrates this design process.

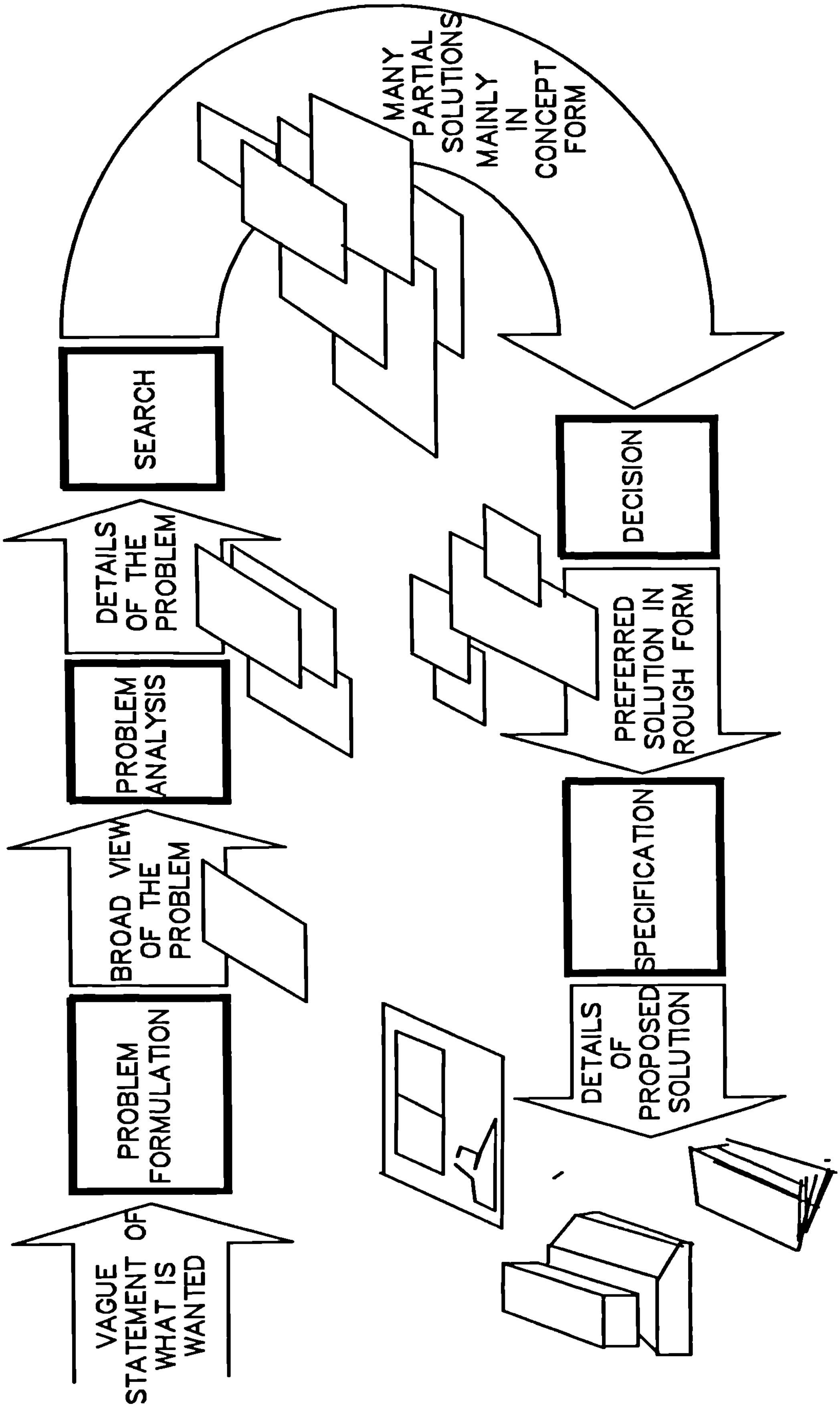


Figure 2.16 The Design Model by Krick

Model by Nigel Cross

In representing his model, Cross [100] expresses the design process in six stages within a symmetrical problem - solution model as shown in Figure 2.17. The six stages are Clarification of objectives, Establishing functions, Setting requirements, Generating alternatives, Evaluating alternatives and Improving details. For each of the stages, a design method is used to achieve the objective in that stage. In the first stage of clarifying objectives, the objectives tree method is used to clarify design objectives and sub-objectives and the relationship between them. The function analysis method is then used to establish function required and the system boundary of a new design at the second stage. In stage three involving setting of requirements, accurate specification of performance required of a design solution is done using the performance specifications method. The morphology chart method is then used at stage four to generate the complete range of alternative design solutions for a product, and hence to widen the search for potential new solutions. In stage five, the design alternatives are evaluated using the weighted objectives method to compare the utility values of alternative design proposals on the basis of performance against differently weighted objectives. The sixth and final stage of improving details, involves using the value engineering method to increase or maintain the value of a product to its purchaser whilst reducing its cost to its producer.

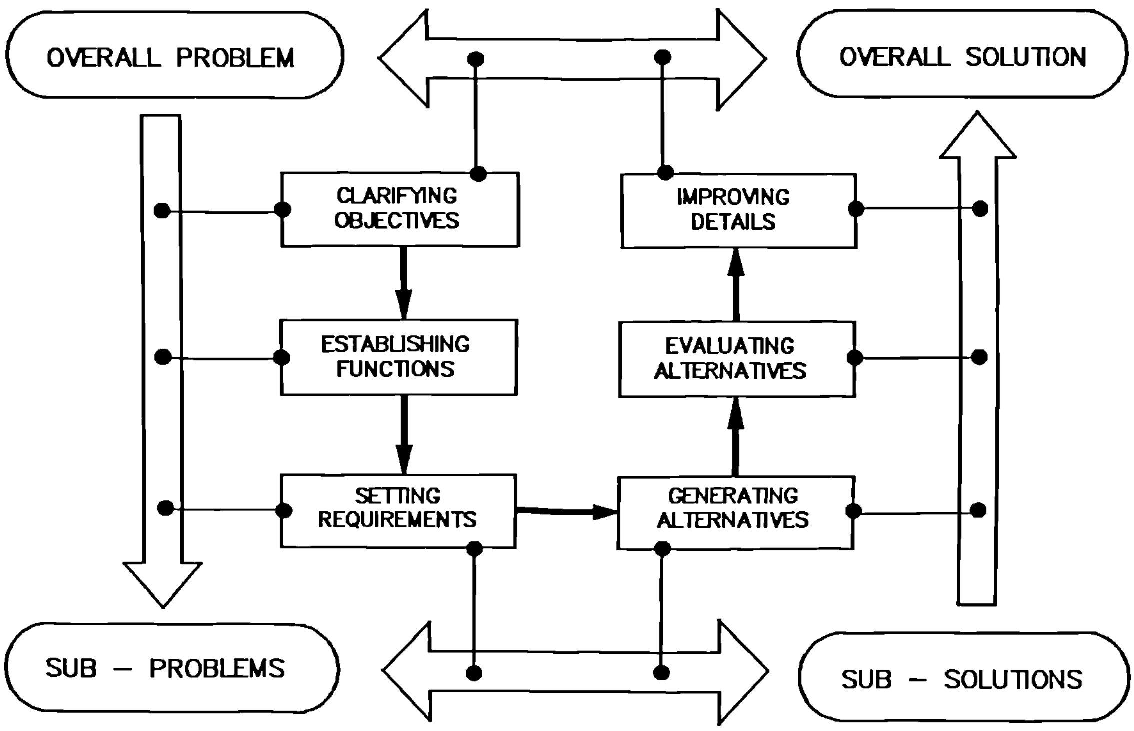


Figure 2.17 The Design Model by Cross

Model by Hubka

The model by Hubka [109] represents the design process in four phases and six stages or steps. These phases and steps are :

Phase 1 Elaboration of Assigned Problem
- Step 1 Elaborate or Clarify Assigned Specification

Phase 2 Conceptual Design
- Step 2 Establish Functional Structures
- Step 3 Establish Concept

Phase 3 Laying Out
- Step 4 Establish Preliminary Layout
- Step 5 Establish Dimensional Layout

Phase 4 Elaboration
- Step 6 Detailing & Elaboration

The procedural model is shown in Figures 2.18.

The model is described in terms of the six steps shown above.

Step 1 - Elaborate or Clarify Assigned Specification

This first step begins with the problem assignment, and entails the following activities:

1.1 Critical recognition of assigned problem
1.2 Establishment of the state of the art
1.3 Analysis of the problem situation
1.4 Examination of possibilities of realisation
1.5 Classification, quantification and prioritization of completed requirements
1.6 Working out of full design specification and finally
1.7 Preparation and planning of organisation for problem-solving.

Step 2 - Establish Functional Structures

This step involves :

2.1 Abstraction of the design specification from step 1 into a black box
2.2 Establishment of technological principle and sequence of operations
2.3 Establishment of technical processes and choice of optimal technical process
2.4 Application of technical systems and establishment of boundaries
2.5 Establishment of grouping of functions
2.6 Establishment and representation of functional structure
2.7 Evaluation, Improval and Verification of functional structures and selection of optimal functional structure

Step 3 - Establish Concept

In this step of the design process, the following activities are undergone :

3.1 Establishment of inputs into, and modes of actions of the machine system
3.2 Establishment of classes of function-carriers, involving morphological analysis
3.3 Combination of function-carriers and examination of relationships
3.4 Establishment of the basic schematic arrangement of concepts
3.5 Evaluation, Improval and Verification of concepts and selection of optimal concept

Step 4 - Establish Preliminary Layout

In step 4, the following activities are carried out :

4.1 Establishment of orientation points for form determination
4.2 Arrangement, re-use, rough form-giving and partial dimensioning of layouts
4.3 Establishment of types of materials, classes of manufacturing methods, tolerances and surface properties, where necessary
4.4 Investigation of critical form determination zones
4.5 Representation of preliminary layout
4.6 Evaluation, Improval and Verification of preliminary layouts and selection of optimal preliminary layout

Step 5 - Establish Dimensional Layout

This step involves the following :

5.1 Deliverance of substantiation for certain design characteristics
5.2 Definitive arrangement, and partial dimensioning for form determination
5.3 Definitive and complete determination of materials, manufacturing methods, tolerances and surface properties
5.4 Optimisation of critical form determination zones
5.5 Representation of dimensional layout
5.6 Evaluation, Improval and Verification of dimensional layouts and selection of optimal dimensional layout.

Step 6 - Detailing and Elaboration

The final step of the design process involves the following activities :

6.1 Deliverance of the substantiated design
6.2 Form determination, definition and complete dimensioning
6.3 Establishment of definitive and complete materials, manufacturing methods, tolerances and surface properties.
6.4 Determination of assembly procedures and states of assembly
6.5 Representation of parts, dimensions, tolerances, surface properties and materials specification
6.6 Creation of assembly drawings, parts lists, and further documents prepared
6.7 Evaluation, Improval and Verification of detailed design of optimal dimensional layout, before eventual release for production.

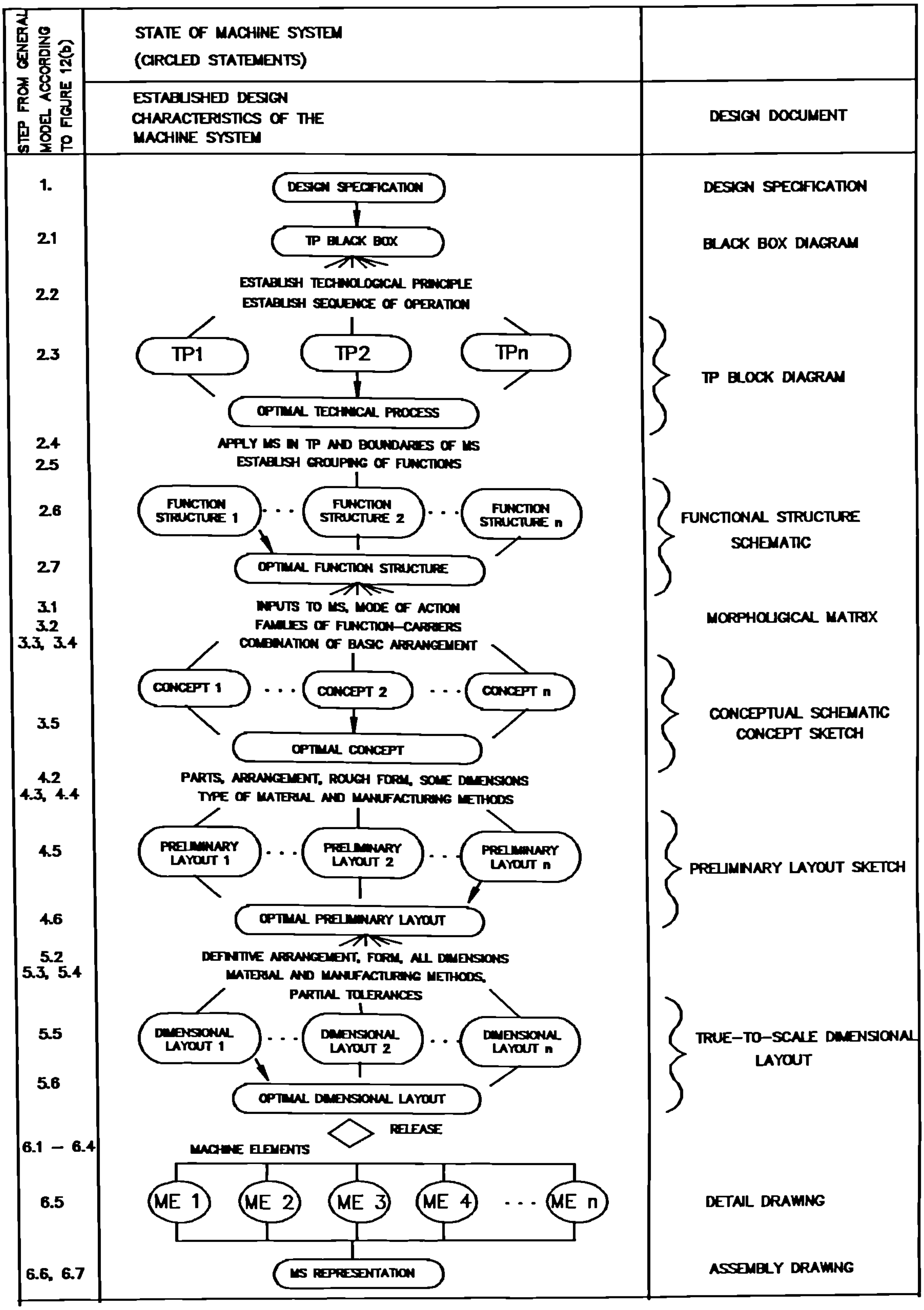


Figure 2.18 The Design Model by Hubka

Model by Seireg [110]

This author considers the ultimate measure of a successful design as the satisfaction of the need with minimum social cost and its acceptance by the user. Design is considered to consist of three basic activities namely, creativity, analysis and decision making. The model proposed by Seireg is as shown in Figure 2.19, and the details are discussed below.

Identification of Need

This is the first step involving the recognition of the need that constitutes the design problem. Here the designer should be sensitive to his environment and seek by careful observation and deduction to establish the exact nature of the need.

Analysis of Need

At this stage, the designer determines if the need as conceived is truly what is needed, by thorough investigation of the conditions producing the need and functional evaluation of the objectives. The resulting outcome of this stage is a formal statement of the design problem.

Information Collection and Organisation

This stage involves the effective search for, evaluation and gathering of both previously acquired knowledge and current information relevant to the problem.

Synthesis of Plausible Alternatives

Most design problem by nature can be satisfied by multiple solutions. This stage hence involves the creative activity of generating all possible alternative solutions before adopting any one. Here, the ability to fit together combinations of principles, concepts and materials in order to produce plausible solutions is most relevant.

Selection of Most Feasible Concept

This stage involves qualitative evaluation of the concepts generated from the previous stage, by comparing the relative merits of the alternative solutions based on a weighted sum of the different objectives.

Modelling or Simulation of the Most Feasible Concept

At this stage, the selected concept from the previous stage, is then modelled for necessary analysis and optimisation.

The Search for Optimal Solutions

This step is regarded as system optimization. It involves a strategy to reach a decision on the combination of parameters which define a feasible system with the highest merit according to a stated objective.

Implementation of Design

This is the stage where all the results of the previous stages are translated into a workable system by the implementation of the design. The implemeted design is then tested and evaluated, in order to ascertain that the original design objectives are fulfilled.

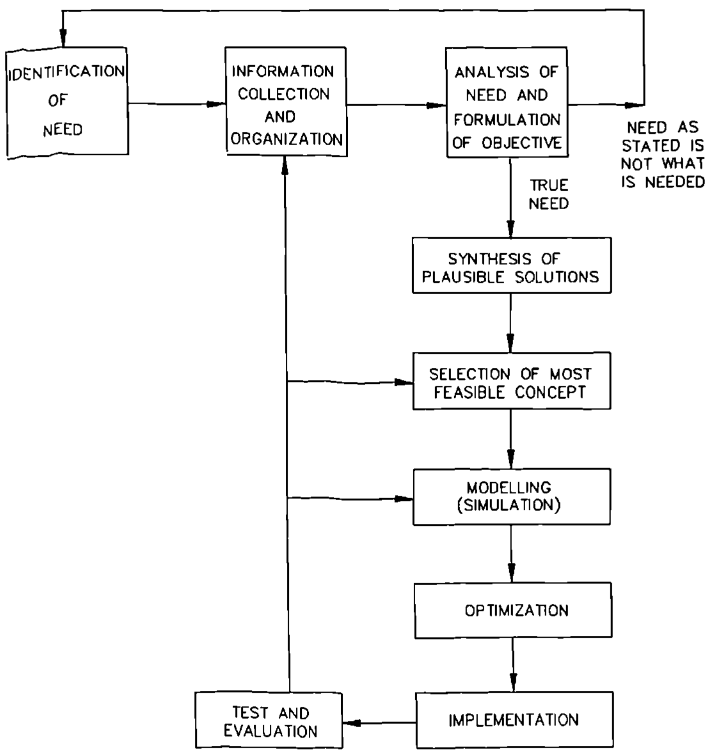


Figure 2.19 The Design Model by Seireg

Model by French

This model is based on the following activities of design [111]: (1) Analysis of the problem, (2) Conceptual Design, (3) Embodiment of Schemes and (4) Detailing. These activities are described below in accordance with Figure 2.20 which shows the model, with circles representing stages reached, or outputs, and the rectangles representing activities or work in progress.

(1) **The Analysis of the problem phase** involves identifying the need to be satisfied as precisely as possible or desirable. The output of this activity, is the statement of the problem, which can have the following three elements : (i) a statement of the design problem proper, (ii) the limitations (constraints) placed upon the solution, e.g. codes of practice, statutory requirements, industry standards, etc., and (iii) the criterion of excellence to be worked to.

(2) **The Conceptual Design phase** takes the statement of the problem and generates broad solutions in the form of schemes. In this phase, engineering science, practical knowledge, production methods, and commercial aspects are given due consideration before taking the necessary important decisions.

(3) **The Embodiment of Schemes phase** involves developing the schemes generated from the previous phase, into greater details, and when there is more than one scheme, a choice has to be made between them. There is usually a lot of feedback from this phase to the conceptual design phase.

(4) **In the Detailing phase**, the selected scheme is worked into finer details, and decisions are made as regards very small but essential points. The end result here is the working drawings of the chosen scheme to be produced.

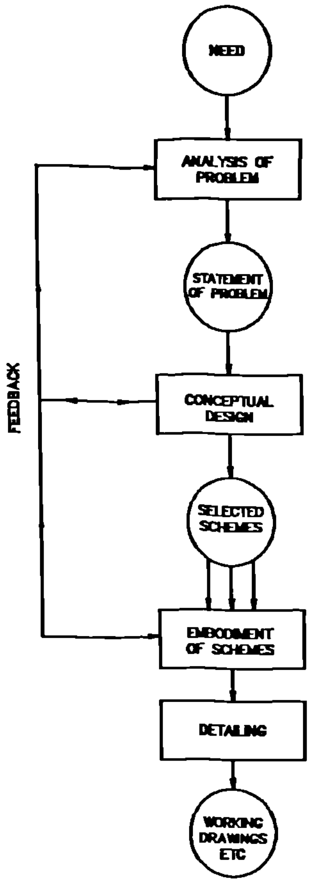


Figure 2.20 The Design Model by French

<u>Model by Sir Alan Harris [112]</u>

This model is based on proposals regarding the teaching of design within the civil engineering discipline. The model consists of five stages, viz : *Appreciation of the Task, Conception, Appraisal of Concepts, Decision, Checking and Elaboration*

<u>Appreciation of the Task</u>
This means discovering what is needed, and to what end and ascertaining what resources are needed and from where. The design starts with finding out what a client wants - This is regarded as the *Total Function*. The designer has to also find out all necessary information about the task, and the conditions affecting its execution.
<u>Conception</u>
In this stage, based on the full digestion of the facts generated from the previous stage, ideas of solutions should begin to emanate. These ideas may be vague, and in some cases may be detailed. They can arise from a leap of intellect (invention) as well as being the result of deliberate techniques like, 'wild or divergent thinking',

'brainstorming', or design team meetings. "During this stage of design, the designer is putting together what he knows of the function of the work with tentative ideas of form, material and method of construction".

Appraisal of Concepts

This stage is where the searching eye based on experience becomes invaluable. Proposed schemes are critically examined to see if they satisfy the needs, can be constructed, how economic both in first cost and in function throughout their working life. Here, hidden snags are examined and possible shortcomings of the design are remedied. Preliminary structural analysis are also carried out to check the broad adequacy of schemes, although close analysis may be necessary for some crucial details.

Decision

After successive operations of conception and appraisal, it then becomes necessary to decide on a particular design scheme. Criteria for decision making may include both simplicity and distinction of the design, as well as constructability.

Checking and Elaboration

This is the stage where the designer makes sure of the adequacy of what is proposed and performs elaboration of necessary details. Here, models can be built and tested. Powerful analytical techniques can also be employed in (1) defining the actions on the structure such as load, temperature difference, corossion etc; (2) analysing the effects of these actions and (3) comparing these effects with a criterion of adequacy. Elaboration at this stage usually proceeds at the same time as checking, with the detailing process varying in extent depending on materials used, contractual arrangements etc. The end result of the design process, is the communication of the detailed design both in the form of drawings and text.

Total Design Activity Model by Pugh [113]

Pugh regards Total design as the systematic activity necessary from the identification of the market/user need, to the selling of the successful product to satisfy that need - an activity that encompasses product, process, people and organisation. The Total Design Activity Model consists principally of a central **design core**, which in turn consists of market (user need), product design specification, conceptual design, detail design, manufacture and sales. The design process in this model proceeds firstly, by identifying a need which when satisfied, fits into an existing or a new market. From the statement of the need, the product design specification (PDS), representing the specification of the product to be designed, is then formulated. The established PDS then acts as a mantle or cloak that envelops all the subsequent stages in the design core, thus acting as the control for the total design activity. Within this model, the design process flows from market to sales, is an iterative one, and recurse can be made to any of the earlier stages, as new ideas and information emerge. This hence causes interactions between the different stages ofthe design core. This model also recognises the fact that, for effective and efficient design do be carried out, it is necessary to utilise various design techniques, to enable the designer/design team to operate the core activity. These design techniques or methods include : (i) discipline-independent ones which relate directly to the design core and can be applied to any product or technology, such as tools for performing analysis, synthesis, decision making, modelling, etc., (ii) specific discipline-dependent techniques and technological knowledge such as stress analysis, hydraulics, thenal analysis, thermodynamic analysis, electronics, etc. This model also takes into account within the overall product development process, the framework of planning and organisation. Thus gaining insight into the way products should be designed within a business structure. The Total Design Activity Model is shown in Figure 2.21.

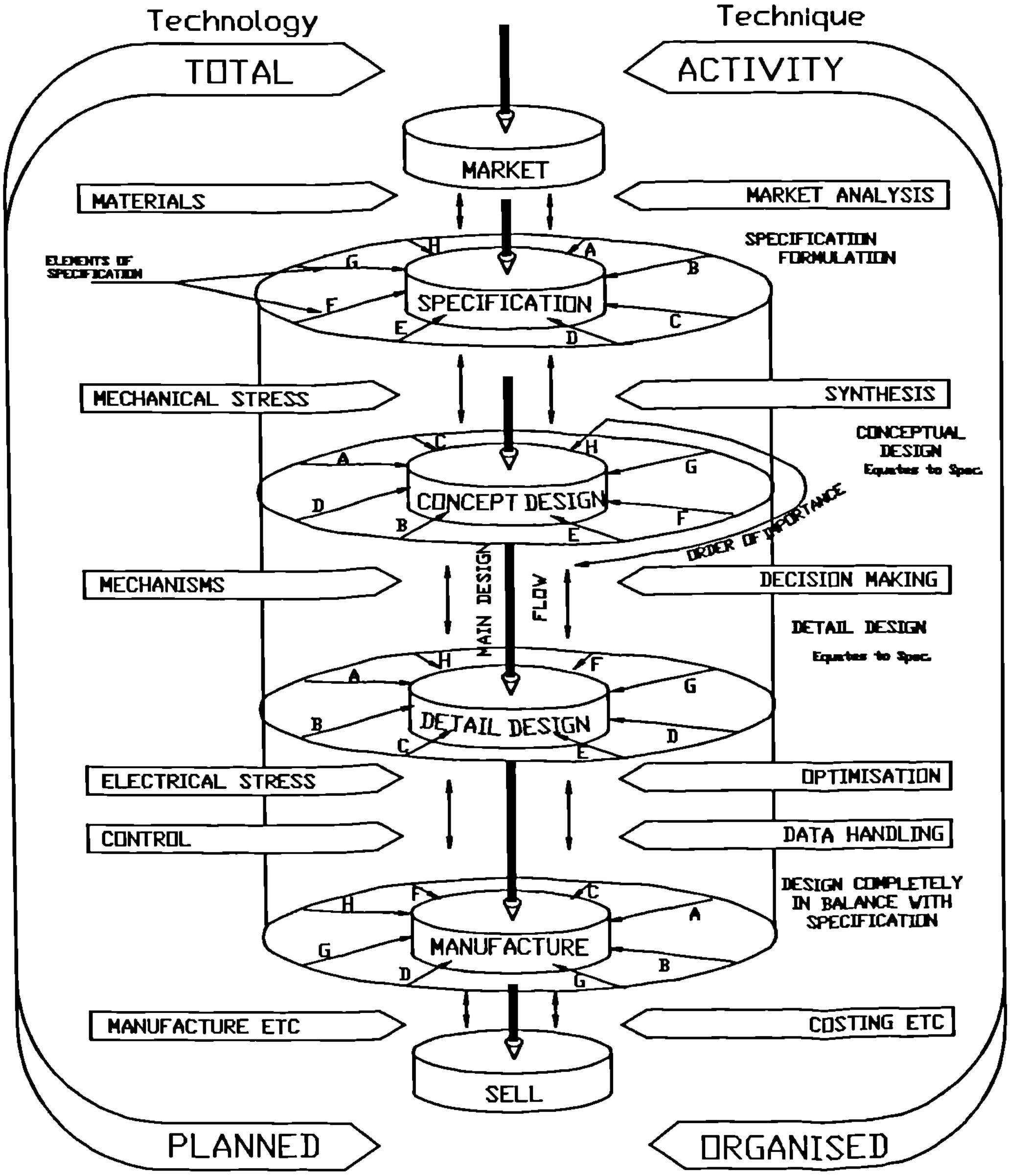


Figure 2.21 The Total Design Activity Model by Pugh

2.3.16 A Critical Appraisal of Prescriptive Models

An in-depth review of the prescriptive models on the design process, show that a majority of them based the procedural steps of their models, on what can be regarded as design activities (or the anatomy of the design process, i.e. analysis, synthesis, evaluation, decision, etc), while others based their procedural steps on what can be regarded as the phases/stages of design (or the morphology of the design process, e.g. conceptual design, embodiment design and detailed design). The models that were based on the phases (morphology) of the design process, include those of Asimow, Pahl and Beitz, VDI 2221, Watts, Hubka and French. With the exception of that of French and VDI 2221, the other models also contained in a more detailed form within each of their design phases/stages, the design activities (anatomy of the design process) which characterised a majority of the other models. The Watts model showed only the two ends of the design phase, that is, abstract and concrete, with the interval between represented by a cyclic (iterative, refining and progressive) process.

The models that were based on design activities, included those by Jones, Marples, Archer, Finkelstein, Krick, Cross, Seireg and Harris. It can also be observed that in all of the models, three key activities were predominant, that is, analysis, synthesis and evaluation. Analysis was mostly related to analysing the design problem, requirements and specifications. Synthesis was concerned with generating ideas, proposing solutions to large or small design problems as well as exploring the design solution space, while evaluation involved the appraisal of design solutions in order to establish whether they satisfied the requirements, specifications and set corporate criteria. The sequence in general also tended to be analysis first, followed by synthesis and then evaluation. In the model by Krick, synthesis was replaced by search and evaluation by decision while in the model by Seireg, evaluation was replaced by selection. The model by Harris represented analysis, synthesis and evaluation, by appraisal of the task, conception and appraisal of concepts, respectively.

It is not surprising that the three activities of analysis, synthesis and evaluation, were predominant, as they represent the core of the design process. If proper analysis of the problem or requirements is not carried out, synthesising solutions will be difficult and inappropriate solutions might be the result. Once plausible solutions are created, there is then a need to evaluate, test and assess their fidelity to the originating requirements and specifications as well as set criteria. Besides the three activities, there are however other necessary activities that should be performed during the design process such as, optimisation, revision, data collection, documentation, communication, selection, decision making, modelling, etc. Some of these activities were included in some of the models.

2.3.12 Prescriptive Models Based on Product Attributes

A majority of product or systems failures can be attributed to either or a combination of the following : (i) incorrect or excessive functional requirements, (ii) continuing alteration to functional requirements, (iii) wrong design decisions and (iv) inability to recognize faulty decisions early enough to rectify them. Thus the existence of unacceptable designs as well as good designs, lends weight to the argument that there should be some features or attributes that can distinguish between good and unacceptable designs. Furthermore, the fine tuning of the later stages of engineering operations, may often have marginal effects on the total outcome of the product, and certainly cannot rectify erroneous decisons made at conception. The foregoing reasoning formed the basis of Suh's axiomatic approach to design based on attributes of the design produced [83].

Taguchi [114], also argues that the total costs at the point of production and at the point of consumption should be minimum for good designs and this should be the goal of product development. He introduces a 'loss function' as an attribute of the product designed which has to be minimised to achieve robust designs. Vasseur, Cagan and Kurfess [115], also present a model with an attribute 'welfare provided by a product' to optimally select a production mode and the quality level of a product.

<u>Taguchi's Quality Loss Function Model</u>

The recent past has witnessed the shift in focus from on-line quality control to off-line quality. This has led to increasing focus on the integration of quality into the early design stage of product development. Ensuring quality by design, thus involve the use of structured off-line methods to determine the design configurations that meet the customer's needs and are robust, where robustness means that product performance characteristics are insensitive to variation in the manufacturing and operating environments [116]. One of the main proponents of off-line quality control is the renowned Japanese, Professor Genichi Taguchi, who introduced the concept of 'quality loss' or 'loss to society'. Taguchi's methodology is based on the precept that the lowest cost to society represents the product with the highest quality, which is achieved by reducing variation in product characteristics. This approach is expressed by what is called the **'Loss Function'**. The loss function is a mathematical way of qualifying cost as a function of product variation. This loss function allows a determination to be made as to whether further reduction in the variation will continue to reduce costs. The loss function includes production costs as well as costs incurred by the customer during use [117]. The simplest form of the loss function is expressed by a quadratic relationship obtained fron a Taylor series expansion , and which can be approximated by :

$$L(Y) = k(Y-M)^2$$

L	=	loss associated with a particular performance characteristic, Y
M	=	the performance target value
k	=	loss parameter $= Lc/D_o{}^2$

where Lc = average loss to the customer when the performance characteristic is not within the limit, Do

Do = customer tolerance limit.

The loss function L(Y) which is shown graphically in Figure 2.22, can thus be defined as the average of the financial loss due to deviations of the product characteristic, Y, from the target function, M, over all customer conditions up to the time required for the product life.

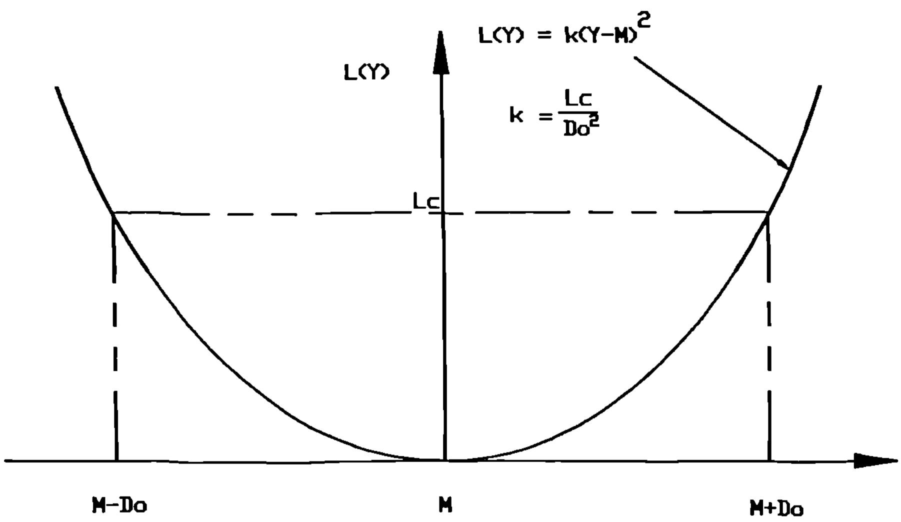


Figure 2.22 Taguchi's Quality Loss Function

Suh's Axiomatic Design Model

The basic premise of Suh's [83] axiomatic approach to design, is that there are basic principles that govern decision making in design, just as the laws of nature govern the physics and chemistry of nature. He describes the design process as a mapping process between the functional requirements FRs in the functional domain and the design parameters DPs in the physical domain. Mathematically this can be expressed as :

$$\{FR\} = [A]\{DP\}$$

In the above equation, the matrix [A] represents the design relationship. In furthering his principles of design, Suh defines two axioms, which are :

(i) Axiom 1 : The Independent axiom

Maintain the independence of Functional Requirements (FRs)

Alternative statement 1 : An optimal design always maintain the independence of FRs.

Alternative statement 2 : In an acceptable design, the DPs and FRs are related in such a way that specific DP can be adjusted to satisfy its corresponding FR without affecting other functional requirements.

(ii) Axiom 2 : The Information axiom

Minimize the information content of the design.

Alternative statement : The best design is a functionally uncoupled design that has the minimum information content.

Associated with these axioms are eight corollaries (having a flavour of design rules) and sixteen theorems (propositions that follow from the axioms or other propositions).

Suh also classifies designs into three categories namely : uncoupled, coupled and decoupled designs. An uncoupled design is a design which obeys the independent axiom and any specific DP can be adjusted to satisfy a corresponding FR. A coupled design has some of the FRs dependent on other functions. When the coupling is due to insufficient number of DPs when compared to the number of FRs, they may be decoupled by adding additional DPs. A decoupled design may have more information content. In the axiomatic approach, the design process is split into four main aspects of : (i) Problem definition, which results in the definition of FRs and constraints; (ii) Ideation or creation of ideas, which is the creative process of conceptualizing and devising a solution; (iii) Analysis of the proposed solution, which involves the process of determining whether the proposed solution is a rational solution that is consistent with the problem definition and (iv) Checking of the fidelity of the final solution to the original needs.

Vasseur et al Product Welfare Model

In their model, Vasseur et al [115] have proposed a classic economic framework to assess the value society attaches to the quality of a product based on engineering considerations. This approach enables the examination of the concept of quality loss from a societal and corporate perspective, known as "Welfare". From the societal perspective, quality is measured in terms of the consumer surplus, that is, the difference between what consumers are willing to pay and what they actually pay, and from the corporate perspective, it is measured in terms of the manufacturer's profit (sales revenue minus cost). The model, taking into account a set of assumptions on consumer's behaviour, defines an optimum quality level, which can be used to maximize the consumer surplus simultaneously with the manufacturer's profit. This model shows that depending on whether the original or actual product quality level is higher or lower than the optimum, either the manufacturer or consumer benefits, without harming the other. It is also shown that the existence of an optimum quality level, explains the propensity of manufacturing companies to produce at the same quality level, within a given market segment. This results in a competition on prices, from which the consumers benefit.

2.3.13 Descriptive Models

Descriptive models emanated both from experience of individual designers and from studies carried out on how designs were created, that is, what processes, strategies, and problem solving methods designers used. These models usually emphasize the importance of generating one solution concept early in the process, thus reflecting the 'solution focussed' nature of design thinking - Cross[100]. The original solution goes through a process of analysis, evaluation, refinement (patching and repair) and development [100, 118,119].

In their paper, Finger and Dixon [99] discuss descriptive models from a different perspective and have identified the research work in this area along two main lines of :

(i) Research based on techniques from artificial intelligence such as protocol analysis. This involves systematically gathering data on how designers design. The activities of the designers are recorded as the design evolves and progresses. Here the designer is encouraged to think aloud.
(ii) Research based on modelling the cognitive process. The aim of this research is to build computer based cognitive models, which describe, simulate, emulate the mental processes and skills used by designers while creating a design. Cognitive models describe the processes that underlie the set of behaviours that constitute a skill. They are specified as a set of mechanisms with defined functionality; each mechanism is described as a process that can transform classes of input into classes of output. The models also specify the interactions among the mechanisms, and generate explanations and predictions about the skills being studied.

<u>Model by March</u>

The model of the design process proposed by March [120], draws on the work of the American philosopher Peirce on the three modes of reasoning, which are : deduction, induction and abduction (production). In rephrasing Peirce's remarks, rational designing is conceived as having three tasks :
(1) the creation of a novel composition - accomplished by Productive reasoning.
(2) the prediction of performance characteristics - accomplished by Deductive reasoning. (3) the accumulation of habitual notions and established values, an evolving typology - accomplished by Inductive reasoning.

Summarily, Production (abduction) creates; deduction predicts; induction evaluates.

In this model the design process begins with the first phase of productive reasoning, which draws on a preliminary statement of required characteristics, some presuppositions, or protomodel about a solution, in order to produce the first design proposal. From the design suppositions and established theory, the first design proposal is then deductively analysed in order to predict the expected performance characteristics. From the predicted performance characteristics, it is then possible to inductively evaluate further design possibilities or suppositions. This cycle is then repeated starting from a revised statement of characteristics, resulting in further refinements and/ or changes in the design proposal.

In this iterative procedure, it is assumed that certain characteristics are sought in a design to provide desired services, and that on the basis of previous knowledge and some general presuppositions or models of possibilities, a design proposal is put forward [120]. The PDI (Production/Deduction/Induction) model described above is shown in Figure 2.23. The figure shows a cyclic, iterative procedure, involving constant refinements and redefinitions of characteristics, design, and suppositions as the composition evolves. The model is considered as representing a critical learning process, in that statements inferred at later stages can be used to modify those used in previous stages and to stimulate other paths of exploration.

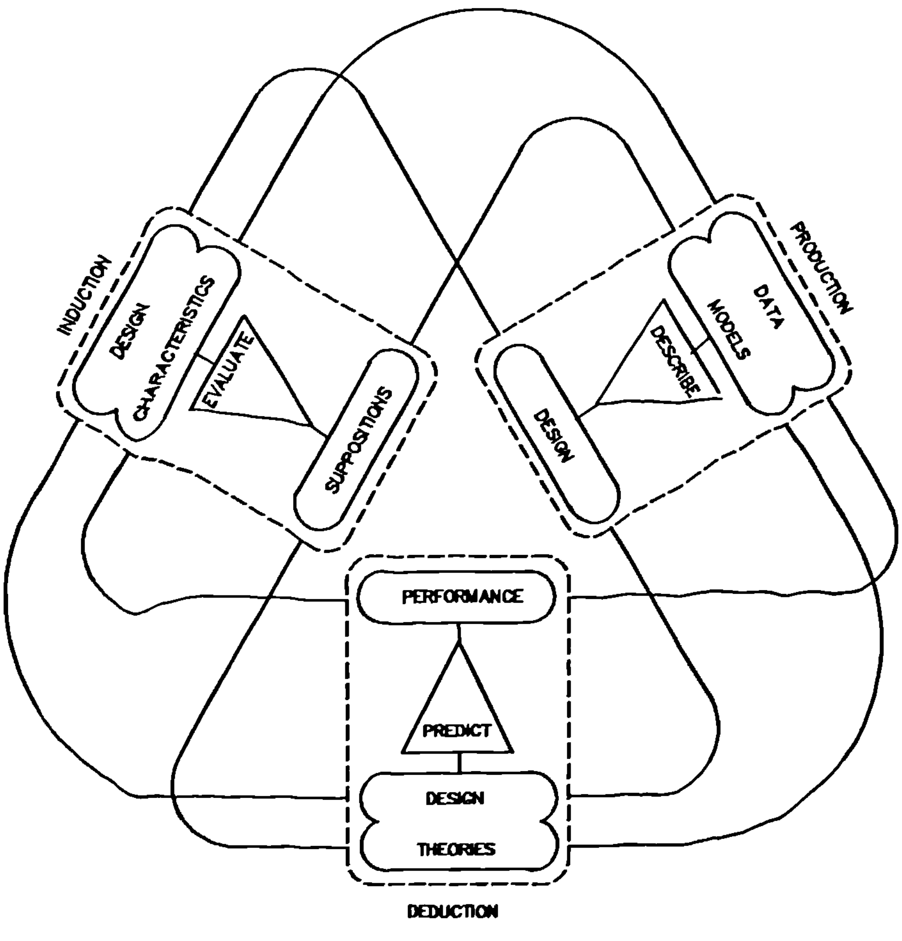


Figure 2.23 The Design Model by March

Model by Matchett

The approach to design as enunciated by Matchett [121,122], is also known as the Fundamental Design Method (FDM). The aim of this approach is "to enable a designer to perceive and to control the pattern of his/her thoughts and to relate this pattern more closely to all aspects of a design situation. The approach adopted by Matchett to design, is built around five thinking patterns. These are : (a) Thinking with outline strategies, (b) Thinking in parallel planes, (c) Thinking from several viewpoints, (d) Thinking with concepts and (e) Thinking with basic elements

Thinking with outline strategies

The idea here is (i) to be able to decide in advance what strategy (that is, a sequence or network of design actions or thoughts) is to be adopted in the design process, (ii) be able to compare what has been achieved in the design project, with what was planned, and (iii) be able to produce strategies for producing strategies.

Thinking in parallel planes

This consists of detached observation of the thoughts and actions of oneself and one's colleagues during the design project, and attention is focused upon the pattern of thought while designing.

Thinking from several viewpoints

Effort here is directed at the solution to the design problem instead of at the process of finding it. This involves stating of objectives by describing the product as something that provides a means of doing something else.

Thinking with concepts

This consists of imagining or drawing of geometric patterns that enable a designer to relate the Fundamental Design Method (FDM) checklists to the pattern of his/her own memories and thoughts. The main purpose of this is to provide the designer with a memorable pattern of the relationship between the design problem, the design process and the solution.

Thinking with basic elements

This thinking pattern is the most rational of the five modes of thinking. The use of basic elements is to make the designer aware of the large number of alternative actions that are open to him/her at each point of decision. These basic elements are considered under seven groups of : (1)*Decision Options* ,*(2) Judgement Options* , *(3) Strategic Options* , *(4) Tactical Options* , *(5) Relational Options* , *(6) Obstacle Options* and *(7) Concept Options* .

The primary roulette is intended to generate a variety of design alternatives. The actions here involves establishing how each part of a design can be eliminated, combined, standardised, transferred, modified or simplified. The secondary roulette is intended to ensure that all changes introduced are compatible with each other and with all the needs. The actions undertaken here involves establishing what effects, demands and restrictions each design item in a set will have on every other item in the set when compared using an interaction matrix. In using the Fundamental Design Method along with the necessary design checklists, the following procedure was recommended [122] : (a) Study the design situation, (b) Then identify provisionally the needs that the design is to satisfy, (c) Identify the primary functional need (that is the need that if not properly satisfied makes the fulfillment of all other needs pointless), (d) Explore alternative principles upon which a means of satisfying the primary need can be based, (e) Complete an outline of a design capable of satisfying both the primary and secondary needs, (f) Review the functional effectiveness of this design, and (g) Review the material and work content in producing the design as well as the component quality.

<u>Model by Gero - Evolutionary Design Model</u> [123]

This model considers the design process as a series of transformations of one state of the design to another state, e.g. transforming function F, structure S, and behaviour B, into a design description D. The evolutionary model is formulated with due cognisance of the use environment E, the resulting product will be exposed to as well as the originating designer's intent I. The resulting model is shown in Figure 2.24 below along with the activities.

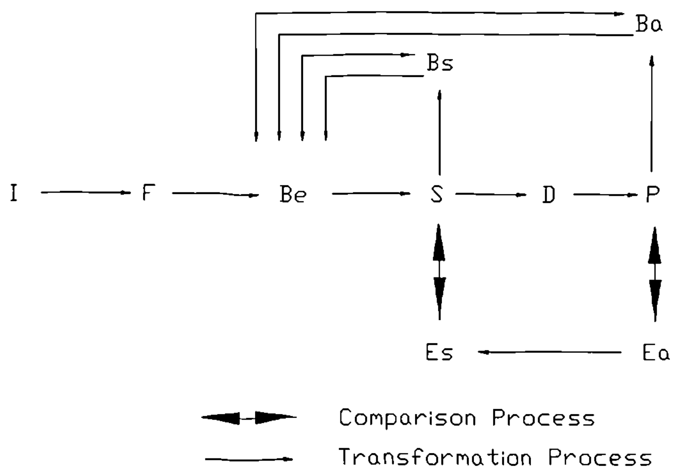


Figure 2.24 The Evolutionary Design Model

The activities in this model are :
formulation or design brief or specification - I ---> F
analysis - F ---> B_e, synthesis - B_e ---> S,
production of design description - S ---> D, manufacture of the product - D ---> P,
simulation - S <---> *Es*, real world interaction - P <---> *Ea*,
evaluation - Bs <---> Be (F), Ba <---> Be (F),
reformulation - Bs ---> Be, Ba ---> Be,
simulated structure performance - S (<---> Es) ---> Bs
actual product performance - P (<---> Ea) ---> Ba.

Where I = designer's intent, F = function (purpose of product), S = Structure (configuration of product's constituents), D = design description, Bs = set of behaviours of structure, Be = set of expected behaviours, Ba = set of actual behaviour of the product, Ea = actual environment, Es = simulated environment, P = actual product. The complete evolutionary design process model then incorporates the cross-over mechanism C, which allows for 'cross-breeding' of different structures Sn from a population of structures to achieve a satisfactory design. The mutation mechanism M, is also added and used to mutate any unsatisfactory design structure S. Both the cross-over and mutation mechanisms are not mutually dependent and are not necessarily both applied at each step of the evolutionary design process. Designing in this model, is based on the concept of an iterative cyclic process of generation and refinement of partial solutions which are evaluated using a model of the environment.

2.3.14 Computational Design Models

Nevill [124] considers that computational models of design play two major roles. Firstly, that they are a necessary part of the development of more effective CAD tools, and secondly, that they play a role of supporting research into design theory and methodology. Computational models considered here focus on mapping function into structure, and which are intended for computer implementation. Within these models design is considered to be a process which maps an explicit set of requirements into a description of a physically realisable product which would satisy these requirements plus implicit requirements imposed by the domain/environment [124]. The development and use of computational models is usually influenced by a number of characteristics of design domains (buildings, circuits, machines, mechanisms, etc). Some of these characteristics include : (i) the natural separability or modularity of the domain, which might facilitate partitioning of the design into subproblems, (ii) the nature of the interactions between subproblems, (iii) the number, size, and complexity of natural subproblems and their solutions, (iv) the amount of a priori knowledge regarding subproblem (and thus solution) structure and interaction, (v) the nature of the problem specifications, (vi) the nature of the solution space and (vii) the nature of the knowledge of the domain available [124].

Models by Agogino et al [125]

The computational models developed by this group, were based on the following propositions of their theory of design, that :

(1) Design is a goal directed constrained activity and thus designers are optimisers with bounded resources, (2) Designers reason qualitatively, mathematically, and to some extent, numerically and (3) Effective interactive design systems must then

reason at all three levels (qualitatively, mathematically, and numerically) and incorporate both constraints and goals.

Figure 2.25 shows the overview of the integrated design framework, which represents their model. The starting point in the framework is the selection of an initial design prototype. The CODESIGNER system in the framework is used to aid the designer in selecting an initial conceptual design and identifying the relevant set of parameters for use in further analysis. The CODESIGNER system offers the designer a structured approach to documenting designs and organizing design information for easy retrieval by designers on the same project or by other future conceptual designers. CODESIGNER also relates a desired behaviour described in one language to other descriptions that effectively communicate to the designer the form of devices that manifest that behaviour. This enables a designer to work from for example, a transfer function, a lumped parameter model, a symbolic description, or other representation that captures the behaviour of interest into a text description, solid model, or other such description that informs the designer of a potential design option.

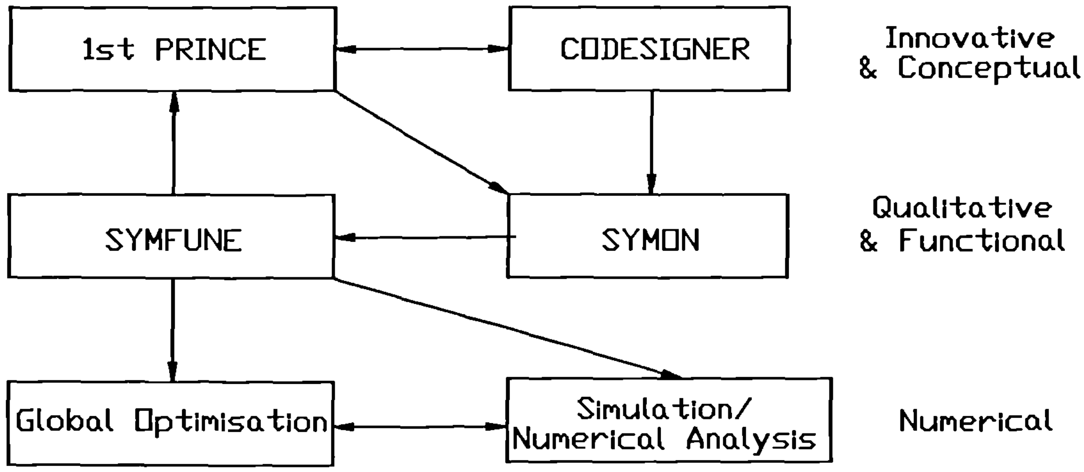


Figure 2.25 Optimally-Directed Design Architecture

The architecture of the CODESIGNER system as shown in Figure 2.26, consists of six elements. The **concept network** contains a collection of linked concepts and models. A concept represents a class of design prototypes such as assemblies, parts, or features, that has associated with it, any number of models which describe it qualitatively, mathematically, or numerically. Once a concept for a device has been chosen, models can then be used to select parameters to describe a device and to impose constraints on the parameters. The **model editors** are used to build new models from the primitives of a modelling language or from existing ones, and in combination with the **concept editor**, they allow the designer to add to or modify the concept network, enhancing it for future designers. The **design database** contains the set of parameters associated with concepts selected by the user, along with the constraints imposed by the models selected, representing the current state of a developing design. Parameters may then be selected based on design requirements, found by constraint propagation, qualitative optimisation or any number of numerical techniques. The constraint manager identifies opportunities for applying any of these strategies and maintains the consistency of the constraints.

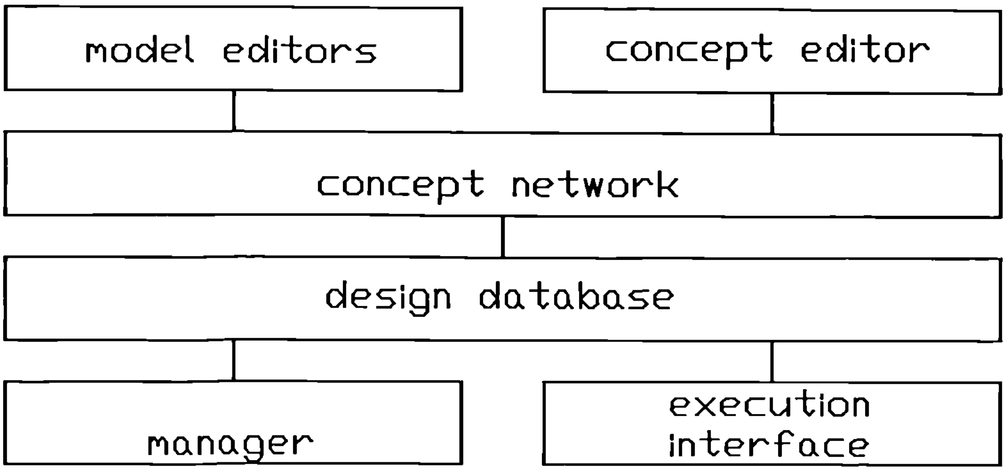


Figure 2.26 The CODESIGNER System Architecture

The CODESIGNER system provides a framework for conceptual design, relying on the designer, previous designers and the system builders to provide the prototypes that are combined to build a composite design. The composite design along with the attached equations and constraints are then ported to an analysis model for numerical simulation or to the SYMON-SYMFUNE systems for parametric or symbolic optimisation. The SYMON-SYMFUNE programs are used to perform qualitative and functional optimisation by means of monotonicity analysis and mathematical functional backsubstitution (using computer algebra).

In SYMON (SYmbolic MONotonicity analyser), a qualitative analysis of the design problem is performed utilising monotonicity analysis, a symbolic approach to non-linear optimisation problems which utilises qualitative first derivative information (i.e., the algebraic sign of the direction of change of a variable) to determine which constraints will be active or inactive for a possible solution to the optimisation problem, thus determining globally optimal search directions. The end result of the analyses is a revelation of potential sets of active constraints whose functional information can lead to optimally-behaved design solutions.

A mathematical functional analysis (implemented in SYMFUNE) can then be performed on each of the cases by backsubstituting the known, active information into the objective function and an optimally-behaved design can be chosen from the final set of designs. The solutions from SYMFUNE are represented as equations in computer algebra which can be machine translated to a variety of numerical analysis programs for detailed refinement. In the situation where the resulting design is unacceptable or new innovated designs are to be explored, 1stPRINCE (FIRST PRINciple Computational Evaluator) is used to create new designs.

The 1st PRINCE program [126] incorporates the qualitative, functional, and numerical levels of reasoning, and utilises monotonicity analysis as coded in the SYMON program to observe the sign of the algebraic first derivative of parameters and make qualitative decisions to direct globally optimal behaviour or detect flaws in the problem formulation. Based on the results of the monotonicity analysis, if deemed important, 1st PRINCE uses mathematical functional information as coded in the SYMFUNE program. When resulting designs are unacceptable or the designer wishes to investigate alternative designs, a higher level reasoning module in 1stPRINCE splits continuous integrals into smaller ranges creating novel design prototypes and allows for innovative designs. Algebraic solutions to the design problem are solved numerically to obtain specific solutions and quantitative comparisons between solutions. In this model, in order to handle nonconvex design optimisation problems, efforts are being directed at global optimisation methods based on stochastic techniques, that is, simulated annealing and variants of the multistart method. The multistart methods have been found to be more applicable to problems with continuous design variables, as opposed to simulated annealing which is generally more effectively limited to discrete combinatorial problems.

Models by Mostow et al [127]

The main thrust of this work is the attempt to develop a domain independent model of knowledge based design. The key research areas involved in this work include : (i) development of top down refinement models with constraint propagation for large scale, realistic problems, (ii) generation and use of knowledge for efficient control of the design process, (iii) adapting models to task and domain requirements , and (iv) knowledge compilation to produce efficient specialised procedures.

The work that has been done involve the integrated exploration of knowledge based, top down refinement with constraint propagation for (1) interactive NMOS digital circuit design (VEXED, BOGART), (2) controlling the design process (DONTE), (3) mechanical rotary power transmission design (MEET, DPMED, SPIKE, Floor-Planner), and (4) domain independent approaches (EVEXED) and knowledge compilation.

VEXED is used for the design of circuits, while BOGART is employed for replaying previous design decisions. DONTE is a program that is used for the intelligent control of the design process, using a top down refinement approach called goal directed planning to plan the design process. The program emphasizes the generation of information during the design process, and attempts to identify and focus on critical design areas (bottlenecks). MEET is the adaptation of VEXED for mechanical systems, and was applied in exploring the design of simple mechanical systems, such as the rotary power transmission system. DPMED uses a general iterative hill climbing approach for parameter value selection and can handle multiple design criteria, while SPIKE employes a bottom up heuristic guided search for selecting gear sequences and Floor-Planner uses a search model to fit gears into available space.

The EVEXED program involved the generalisation of VEXED and MEET into a domain independent shell. This work has progressed along two main lines viz: (a) a reconstruction of existing specialised algorithms and (b) the automatic transformation of design knowledge into an algorithm. Approaches include the use of constraints to guarantee generation of only acceptable alternatives, use of a generate, test and patch approach and the generation of abstract or rough solutions followed by refinement.

<u>Models by Dixon [128]</u>

This work focussed on design problems in which the design variables are known and their values are to be determined (parametric design) and for which an initial trial design is readily obtainable. The emphasis is on the determination of general concepts and methods which are domain independent. In the development of this project, two models were adopted, that is, *iterative redesign and iterative respecification*. The iterative redesign model was used to explore design problems which were manageable without decomposition, while the iterative respecification model was applied to design problems in which decomposition is given a priori.

<u>Iterative Redesign</u>

This project resulted in the development of the programs DOMINIC 1 and 2. The iterative redesign model incorporated in these programs, involves the evaluation of the current design with regard to performance parameters and constraints, the testing of results for acceptability, and revision of parameter values (redesign) based on use of dependencies between design variables and performance parameters. This is a hill climbing approach in which one performance parameter at a time is selected and its value modified. Dominic does not seek an optimal solution, but rather a satisfactory one. DOMINIC 1 solved problems in two different domains, but there was a need to improve its control. This led to the development of DOMINIC 2, which is capable of handling discrete variables and modifies its strategy based on problem solving progress. DOMINIC 2 monitors its problem solving performance by recording, for a number of iterative loops, (1) the design variable modified and how much, (2) the target performance parameter and the consequent satisfaction change, (3) active constraints, (4) the current design, and (5) the overall satisfaction level resulting. This record is then analysed to determine if the design has improved its level of satisfaction and if unproductive efforts are present.

Iterative Respecification

In this project, the concept of iterative improvement of designs, was extended to enable the handling of more complex problems which benefit from decomposition. The iterative respecification model was designed for parametric design of mechanical systems, in which a fixed prior hierarchical decomposition of the problem is assumed. This approach combines ideas from top down refinement, backtracking and iterative redesign, and explores the use of iteration between levels of a hierarchy. Communication within the model is limited to parent-child paths.

Dimensional Variable Expansion Model

Dimensional Variable Expansion (DVE) has been proposed by Cagan and Agogino [92] as a formal approach to design space expansion and design innovation. DVE is considered to be a rigorous approach which classifies expansive behaviour of variables and constraints, formalising a description of such that automation of constraint expansion is made possible. DVE begins with an initial problem formulation as an optimisation problem, with an objective function and a set of equality and inequality constraints, called primitive-prototype. The whole design information of the initial design concept, is modelled in the primitive-prototype.

Within the DVE framework, a design space consists of a set of design variables which are bounded by the set of equality and inequality constraints. These design variables are of five types. They are : (i) coordinate variables - which represent the co-ordinate axes of the physical object, (ii) dimensional variables - these are design variables associated with coordinate variables, and define the boundary or sub-regions of the geometrical design object, B (iii) a System variable - defined within a subspace of B as an integral quantity over a vector of variables formed from the elements of a subset of the design variables, (iv) region variables - these are defined over the space of B as charactersitics of a region which are not represented as integral quantities and (v) assignment variables - these are used to designate a global quantity or a temporary variable associated with a system variable, used in modelling a design problem.

DVE expands the design space along a variable which is critical (one which influences the objective function and which, when expanded, will create new variables which will also influence the objective function). Optimality conditions are then used to determine which variables are to be expanded, and the regions should be subjected to property modifications. During the process of expanding over a critical dimensional variable by DVE, the geometric design object is expanded into subregions and discontinuities are allowed across the coordinate axis, thus introducing new variables. It is these discontinuities which cause design innovation.

In DVE, the new designs or prototype generated, are usually optimally directed, where optimally directed design is an approach to design that attempts to determine optimal regions of the design space by directing the search toward improving the objectives and eliminating suboptimal or dominated regions of the design space. The DVE technique can also be used to expand the degrees of freedom of a design (that is, the number of variables which may be independently modified within a design problem), and designs with unique features can be derived.

DVE although rooted in optimisation theory, has been presented as a domain-independent technique with a formal theoretical framework that enables its general automation.

Case Based Design [129]

Design based on case-based reasoning (a well defined paradigm in artificial intelligence) is an emerging approach in design research. Case-based reasoning which is based on the premise that humans reason from specific experiences rather than by following a set of general guidelines, relates a current situation to the closest most specific experience in memory and uses the experience to solve the problem at hand. Hence case-based design can in fact be regarded as a memory-based approach. Key factors in case-based reasoning are the storage of cases as complete patterns of experiences including the reasoning process, the ability to be reminded of the most appropriate case and the application of that case to the current situation. Case-based reasoning uses the strategies of modification and repair to effect modifications to case applications when there is no exact match of the case with the current design situation. As a result of these modifications, new cases are produced which are variations of the previous case, or in some extreme circumstances, as entirely new cases arising from severe modifications. Case-based reasoning thus incorporates a learning capacity in the form of new cases being incorporated into a dynamic case base. Using existing cases of designs involves a searching process based on indexing cases with regards to various factors such as goals and attributes. The efficiency of the search process usually depends on the efficiency of the indexing cases. Retrieving a case is based on pattern matching, that is, matching a required pattern of requirements to an existing set.

Case-based design involves key processes of *search, match, retrieve, select, modify, repair and store.*

Search - This is the process of searching the case base for an appropriate design case based on a given problem description or requirements including functions to be achieved, required behaviour performances, the design environment and constraints on values of structure variables. The searching process can be sequential, parallel or direct using an indexing mechanism. Indexing must be done on the function, behaviour, structure and context features of the design.

Match - A match occurs when the search criteria features corresponds to one or more cases in the case base. The appropriate cases then represent candidates for consideration. Perfect matching which rarely occurs represent the situation where the required features are found exactly in a case. The likely situation of partial matching occurs when some of the features are matched or the features are matched to some degree.

Retrieve - When a particular case or cases from the case base matches the search criteria features to some defined degree, they are then retrieved for consideration. This may involve a display of these cases to the designer or users for perusal and consideration.

Select - This is performed to select a single case as the basis for determining the design solution. If only part of a design case is required, then several design cases may be selected, and the necessary parts of each extracted. In either of the above situations, the 'best' matching design case should be selected, on the basis of the most similar or the most useful match. The selection process can be performed by the system or by the users after consideration of an appropriate set of candidates retrieved by the system. Selection by the system based on partial matching entails factors such as the importance of the features matched as well as how close they are matched.

Modify - This occurs in situations where a selected design case does not match the design requirements sufficiently, thus requiring some modifications. This involves the replacement of variables with other variables or simply the alteration of some values of variables.

Repair - In many design situations, the performance of a modification to an existing design case based on substitution of variables or modification of values, usually causes some performance failure in some other behaviour or function. In this situation, two main strategies can be employed. Firstly, an alternative design case can be selected based on the new information known regarding the necessity for modifications and the effects of modifications, or secondly the current selected design case is modified in such a way as to make it acceptable. This second approach is what is known as the process of repair in case-based reasoning.

Store - The completion of the necessary modifications and repairs to a selected design case, results in the generation of a new design case. This new design case, when considered to be sufficiently important as a design experience different from existing design cases, is then stored in the case base with appropriate indexing. In other situations where the failures of solutions are seen as important pieces of information to the anticipation of future problems, such cases can also be stored or noted in the design case base.

2.3.15 Design Methods

During the different phases of a design project and through the various stages of the design process, a number of design aids, tools and support systems are used, in order to arrive at a realizable product and/or process. These tools and aids are what are generally regarded as design methods. Design methods generally help to formalize and systematize activities within the design process and externalize design thinking, that is, they try to get the designers' thoughts and thinking processes out of the head into charts and diagrams[100]. Hubka [130] defines a design method as "any system of methodical rules and directives that aim to determine the designer's manner of proceeding to perform a particular design activity, and regulate the collaboration with available technical means ...". Design methods were also considered to exhibit certain characteristics in terms of their usage, such as : the goals the methods serve, their general applicability, conditions under which they can be used, whether the methods are intended for single designers or for design teams, their origins, how they function (modus operandi) and the time demanded by the methods. Taking into account the above characteristics, design methods can be classified under the following broad headings :

(i) Methods intended to provide basic improvements in the way designers work, their effectiveness as well as eliminating errors in the thought process.
(ii) Methods that act on the creative characteristics of the human being.
(iii) Methods that attempt to describe and master the problem situation by means of strict logic and mathematics.
(iv) Methods that prescribe methodical rules and regulations, which can significantly increase the overall probability of success.
(v) Methods based particularly on the knowledge of the artifact being designed.
(vi) Methods which encourage the use of technical means and aids, and aim towards automation of that part of the design process.
(vii) Combinations of the above methods appropriate to the existing situation.

Jones[122] in his book, gives a description of thirty five design methods. Over the years, several other design methods have emerged, which are currently being used for technological advancement of products and/or processes. These methods are shown in the proceeding subsections.

<u>**Design Methods by Jones**</u>

Prefabricated Methods

<u>Systematic Search</u>
Aims : To solve design problems with logical certainty.

Value Analysis
Aims : To increase the rate at which designing and manufacturing organizations learn to reduce the cost of a product.

Systems Engineering
Aims : To achieve internal compatibility between the components of a system and external compatibility between a system and its environment.

Man - Machine System Designing
Aims : To achieve internal compatibility between the human and machine components of a system and external compatibility between the system and the environment in which it operates.

Boundary Searching
Aims : To find limits within which acceptable solutions lie.

Page's Cumulative Strategy
Aims : To increase the amount of design effort that is spent on analysis and evaluation

CASA (Collaborative Strategy for Adaptable Architecture)
Aims : To enable everyone concerned with the designing of a building to influence decisions that affect both the adaptability of the building and the compatibility of its components.

Strategy Control

Strategy Switching
Aims : To permit spontaneous thinking to influence planned thinking and vice-versa.

Matchett's Fundamental Design Method (FDM)
Aims : To enable a designer to perceive and to control the pattern of his thoughts and to relate this pattern more closely to all aspects of a design situation.

Methods of Exploring Design Situations

Stating Objectives
Aims : To identify external conditions with which the design must be compatible.

Literature Searching
Aims : To find published information that can favourably influence the designers' output and that can be obtained without unacceptable cost and delay.

Searching for Visual Inconsistencies
Aims : To find directions in which to search for design improvements.

Interviewing Users

Aims : To elicit information that is known only to users of the product or system in question.

Questionnaires

Aims : To collect usable information from the members of a large population.

Investigating User Behaviour

Aims : To explore the behaviour patterns and to predict the performance limits of potential users of a new design.

Systemic Testing

Aims : To identify actions that are capable of bringing about desired changes in situations that are too complicated to understand.

Selecting Scales of Measurement

Aims : To relate measurements and calculations to the uncertainties of observation to the costs of data collecting and to the objectives of the design project.

Data Logging and Data Reduction

Aims : To infer and to make visible patterns of behaviour upon which critical design decisions depend.

Method of Searching for Ideas

Brainstorming

Aims : To stimulate a group of people to produce many ideas quickly.

Synectics

Aims : To direct the spontaneous activity of the brain and the nervous system towards the exploration and transformation of design problems.

Removing Mental Blocks

Aims : To find new directions of search when the apparent search space has yielded no wholly acceptable solution.

Morphological Charts

Aims : To widen the area of search for solutions to a design problem.

Methods of Exploring Problem Structure

Interaction Matrix
Aims : To permit a systematic search for connections between elements within a problem.

Interaction Net
Aims : To display the pattern of connections between elements within a design problem.

AIDA (Analysis of Interconnected Decision Areas)
Aims : To identify and to evaluate all the compatible sets of sub-solutions to a design problem.

System Transformation
Aims : To find ways of transforming an unsatisfactory system so as to remove its inherent faults.

Innovation by Boundary Shifting
Aims : To shift the boundary of an unsolved design problem so that outside resources can be used to solve it.

Functional Innovation
Aims : To find a radically new design capable of creating new patterns of behaviour and demand.

Alexander's Method of Determining Components
Aims : To find the right physical components of a physical structure such that each component can be altered independently to suit future changes in the environment.

Classification of Design Information
Aims : To split a design problem into manageable parts

Methods of Evaluation

Checklists
Aims : To enable designers to use knowledge of requirements that have been found to be relevant in similar situations.

Selecting Criteria
Aims : To decide how an acceptable design is to be recognized.

Ranking and Weighting
Aims : To compare a set of alternative designs using a common scale of measurement

Specification Writing
Aims : To describe an acceptable outcome for designing that has yet to be done.

Quirk's Reliability Index

Aims : To enable inexperienced designers to identify unreliable components without testing.

Other Design Methods

These include the following : (1) Design Axioms, (2) Design for Manufacture (3) Design for Assembly, (4) Design for Disassembly, (5) Design for the Environment, (6) Design for Cost, (7) Design for Maintenance, (8) Design for Reliability, (9) Design for Testing, (10) Design for Serviceability, (11) Failure Mode and Effect Analysis (FMEA), (12) Robust Engineering Design, (13) Taguchi Methods, (14) Manufacturing Process Design Rules, (15) Finite Element Analysis (FEA), (16) Group Technology, (17) Fault Tree Analysis (FTA), (18) Computer Aided Modelling (Solid Body Modelling), (19) Sketching Input, (20) Design Retrieval (Design Databases)

2.3.16 A Review of Computer Based Design Systems

In the recent past, there has been several issues raised in the design and research community, about the limitations of currently available geometry based computer aided design (CAD) systems, and their failure to accomodate other aspects of the design life-cycle ranging from problem definition and specifications right down to production planning. Preliminary work in this regard (although limited) has focussed on developing more integrated and robust design systems that can support not only geometric modelling, but also accomodate various design models, design methods and techniques as well as providing sufficient flexibility for the designer to innovate and be creative. A major intention of such systems, is for them to be able to handle varying forms of design information in addititon to geometric data. The proceeding sections will involve the discussions on several of these systems including those being developed in a number of the engineering design centres within the UK, who are sponsored by the Science and Engineering Research Council (SERC) Design Initiative.

Integrated Design Environment

The Engineering Design Centre at the University of Newcastle are concerned with the generic design processes and their integration, with particular reference to large made-to-order (MTO) products, such as power plants, offshore facilities and aerospace products [131]. Within the design environment, exists, several functionality and resource components, consisting of (i) Generic Functionality - which are applicable throughout the design process, regardless of the specific design task, e.g. design process management, (ii) Specific Functionality - relates to specific tasks in the design process, e.g. solid modelling, (iii) Tool Resource - which are third party software employed to deliver enabling technology for addressing specific design tasks, (iv) External Information Resource - such as company specific databases and (v) Programming Language Resource - which include languages like FORTRAN, Modula-2, Prolog and Expert systems shells. In order to facilitate the integrated use of these components in the design environment, as well as implement it in a concurrent and distributed architecture, the integrated system architecture, consisting of four layers as shown in Figure 2.27, has been established. This integrated platform is based on a networked workstation environment, using standard application programmer's interface (API), portable tools and standard programming language bindings, in an open architecture. The four layers are : (i) The Distributed Design Protocol, (ii) The Design Process Management, (iii) The Design Functionality and (iv) The Design Repository.

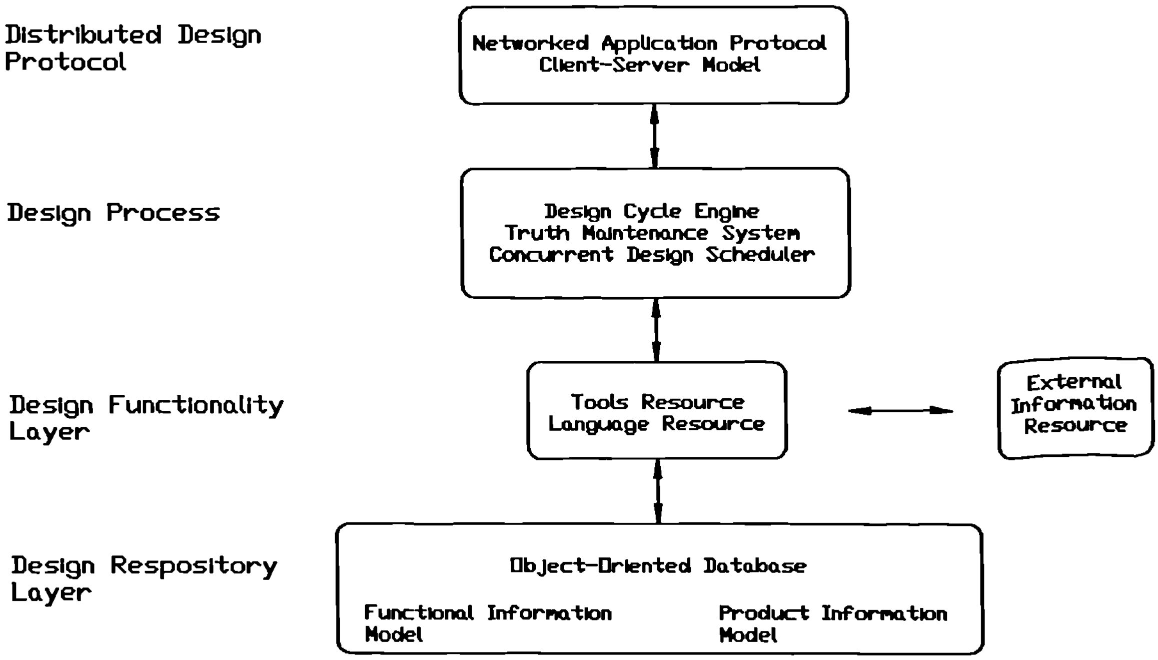


Figure 2.27 Integrated Design Environment : A Layered Architecture

The Distributed Design Protocol layer enables communication between computationally (and physically) distributed design agents. It also enables the inter-process communication between the design activities and the product database.

The Design Process Management layer is used to instantiate and manage the design process. It consists of the design cycle engine (DCE), an assumption-based truth maintenance system (ATMS) and an event scheduler. The DCE is a finite state machine representing the possible design states and transition paths between them. It is used to model and instantiate the design process and its underlying design activities. The DCE is implemented using Prolog and C. The ATMS is used to perform non-monotonic reasoning on the truth relations between design decisions and their consequences in the design content [132]. The ATMS is a directed acyclic graph of nodes representing design assumptions and decisions, with directed arcs between nodes being the dependency vectors. This module is based on the Edinburgh simple ATMS [133]. The event scheduler is used to schedule the design activities, and to maintain temporal relations between them and to keep track of the current state of the design process. The blackboard model and other approaches are being considered as possible ways of implementing the scheduler.

The Design Functionality layer includes proprietary software and programming languages, which are interfaced with the external information resource. It also includes expert and knowledge based systems.

The Design Repository layer encapsulates the design content of a product in terms of a product information model and a functional information model (which reflects the task-specific views of product information). This layer is being implemented via an object-oriented database, in addition to a C+ + class library. Product information modelling is carried out based on the ISO 10303 Standard for the Exchange of Product Model Data (STEP).

The Integrated Design Framework

The design system under development at the Cambridge University Engineering Design Centre, is called The Integrated Design Framework (IDF) [134]. This system is described as an open and flexible computer environment, which is based on Framework Support Modules (FSMs). These FSMs are being developed to aid designers in specifying requirements, synthesising and evaluating solution concepts, performing embodiment design, optimising the configuration and manufacturing their products. The architecture for this system is shown in Figure 2.28.

A Blackboard system exists within the IDF, whose role is to provide a structured communication media between the designer, specialist design tools (FSMs) and an evolving product model, which will record the product and process data generated, while using the FSMs. The Blackboard system contains a control and a domain blackboard. The control blackboard supports problem solving, and the entries on the control blackboard represent what actions are desirable, feasible and actually performed at each point in the design process. Entries on the domain blackboard represent the development of design objects generated during the design process.

The design strategy Framework Support Modules constitute IDF's design process model, based on design matrices developed from prescriptive models of design and descriptive studies of design. The design matrices correspond with the context of the project within different stages and enable tracking of the design history and its rationale. Each FSM is linked into IDF by an FSM agent program that matches the FSM shell data against Blackboard states and activates the FSM whenever there is a match. One of the aspects of IDF development, is the linkage between control/domain Blackboards and the FSMs. This linkage defines the design context in which the FSM would be best applied by engineers for a problem state.

The basic design cycle of the sysytem is driven by a scheduling algorithm that (i) identifies FSMs triggered by the current design state and (2) maps FSMs output onto the control/domain Blackboard partition to produce a new design state at the next design cycle. The model of communication within the IDF is based on the IDEFO model, developed within the ICAM project of the US Air Force [135]. The model provides a description of system modules in terms of a hierarchy of functions, which are decisions, actions or activities undertaken within the modules.

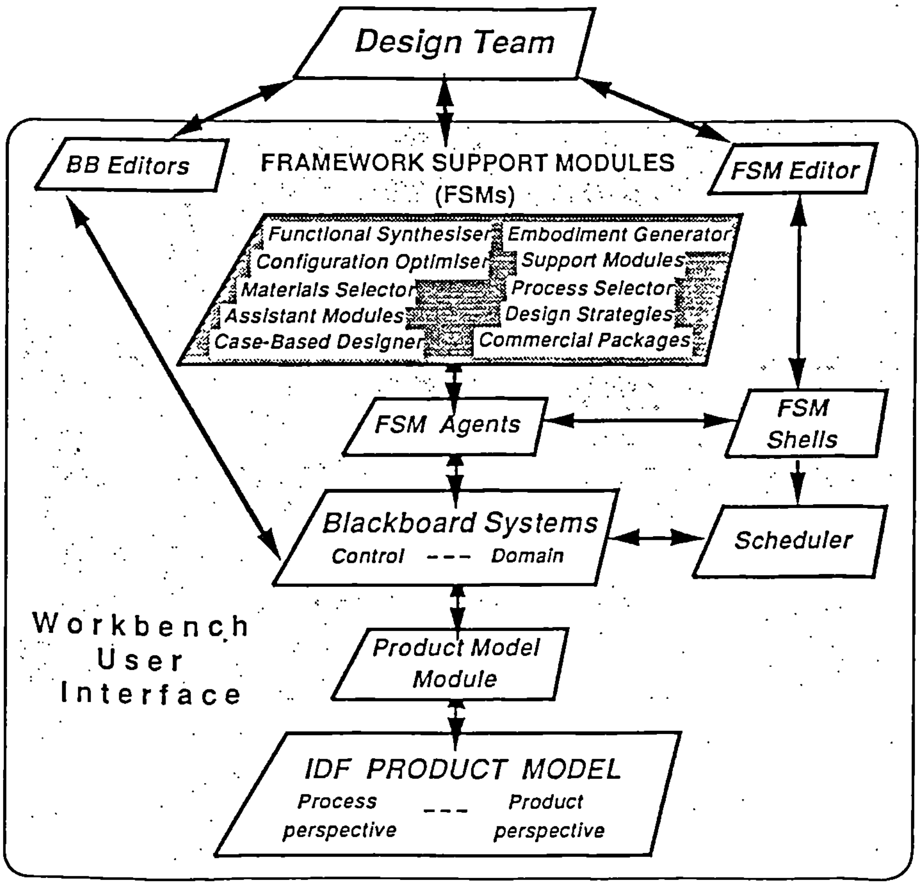


Figure 2.28 Architecture of the Integrated Design Framework

The Scheme builder

"SCHEMEBUILDER", the design system under development at Lancaster University Engineering Design Centre [136, 137], is a conceptual design tool aimed at guiding designers through the vast range of design options available to them. The aim is to facilitate the exploration of alternative conceptual schemes with an appropriate allocation of function between mechanical, electronic and software elements. Schemebuilder aims to provide for the creation of a model of the system to be designed, whilst giving advice on appropriate means. The solution approach adopted uses Bond Graphs to classify individual components both by their function and the type of ports they possess. This structure enables a user to select components on the basis of their function and/or by specifying required port types and attributes. Schemebuilder presently recognises two fundamental port types, namely :

signal ports - these communicate either significant generalised 'effort' or 'flow' but not both

power ports - these are capable of simultaneously communicating significant effort' and 'flow'

The Schemebuilder also contains the Browser/Selector facility, which is used to search the component database using functional specifications and/or port types as an index. This facility also allows the application of various filters which are used to exclude non-matching components, disqualification of particular component classes and the ranking of alternatives by cost or weight. Schemebuilder currently contains about 150 mechatronic component types, modelled parametrically with details of function, spatial requirements, cost and weight, as well as in terms of the power and signal ports that they possess. These components can be displayed graphically to a user and may be browsed and selected using the browser/selector facility. Selected components can then be instantiated and placed on a "Building Site" window. In this window, the positions of individual components and their ports can then be manipulated and connections made between them, to form complete schemes. Associated with the Schemebuilder is the Layout module, which supports the preliminary embodiment phase of design. It is used to quickly generate 3D solid geometries of selected schemes from the 'building site'. It uses stored component drawings with dimensions given by parametric formulae.

In order to provide help on using Schemebuilder, document component properties and use, advise on principles of good design as well as record user/designer actions for subsequent playback and analysis, three additional modules were developed, that is, the hypertext user interface, the userlog and the design advice expert system. The hypertext user interface is used to visually display the facilities for advice, replay, documentation and help, using a stack of screens (cards) containing text and graphic entities as well as links to other cards. The userlog stores a record of events, which are sent automatically to it as messages (which are generated by the designer's actions in the Schemebuilder system). This enables an action replay on a graphicmap flagging key design decisions with text messages. The design advice expert system consists of a knowledge-based system capturing the expertise of both in-house and external engineers in the form of design principles and rules. The system architecture for the schemebuilder is shown in Figure 2.29.

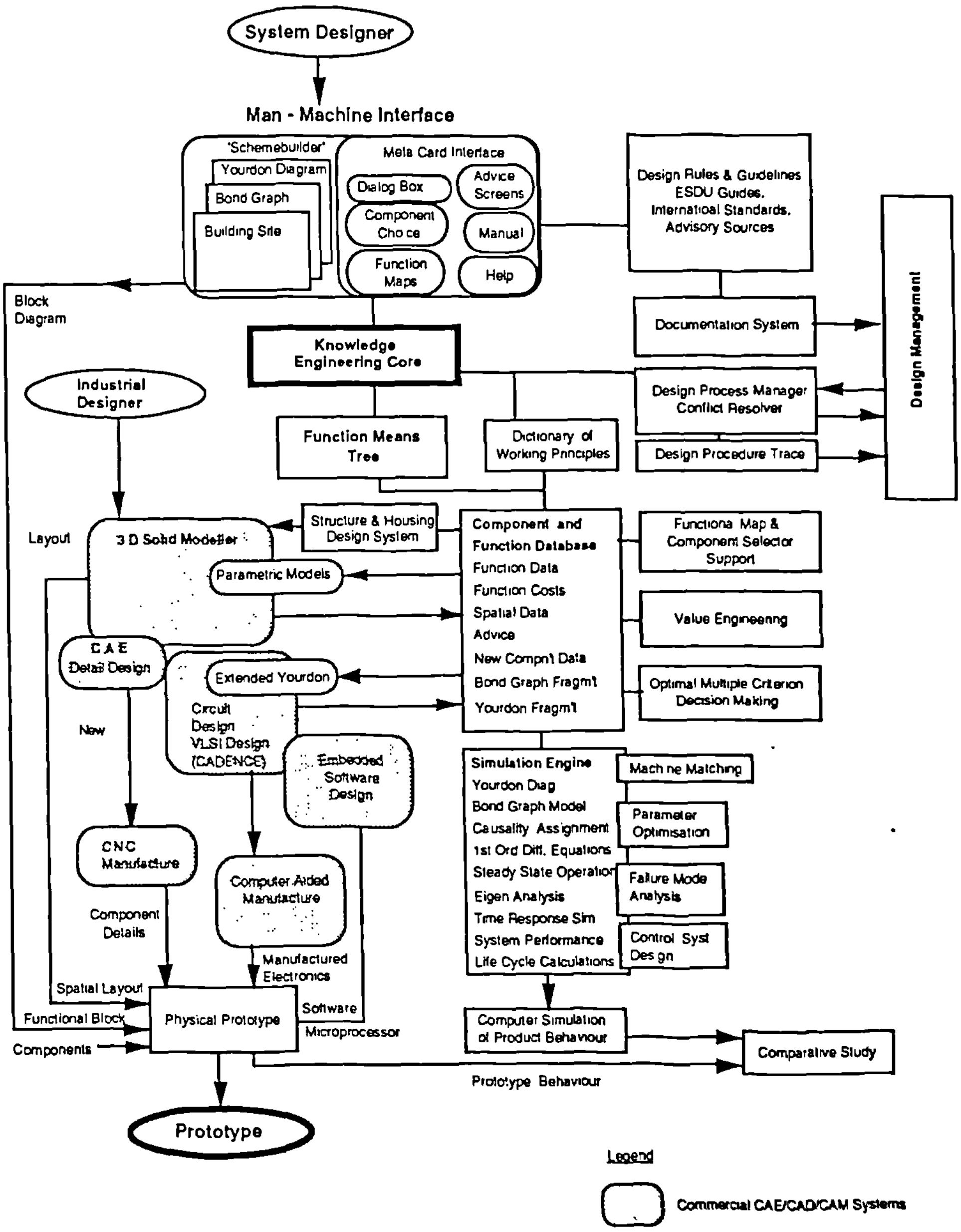


Figure 2.29 Architecture of the Schemebuilder Environment

Anticipated application areas of the Schemebuilder system include inter-alia: assisting in the production of specifications, providing advice on available technologies, producing system models for dynamic simulation, monitoring system integrity, checking for continuity and matching and design and modelling of casings and support systems structures. This will involve the integration of several existing software tools. The Schemebuilder system runs on the Sun workstation and it is being developed within the UNIX/X- window environment. The software development environment chosen is the Common Lisp based Knowledge Engineering Environment, KEE, a large artificial intelligence shell.

Vehicles Knowledge-Based Design Environment

The Vehicles design environment [138] which is being developed by the Aerospace corporation, Los Angeles, California, USA, is a knowledge-based system with an interactive environment built to enhance, assist, simplify, and expedite design activities under the guidance of designers. Vehicles has been built and used in parallel with traditionally coded simulations, to study tradeoffs during the conceptual design phase of several different satellite-architecture studies. It is being developed to support varied styles of design, handle and evaluate multiple designs, provide meaningful status reports on the results obtained and on the degree of completion of a design, provide a variety of analysis tools and to create an open and extensible architecture in which new models, tools, and design concepts may be added. The analysis tools include : equation solvers, sensitivity analysis tools, parametric analysis and what-if analysis tools. The design system also contains subsystem sizing and performance models and a historical database (of satellites that have already been built).

The kernel of Vehicles is written in Quintus Prolog and C on a Sun workstation. The user interface is being built using X-windows, C + +, and the Interviews widget set. It also has links to existing software tools (in Fortran and C) for graphics and engineering analysis.

An Integrated CAD System [139]

This system was developed based on a model for the process of machine design, taking into account the requirements of Computer-aided design (CAD), and consists of three main elements, that is, Product-defining models (PDM) that support modelling of product properties, Product-defining data (PDD) that adjust the overall, general models to an appropriate size, given the design problem and Operational principles (OP) that select the design methods with all the necessary information about the process.

The resulting integrated and flexible CAD system consists of four main modules, which are : (i) The task analysis processor (AAP), (ii) The solution coordination processor (LKP), (iii) The design management system (KLS) and (iv) The database (DB). The flexible and continuous CAD system is shown in Figure 2.30.

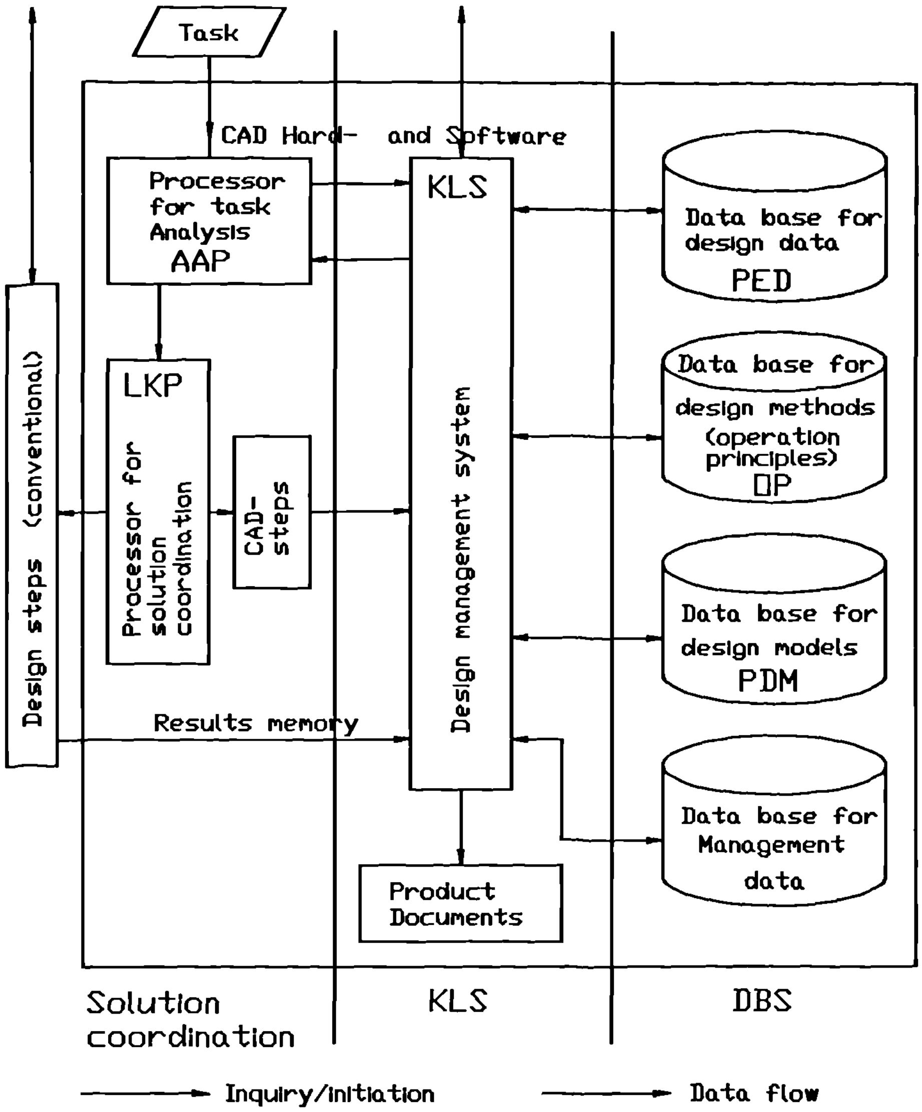


Figure 2.30 The Structure for a System of Flexible and Continuous CAD

The Task Analysis Processor (AAP)

This supports the search for suitable specifications and available information of recently designed systems, with regard to the current problem. It also facilitates the presentation of information about likely design problems stored in the database, as well as the determination of necessary tasks for the conceptual design, embodiment design, refined design, and final layout.

The Solution Coordination Processor (LKP).

This module enables the design process by utlising the information gathered by the task analysis processor. The LKP is the tool that informs the designer about the installed hardware and software, about the preparation of input data, and about the operation of the computer.

The Design Management System (KLS)

This module is the kernel of the integrated CAD system, and it coordinates the design models (PDM), the design data (PDD) and the principles of the design operation (OP). The structure of the design management system consists of the following components: The *structural part* which organises the sequence of necessary design steps. It is established by the operation structure (OS). The OS is a list of clues, referring to the design methods, stored in the appropriate database.

The *operational part* which coordinates the interaction between methods, models, and data for a certain design step, and is established by the operation processor (OPR). The OPR is the program that organises all the components of the design process during the current design step.

The *Database* (DB) - During the design process, the data generated and accumulated are stored in a general database, which is subdivided into the following subdatabases:
(1) Database for design data that stores all process external data (PED) such as data about: (a) standards that influence geometry, (b) geometry and material of semi-manufactured and finished products, (c) geometric and material properties of standard components, purchased components, etc, and (d) text and graphic information about standards, guidelines, regulations, rules, etc.
(2) Database for design methods that stores all design methods
(3) Database for design models that stores all design models
(4) Database for management data.

The primary goal of this system, is to ensure continuous design with data flowing from one solution or design step to the next.

The Design System MFK [140]

The primary objective in the development of this system, was to on the basis of a traditional CAD system, support the designer/design team through an object-orientated component description offering an integrated knowledge-based analysis of the component. The architecture of this system is shown in Figure 2.31. The system consists of an information-generating synthesis part and an information processing analysis part. Also included is the component model module, which adjoins both the synthesis and the analysis parts. Contained within the component model, is the product defining data. Connected to the design system, is a CAD system, through an interactive/procedural interface.

In the synthesis part of the system, the designer is offered an object-orientated description of components, by allowing the access of design elements through a design module. These description elements are structured hierarchically, in the form of basic design elements, design elements and building blocks. The functions used to manipulate the design elements are available within the design module, which are : generate, modify, delete, load and save.

In the analysis part, the information modules allow access to an extensive knowledge base, which contains facts, methods and experience that are necessary for the completion of an individual problem analysis. The information modules referred to above provide three main functionalities of diagnosis, consultancy and correction. The diagnosis module is used by the system to assert whether the design rules have been violated or not. The consultancy module is used to advise the designer, based on the diagnosed errors, about several alternative suggestions for getting rid of the errors. The correction module is used to automatically perform any correction suggested and approved by the designer.

The component model, which is the interface between the synthesis and the analysis parts, contains all the product defining data specified by the designer in relation to geometry, technology, function and organisation. The component model, is in the form of hierarchically structured parameter data, which is the abstract representation of the real three-dimensional component. The component model is automatically generated during the synthesis procedure.

This design system allows different types of analysis to be performed, such as : design for production, tolerance analysis, cost and stress calculations as well as component search.

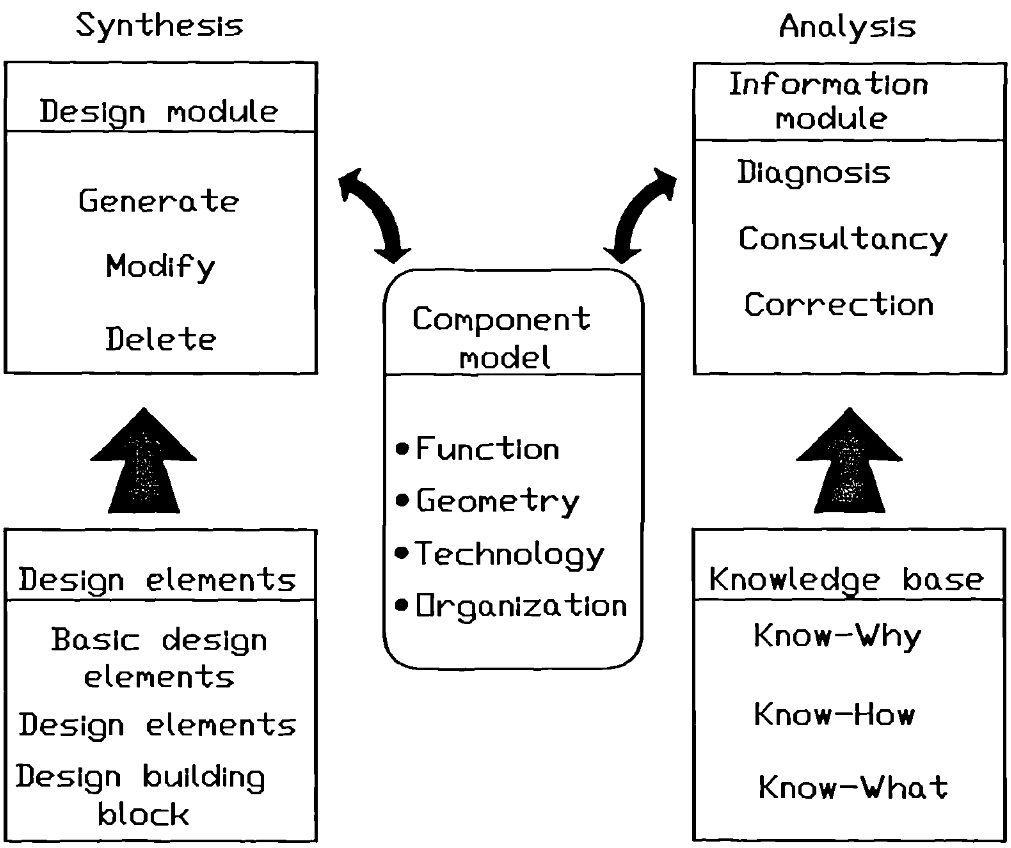


Figure 2.31 Architecture of the Design System MFK

Intelligent Integrated Interactive CAD System

The philosophy of the Intelligent Integrated Interactive CAD (IIICAD) system [141], is that it should support the designer/design team in the entire design process using unified models with rich functionalities for various design activities. The system should also have models of the design object which should exhibit maximum similarity to the designer's own images about them.

The architecture of this system and its components is shown in Figure 2.32. The supervisor SPV is at the core of IIICAD and controls all the information flow. It also adds intelligence to the system by comparing user actions with scenarios which describe standard design procedures, as well as performing error handling when necessary. The Integrated Data Description Schema (IDDS) regiments the databases and knowledge bases and relieves the user from the burden of specifying where and how to store and retrieve data. The knowledge base (KB) is divided into two parts: an object store, where all design artifacts or objects, and their parts as well as their internal structure are stored, and a rule store, which contains the facts and relations between the objects defined in terms of definite program clauses as in prolog. The rule store is used most dynamically during the design process. The knowledge base can contain both procedural and declarative descriptions of the object being designed.

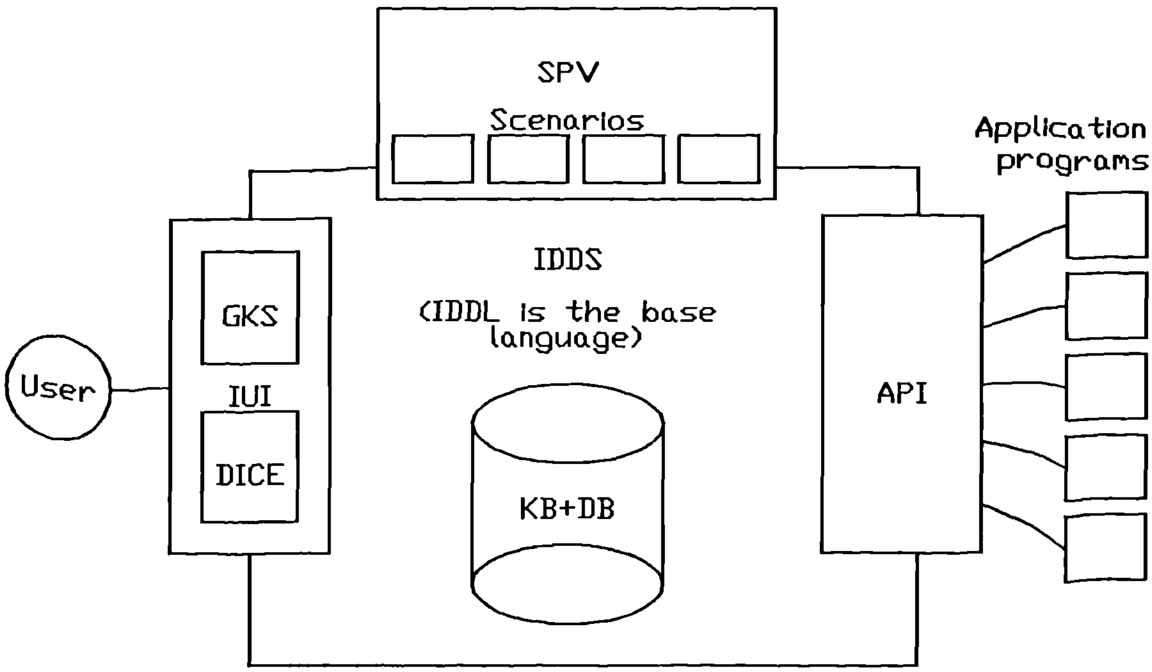


Figure 2.32 IIICAD Architecture

The IDDS has a kernel language called the Integrated Data Description Language (IDDL), which is used by all the system elements. IDDL is the means used to code the design knowledge and the design objects in order to guarantee integrated descriptions systemwide. IDDL is based on logic and the concepts of knowledge engineering.

As shown in the figure, IIICAD also has a high level interface called the Intelligent User Interface (IUI) which is also driven by scenarios written in IDDL. IUI accepts messages from the other subsystems and sends them to other lower level interface systems such as DICE (Dialogue Cells) and GKS (Graphical Kernel System).

The Application Interface (API) is used to secure the mapping between the central model descriptions about a design object and the individual models used by application programs. Such application programs can include programs for : (a) conceptual designs and handling of vague or fuzzy information, (b) consultation and problem solving for engineering applications, (c) basic/detailed designs including geometric modelling, (d) finite element analysis and other engineering analysis activities and (e) product modelling.

<u>Design Fusion Project - CMU EDRC [142,143]</u>

The primary goal of this work, is to infuse knowledge of downstream activities of product development, into the upstream design process so that designs can be generated rapidly and correctly. The design space is hence viewed as a multi-dimensional space in which each dimension represents a different life-cycle objective such as fabrication, testing, serviceability and reliability. In practice, these various dimensions may interact or conflict. This system thus aims to provide an intelligent aid to the designer which would help in the understanding of the interactions and trade-offs among these different, and conflicting requirements of a product or system. This computer based system surrounds the DESIGNER with expert modules that provide continuous feedback based on incremental analysis of the design as it evolves. The expert modules are called PERSPECTIVES and can be used to generate : (i) comments on the design, (ii) information that becomes part of the design,(iii) and portions of the geometry. These perspectives represent a collection of modules (fabrication, assembly, etc) that interact with one another and with the DESIGNER.

The Design Fusion system is a computer based, blackboard architecture that can be used by a single designer. The system is based on three underlying concepts : integrating life-cycle concerns through the use of views from multiple perspectives (manufacture, distribution, maintenance, etc.); representing the design space at different levels of abstraction and granularity through the use of features (attributes that characterise a design from the viewpoint of any perspective); and using constraints to guide the design [144]. It integrates the perspectives (expert modules) around a dynamic, shared representation of the design. The shared representation includes geometric model of the design as well as the features, the constraints, and the design record. The design record contains the design decisions that led to the creation of a constraint or feature. The perspectives are co-ordinated through a blackboard architecture which uses a user driven control structure. Thesystem architecture for the design fusion system is shown in Figure 2.33.

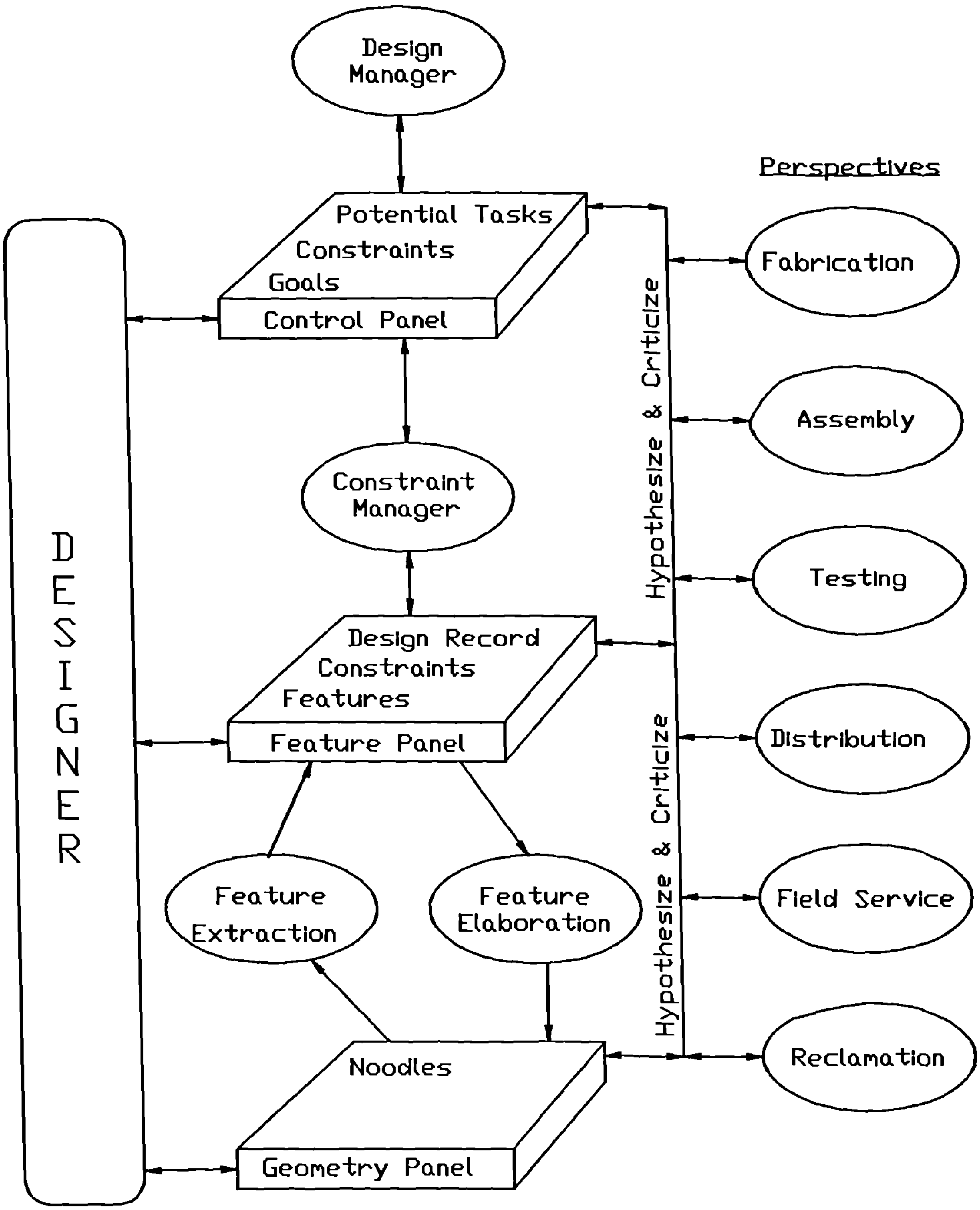


Figure 2.33 The Design Fusion System Architecture

2.3.17 Summary

This report represents a comprehensive survey carried out on design philosophies, models, methods and systems. The discussions examined different definitions of design, various features of the design process, types of design and product classifications, the concept of design goals as well as design philosophies, models, methods and systems. From the survey, it is apparent that, there are as many approaches to design, as the number of engineers and designers. These different approaches to design, represent the various schools of thought in the design community, and they are expressed in the form of design models, which are either prescriptive or descriptive. The prescriptive models represent the design process in phases and/or stages, and tend to prescribe how the design process should be carried out, in an algorithmic and systematic way. They also looked at the design process from a global perspective, while progressing from an abstract (conceptual or feasibility) phase to a concrete (detailed) phase. The descriptive models on the other hand were based on the strategies used by designers, and focuses on designers' actions and activities during the design process. These models usually encourage early generation of single solutions, reflecting the solution focussed strategy, as well as involving many iterative activities in the design process. A majority of the models, especially the prescriptive ones, favoured the establishment of design requirements in a solution neutral form, at the first stage of the design process.

The survey also included a listing of the design methods used in most of the models, as well as currently emerging ones. In general, these design methods consisted of both manual and computer based design aids, tools, techniques and support systems, used during the design process, to : (i) arrive at a realizable product, (ii) formalize and systematize activities within the design process and (iii) externalize design thinking in charts and diagrams. More recently, efforts are being made by various researchers, in the bid to develop more comprehensive design systems, which would include not only the traditional geometric aspects, but also take account of other design activities right from design concept to manufacturing. Some of this on-going work have also been discussed under computational design models, and computer based design systems. The aim of the survey was to examine the whole body of issues related to design theory and methodology, and to identify the key features amongst the issues discussed which will be contributory to the development of Design Function Deployment. Such beneficial features have been identified and will be discussed in detail in chapter 3. Current research in engineering is encouraging, although a lot of activities are still focussed on the analytical and computational aspects of design. It is highly desirable that more research will start to focus on the synthesis aspects of design, examining ways to support creativity, innovation and generation of conceptual design solutions. It is in this aspect, that there would be opportunity for design and manufacturing companies to excel.

2.4 A STATE OF THE ART REPORT ON CONCURRENT ENGINEERING

2.4.1 Introduction

The current pervasive worldwide competition is placing an increasing pressure on manufacturers to be more responsive to change.

The manufacturer who can utilize the latest technologies and materials in both product and means of production as well as getting the product to the market in the shortest lead time is at a clear advantage. It allows maximum benefits to be gained from one generation of a product before the next generation comes to the market place. Since there are many other competing alternatives in the market, the product has to be 'right first time' in addition to the need for continuous improvement of the design process. The measurables of improvement to designs are (a) better function (b) lower cost (c) shorter lead times and (d) better quality or fitness for purpose. Hartley [145] states that each new product should

(i) Be the product customers want at the price they are prepared to pay
(ii) Reach the market on time without exceeding the budget and in 25 to 33 percent less time than at present
(iii) Be designed with the highest level of quality and reliability from the outset
(iv) Be easy to manufacture in high volumes from job one, on machinery that is flexible enough to cope with possible changes
(v) Contain the smallest number of parts and be designed for ease of assembly
(vi) Reach sufficient production volume quickly enough to reach the break-even point early.

2.4.2 The Traditional Product Development Process

Thetraditional design and product development cycle is a serial process which proceeds sequentially starting from marketing and sales who through their interaction with customers, identify the need for a product and the necessary information is passed on to the design department. The design department on completing the design, pass it on to manufacturing palnning, who then pass it to production. As a result of this compartmentalisation of activities, this approach to product development is popularly called the 'over the wall' approach to design and product development. In this approach, changes to the design are necessitated by the need for manufacturing, sales and service departments to accomodate their requirements which were previously overlooked during design. This involves an iterative process with many repetitions before an acceptable design is developed. Some of the difficulties associated with this traditional sequential method, has been identified by Bedworth et. al [146] as follows :

(a) Design alternatives are quickly eliminated in the interest of time and one particular idea is pursued.

(b) The definition of design detail is costly in labour hours. Even with CAD/CAM tools much manual effort is needed.
(c) The design process is characterised by a rigid sequence of design decisions.
(d) Producibility and supportability issues are not considered until relatively late in the process, when a design change may be very costly.
(e) Production planning, support analysis, maintenance and reliability are considered separately from the design process.
(f) Design data is fragmented. Documentation includes CAD files, dimensioned components, sketches, process drawings, 3D solid models etc. It is difficult if not impossible to maintain consistency at all times across these representations.
(g) Information is lost as the design progresses. The design intent may be lost by the time the documentation gets to the producibility experts.
(h) Designers are usually not aware of cost information, so they cannot intelligently set cost reduction as a realistic goal. There are no tools for estimating costs as there are for other domains, and when costs are calculated, it is often too late to make major design changes.

2.4.3 Concurrent Engineering - Definitions and Benefits

Concurrent Engineering, sometimes called simultaneous engineering, or parallel engineering has been defined by several authors. Some of them are given below:

Broughton [147] defines simultaneous engineering in the following way: "Simultaneous Engineering attempts to optimize the design of the product and manufacturing process to achieve reduced lead times and improved quality and cost by the integration of design and manufacturing activities and by maximising parallelism in working practices".

Evershiem [148] defines simultaneous engineering as "an organisational strategy with the idea to shorten the time of product design by simultaneous planning of product and production. Retailers and buyers of the means of production are working together during the product design phase, so that demands on the means of production are specified at the very earliest moment. The results are shorter innovative times and lower costs".

Brookes and Gatehouse Limited [149] defines Simultaneous Engineering as "the bringing together of many design techniques with the concurrent consideration of all constraints when making a design decision".

Stephanon and Spiegl [150] define Simultaneous Engineering as "a way of conducting engineering operations so that all functional considerations from design to manufacturing are taken into account and solutions to potential problems are developed as early as possible".

Ellis [151] "Concurrent Engineering is a name given to the process of paralleling design engineering with the steps taken further down the implementation chain, so as to save time in launching products and improve communication over serial progression from department to department".

Lake [152] "Concurrent Engineering - The set of methods, techniques and practices that: (i) cause significant consideration within the design phases, of factors from later in the life cycle, (ii) produce along with the product design, the design of processes to be employed later in the life of the product, (iii) facilitate the reduction of the time required to translate the design into distributed products, (iv) enhance the ability of products to satisfy users' expectations and needs".

Lake [152] "Concurrent Engineering - The application of systems engineering principles and of management approaches. Integrated engineering of products and their associated production and logistics processes with the objective of providing a product and production process that is robust during manufacturing and customer use".

Winner[4] "Concurrent Engineering is a systematic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause the developers, from the outset, to consider all elements of the product life cycle from conception through disposal, including quality, cost, schedule, and user requirements"

Ramana Reddy et al[153] "Concurrent Engineering (CE) is a systematic approach to integrated product development that emphasizes response to customer expectations and embodies team values of cooperation, trust, and sharing".

Kannapan [154] "Concurrent Engineering, in the ideal case, brings to bear all the concerns throughout the product life cycle concurrently during product design. The strategy of concurrence provides an opportunity to address the source of conflicts between design agents representing the concerns of different engineering disciplines, functionality, marketability, manufacturability, maintainability etc. early in the engineering process".

Knight & Jackson [155] "Simultaneous engineering is the concurrent development of project design functions, with open and interactive communication existing among all team members for the purpose of reducing lead time from concept to production launch".

Walklet [156] " Simultaneous engineering is a process in which appropriate disciplines are committed to work interactively to conceive, approve, develop, and implement product programs that meet predetermined objectives. The keys to simultaneous engineering are : breaking down barriers, pooling resources, and getting input from those affected by decisions".

Cleetus [5] "Concurrent engineering is a systematic approach to integrated and concurrent development of a product and its related processes, that emphasizes response to customer expectations and embodies team values of cooperation, trust and sharing in such a manner that decision making proceeds with large intervals of parallel working by all life-cycle perspectives early in the process, synchronized by comparatively brief exchanges to produce consensus".

The above definitions of concurrent engineering, do point to the benefits and advantages that can be realised and gained by adopting not just its philosophy, but also its principles and necessary techniques for its implementation. A close examination of these definitions also reveal certain issues that can be considered as goals of concurrent engineering, and which are pervasive in virtually all the definitions on concurrent engineering. Such issues include the need to : (1) improve and maintain the quality of the product, (2) reduce product development lead time, (3) reduce product development costs, (4) integrate the design of a product and the associated manufacturing and production processes, (5) consider all life cycle issues (both downstream and upstream) which affect product design, (6) resolve and manage conflicts and tradeoffs in the early stages of design, (7) breakdown barriers between product development functional disciplines, (8) encourage the integration and use of all company resources, (9) respond proactively to customers and their needs, and (10) parallel the design process.

Within the sphere of product development, the adoption of the concurrent engineering approach, leads to several benefits, some of which are discussed below.

(i) Segregation, isolation and the 'over the wall' syndrome are virtually removed from the company, with the result that all the various divisions can now work together in an integrated manner to achieve a more productive team, (ii) The overall product development process is shortened as steps along the way are handled in parallel. Thus improving time to market, (iii) The number of product iterations is also reduced, (iv) There is also a reduction in design errors, engineering and design changes and product re-engineering as more information is available at the design stage, (v) New products, with better customer satisfaction, lower costs and higher quality can be released early into the market, (vi) Design can be done more 'right first time' and (vii) Companies can be more responsive to the customer.

2.4.4 Principles and Goals of Concurrent Engineering

In looking at the environment for concurrent engineering, krishnaswamy [157], discusses concurrent engineering as a process driven by a company's Resources (Marketing, Design, Manufacturing, Sales, Finance, Service, etc.), controlled by four

Key Dimensions of (Organisation, Communication Infrastructure, Requirements and Product Development), and operated by five forces of change (Technology, Tools, Tasks, Talent and Time). The four key dimensions define the environment for concurrent engineering, for which the five forces of change have to be transformed and well managed by a company's resource base, in order to respond to customer demands and produce high quality products that are released on time to the market.

In the many discussions and reports on concurrent engineering, certain fundamental and underlying issues that constitute a foundation for it, have been prevalent in literature. These issues which have been highlighted by several authors, as principles and goals aimed at the realization of concurrent engineering [158-164], are categorised and discussed below :

Requirements and Specifications

1 Consider the customer requirements with the degree of importance.

2 Consider the legal, statutory, use environment, and maintenance requirements.

Conceptual Solutions

4 Identify product concepts that are inherently easy to

5 Focus on component design for manufacturing and assembly.

6 Convert concept to manufacturable, salable, usable design by stating all constraints.

7 Reduce number of parts.

8 Integrate the manufacturing process and product design that best match needs and requirements.

9 Anticipate fabrication and assembly methods and problems.

10 Perform continuous optimization of product and process.

11 Carry out design for producibility and usability study.

Interchangeability

12 Identify subassemblies.

13 Increase interchangeability between models.

14 Define subassemblies to allow models to differ by the subassemblies.

Standardisation

15 Standardise fastener types and sizes; use low cost, irreversible fasteners only where a skilled service person would work.

16 Break down products and processes into self contained modules and assembly lines.

17 Identify difficult process steps for which costs and process times cannot be predicted.

18 Use existing processes and facilities so that product yield is high.

Design for assembly

19 Consider design, fabrication and assembly process.

20 Design assembly sequence.

21 Make assembly easier by minimising setups and re-orientations.

22 Design parts for feeding and insertion.

23 Integrate quality control strategy with assembly.

24 Design factory system to fully involve production workers in the production strategy, operate on minimum inventory, and integrate with vendor capabilities.

Tolerances

25 Adjust tolerances to eliminate failures during assembly.

26 Design each part so that tolerances are compatible with assembly method and fabrication costs are compatible with cost goals.

27 Improve robustness of product and processes

28 From the start, include all domains of expertise as active participants in the design effort.

29 Resist making irreversible decisions before they must be made.

30 Integrate Quality control strategy with assembly.

31 Identify testable areas and tests.

32 Consider issues relevant to mass production as well as small lot sizes.

System Thinking

33 Look at a wide range of issues simultaneously during the early design phase of product development.

34 Separate concerns by breaking away from the linear thinking and organise around an outcome rather than tasks.

35 Co-evolve requirements and the whole system design.

36 Use multiple cycles of prototyping (electronic mockups) to build up experiences for better management decision and/or goal definition.

37 Perform design of product and process taking into account the entire product life cycle.

Continual Improvement

38 Do it right and do it better.

39 Focus on the improvement of both product and process.

40 Be proactive and not reactive with respect to quality.

Empowerment

41 Empower individuals and teams to realise their full potential.

42 Support empowerment by ensuring a shared vision in the team, to align individual or team purposes with the organisation's objectives.

43 Locate multidisciplinary teams physically together to facilitate better communication.

Inclusion

44 Deploy interdisciplinary teams with a focus on the customer.

45 Locate all stakeholders, assign their roles and define communication channels among them.

Reification

46 Facilitate an open exchange of information, to make explicit the basis of judgement and to communicate reasons and rationale for decisions.

47 Encourage team members to speak openly without fear of reprisals and help to breakdown defensive game playing.

2.4.5 Implementation and Realization of Concurrent Engineering

It is increasingly becoming evident that, although concurrent engineering is gaining grounds in industry, the necessary benefits can only be realised, by the establishment of procedures and processes, and the development of resources, tools and techniques, as well as systems to support its concepts and principles. By integrating all these features, concurrent engineering strives to create successful new products by bringing together as early as possible in the development cycle a company's resources and its experience in design, development, marketing, manufacturing, service and sales. These capabilities are then focused on developing and manufacturing a high quality, low cost product that satisfies the customer [165].

Several authors have proposed approaches to be adopted for the implementation and realisation of concurrent engineering, and these are discussed below [28,166-168] :

Realization of simultaneous engineering involves the following: (1) Focus on the Customer, (2) Integration of the Organisation, (3) Use of Design and Product Development Teams, (4) Employee Involvement and Participative Management, (5) Competitive Benchmarking, (6) Focus all Activities on Quality, Cost and Delivery, (7) Concurrent (Parallel) Product Development Process, (8) Integrated Design of Product, Manufacturing and Support Processes, (9) Strategic Relationships with Suppliers, (10) Integrated Tools and Techniques, (11) Use of project management, (12) Consideration of new technologies, (13) Synchronization of information and (14) Use of computers.

<u>Focus on the Customer</u>

There is need for better understanding of customer demands, and thus more frequent communication is encouraged. Emphasis should be on customer's responses to products, rather than on internal corporate metrics. The use Quality Function Deployment will aid in eliciting, categorising and prioritising customer requirements and translating them into measurable design requirements or quality characteristics, which help to achieve the customer requirements.

Integration of the Organisation and use of Product Development Teams

A key aspect of ensurring closer co-operation within a company as well as realising the benefits of concurrent engineering, is the integration of all the functional divisions who have a part to play in product development. The major defects of the sequential design method as described in section 2 arise from here because producibility and supportability issues are considered late or only after costs are committed and production planning, support analysis, maintenance and reliability are considered separately from the design process.

The fundamental purpose of the team is to address the breadth of concerns involved in the life-cycle of a product. It involves humans working together and it cannot be over-emphasised that the characteristics of the team players greatly decides the success of the team. The importance of this aspect, has been emphasised in literature [169-174] available on this subject. They describe essentials such as a shared sense of responsibility and goals, availability of information, a high level of communication, a sense of trust, an ability to resolve conflicts, inclusion of people with different styles of thinking and decision making, leadership, senior management involvement and committment, future focus, an inter- disciplinary mix of stakeholders, creativity, an understanding of risks, an equitable reward structure, the value of rapid response and the need for ways to calculate the effectiveness of output. To assure that a company gets pass the starting gate in their team effort, two items are paramount: organizational fit and selecting the right players".

The complexity of today's product and production processes, demands contributions from various sections of the organisation such as marketing, finance, design, manufacturing, assembly, sales, after sales service, maintenance as well as from customers and suppliers of parts, materials and manufacturing equipment. These contributions are effectively co-ordinated by having representations from these sections, in the design or product development team. Such representations should be both multifunctional and multilayered.

Several authors [175,176] have listed the necessary potential participants in a product development team for the successful implementation of concurrent engineering. The following is a compilation of these teams: (1) marketing, (2) customers, (3) finance, (4) aesthetics or styling, (5) design or product engineering, (6) manufacturing engineering, (7) advanced purchasing, (8) scientific institutions or research and development, (9) internal company departments, (10) suppliers of components and parts, (11) suppliers of materials, (12) suppliers of manufacturing equipment, (13) dealers, (14) insurance companies, (15) relevant government departments eg. Department of Transport, (16) central engineering laboratories and (17) experiment and prototype engineering

Several advantages have been attributed to the use of product development teams. The major ones as discussed by Clealand and Bursic [177], are listed below:
(i) Increased capacity for global competition and ability to get lower costs high quality products to customers on time, (ii) Fewer design errors and mistakes and a reduced number of engineering changes, (iii) Reduction and possible elimination of design reviews, (iv) Enhanced communication and co-operation among designers, managers, and other professionals involved in the product development process, (v) The resulting design is more simple, with reduced number of parts to be manufactured and assembled and (vi) Greater employee involvement in the organisation.

Employee Involvement and Participative Management

The organisation should ensure that the full talents of employees are utilised, and some responsibility decentralised to local areas of expertise and actionability. The aim should be for a multilayered organisation, with communication networks occurring both horizontally and diagonally on a need basis, without the constraint of the vertical tree like structure of the organisational charts.

Competitive Benchmarking

There should be a strong emphasis on competitive benchmarking of company products against the best of competition. Each process and sub-process within product development should also be benchmarked with the motive of continuous improvement.

Focus All Activities on Quality, Cost and Delivery

All activities of the development process for a new product should be driven by the prime goals of quality, cost and delivery (development schedule). Thus overcome the conflicts that might arise from local objectives.

Concurrent Product Development Process

The dominant theme here, is the elimination of the 'over the wall' transactions, as well as non-value adding activities and thus streamlining the design process. Product design, manufacturing and production capability, field support capability and quality are developed together as one system, with decisions made in one tradeoff space. Design activities that are reasonably independent of each other can also then proceed in parallel. The design process also includes disciplined design reviews and convergence on the final set of design decisions. It is also necessary to freeze earlier decisions in order to minimize rework.

Integrated Design of Product, Manufacturing and Support Processes

A key part of the ethos of concurrent engineering is to integrate the design of the manufacturing and production planning processes as well as other product support processes with the design of products. This will help to optimise the performance, availability and life cycle costs of the product. This hence requires the development and application of guidelines for issues such as manufacturability, maintanability, serviceability, assemblability in association with corresponding formal tools. At a higher level, corporate business objectives should also be integrated with design. A major issue to be addressed in the concurrent design of product and process, is the development of techniques for rapid generation and evaluation of manufacturing process plans during the product development process [178].

Strategic Relationships with Suppliers and Cooperation of Companies

In the process of manufacturing a product, raw materials, components and parts, sub-assemblies, manufacturing means and the related technologies from different manufacturers and suppliers are brought together. Suppliers know their product technology, product application and process constraints best. It therefore follows naturally that to make the best use of their expertise in the product under development, they should be involved in the design, right from the very start of the project. It is also advisable to reduce the suppliers base to the required minimum, and to focus more attention on long term relationships.

Integrated Tools and Techniques for Concurrent Engineering

In order to cope with the increasing complexity of the product development process, as well as providing enabling technologies for concurrent engineering, many existing engineering tools are being developed and integrated in emerging design systems. Such tools can be classified into [179] : Requirements elicitation and generation tools, Analysis/Design optimisation/Integration tools, Rapid Prototyping tools, Manufacturing and Production control tools, Test/Assessment tools, Supportability tools and Communication tools.

There are also several tools and techniques which assist design and manufacturing engineers to improve the product development process. Some of these tools include : Function Analysis, Sketching input for Solid modelling, CAM tools, Circuit analysis and design, NC verification, Solid modelling and Finite Element Analysis, Optimization, Design for Cost, Value Engineering, Design for Manufacture, Design for Assembly, Materials Selection, Design for Safety, Design for Environment, Design for Ergonomics, Design for Reliability, Design Rulebases and Databases, Failure Mode and Effect analysis (FMEA), Robust Engineering Design and Taguchi Methods. Some of these tools can be integrated to work with a common product model, in the analysis and refinement of product and process design data, while others can be used to simulate product performance using the product electronic mock-ups.

They should be employed early in the product development process, in order to develop a more mature design, as well as to reduce the number of time consuming design/build/test iterations for mock-ups and physical prototypes.

Use of Project Management

The adoption of the concurrent engineering approach to product development, results in engineers anddesigners having to cope with larger amount of design data and information. There is hence a need to adopt techniques for planning, organising and monitoring the large and complex network of design tasks, arising in product development. In this regard, Evbuomwan et al [180] have reviewed some project management techniques such as Directed Graphs (DG), Project Evaluation and Review Technique/Critical Path Method (PERT/CPM), Structured Analysis and Design Technique (SADT) and the Design Structure Matrix (DSM) [181-184]. Their advantages and limitations were highlighted.

The main advantage gained through proper project management is reduced development time, achieved by removing activities or groups of activities in the critical path and performing them in parallel in a non-critical path. The degree to which design activities can be scheduled simultaneously depends on the quality of clusters and the nature of precedence constraints. In the absence of precedence constraints mutually exclusive groups can be scheduled in parallel.

The concept of decomposing into modules and activities have the following advantages [184, 185]: (1) Separation of the overall design into groups of modules and activities, (2) The group of modules do not have to correspond to the traditional organizational structures. For example vehicle body design group and transmission design group, (3) Potential activities that might be performed simultaneously are detected, (4) Complexity of management of the design task is reduced and (5) Simplification of the design task and reduction of the design cycle.

Consideration of New Technologies

Several companies have by use of modern materials and manufacturing processes, given their products significant competitive edge in the market. There are two main areas in which substantial development has been achieved. They are : (i) New materials arising from improved materials technology and (ii) Innovative manufacturing processes. The increasing effect of the use of modern materials and processes includes inter-alia : the increase of new product options for manufacture, improvement of product performance and efficiency, and cost reduction. Very significant benefits that have arisen from the use of new materials and technologies, have come about from the recognition of the importance of concurrently integrating design, materials and manufacturing process selection at the early stages of product development. This fact suggests that materials selection should not be an isolated activity during the design process, but new materials and technologies should introduced at the very early stages of the design process particularly for new and innovative designs, when the generation of viable solutions is still in progress and no particular design concept has yet been chosen.

Synchronization of Information

The information generated in a modern design process, is significantly more than that of the past, and needs to be properly managed. Such information can be used repeatedly in manufacture, inspection and quality control, packaging, commissioning, etc. It is therefore important to capture the data and information in the first instance, store them and use them whenever necessary. The manufacturing data preparation includes design engineering, process planning, NC programming, robot programming, inspection programming and so on. Presently they are (i) prepared by human beings (ii) computer assisted (iii) often produced off-line and with (iv) no interactions. But a well advanced system of the future should be (i) prepared by computers (ii) assisted by humans (iii) produced on-line and (iv) an integrated system with interactive facilities. The current move within the concurrent engineering field, is the evolution of environments that can support multiple users, applications and databases, all working together to optimise the product development process [186].

The sharing of common design information, will require the development of a central database integrated with a network of distributed/parallel databases. The databases must be designed to allow concurrent access to the design information by all interest groups (design, manufacturing, production, etc.). Lewis [187] discusses four basic elements in the implementation of such databases. These are (i) common, system-wide schema definitions for shared objects, (ii) a heterogeneous collection of databases and files for permanent storage of the objects, (iii) protocols for interchanging and sharing objects across multiple systems, and (iv) wrappers for making tools, utilities, and databases available as standardised system services. These help to remove unnecessary interfaces between the groups, provide more current and consistent information, facilitate timely feedback on the design as well as reducing delays. For collocated design teams, a single shared database tends to be sufficient, while for geographically dispersed teams, there would be a need for distributed database systems to handle the multiple design information within an enterprise [188].

Synchronization of information can also be achieved with the use of modern information technology tools for data exchange and transfer like modems and other teletransmission aids. Such single repository of product data helps to minimize data handling, redundancy and errors as the design evolves. A proper synchronisation of data is a pre-requisite to 'Just-in-time' manufacturing.

Use of Computers

Hartley [145] states that " To maximise the benefits of simultaneous engineering, the current trend towards increased Computer Aided Design and Manufacturing must be exploited ". He continues by writing that " with the right combination of hardware and software, design and stress engineers can work in parallel, and far fewer prototypes need be built, and lead times can be cut dramatically ". In addition, with proper interfaces, all further advantages provided by the specialist techniques, can be realised in a parallel set up.

2.4.6 Constraints on the Implementation of Concurrent Engineering

While recognising the many benefits of concurrent engineering, it is equally important to note certain constraints that will influence the realisation of these benefits, and which should be taken into account in the process of its implementation. Some of these major constraints as highlighted by Professor Peter Hills [189], are discussed below :

(i) Incomplete requirements - This can be due to external sources in the form of performance requirements mismatch and incomplete interface definition as well as internal sources in the form of skill inadequacies, available technical knowledge vis-a-vis intellectual property rights (IPR), (ii) Resource profiling, and (iii) Possible Risk Sources : (a) technical - due to either technology performance default or design team skill deficiences); (b) different projects competing for the same resource and/or non-availability of tools or facilities when required; (c) commercial - due to funding profile constraints and contract security (d) financial and/or present economic environment; (e) Procurement due to delivery lead time default.

Sharon [190] also highlights three possible bottlenecks to the implementation of concurrent engineering. These are (i) the need to provide a capability for early decision making in product development, (ii) the need for facilitating technologies to support feedback (e.g. rapid prototyping) and (iii) the need for high calibre people who must be experts in their field and also knowledgeable in other areas in order to communicate effectively in the design team.

2.4.7 Software Support for Concurrent Engineering

Current research activities in concurrent engineering are focused on the development of software tools to aid designers in achieving and realising not only the many benefits of concurrent engineering, but also to make the design and product development process, more effective and productive. A number of these activities are reported below. Kannapan and Marshek [155] in their research work, have proposed a concurrent design schema where intelligent design agents representing different concerns in the product life cycle can negotiate using utility functions to resolve conflicts on the value of shared parameters. The utility functions associated with each design agent are defined on values of design parameters controlled by a design agent and propagated to a shared parameter when conflicts arise. The method adopted for resolving conflicts, was adapted from the Nash and Kalai-Smorodinsky [191] solutions developed in the literature for industrial and social negotiation/bargaining. Their concept was further illustrated with an example of concurrent engineering of a poppet relief valve, by demonstrating the process of design negotiation and resolution of parameter conflicts between a valve design agent, a helical-spring design agent and a pipe-enclosure design agent.

Sobolewski et al [192] in their work at the Concurrent Engineering Research Centre based at West Virginia University, Morgantown, WV, USA, are developing a concurrent engineering environment based on a concurrent engineering database architecture. The architecture is represented in four levels as shown in Figure 2.34.

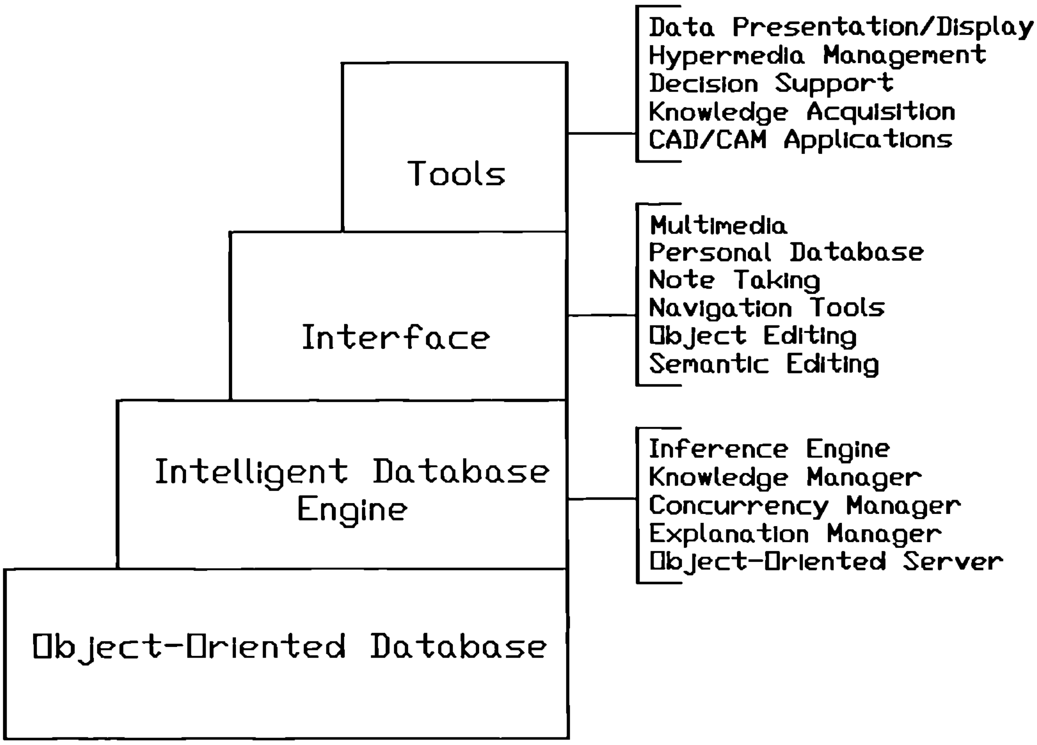


Figure 2.34 The Concurrent Engineering System Architecture [192]

The first level, that is, the bottom level, is the object-oriented database which is being developed to provide an engineering environment that supports real-time inspection and modification of enterprise data (information on product form, function, manufacturing processes, analysis results, constraints, etc). The second level, above the object-oriented database level, is the intelligent database engine. This consists of an inference engine, knowledge manager, concurrency manager, explanation manager, multi-media manager, and object-oriented server. The third level is the high-level user interface, which contains a set of representation tools that enhance the functionality of the engineering environment.

The fourth level is the high-level tools, which provide the user/designer with facilities for intellegent search, data representation, data quality and integrity control. This level also contains CAD/CAM/CAE applications and knowledge-based tools, such as DICEtalk (used for solving various engineering tasks) as well as intelligent system design tools used for designing the concurrent engineering intelligent database.

Sriram and Logcher [193], have also reported on their on-going research at the MIT Intelligent Engineering Systems Laboratory, involving the development of a Distributed and Integrated Environment for Computer-Aided Engineering (Dice). Dice is being envisioned as a network of agents or knowledge modules that communicate through a shared workspace. As part of the Dice effort, several research issues, are being addressed. These include *frameworks, representation issues, organisation issues, negotiation/constraint management techniques, transaction management issues, design methods, visualisation techniques, design rationale records, interfaces between agents and communication protocols*. The Dice framework is a collaborative agent based architecture, which uses object-oriented database management techniques, artificial intelligence, cognitive science, organisational theories and user interfaces.

Representation issues deal with the development of product models for communicating design information across disciplines. Organisational issues involve strategies for organising engineering activities for effective use of computer aided tools. Negotiation/constraint management techniques deal with conflict detection and resolution between various agents. Transaction management issues deal with the interaction issues between the agents and the central communication medium. Design methods represent the techniques used by individual agents. Visualisation techniques include user interfaces and physical modelling systems. Design rationale records are used to keep track of the justifications (design rationale and intent) generated during the design. Interfaces between agents support information transfer between various agents, while communication protocols are used to facilitate the movement of objects between various applications. The Dice project is being implemented in Cosmos (C++ Object-Oriented System Made for Expert System Development), a C++ based knowledge representation language.

Ramana Reddy et al [153] highlighting the dearth of tools for enabling real-time communication among team members working in a heterogenous, geographically distributed environment, proposes the notion of virtual tiger teams, which consists of a geographically scattered team of experts who use a computer-supported environment to collaborate over a network. In this regard, a layered architecture of different types of computer technology, which should be integrated to provide such a collaborative environment, was proposed. This architecture, utilises advances in database and networking technology, groupware, multi-media, and graphical user interfaces, in creating a collaborative environment which transcends the barriers of distance, time, and heterogeneity in computer equipment. The aim is for this environment to enable any team member to spontaneously communicate, (and thus collaborate) with any other member or group. This proposed architecture consists of the Activity layer, the Transaction layer, the Collaboration services layer, the Enterprise information model layer and the Network layer.

The activity layer represents the activities carried out by the concurrent engineering teams. The transaction layer consists of fundamental activities such as looking up information scattered within the enterprise, through heterogenous data and knowledge bases, carrying out various computations, communicating and sharing design information, carrying out negotiations with other groups, until a consensus is reached, as the design evolves, making necessary group design decisions, and archival (storage and retrieval) of design informations. The collaboration services layer provides services to support the various transactions and activities of the team members. Such services include those for : (i) collocation - ensuring effective communication between geographically dispersed members, (ii) coordination - issues relating to group decision making and negotiations, common visibility of activities and data, planning and scheduling of activities, design change notification, and constraints management, (iii) information sharing - issues relating to common data representations, transparent access to information in a distributed heterogenous system, version and concurrency control and management of replicated data, (iv) corporate history management - issues relating to electronic capture of design rationale and intent, indexing, linking and storage of design information and archiving of design decisions, (v) integration - facilitating access to engineering tools and services in a transparent manner across the enterprise and encapsulating existing tools and services for use across the network. The enterprise information model layer, takes into account the need to make available enterprise (company) information to support the earlier services (layers), and uses enterprise information models characterises the company's product, the processes adopted to make it, and the resources available to the company. The network layer which is the kernel of the virtual team environment, represents advances in communications technology and distributed computing, such as directory services, interprocess communication and remote procedure calls.

Cutkosky et al [194] are also developing the Palo Alto Collaborative Testbed (PACT), a concurrent engineering infrastructure that encompasses multiple sites, subsystems, and disciplines. PACT is being developed as a testbed for knowledge-sharing research, based on emerging artificial intelligence techniques and data-exchange standards (Product Data Exchange Using Step/Standards for the Exchange of Product Model Data - PDES/STEP). The PACT architecture is based on interacting agents, which rely on shared concepts and terminology for communicating knowledge across disciplines, an interlingua for transferring knowledge among agents, and a communication and control language that enables agents to request information and services. The PACT system consists of four main tools. These are NVisage, DME (Device Modelling Environment), Next-Cut and Designworld. The NVisage tool is a knowledge-sharing technology that enables each engineering tool to encode and maintain its own separate model of a design, while intertool communication mechanisms maintain consistency among the models. DME is a model formulation and simulation environment. It is used to help electromechanical-device designers to experiment with alternative designs, by providing feedback about the implications of design decisions. It also helps in documenting designs for future use and operates on multiple knowledge bases. Next-Cut is a prototype system for concurrent product and process design of mechanical assemblies, and it consists of several modules that surround the representations of design artifacts, process plans, and tooling. It includes modules for feature-based design of components and assemblies, tolerance analysis, kinematic analysis and synthesis, geometric analysis, and CNC process and fixture planning. Designworld [194, 195] is an automated prototyping system for small-scale electronic (digital) circuits built from standard parts. The design for a product is entered into the system via a multimedia design workstation. A dedicated robot cell then automatically builds the product. Designworld consists of 18 processes which perform various product life cycle tasks including design solicitation, simulation, verification, diagnosis, test and measurement, layout, assembly planning, and assembly execution. Each of the processes communicate with its peers via an agent communication language (ACL) messages.

Other research work currently on-going include the development of a constraint based software for concurrent engineering as reported by Bowen and Bahler [196]. The aim of this work is to develop a concurrent engineering-oriented language based on the notion of constraint networks. The attraction of using constraint networks, is that they can support multi-directional inference. This means that a constraint network can capture the impact of a decision made concerning one phase of a product's life cycle by an expert in that phase, on the other phases of the life cycle. Tong and Gomory [197] also report on their work involving the development of a knowledge based computer environment that supports the concurrent engineering of small electromechanical appliances, such as kitchen and lighting applicances. This is implemented by integrating and providing active assistance for specification acquisition, conceptual design and redesign, and qualitative simulation. Gupta et al [198] are also developing a methodology for the concurrent evaluation of the machinability of a part or component, during the design stage. This involves systematically generating and evaluating several machining alternatives. The results of the analysis are then used to (i) provide feedback to the designer about the machinability of the design so that problems related to manufacturing can be recognised and corrected, while the product is being designed and (ii) provide information to the manufacturing engineer for use in developing process planning alternatives, depending on machine tool availability.

Dewan and Riedl [199] have also reported on the development of FLECSE (Flexible Environment for Collaborative Software Engineering), a multimedia environment, to support concurrent software engineering, by providing improved interaction between software engineers/and development teams. The system contain a series of tools that are employed to increase concurrency, in the various phases of the software development life cycle. These tools enable structured interaction, provide multimedia support and teleconferencing, concurrency and version control, caching of user interface state and collaborative debugging.

Hutchison and Amundsen [200] in their paper, discuss a concurrent engineering environment for electronic circuit design. This system is considered to be a supportive, intelligent environment for electronic circuit design, which provides customisability features to improve the designer's productivity and a knowledge base to improve the quality of the design by the consideration of producibility, testability, reliability, maintainability and manufacturability issues in parallel with function, in the design decisions.

2.4.8 Summary

Concurrent engineering, since its emergence in the design, manufacturing and research community, has gained wide acceptance. This is made evident by the ample research activities and programmes currently going on in industry and academia, particularly in Europe, USA and Asia. The aim of this state of the art report has been to fully elucidate in a comprehensive manner, the whole spectrum of issues and activities associated with concurrent engineering. In this regard, the discussions have focussed on what concurrent engineering is seen to be and what it should be, its principles and goals as well as the necessary implementation issues that have to be addressed in order to reap the benefits of concurrent engineering. There was also an examination of issues that might act as bottlenecks or constraints to the implementation of concurrent engineering in practice. This section concluded by discussing current on-going research activities that are focussed on developing software tools, techniques and environments for the realisation of concurrent engineering, in design and engineering enterprises. From this research into concurrent engineering, it is worth noting that the early work on concurrent engineering were mainly biased towards the philosophy and management aspects with little discussions on the technological aspects. However, there are encouraging signs as regards the efforts now being made to address the issues which are required to realise and implement concurrent engineering in practice. There is still much work to be done particularly in the area of developing integrated frameworks to provide a platform for concurrent engineering. The requirements of such a system have been elicited during the course of this research, and are discussed in detail in chapter 4. The key features of concurrent engineering, that were also extracted for the development of Design Function Deployment as a concurrent engineering system, will also be discussed in chapter 3.

2.5 GENERAL SUMMARY

The essence of this chapter has been to discuss in detail, the research carried out into the three main contributors to the evolution and development of Design Function Deployment, namely : Quality Function Deployment (QFD), Design Philosophies, Models, Methods and Systems, and Concurrent Engineering. Each of these three topics, represent major research areas in their own right, and it was therefore necessary to devote separate sections in this chapter to discuss them. QFD which was the starting point was examined to establish its suitability for supporting the integration of the quality function into engineering design. The research work carried out involved looking at the historical development of QFD, the formalisation process, and the various enhancements and extensions made to it since it's introduction to the USA and Europe. A number of QFD applications, were also discussed as well as directions for further research. From the research into QFD, the useful features for the development of DFD, were identified and are discussed in chaper 3. The research into design philosophies, models, methods and systems involved looking at the nature, definitions and varieties of design, design theory and design methodology. Several types of design models, methods and systems were identified and discussed. The important features from the survey, which were extracted for the purpose of DFD, are further elucidated in chapter 3. The detailed investigation of concurrent engineering, involved looking at the various definitions, principles, benefits and goals of concurrent engineering. The core part of the discussion was focussed on the issues that need to be addressed in order to realise and implement concurrent engineering in practice, such as integration of tools, synchronisation of design information, use of new materials and technologies, focussing on the customer, quality, cost and delivery, paralleling the design process, bringing downstream issues upstream, etc. The useful features of concurrent engineering for the purposes of DFD were also identified and are discussed in the next chapter.

CHAPTER 3

EVOLUTION OF DESIGN FUNCTION DEPLOYMENT

3.1 INTRODUCTION

This chapter discusses the research issues that contributed towards the development of Design Function Deployment, DFD. It commences with discussions on Quality Function Deployment (QFD), and continues with the work done on design philosophies, models, methods and systems, and concurrent engineering. The key features of the above topics which were drawn upon in the evolution and development of DFD, are discussed.

The research work that led to the development of Design Function Deployment, DFD, was based on two prime objectives of :

- ¤ Integrating the quality function into engineering design and
- ¤ The development of a generic design methodology in line with the 'Design It Right' philosophy and within the context of the Concurrent Engineering paradigm.

In addressing the first objective, Quality Function Deployment, a technique attributed to the Japanese, and which is used primarily to translate demanded customer requirements into measurable design requirements (quality characteristics), was the starting point. In looking at QFD, a detailed study was carried out on the various stages and processes involved, based on the training manual of the American Supplier Institute (ASI) [201] in particular. The study involved looking at the philosophy behind Quality Function Deployment, the main concepts as well as the usage of the technique. The full details of the research into QFD, has been reported in section 2.2 of chapter 2.

As the work progressed, it soon became apparent that QFD offered a number of advantages, while also exhibiting some limitations. It was also observed that QFD, although progressed through four main stages in the deployment process, did not adequately provide a framework for a generic design methodology. QFD was found to be mainly suited for conceptually static products (i.e. products whose fundamental concepts were fixed) and there was no provision to handle complex and conceptually dynamic products (i.e. products with ample room for conceptual innovation). QFD is effective in handling existing products which can be compared with those of competitors, with the resultant effect being the development of relative designs. For the development of entirely new designs, QFD will be much less adequate.

Furthermore, although the word 'quality' is pervasive in QFD, the issues associated with and referred to, for example, as quality elements, are influenced significantly by design, and can thus be regarded as design functions. Hence the term 'Quality Function' can be replaced by 'Design Function'. In the final analysis, during product development, whatever factors or parameters were regarded as quality characteristics, had to be **'designed into'** the product or system.

In view of the foregoing and in order to address the second objective, it became necessary to have a strategic switch in paradigm. Thus the concept of Design Function Deployment was born. This then led to further research into the main issues of design philosophies, models, methods and systems, involving a detailed survey of the above in relation to the work done in engineering design and research in engineering design, within the last four decades, as reported in various research journals, design textbooks, and several proceedings of international conferences on engineering design and design methodology. Such conferences include those that were held in the UK, Europe and the USA. From this review, it was found that the design models proposed to date, aggregated into the following types :

(a) Prescriptive models based on the design process
(b) Prescriptive models based on product attributes
(c) Descriptive models and
(d) Computational models

These models have been discussed in detail in sections 2.3.10 - 2.3.14 of chapter 2.

In the case of design methods, they tended to cover various aspects of design synthesis, analysis, evaluation, decision making, optimisation, modelling and simulation. A number of these methods were more suitable for the detailed design stage, while others were more amenable to the early stages of conceptual design, in the product development life cycle. Some methods were also found to be based principally on scientific, numerical or computational principles as well as being very quantitative, while others involved the use of subjective judgement, and thus were more qualitative in nature. Others also involved the use of experience and heuristics, and were thus akin to rule or knowledge based systems.

The recent past, has also witnessed a significant amount of emphasis being placed by researchers and designers on the provision of integrated design systems, that would provide facilities beyond the traditional computer aided design (CAD) systems, which are predominantly geometry based. Such systems are looking at providing support for the designer at the early stages of design, as well as supporting co-operative (team based) design. Section 2.3.16 of chapter 2 shows the details of the review done on this subject, including on-going activities in the UK.

Concurrent with the research into design models, methods and systems, another research was also embarked on. This involved a detailed investigation of concurrent engineering in relation to engineering design. Within the last three to five years, the term concurrent (sometimes referred to as forward, simultaneous or parallel) engineering, has been pervasive in various technical literature and magazines. Many companies have also been using it as a marketing phrase, and several definitions have been proposed as representing concurrent engineering. The research into concurrent engineering, involved issues relating to the various definitions of concurrent engineering, principles and goals of concurrent engineering and impact of product teams and management. The advantages of concurrent engineering over the traditional serial and the 'over the wall' approach to product development, were strongly emphasized. Current activities as regards the realisation of concurrent engineering, including tools and software development efforts aimed at supporting the concurrent engineering philosophy, as well as the development of design frameworks for its implementation, were also extensively studied. The research carried out on concurrent engineering has been discussed in detail in section 2.4 of chapter 2.

The foregoing represents the research activities that led to the evolution of Design Function Deployment, DFD. The focus of DFD is in the provision of a generic design framework for the realisation of concurrent engineering. DFD emerged with the consideration of the many benefits provided by Quality Function Deployment, Design Models, Methods and Systems and Concurrent Engineering. These benefits are discussed in the proceeding sections.

3.2 KEY FEATURES OF QUALITY FUNCTION DEPLOYMENT

The key features and benefits of Quality Function Deployment (QFD) can be discussed under the following headings [202]: (a) General features, (b) Customer focus, (c) Reduction of implementation time, (d) Promotion of teamwork and (e) Documentation.

3.2.1 General Features

(1) The use of charts and matrices for capturing customer requirements and translating them into design requirements, represents the fundamental feature of QFD. In the subsequent stages of the QFD process, the design requirements are translated into parts characteristics, which are translated into manufacturing process parameters in the third stage. The fourth stage then involves the translation of the manufacturing process parameters into production operations parameters.

(2) The above process enables traceability in the design process through the cascade of the QFD stages, with the links acting as a connecting thread running from customer requirements to production planning.

(3) Another key feature of QFD, is the capture of alphanumeric (textual) data about

the design in progress. Such design information include those representing the intent of the design, its evolution process, as well as the rationale (reasonings) behind decisions made in the design process.
(4) QFD also assists in setting design goals and targets.
(5) Another principal feature of QFD, is the capability for the enhancement of quality, planning of quality and designing quality into the product, process or system under development.
(6) The use of QFD, helps the design team to focus on customer satisfaction, as well as when possible, customer delight.
(7) QFD provides a useful basis for the implementation of the Total Quality Management (TQM) strategy within an enterprise.
(8) The QFD technique also provides the capability for product planning, early in the development process.
(9) The cascade of the QFD stages helps to also provide a structured approach to designing quality into a product or system.
(10) The QFD charts act as databases for building and storing design knowledge and information during the product development process.
(11) QFD also provides a framework for taking into account, the different customer groups (interest groups) that might exist in the development of largescale/complex products, process or systems.
(12) The use of QFD helps to identify key attributes of a product that really matter under particular circumstances of requirements.

3.2.2 Customer Focus

(1) The use of QFD helps the designer/design team to adequately focus on customer requirements and ensures the deployment of the voice of the customer throughout the design process. This ensures the formal establishment of customer requirements and proper definition of the design problem.
(2) QFD enables the determination of importance ratings of customer requirements and design requirements using the 'quality plan' and the relationship matrix charts respectively. This process enables the prioritisation of both customer and design requirements.
(3) Within QFD, competitive information can be effectively used to assist in competitive benchmarking of own/existing products against those of competitors.
(4) QFD also provides a framework for responding to changing customer requirements, as well as helping to prioritize resources and identifying items that can be acted upon.
(5) The use of the QFD charts helps in structuring resident experience (corporate/company knowledge) and information.

3.2.3 Reduction of Implementation Time

(1) The use of QFD leads to decreases in mid-stream and downstream design changes, as downstream issues are taken into account during the product planning stage.
(2) It also helps to limit post-introduction problems and enables designing product right first time.
(3) The use of QFD also helps in avoiding future development redundancies as well as identifying future application opportunities.
(4) The use of the QFD technique also helps to surface missing and/or ignored assumptions, throughout the product development process.

3.2.4 Promotion of Teamwork

(1) The activities performed using QFD is consensus based and the process assists in more honest and objective decision making. This results in better teamwork.
(2) The development process in QFD helps to identify actions at interfaces as well as ensuring communication across these interfaces between departments and divisions within companies. This results in significantly improved communication.
(3) The QFD process helps to create a global view out of details.

3.2.5 Documentation

(1) The QFD process including the resulting charts enables the documentation of the rationale for the design decisions made, represented in the form of the generated textual design information.
(2) This makes it easy to assimilate the documented design information.
(3) The documentation process using the QFD charts helps to add structure to the design information, enabling adaptation to design changes, and thus providing a 'living document'.
(4) The QFD process also provides a framework for performing sensitivity analysis and the examination of what if scenarios.

In the development of Design Function Deployment (DFD), the key features utilised from QFD include the following :

(i) The focus on customer requirements as well as the design problem, is also a key feature included in DFD. DFD also uses similar charts to those of QFD, which have been adapted to suit the design process in DFD.
(ii) The concept of linking the charts is adopted. In DFD, this link runs through from stage 1 to stage 5 of the design process.
(iii) The setting of target values for the design requirements in QFD is also adopted in DFD. This aspect has however been extended within DFD to support the Robust Engineering Design (RED) exercise.
(iv) The determination of importance ratings throughout the deployment process in

using the QFD charts, is also adopted in DFD. In DFD however, unlike in the QFD process, where the ratings are used to screen design functions from one stage to another, they are used for focussing and guiding the design process. All the derived design functions are carried on to subsequent stages.

(v) The use of charts to store design information is also adopted in DFD. Here, the DFD charts and the information contained in them, represent core part of the textual model within the DFD product modelling environment.

3.3 KEY FEATURES OF DESIGN MODELS, METHODS AND SYSTEMS

The important features of design models, methods and systems that contributed to the evoluation and development of Design Function Deployment are discussed under the following headings : (a) The design process, (b) Design methods, (c) Design classification, (d) Product classification, (e) Design models, (f) Design activities, (g) Design for quality and (h) Design systems.

3.3.1 The Design Process

(1) The structured and systematic process of design helps to give insight into the multiple facets of the nature and features of the design process, such as : (i) Opportunistic tendencies - Top down, bottom up and middle out (as in QFD), (ii) Exploratory process of design, (iii) Investigative process of design, (iv) Creative process of design, (v) Logical process of design, (vi) Decision making process in design, (vii) Iterative process of design and (viii) Interactive process of design

(2) An important feature of design is the representation of design phases and stages in a planned and controlled manner. This helps in the control of the design process as well as providing a framework for the implementation of concurrency in design.

(3) An examination of the different approaches and models of the design process helps to give insight into the stages of thought and phases in design. This is exemplified below.

Table 3.1 Design Phases

	Types of design phases			
Stages of thought in design	Feasibility	Divergent	Abstract	Conceptual
	Preliminary	Transformation	Pseudo Concrete	Embodiment
	Detailed	Convergent	Concrete	Detailed

(4) The investigation of the design process, also helps in the understanding of the nature of design goals, the interaction between design goals and the strategies for controlling interacting design goals.

3.3.2 Design Methods

(1) the research into design methods helps to (i) highlight the importance of design methods that can be used during the design process, (ii) highlight the function of design methods and (iii) give insight into the nature, goals, general applicability, origins and classifications of design methods.
(2) The use of appropriate design methods helps to (i) support and augment the design activity in each of the design phases and stages, (ii) relieve the designer/design team of mundane tasks, improve design speed as well as enabling flexibility and versatility during the design process, (iii) support the multiple representation of the knowledge, information and characteristics about the design artifact (product).

3.3.3 Design Classification

In order to adopt a right approach to design and to use appropriate design tools and techniques, it is needful to identify the nature of the design problem under consideration. A key feature of research into design is the insight given into the nature or types of design problems. These include : (i) Routine Designs, (ii) Redesigns (Adaptive Designs and Variant Designs), (iii) Non-Routine Designs (Innovative Designs and Creative Designs).

3.3.4 Product Classification

Products resulting from the various types of designs shown in section 3.3.3 above, cab be classified into the following : *Static products, Dynamic products, Overconstrained Products, Underconstrained products, Clean sheet products (genesis, radical and new products), Market segment entry products and Customised products.*

This classification, enables the designer/design team to identify the status and classification of the product being designed. This enables appropriate response to the use of available techniques, new materials and technologies as well as proactively responding to both customers and competitors.

3.3.5 Design Models

These represent one of the fundamental aspects of research into design theory and methodology. It also represents how design is done in practice as well as how it could be done. This research into design models thus helps to give insight into their nature, types and characteristics. They can be categorised into :
(1) Prescriptive models - They consist of both design process based and product attribute based. They also (i) prescribe how to carry out design, (ii) emphasis is on a structured and planned approach to design and on proper problem definition and (iii) encourages improved ways of doing design.
(2) Descriptive models -relates to designer actions and activities during design and

(3) Computational models - involving the use of mathematical, symbolic and artificial intelligence techniques.

3.3.6 Design Activities (Tasks)

During the design process, irrespective of the particular design model employed, many design activities (tasks) are performed in realising a tangible product, process or system. Such design activities generally emphasized, include the following : Analysis of the problem (problem investigation and definition), Synthesis, Evaluation, Decision making, Optimisation, Search/Data Collection, Revision of design, Communication, Modelling, Decomposition, Simulation, and Documentation.

These activities represent what is termed the anatomy of the design process.

Design activities also consist of the following key features
(i) the use of decision strategies and actions such as combining, standardising, transferring, modifying and simplifying of design parameters, parts, components or subsystems in resolving interactions between them during the design process.
(ii) the emphasis on the need to develop performance specifications in a solution neutral form (i.e. without reference to any particular design solution).
(iii) the emphasis on the need for the expansion, exploration and investigation of the design solution space and
(iv) the emphasis on design innovation

3.3.7 Design For Quality

A major development over the last few years in the design and manufacturing industry, is the increasing demand for quality products and focus on the quality function in design. This factor has shifted emphasis from traditional quality assurance issues to the need for tools and techniques that can be employed to ensure the design of quality into a product, from the perception of the customer and user. Key issues to be addressed here involve :

(1) The need to design product for robustness/designing against variation in manufacture and use environment.
(2) The need to focus on ensuring quality by design.
(3) The need to encourage the use of structured off-line methods for robust design involving statistical techniques like Taguchi methods, experimental design, response surface methods, etc.
(4) The use of design principles (axioms) to guide and govern decision making in design.
(5) The need to focus on the reduction of loss to society by design, as well as the need to focus on the improvement of the value society places on the quality of the product.

3.3.8 Design Systems

Besides the foregoing features, the research into the development of design system has raised issues bodering on the requirements demanded of such systems by designers and engineers. Such features currently demanded of design systems include :

(1) The need for integrated and robust design systems that can support the handling of varying forms of design information right from the early stages of conceptual design through manufacturing process planning to production, packaging and eventual disposal.

(2) The need for an integrated and coherent product modelling environment.

(3) The need to provide capabilitties for design tradeoffs and resolution of conflicts as well as constraints management.

(4) The need for a product database management system.

(5) The need for an interactive user interface system.

(6)The need for a shared representation of design information and for designing from various perspectives and views.

The foregoing discussion in this section on the key features of design models, methods and systems, represent the issues that contributed to the evolution of DFD. The major contributors to the devlopment of DFD, amongst these, are :

(i) Design models, particularly those based on the prescriptive approach. The other aspects of the descriptive and computational approaches are also accounted for. The design process in DFD is represented by a prescriptive model.

(ii) Design methods - In the development of DFD, the necessary design methods that will support design activities throughout the life cycle of the design process of DFD, were accounted for in the system.

(iii) the representation of the design process in phases and stages in a planned, structured and controlled manner, was also a key factor in the development of the DFD philosophy, and design model

(iv) Design for Quality - this represents a major ethos in the development of DFD, covering aspects dealing with the integration of the quality functions into both the design process and the product

(v) Design systems - relating to aspects of developing an integrated design system encapsulating necessary design methods, knowledge-bases and databases to support the design activities performed in DFD.

3.4 KEY FEATURES OF CONCURRENT ENGINEERING

From the research into concurrent engineering, the key features considered for the development of Design Function Deployment, are discussed under the following categories : (a) General issues, (b) Integration, (c) Reduction of lead times, (d) Customer focus and (e) Team support.

3.4.1 General Issues

The key features considered here include :

(1) The need to provide a capability for the creation and development of design solutions that are not only functional, but easy to manufacture and assemble.
(2) The encouragement and provision of the capability to use electronic mock-ups in a concurrent (simultaneous) manner during product development.
(3) The encouragement to use new materials and technologies.
(4) The need for the use of computer hardware and software.
(5) The need to focus on quality, cost and delivery.
(6) The need to emphasize the employment of software support to realise and implement the benefits of concurrent engineering.

3.4.2 Integration

This involves :

(1) Bringing other downstream lifecycle issues upstream. Emphasis here is on the integration of downstream manufacturing and use considerations into design at the early stages of product development
(2) Emphasis on the need for synchronisation of design information, involving the use of a common and integrated database vis-a-vis the master modelling concept. Thus enabling the sharing of common design information and removing interfaces between disciplines.
(3) Adopting a concurrent optimisation approach to the design of a product and its manufacturing process.
(4) The concurrent consideration of all constraints, early in the design process.
(5) Ensuring the integration of all activities that influence product development.
(6) The use and integration of concurrent design tools and techniques, within a design system.

3.4.3 Reduction of Lead Times

Other major features and goals of concurrent engineering are :

(1) The emphasis on concurrent/parallel design within the product development process, and the use of appropriate project management techniques.
(2) Ensuring the reduction of design lead time.
(3) Emphasis on the need to manage and control the design process to ensure concurrency as well as optimal utilisation of manpower, resources, tools and time.

3.4.4 Customer Focus

The adoption of the concurrent engineering approach to product development also helps in enabling :

(1) Focus on customers and rapid response to their expectations by the design and product development team.
(2) The establishment of customer requirements and design specifications and constraints.
(3) The performance of competitive benchmarking of both the product as well as the design process against those of competitors..

3.4.5 Team Support

Key features and principles of concurrent engineering also include :

(1) Provision of a framework for co-operative and collaborative design teams.
(2) Provision of an effective communication framework.

The foregoing represent the key features of concurrent engineering that were considered in the development of Design Function Deployment. The major contributiing features include :

(i) the need to cater for and integrate design tools and techniques for the creation, analysis and development of desig nsolutions as well as other life cycle issues, such as manufacture, assembly, serviceability, recyclability, etc.
(ii) the use of new materials and technologies in the early stages of product development.
(iii) carrying out the concurrent (parallel) design of both the evovling product and its associated manufacturing processes.
(iv) the need to capture all constraints from the various interest groups in an organisation that will influence the product development process.
(v) bringing upstream to the early stages of design other downstream and life cycle issues.
(vi) the need to focus on customer requirements and establishing more accurately design specifications and constraints.
(vii) the provision of a framework and communication protocols to support co-operative and collaborative design teams.
(viii) synchronisation of all design information both (textual and geometric) as well as the use of a common geometric database, based on the 'master modelling' concept.

3.5 SUMMARY

The discussions in this chapter have focussed on the key features of the three main research areas namely : Quality Function Deployment, Design Philosophies, Models, Methods and Systems, and Concurrent Engineering, which contributed to the evolution and development of Design Function Deployment (DFD). Quality Function Deployment (QFD) was the starting point, and contributed to the aspects relating to stage 1 of the DFD design model involving the process of capturing, categorising and prioritising (quality plan) customer requirements that are then translated into specifications and constraints, along with setting their target values. This enables the product development team to focus on the important requirements of the customer. The QFD charts also acted as precussors to the DFD charts, which are used for capturing and documenting textual design information. The deployment stages in QFD, also acted as a starting point to the development of the DFD design model. In the case of design philosophies, models, methods and systems, the main contributing features are (i) expanding the deployment stages of QFD into a more robust prescriptive design model, (ii) developing specifications in a solution neutral form, (iii) generating the design solution space, and (iv) the use of design methods. These features together contributed to the development of the DFD design model, as well as some of the design methods incorporated into it. Concurrent Engineering, the third area, represents a predominant contributor both in terms of philosohphy and development of DFD. The key features of concurrent engineering that contributed to DFD, are : (i) the integration of all life cycle aspects of the product development process as well as the corresponding tools, (ii) providing a framework for paralleling the design process, (iii) integration of concurrent design tools and (iv) the use of new materials and technologies in design, amongst others. This led to the development of the overall structure of DFD consisting of a design model, auxiliary design tools and techniques and supporting knowledgebases and databases that capture the life cycle issues (manufacture, assembly, testing, environment, service, maintenance, cost, quality design, etc) of the resulting product.

The foregoing show the chronological sequence of the evolution of DFD. Besides the three main contributors to DFD discussed above, a lot of innovative thinking and ideas of the author were drawn upon in the full development of Design Function Deployment, details of which are discussed in the subsequent chapters of this thesis.

CHAPTER 4

REQUIREMENTS FOR A CONCURRENT ENGINEERING DESIGN SYSTEM

4.1 INTRODUCTION

This chapter discusses the issues that relate to the requirements for a concurrent engineering design system. It begins by discussing the need for a concurrent engineering design system and then looks at the goals and objectives of such a system. The remaining part of the chapter then focusses on the details of the requirements for a concurrent engineering design system. This is discussed under the broad headings of :

- ¤ General requirements
- ¤ Designers, Users and System requirements

In discussing the requirements for a concurrent engineering design system, the issues to be addressed in the implementation and realisation of concurrent engineering as discussed in section 2.4 of chapter 2, are also accounted for, under these two headings.

4.2 THE NEED FOR A CONCURRENT ENGINEERING DESIGN SYSTEM

There has within the last 5 to 10 years been the emergence of several computer aided design (CAD) systems developed supporting both 2D and 3D designs. These systems although have grown in popularity, however still exhibit some fundamental limitations and hence do not fully satisfy the needs of engineers and designers. Such limitations include their inability to support the designer throughout the whole life cycle of the design process. They are mostly geometry oriented, and tend to come into play when the major design decisions have been made between the problem definition and requirements analysis stage and conceptual design. Furthermore, conventional CAD systems work mainly with primitive geometric objects (e.g. points, lines, etc) while designers usually think and work with higher order design objects (e.g. components, bearings, gears, walls, sensors, motors, etc). They do not support design synthesis and information processing, and as a result, mainly act as electronic drawing boards. Another limitation of conventional CAD systems, is their inability to represent intent of the design (the expression of the original objectives of the design) as well as the rationales behind the design decisions made. These systems do not also provide the capability to accomodate and represent downstream manufacturing and process planning information in an integrated manner. Such information are usually in a very limited way contained within separate CAM/CAE systems, and thus information has to be transcribed between the CAD and the CAM/CAE systems. It is widely known and accepted that although these conventional systems are able to handle the graphical and geometrical information of designs, they generally do not also possess any capability to represent in an integrated manner, the vast amount of textual (alphanumeric data) information generated for a design throughout the design life cycle. The tendency is that the textual and geometric information about a design (if they are fully available), tend to be contained within separate locations resulting in the fragmentation of such information.

The consideration of the above limitations of conventional/traditional CAD systems in addition to the demands currently made by designers in industry, makes imperative the need to develop a comprehensive design system to not only cope with the above needs but also provide a platform to support the concurrent engineering principles and practices and thus enable the production of innovative, robust and customer led products. Such a system should provide a framework and infrastructure for engineering information processing. It should also support a more powerful product information representation for designers and manufacturing engineers as well as all the product life cycle perspectives from requirements analysis, product planning through conceptual design, detailed design to manufacture, packaging, installation, use, service and eventual disposal. Another driving force for a concurrent engineering design system is attributed to the fact that, a large number of specialised design tools are required during the design phase to meet the needs of the complex design tasks of design analysis, design synthesis, manufacturability, assemblability, cost estimation, process planning, etc., necessary for the examination of the product's life cycle issues. These tools however in order to ensure that the design can be globally optimised, need to be integrated within a design environment where they can all interact and/or cooperate as well as providing a variety of functions and services to designers. Thus obviating the need for a great deal of time (which can be spent on design) to be spent on integrating these diversified tools in a single working CAD environment [203, 204].

4.3 GOALS OF A CONCURRENT ENGINEERING DESIGN SYSTEM

The goals of a concurrent engineering design system, represent the purpose and the 'raison d'etre' of modern design systems. Such goals relate to issues associated with both the design and product development process and the resulting artifact or product. The major issues which constitute these goals are discussed below [205, 206].

(1) A major goal of a concurrent engineering design system is to enable the development of quality products at prices that customers are prepared to pay. Subgoals under this goal include : (a) the maintenance of focus on quality on both the design process and the product and (b) the maintenance of team morale, communication, co-operation and a sharing culture.

(2) Another goal of a concurrent engineering design system is to enable early time to market of products and the reduction of product development lead times. Sub-goals under this goal are : (a) the enablement of designers and design teams to work concurrently in harmony, (b) the improvement of access to design information, (c) the improvement of communication among co-designers and within design teams and (d) the enablement of rapid evaluation of design alternatives.
(3) The satisfaction of customer requirements and expectations, represents another major goal of a concurrent engineering design system. The subgoals under this goal include the need to : (a) understand customer requirements and expectations, (b) articulate customer requirements and expectations in a way that makes them easy to implement and (c) ensure that the focus on customer requirements and expectations is maintained throughout the development process.

Other goals of a concurrent engineering design system include :
(4) The continuous improvement of the design and product development process.
(5) The integration of all related lifecycle issues throughout the product development process.

The essence of concurrent engineering is to be able to develop complex products with good quality, within short lead times and at low cost.

4.4 REQUIREMENTS FOR A CONCURRENT ENGINEERING DESIGN SYSTEM

This subsection discusses the whole body of issues that constitute the desiderata of a concurrent engineering design system. They are discussed under two main headings of (a) General requirements and (b) Designers, Users and System requirements.

The **General requirements** are discussed under the subheadings of : (i) User friendly requirements and (ii) Concurrent engineering support requirements.

The **Designers, Users and System requirements** are further discussed under the subheadings of :
(i) User interaction requirements, (ii) Design process management requirements, (iii) Design artifact (product) requirements, (iv) Design team support requirements, (v) Manufacturing requirements, (vi) Product life cycle requirements, (vii) Design life cycle requirements, (viii) Computer software requirements, (ix) Requirements capture requirements, (x) Design tools and techniques requirements and (xi) Design knowledge capture requirements

4.4.1 General Requirements [207 - 214]

User Friendly

These requirements are associated with the demands echoed by designers and users of design systems as they interact with them. They require that :

(1) The system should be easy to use and update.
(2) The system should exhibit designer orientedness, openness, configurability, simplicity and efficiency.
(3) The system should support interactiveness and flexibility in the design process.

Concurrent Engineering Support

It is a fundamental requirement that a concurrent engineering design system should support the philosophy, principles and practices of concurrent engineering. Hence a concurrent engineering design system should :
(1) Fully support concurrent engineering practices and be flexible to allow a concurrent engineering approach to design.
(2) Contain several facilities which coordinates the entire design process that complies with the concurrent engineering philosophy, i.e. (a) enabling the integration of all lifecycle aspects of the product (both upstream and downstream) and (b) enabling paralleling of the design process.
(3) Provide an interactive environment for the design process, that enables designers to control available resources in the form of data, knowledge, methods and algorithms.
(4) Enable designers/design team to work concurrently and to consider the interactions and tradeoffs among different and conflicting requirements.
(5) Improve communication among members of the design team.
(6)Provide a group problem-solving environment in which knowledge-based systems can contribute to the design process.
(7) Provide an environment that supports the synchronisation of design information.

4.4.2 Designers/Users/System Requirements

This subsection discusses the requirements (needs, perceptions, wants, expectations) of designers (individuals or groups) which are expected from a computer based concurrent engineering design system. These requirements which represent the issues that have been consistently highlighted by many designers, engineers and researchers, are discussed under the broad headings of : (1) User interaction requirements, (2) Design process management requirements, (3) Design artifact (product) requirements, (4) Design team support requirements, (5) Manufacturing requirements, (6) Product life cycle requirements, (7) Design life cycle requirements, (8) Computer software requirements, (9) Requirements capture requirements, (10) Design tools and techniques requirements and (11) Design knowledge capture requirements.

User Interaction Requirements

The requirements here are that :

(1) The system should support interactive design, providing the designer with the flexibility to be actively involved in the evolution of the design.
(2) The system should have a user friendly and easy to use graphical user inteface. This should enable the designer to traverse from one tool to another and from one design state to another, with minimal effort.
(3) The system should provide a user interface for visualisation of multiple and alternative views of the design.

Design Process Management Requirements

These are the requirements relating to the needs of managing the design process in a controlled and planned way, and they are discussed below.

(1) Design Process Control.
These requirements relate to the control of the design process, and they require that:
(a) The system should contain facilities for formalising various kinds of design goals as well as for the flexible control of the design process.
(b) The system should provide a control mechanism for the design events during the entire design process.
(c) The system should provide design process control and communication facilities such as version control, status control, design database hierarchy control, design validation and design notification.
(d) The system should support scheduling and control of design activities performed during design.
(e) The system should support tools for truth maintenance, that is, preserving deductions (including design decisions made) and retaining the justifications for the deductions, during the design process.
(f) The system should contain a prescribed minimum number of steps in its design model and be flexible enough to allow specific paths to be followed by the designer/design team.
(2) Design Iteration - The system should enable and facilitate iteration in decision making during the design process.
(3) Design Communication
(a) The system should enable and facilitate high quality, sustained and coherent communication concerning the reasoning about the design situation, the design and its implementation.
(b)The system should provide communication facilities for the design tools and data within the system as well as supporting the communication of design data in various formats.

(c) The system should provide facilities for communication between designers as well as between designers and the design tools.
(d) The system should support communication between distributed design agents (representing different design and/or engineering disciplines) involved in the product development process, and also provide facilities to improve access to design information. (e) The system should provide a computing environment that enables sharing of electronic information freely among team members in order to facilitate sharing of ideas, iteration of designs and propagation of design changes.
(4) Partitioning of Design Tasks - The system should provide effective procedures and techniques for partitioning design tasks into perceivable steps.
(5) Feedback
(a) The system should provide facilities for accomodating feedback of evaluation results to guide the design generation process.
(b) The system should provide continuous feedback based on incremental analysis of the design as it evolves.
(6) Design Solution Space Generation Requirements
(a) The system should provide effective procedures and techniques for the creation of design solutions which are systems of perceivable units and which are easy to modify.
(b) The system should provide facilities to enable the generation and exploration of the design solution space.
(c) The system should provide effective strategies for ordering of design decisions such that the designer has opportunities to explore the solution space before deciding on a particular solution.
(d) The system should provide the facility for rapid generation of ideas and creation of design alternatives and conceptual solutions.
(e) The system should enable the performance of multi-criteria optimisation of the evolving design in a cohesive manner.
(7) Abstraction of Design Stages
(a) The system should enable designers to work on different abstract levels of the corresponding design phase, as well as accomodating the representations of the abstractions of the design process.
(b) The system should accomodate the representation of the intermediate states in the design process.
(c) The system should support modelling both of the design process and the product being designed.
(8) Multiple Design Models
(a) The system should accomodate alternative design styles and support several design models.
(b) The system should support product models needed for communicating information across disciplines. Thus supporting multiple levels of abstraction and different

functional views as well as multiple levels of geometric representations.

(9) Design Data Management (Transfer/Exchange)

(a) The system should provide facilities for transfer of design data between design tools.

(b) The system should support instantaneous, accurate and flexible data interchange among CAD/CAM/CAE systems.

(c) The system should enable transfer of design information between CAD/CAM and CAE systems, as well as design configuration control systems. Translations should be minimised in order to preserve the original data as much as possible.

(d) The system should provide facilities for information transfer between various design agents (disciplines) involved in the product development process.

Design Artifact (Product) Requirements

(1) Focus on Quality

(a) The system should enable the designing of quality into the product.

(b) The system should help in maintaining focus on quality both from the perception of the external customer/users as well as the internal customer (manufacturing, production, etc).

(c) The system should enable the robust design of products and/or systems.

(2) Decomposition of Products - The system should provide effective procedures and techniques for the partitioning of large scale and complex designs into reasonably smaller components

(3) Abstraction of Product Design

(a) The system should provide for the representation of the design solution at various levels of abstraction (e.g. function, structure and behaviour of product), starting from the most general and ending at the most specific.

(b) The system should support feature based design approaches

(4) Multiple Design Views

(a) The system should provide support for multiple views of the design (textual, geometric, functional, behavioural, structural/physical).

(b) The system should maintain multiple representations to support the different perspectives and design activities at multiple levels of details.

(c) Integration of Textual and Geometric data - The system should handle a diversity of numeric, graphic, symbolic and geometric design parameters in an integrated manner.

Design Team Support Requirements

(1) The system should provide the capability to support, model and control the co-operative process of design teams.

(2) The system should allow multiple designers to work on a design simultaneously.

(3) The system should support multiple design experts and knowledge based systems.

(4) The system should support negotiation tools, such as a common simulation environment and on-line graphical group interaction, in order to expedite the design change negotiations.
(5) The system should provide a framework to support the different forms of transactions carried out by members of the design team, such as computations, communications, negotiations, decision making and so on.
(6) The system should support design information sharing between members of the design team.
(7) The system should support an integrated, shared, dynamic and domain neutral (independent) representation of the evolving design.
(8) The system should provide facilities to support collaborative participation of teams in critiquing designs, planning and executing design changes, recording design rationale, maintaining agendas of unfinished business such as evaluations of design, suggestions about goals and constraints of design changes, effects of proposed changes on other aspects of the design, changes under consideration or implementation, and the divisions of responsibility.
(9) The system should support group interaction of engineering and design teams having different technical backgrounds and geographical locations. These interactions could be centralised or distributed, synchronous or asynchronous, technical or administrative, and involve both people and machines.

Manufacturing Requirements

(1) The system should provide facilities for supporting manufacturing process and production planning and modelling.
(2) The system should provide the capability for intelligent computer aided process planning.
(3) The system should provide the capability for manufacturing process selection.

Product Life Cycle Requirements

The system should enable the consideration of all the life-cycle issues of the product development process, i.e. design, manufacture, use and disposal.

Design Life Cycle Requirements

(1) The system should support the designer throughout the whole design process. That is, it should provide for computer support of all the design phases, i.e. concept, embodiment, and detail design stages.
(2) The system should be a well organised software infrastructure which properly guides designers through the entire design process so that they can reach the design of maximum quality that can be produced at the lowest cost in the shortest time.

Computer Software Requirements

(1) The system should enable the use of computer hardware/software as enabling and intelligent aids in the design process.
(2) Open Architecture
(a) The system should provide an open and extensible architecture, be more domain independent and allow the addition of new tools and information.
(b) The system should be integrated with existing CAD/CAM/CAE systems in a coherent way to provide a variety of functions and services to designers.
(c) The system should be such that it is easy to integrate necessary tools into it, in such a way that minimal effort is required to move from one tool to another.
(d) The system should be capable of supporting a large population of tools and corresponding design and manufacturing data models.
(e) The system should accomodate a distributed computing environment with appropriate levels of security based on open networking architecture to allow teams to work transparently.
(f) The system should have a common interface structure so that users can move from one computer application tool to another with minimal learning, i.e. allowing the user to concentrate on the tool and not the interface.
(3) Software Infrastructure
(a) The software infrastructure of the system should support both procedural and object-oriented software development approaches as well as necessary knowledgebase and database systems' development.
(b) Knowledgebases
(i) The system should provide facilities for the generation, representation and capture of the knowledge (building blocks) within a particular design domain from which solutions can be constructed.
(ii) The system should support the efficient use of design knowledge, methods and procedures.
(iii) The system should have knowledgebases for storing and retrieving corporate knowledge and other relevant design information.
(c) Design databases for information storage and retrieval

The system should :

(i) contain a consistent database management system, a uniform graphical user interface and a design management facility.
(ii) provide facilities for browsing and navigation aids when searching for information.
(iii) provide facilities for storage and retrieval of design information.
(iv) provide facilities for archiving (storing as well as retrieval) of corporate design history (knowledge) of previous designs.

(v) support large product as well as component information databases.
(vi) have a central intelligent database which provides a flexible way of accessing, displaying, interpreting and distributing (sharing) product information to all team members.
(d) Materials Selection - The system should support materials selection in the design process in an integrated and concurrent manner.

Requirements Capture Requirements

(1) Requirements Analysis and Establishment - The system should :
(a) provide effective procedures and techniques for the precise establishment of customers/users' requirements and desires of a product :
(b) help in the understanding of customer requirements and expectations and in articulating them in such a way that makes them easy to implement.
(c) help to ensure that the focus of the designer/design team on the customers' requirements and expectations is maintained throughout the product development process
The system should enable designers/design team to be more responsive to customer needs and be more capable of improving time to market.
(e) The system should enable the translation of the customer's vision (which may be emotional, vague, subjective and qualitative statements) into physical reality by the design team.
(2) Requirements Representation
(a) The system should provide facilities for the representation of requirements and function domain of a given design problem.
(b) The system should provide facilities for the creation, revision and change of design requirements and specifications.

Design Tools and Techniques Requirements

(1) The system should provide effective strategies and techniques for selecting, developing and using computer design tools for all algorithmic activities within the design process.
(2) The system should provide effective strategies and techniques that help designers to be more creative and to make value judgements.
(3) The system should provide evaluation strategies for assessing design proposals.
(4) The system should enable various forms of analyses to be performed throughout each design phase and accomodate a variety of design analysis tools.
(5) The system should provide support for rapid handling, analysis and evaluation of multiple design alternatives.
(6) The system should provide facilities for generating designs, establishing criteria for comparing them, evaluating them according to the criteria, choosing an acceptable design and revising the design choice if the chosen design is unsatisfactory.

(7) The system should support all necessary engineering analysis and optimisation tools

(8) The system should provide support for computer aided geometric and solid modelling tools.

(9) The system should provide a high degree of flexibility to promote creativity and quality engineering and design, in the optimisation of subsystems, components and parts of a design.

(10) The system should support design tools and databases that provide a variety of evaluation and trade-off assessments. Thus allowing design iterations, but with the design team maintaining consensus over global changes.

(11) The system should support design methods that can be used by designers/design team throughout the product development process.

(12) The system should support concurrent engineering tools that are able to incorporate multiple perspectives, support multiple stages, and work with multiple participants.

Design Knowledge Capture Requirements

(1) Capture of design history.

(a) The system should provide facilities to support and capture the progress and evolution of the design process and of the degree of completion of a design.

(b) The system should accomodate the representation of both the design decisions as well as the rationales for the design decisions.

(c) The system should enable and facilitate the documentation of design logic and decisions.

(d) The system should provide facilities for capturing the design history including the record, tracing and review of design decisions.

(2) Capture of design intent and rationale

(a) The system should provide facilities for capturing the design intent.

(b) The system should enable the performance of effective data management, to preserve the design data relationships, capture the dependencies and the important characteristics of the design. The information infrastructure should enable continuous access and guarantee integrity of the product information.

(c) The system should support an integrated vision of how the product moves from design conception to manufacturing and other support processes at the later stages of the product development life cycle.

(d) The system should provide facilities for keeping track of the justifications (design rationale and intent) generated during design (or other engineering activities) both at the individual level and at the team level.

(e) The system should support electronic capture of design intent, rationale and evolution of a product from inception up to retirement.

(f) The system should support visualisation aids for initial sketching of design ideas.

This will be useful for exploring ideas, communicating information between other designers, establishing scale, proportions, clearances, etc, and for extending working memory which can be retrieved when needed, visually.

4.5 SUMMARY

In order to develop a concurrent engineering design system, it is needful to understand the requirements of such a system. The intent of this chapter therefore has been to fully elucidate and enumerate the requirements and demands of a concurrent engineering design system. These requirements have been articulated and discussed from the viewpoint of the designer or user of such a system, under several headings. These headings include user friendly requirements, concurrent engineering support requirements, user interaction requirements, design process management requirements, design artifact requirements, design team support requirements, manufacturing requirements, product life cycle requirements, design life cycle requirements, computer software requirements, requirements capture requirements, design tools and techniques requirements, and design knowledge capture requirements. In addition, the system requirements in relation to the software implementation of a concurrent engineering design system have also been discussed.

CHAPTER 5

DESIGN FUNCTION DEPLOYMENT

5.1 INTRODUCTION

Designing quality into products, processes or systems, exploiting the benefits of new materials and emerging technologies, using necessary CAD/CAM/CIM tools and implementing concurrent engineering practices, tools and techniques in a modern product development and manufacturing setup, represent fundamental prerequisites as well as the route to success in today's extremely competitive industrial climate. To achieve this, requires the development of a support environment which provides an integrated framework that ensures the concurrent maximisation of design and manufacturing innovation and other life cycle issues during product development. In this regard, Design Function Deployment (DFD) has been developed to satisfy the requirements of such a system, and as a comprehensive design system to meet the challenges of the 90's and the next century. To realize the full benefits of DFD, it will be developed as a computer based design system. There were three main contributors to the evolution and development of DFD, namely : (1) Quality Function Deployment (QFD), (2) Design Philosophies, Models, Methods and Systems and (3) Concurrent Engineering.

This chapter which represents the core of the thesis, will discuss what Design Function Deployment (DFD) is, its principles and goals, as well as the need for it. The structure of the DFD system, the design model and methods incorporated within it, will be fully elucidated. Further discussions will also include the capabilities and potentials of DFD, with regards to the generation of the design solution space, the explication of the three-dimensional design matrix encapsulated in it, the integrated product modelling environment within it, the process of capturing design knowledge during the design process as well as the concurrent management of the design process in DFD. The approach adopted in DFD as regards materials and manufacturing process selection and multi-criteria design optimisation, will also be discussed. The chapter will conclude by describing the process of designing using DFD in a manual form.

Design Function Deployment - Definitions

¤ Design Function Deployment, DFD, can be defined as a comprehensive design system, which incorporates the features of a prescriptive design model and associated design methods for the integration of manufacturing, use and other downstream issues into the design and thus enabling a 'Concurrent Engineering' approach to product/process development.

¤ The design model in DFD provides a systematic approach for the optimal translation of stated (explicit) and latent (implicit) customer requirements and designer intentions into identifiable design functions (specifications and constraints). It also helps to preserve traceability to the original customer requirements throughout the design, manufacture and use stages in the product/process. It ensures that the product is properly conceived at the design stage for manufacture and use. The customer requirements include the functional requirements and others such as durability, price, reliability, maintainability, cost of maintenance, aesthetics, safety, ease of use and environmental concerns.

¤ It enables the building of quality into the design and hence into the product/process, by developing a quality plan (prioritisation of customers' requirements) derived from the customers' requirements.

¤ DFD defines product/process development as the process of designing something that the customer wants and can be manufactured to a standard and price acceptable to both customer and manufacturer.

¤ By taking the design process through predefined stages with predefined standards for the outputs, DFD has a built-in Total Quality Management approach to product/process development.

¤ In DFD, the designer is presented with the opportunity to consider and resolve early (upstream) in the design cycle conflicting design requirements and constraints with other associated issues such as manufacturing, cost, maintainability, reliability, tooling, assembly and disassembly, market and sales considerations, etc, as well as the consideration of the mechatronics approach during the design process, in a concurrent manner. DFD thus provides a concurrent engineering framework and helps to shorten the product development process by reducing the number of design iterations and hence cost. It also ensures that by getting the design 'right first time', a high quality, low cost product which satisfies and possibly delights the customer, is developed [215, 216].

5.2 GOALS OF DESIGN FUNCTION DEPLOYMENT

The principal goals of DFD consist of the following : (1) to recognize the importance of customer requirements, (2) to record all relevant data, to ensure satisfaction of specifications, and for future reference, (3) to ensure the change from the 'over the wall' approach to team approach in design and product development, (4) to provide a Platform for 'Concurrent Engineering', (5) to generate the Design 'Solution space', (6) to facilitate Design Retrieval, (7) to maximise knowledge about performance at the design stage, (8) to minimise or eliminate downstream engineering changes, (9) to establish robustness of design, (10) to ensure reliability and safety, (11) to take advantage of new materials and technologies, (12) to ensure decisions are made after evaluating their cost implications, (13) to ensure quality through design and (14) to go into manufacturing with absolute confidence of performance and market appeal.

5.2.1 Recognising the Importance of Customer Requirements

Since the purpose of a product is to satisfy a need of a user, customer or society at large, it is important to understand and fulfill the need, and the success and failure of the product greatly depends on the degree of fulfilment. The aim of DFD is to ensure that all requirements of customers are captured as well as those of both external (i.e. customers from the society outside the company) and internal (i.e. customers within the company, e.g. the assembly department is a customer to component manufacturing department, who are customers to design). These requirements are then refined, classified and prioritized.

5.2.2 Recording All Relevant Data

This is done to ensure satisfaction of specifications and for future reference. There is a need therefore for a systematic top down method for recording the design process. During the design process, the relevant design information (both textual and geometric) that need to be captured include the following : (i) Customers' explicit and implicit requirements, (ii) Company's requirements, (iii) Manufacturing requirements, (iv) Use, abuse and maintenance requirements, (v) Statutory requirements, (vi) Any other requirements which form part of the specifications, (vii) Specifications and constraints derived from these requirements, (viii) Conceptual solutions or Architectures proposed, (ix) Layouts or Detailed solutions for each architecture considered, (x) Materials, Parts and Manufacturing processes for the layouts and (xi) Production Plans for large and small volumes of manufacture. The textual information are captured by the DFD charts, while the geometric information will be captured by the solid modelling system and design database.

5.2.3 Need for Change from The 'Over The Wall' Approach

In the traditional 'over the wall' approach, shown by Figure 5.1, product development is performed sequentially by various departments. Some deficiencies associated with this 'over the wall' approach include : (i) lack of effective communication, shifting responsibilities for faults and blaming other departments, (ii) a need to manage the interfaces between departments with inspection and controls, (iii) an inward looking focus which loses sight of the external customer and (iv) fragmentation of design data. Some benefits of the integrated team approach are : (a) better decisions are made because more options become available, (b) progress is quicker as the functional barriers are removed, (c) greater task focus which helps to ensure that the product is developed as specified by the customer, and (d) increased morale and motivation.

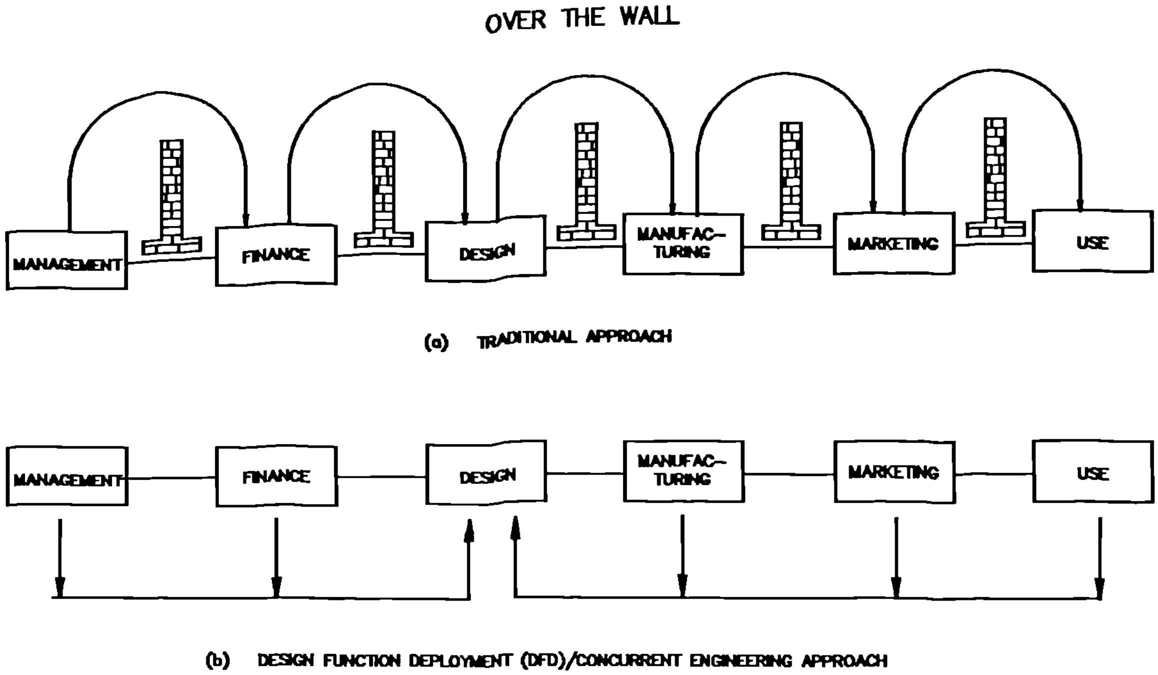


Figure 5.1 Traditional versus Concurrent Engineering Approach to Product Development

5.2.4 Essential Inputs for Concurrent Engineering

DFD emphasizes the need to include all key players involved in the product development process. This will include : (i) customers, (ii) design engineers, (iii) manufacturing engineers, (iv) marketing, (v) purchasing, (vi) finance, (vii) principal suppliers of manufacturing equipment, and (viii) principal suppliers of components. The team approach to design is further enhanced by the advent of computers as a member of the design team which enables the intimate coupling of the best characteristics of man and computer.

5.2.5 The Implementation of Concurrent Engineering

A major objective of DFD is to provide a platform for the implementation of concurrent engineering. In this regard, issues that need to be addressed, involve the following : (1) Establishing a focus on the client, customer or user of a product, (2) Integration of the organisation, (3) Use of design and product development teams, (4) Employee involvement and participative management, (5) Competitive benchmarking, (6) Focussing all activities on quality, cost and delivery, (7) Adopting a concurrent (parallel) product development process, (8) Integrating design of product with manufacture and support processes, (9) Establishing strategic relationships with suppliers, (10) Integrating computer aided tools and techniques, (11) Use of project management techniques, (12) Consideration of new materials and technologies, (13) Synchronisation of design information and (14) Use of computer hardware and software.

5.2.6 Generation of The 'Solution Space'

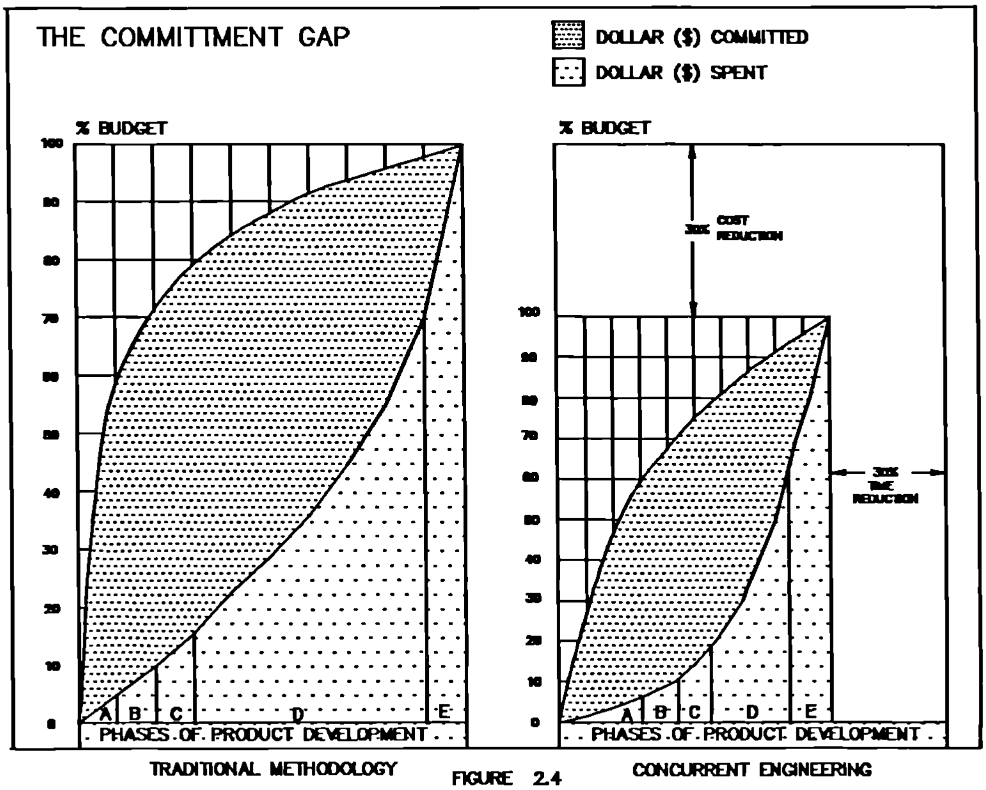
The act of designing can be described as the process of identifying a particular solution amongst a group of viable solutions to a design problem, that has been defined by a set of requirements. The design process can hence be seen as the mapping process between the 'requirements space R' and the 'solution space S'. This is discussed in detail in section 5.8 of this chapter. The mapping D is the design process. The mapping D:R to S can be a one to several or several to several mapping, and the process of designing should identify the optimal one that would suit a given set of circumstances. With the advent of the current CAD era where man and computer work as a team, one complementing the other, visualisation of the entire spectrum of viable solutions (solution space) is necessary before identifying the preferred solution.

5.2.7 Benefits of Design Retrieval

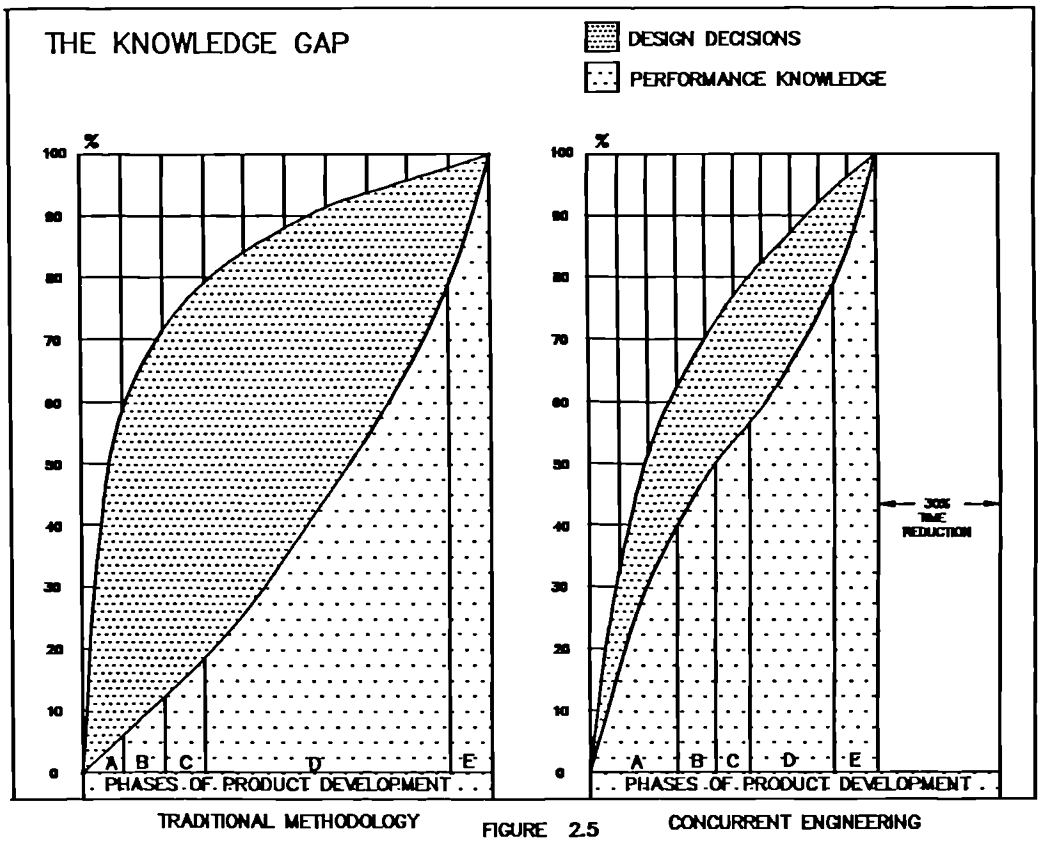
One of the key aims of DFD is to support the use of previous proven designs as well as using standard components and parts at the detailed design stage. The experience gained during the process of design in the past, can also be used in current designs, and current design experience can be retrieved for next generation designs if stored.

5.2.8 Maximise The Knowledge About Performance

In the traditional method nearly 70-80% of cost is committed during the early phase of product development as shown by the convex line in Figure 5.2a. The actual amount spent is shown by the concave line. In the use of DFD, the aim is to reduce the gap between the commited and the actual amount spent, by delaying critical decisions as much as possible. In Figure 5.2b, the knowledge on performance and design decisions during product development is shown in Figure 5.2b. In the traditional method, the gap between the design decisions and performance knowledge is large. In DFD, the aim will be to reduce the gap between design decisions and knowledge about performance, by integrating all groups involved in the product development process and making available upstream at the early stages of product development, knowledge about downstream issues that impinge on design of the product. The essence of this is to increase performance knowledge at the stage where significant costs are committed.



(a)



(b)

Figure 5.2 Cost Commitment and Maximisation of Performance Knowledge

Courtesy of Pareti [282].

5.2.9 Minimise Downstream Engineering Changes

Another aim of DFD is to reduce as much as possible costly downstream engineering changes of a product. In Figure 5.3 comparing engineering changes to the design of a product between a typical Japanese and western company, the number of changes at the design stage was higher in the Japanese company because of the concurrent (simultaneous) consideration of the downstream manufacturing and use processes at the design stage. Thus minimising or eliminating changes during launch and use, unlike in the western company which was still making changes at these stages. The minimisation of downstream engineering changes, has a positive knock on effect of maximising performance knowledge at the early stages of design. DFD hence by providing necessary life cycle design methods ensures that this is achieved.

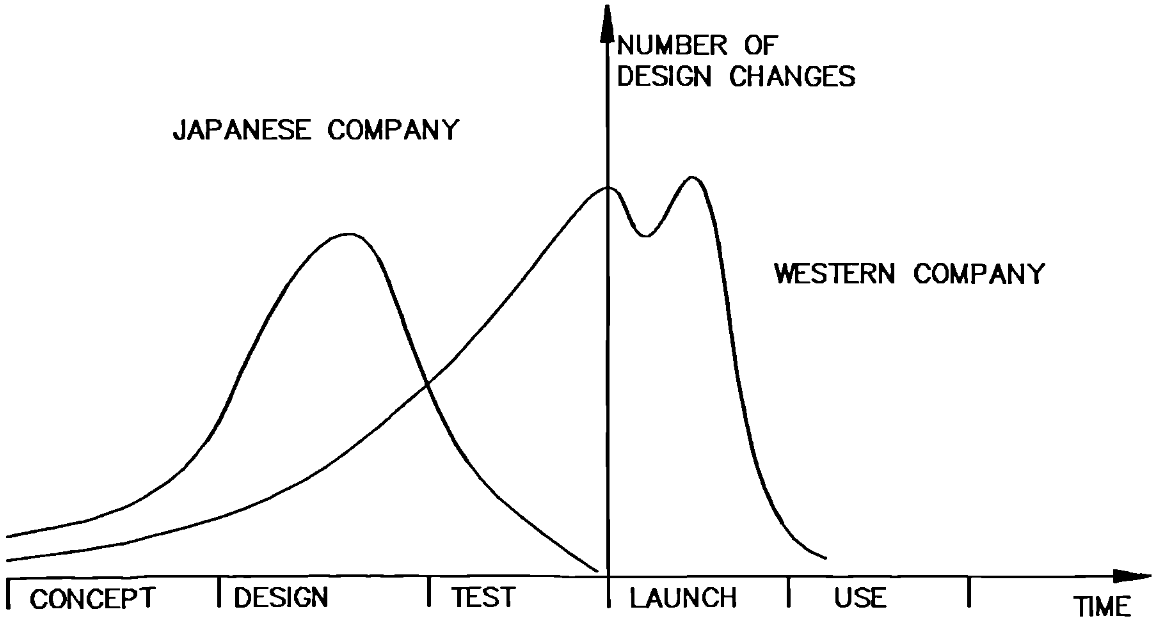


Figure 5.3 Comparison of Design Changes

5.2.10 Robustness of Design

In addressing the issue of design and quality, DFD aims to incorporate into the design process, quality tools that support robust engineering design. A design is said to be robust when its variation in performance is minimal against variations in the design, manufacture and use stages of the life cycle of the product as shown in Figure 5.4. Robustness is achieved by two major methods : (i) Designing out the unwanted operating regions and (ii) Favourably using the interactions between the noise and control factors in the design. Robust Engineering Design is hence the process of arriving at the target specifications for product performance with minumum variation or noise. This may be achieved by using the results from design and analysis of experiments and by performing muti-criteria optimisation.

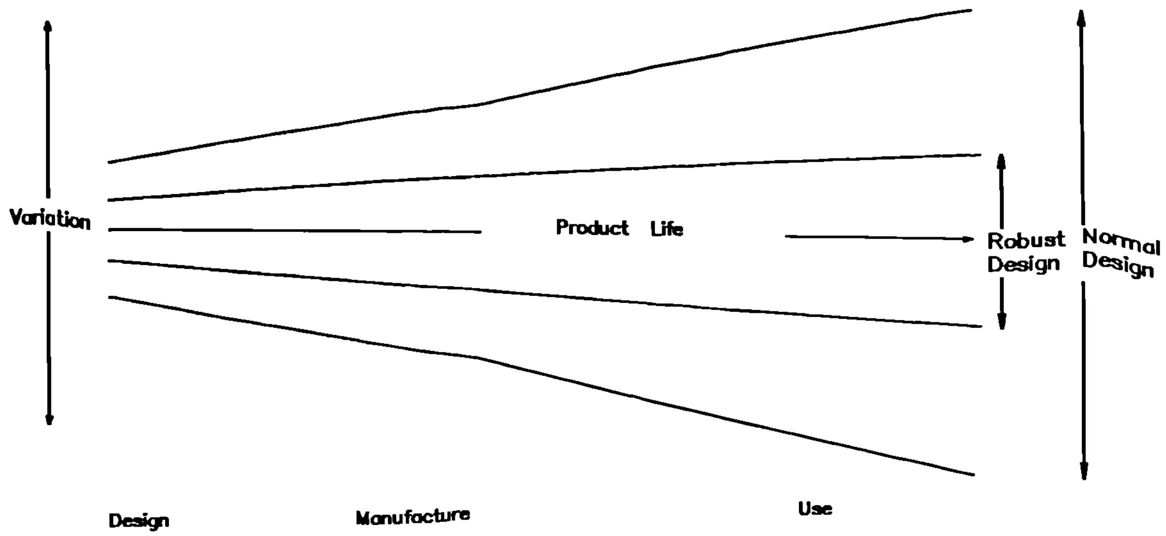


Figure 5.4 Illustration of Robust Engineering Design

5.2.11 Reliability and Safety

Reliability is the measure of the ability to produce the performance characteristics within specified tolerances for a specified time. This is achieved by : (i) using Mean Time Between Failures (MTBF) analysis and using components with high MTBF and (ii) using extra redundant paths to ensure operation in the event of the failure of a component. Safety requirements include safety both to personnel and equipment in the event of failure of component or unit. This is achieved by : (i) using Failure Mode and Effect Analysis (FMEA), Fault Tree Analysis (FTA) and (ii) taking appropriate steps to ensure high reliabilty for components and units having high risk

5.2.12 New Materials and Technologies

The development of materials science in the recent past has been rapid and the continuing appearance of new materials with novel and exploitable properties expands the range of choice. There are about 40, 000 to 80,000 materials available for the designer to choose from [283]. The selection of materials is an optimisation process involving many conflicting requirements, and it is also influenced by the shape/geometry and size of the components as well as available and suitable manufacturing processes. The use of new materials and technologies provide the designer with an opportunity to innovate and maximise performance. DFD hence encapsulates a materials selection process within its structure.

5.2.13 Evaluation Of Cost Implications

Design for cost is the process of bringing up to the early stage of product development enough information on costs to enable the designer to use them in decision making.

This ensures that cost is given equal status with other more functional aspects of design. DFD addresses the issue of cost estimation based on the concept of 'attribute-based costing'. Such attributes include, function, feature and activity. *Function based costing* lends itself to the evaluation of a design at the conceptual level. *Feature based costing* lends itself to the evaluation of a design at the parts level and *Activity based costing* permits detailed costing at the production planning level. During product development, certain activities will generate more costs or may be more sensitive to design decisions. These cost drivers should be identified and included in the costing model, as shown in Figure 5.5.

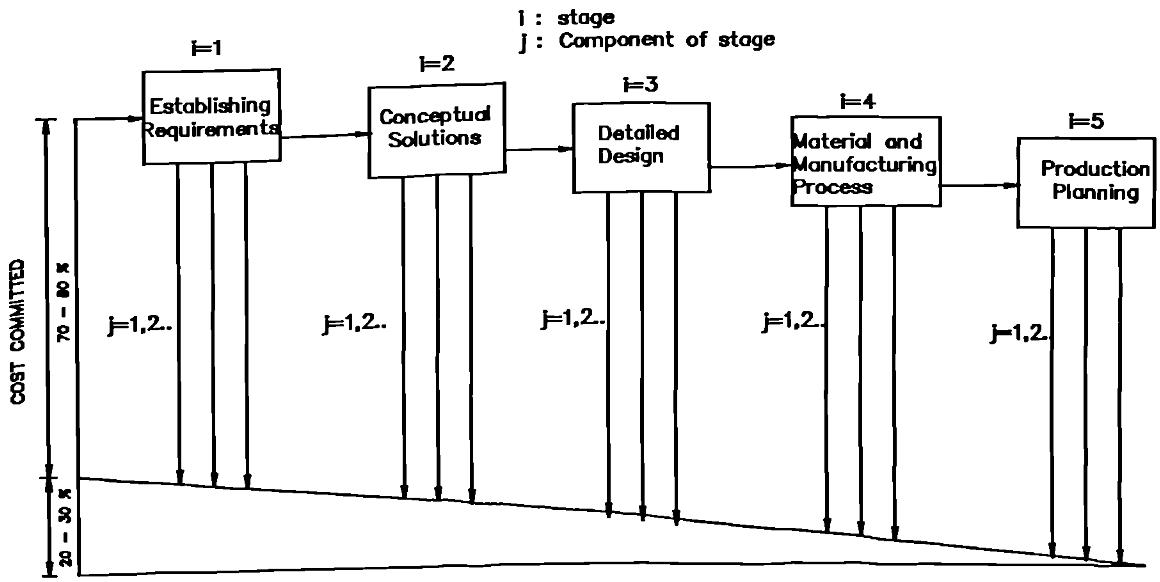


Figure 5.5 Cost Drivers in Different Stages of DFD

5.2.14 Quality Through Design

DFD treats quality as "The marriage between customer importance rating, which comes from customer requirements weighting, the bench mark position desired by the design team in the market, market relative importance, and the design team's technical design functions, prioritized to give maximum advantage". Quality is an obvious parameter to optimize for, but this implies a method of measuring quality. Taguchi's loss function is one of such measures. DFD ensures that design requirements are met and that the customer requirements are satisfied. Design is the prime driver for ensuring quality in products.

5.2.15 Design It Right First Time

A key ethos of DFD is to ensure that design is done 'right first time' and hence commence manufacturing with absolute confidence of performance and market appeal. To do this, DFD aims to ensure that : (i) customer requirements are properly studied and evaluated, (ii) design requirements are established and evaluated, (iii) all viable conceptual solutions are studied, (iv) the viable concepts are worked into viable layouts to the details of parts, (v) materials and associated manufacturing processes are then explored and (vi) production plans for all viable layouts with their associated materials and manufacturing processes are properly developed.

5.3 THE STRUCTURE OF THE DESIGN FUNCTION DEPLOYMENT SYSTEM

The structure of DFD as shown in Figure 5.6, encapsulates all the modules within it under an overarching structure consisting of three levels. The three levels provide the platform for design activities involving the use of specialist programs, databases and rulebases in a computer implementation.

5.3.1 Level 1of The DFD Structure - The Design Model

This consists of five stages representing the prescribed minimum path of the design process in DFD. It ensures the coherent structure of the different activities performed in each stage. The five stages represent the prescriptive approach to design in DFD. The overarching structure of DFD in level 1 and its interface with the modular structure in level 2, allows for the design activities suited for a descriptive design approach.

5.3.2 Level 2 of The DFD Structure - The Design Methods

This level constitutes the modules which bring the expertise in engineering and science into design. The modules permit the concurrent analysis of the design functions at the design, manufacture and use stages of the product development life cycle. The structure also permits the addition of other necessary modules which are not presently envisaged. Each of these modules need to communicate with each other in an integrated manner. The implementation of this facility, constitutes a key aspect in the software development of DFD.

5.3.3 Level 3 of The DFD Structure - The Knowledge and Databases

This level consists of databases, knowledge/rulebases associated with modules in level 2. These modules are used to store various information in the form of materials and manufacturing process data, design rules as well as capturing previous design knowledge and information. Such information can then be retrieved whenever required during the design process.

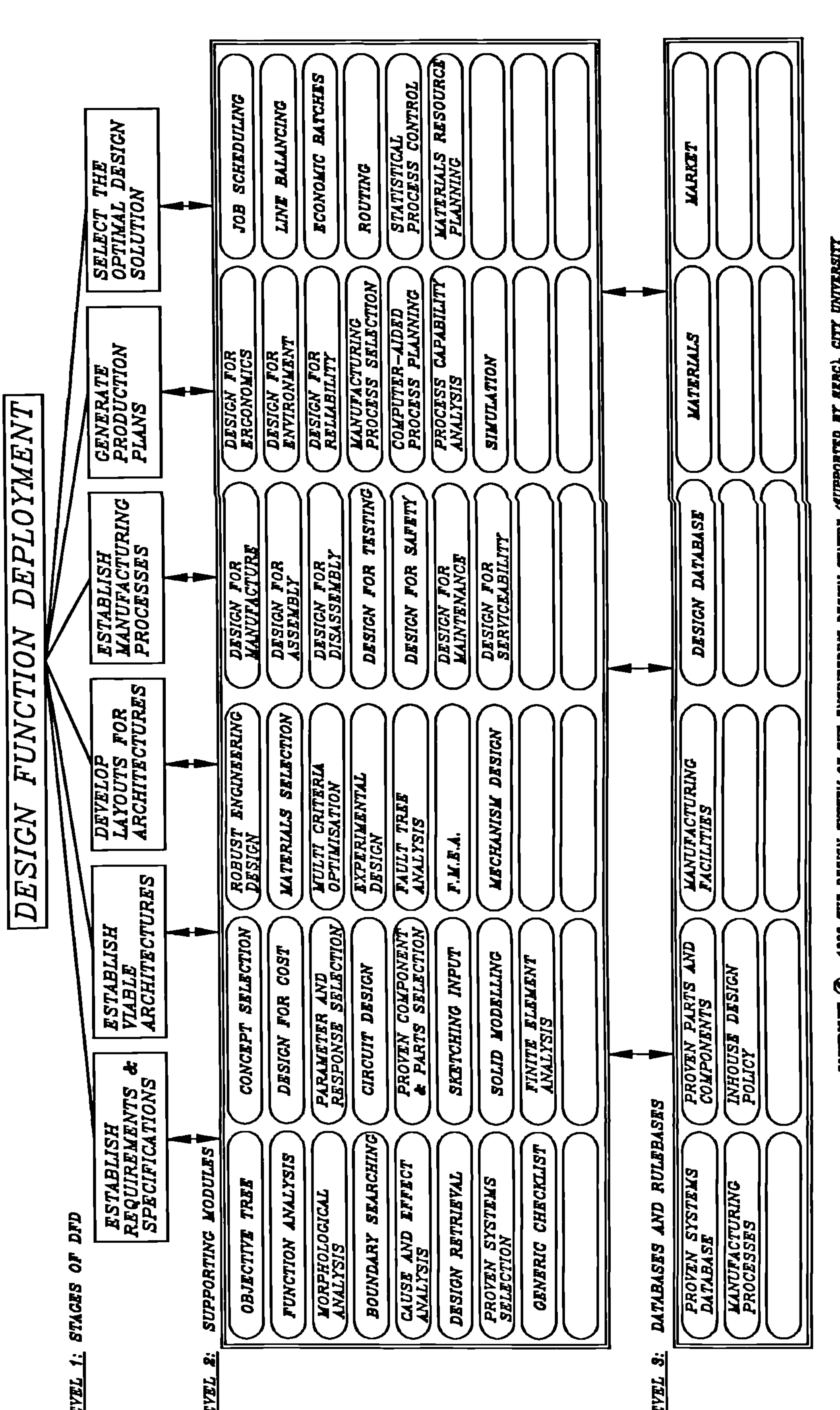


Figure 5.6 The Structure of the Design Function Deployment System

5.4 THE DESIGN MODEL OF DESIGN FUNCTION DEPLOYMENT

5.4.1 Design, Design Activity and Design Process

The process of design involves the development of a scheme that gives concrete form to some design goals. In doing this, the design activity has to (i) recognize the goals or purposes of the product or system, (ii) shape and model the design objects, (iii) create the forms of the design objects in accordance with the design goals and (iv) evaluate and determine the forms of these objects to make their contents comprehensible.

5.4.2 Design Models

To carry out the above and other design activities, the design process hence requires a structured approach in the form of a design model. Such a design model should communicate the structure of the design problem, and the essence of the problem solving process. It should also make explicit to the learner, the problem and the process that most designers assimilate inductively and apply intuitively [217]. Design models can be regarded as descriptions of the sequence of activities that take place in the design process. They are often drawn in the form of flow diagrams, with feedbacks showing the iterative returns to the earlier stages and can be descriptive or prescriptive in nature. **Descriptive models** simply describe the sequence of activities that typically occur in designing. Descriptive models are 'solution focussed', while **Prescriptive models** attempt to prescribe a better or more appropriate pattern of activities. They tend to be both 'problem focussed' and 'solution focussed'. In a 'problem focussed' approach the solution and problem are both developed in parallel leading to a creative redefinition of the problem. Prescriptive models are predominantly design process based, and a few are based on product attributes.

5.4.3 The Design Model in Design Function Deployment

The design model in DFD, is a 'prescriptive' model with adequate flexibility to support the 'descriptive' approach. It leads the design team through a 'minimum prescribed' path. This model, as shown in Figure 5.7, is represented in five stages, which involve :

- ¤ Establishment of Customer requirements and determination of design specifications and constraints in a solution neutral form.
- ¤ Synthesis of different conceptual solutions called architectures.
- ¤ Development of combinations of alternative subsystems within each promising architecture called layouts.
- ¤ Establishment of different materials and manufacturing process combinations for each part in each viable layout.

¤ Development of production plans for each viable layout, material and manufacturing process combinations.

The model also accomodates any necessary iterative feedback during the design process, from stage to stage.

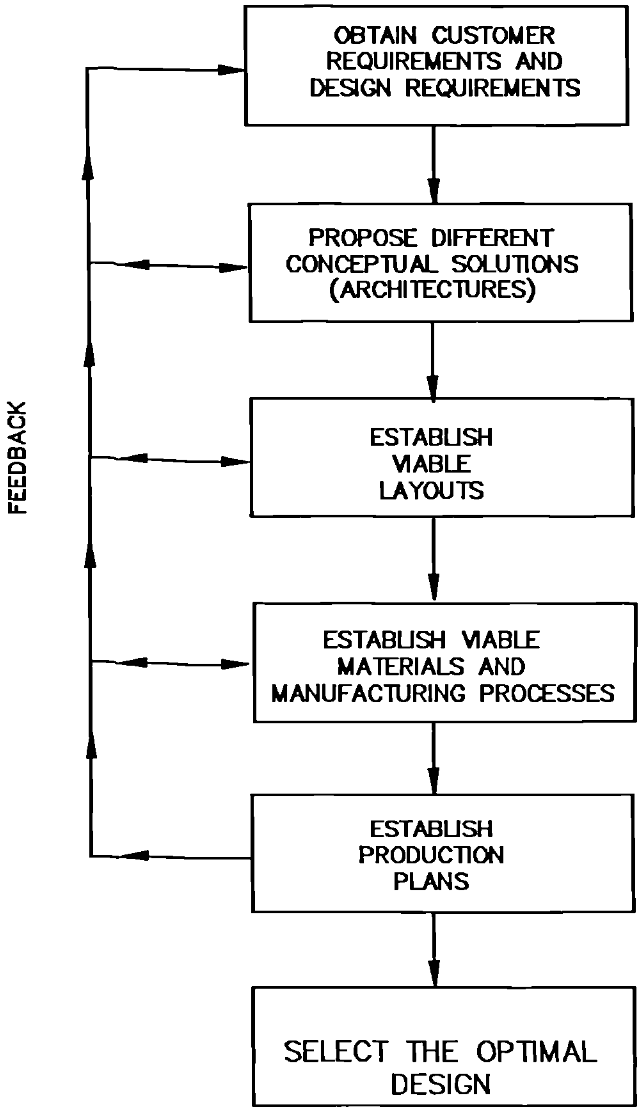


Figure 5.7 The Design Function Deployment Design Model

<u>Stage 1 of the DFD Design Model</u>

This stage first of all involves eliciting customers' requirements in their own words, and then including other requirements such as manufacturing constraints, use environment requirements, company requirements and statutory requirements. This list is then analysed and grouped into primary secondary and tertiary requirements. The tertiary requirements are then relatively rated according to their inportance. These rated requirements are then translated (deployed) into design specifications and constraints in a solution neutral form which are called stage 1 design functions.

Stage 2 of the DFD Design Model

In this stage, using the stage 1 design functions, several different design concepts (architectures) are proposed and developed to satisfy them. Some of these solutions may not be realisable, and others may be inefficient. The conceptual solutions are then analysed and evaluated, and the most promising ones are selected for development at the next stage.

Stage 3 of the DFD Design Model

This stage involves taking each of the viable (chosen) conceptual solutions from stage 2, dividing them into subsystems and then developing alternative solutions to each of these subsystems. The combinations of these alternative subsystems yielding complete solutions are recorded. These complete solutions are called layouts, belonging to the parent architecture. There can be several such layouts belonging to one architecture. At the end of this stage, some form of geometry should start to emerge for each layout.

Stage 4 of the DFD Design Model

In stage 4, each part in each layout is considered. Taking into account the part characteristics from stage 3, potential materials are selected for each part. On selecting the material, the corresponding manufacturing process, is determined. The final form or geometry of each part will depend on the material and its associated manufacturing process. Several methods can be used to manufacture a particular part, and an optimal manufacturing process has to be selected. This depends on many factors and it is a multi-criteria optimization process.

Stage 5 of the DFD Design Model

This final stage is the production planning stage. At this stage, the detailed design of the product would have been completed. The activities here then involve the planning of : batch sizes, necessary quality control checks, parameters to be controlled, inspection plans, etc. This is influenced by facilities available, costs and the quantity to be produced.

5.5 THE MAIN DESIGN FUNCTION DEPLOYMENT CHART

There are seven groups of information generated at each of the first four stages of Design Function Deployment.

They are stored in a QFD style chart which is called the main DFD chart. It houses information on : (1) The 'Whats', that is, the parameters that are to be deployed or need to be satisfied, (2) The 'Hows' or the 'Design Functions' which deploy the whats, (3) The relationship between 'whats' and 'Hows', (4) Target information about the 'Hows', (5) How the design functions or the 'hows' interact among themselves, (6) How the design functions or 'hows' themselves are rated according to their absolute importance, (7) How the design functions or 'hows' themselves are rated according to their relative importance. Each of these information set is represented by a block in the main DFD chart (for stage 1), as shown in Figure 5.8.

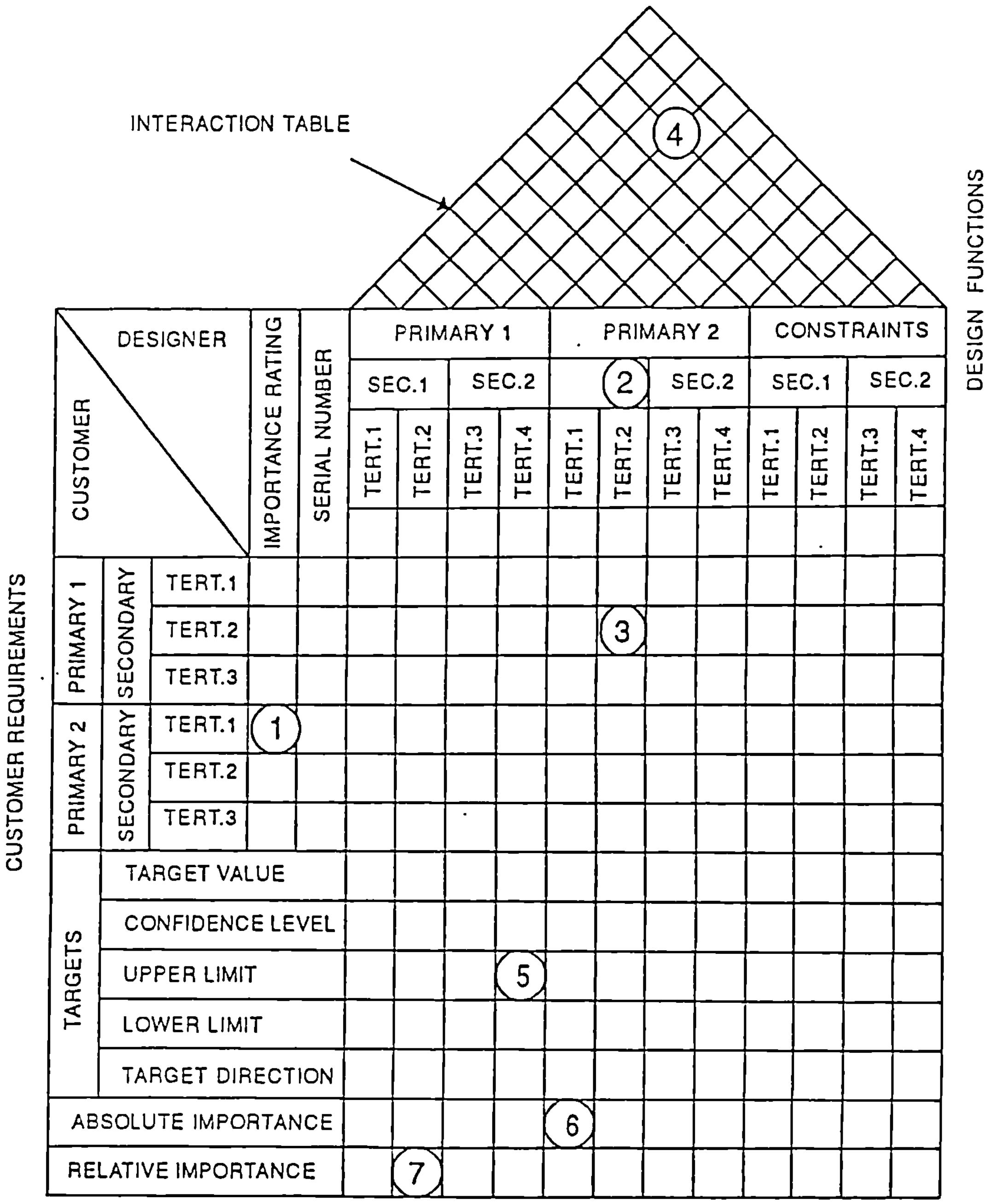


Figure 5.8 The Main DFD Chart

5.6 THE DESIGN METHODS OF DESIGN FUNCTION DEPLOYMENT

This section describes in summary, a majority of the auxiliary design methods, which are constituted within level 2 of the DFD structure. It should be noted that this list is not exhaustive, and DFD provides room for the addition of other useful and emerging design methods. The various stages within DFD, where these modules are most applicable, have also been highlighted.

5.6.1 Objective Tree

This module involves the clarification of design objectives and sub-objectives, and the relationships between them. The procedure involves :

(a) Preparing a list of design objectives, which are obtained from the design brief, from questions to clients and from discussions in the design team.

(b) Categorising the list of objectives into sets of higher-level and lower-level objectives, in a hierarchical manner.

(c) Drawing diagrammatic trees of objectives, which show the hierarchical relationships and interconnections.

5.6.2 Functional Analysis

The aim of this module is to establish functions required, and the system boundary of a new design. The procedure involves :

(a) Expressing the overall function for the design in terms of the conversion of inputs into outputs .

(b) Breaking down the overall function into a set of essential sub-functions. The sub-functions comprise all the tasks that have to be performed inside the 'black box'.

(c) Drawing a block diagram showing the interactions between sub-functions. The 'black box' is made transparent, so that the sub-functions and their interconnections are clarified.

(d) Drawing the system boundary, which defines the functional limits for the product or device to be designed.

(e) Searching for appropriate components for performing the sub-functions and their interactions.

The above two methods will be particularly useful in stages 1, 2 and 3 of DFD.

5.6.3 Morphological Analysis

Morphological analysis which derives its name from the dictionary definition: 'pertaining to the study of an organised system or form', can be considered to be a systematic approach to discovery and invention. It is used to force divergent thinking and to safeguard against overlooking novel solutions to a design problem [122]. It also obliges the designer/design team to think of several solutions for each major functional requirement and how they can be combined to form an explosion of solutions. The Morphological analysis involves the use of two complementary techniques viz : The Morphological Box and The Morphological Tree [218, 219]. The Morphological Box is used for generation of solutions without imposing constraints or bias, and it is more compact and efficient, while the Morphological Tree is used for describing alternative solutions with the imposition of constraints, but it gives better visual representation than the Morphological Box. In the example shown below, the matrix comprises a single left-handed column in which are listed the parameters essential to the design i.e. what the design must be or must have. To the right of each element in the column is a row containing the possible ways of achieving that particular parameter. The example of a forklift truck to be used in a warehouse is given below. The design parameters resulting from the specification would include: (i) means of support which allow movement across the floor, (ii) means of steering the vehicle, (iii) means of stopping, (iv) propulsion, (v) power unit, (vi) power transmission, (vii) lifting mechanism and (viii) facilities for operator.

A morphological chart showing these parameters and some possible ways of satisfying them is shown below:

Table 5.1 A Morphological Chart for a Fork Lift Truck

Parameters	**Possible Solutions**				
Support	wheels	air cushion	tracks	slides	spheres
Steering	turning wheels	rails	air thrust		
Stopping	reverse power	brakes	blocks under wheels	drag a weight on the floor,	
Moving	air thrust	power to wheels	hauling along cable	linear induction motor,	
Power	electric	bottled gas	petrol	diesel	steam
Transmission	hydraulic	gears and shafts	belts or chains	flexible cable,	
Lifting	screw	hydraulic ram	rack and pinion	chain or rope hoist,	
Operator	seated at front	seated at rear	standing	walking	remote control

This module is very useful in provoking the thinking process when proposing architectures. One could take the specifications and start developing them into conceptual solutions. Morphological analysis will be used mainly in stages 2, 3 and 4 of DFD.

5.6.4 Solid Modelling/Master Modelling

The 'Master Model' concept is now emerging as the way forward in CAD/CAM. It is based on a three dimensional solid modelling software that serves as a central database containing all geometric and non-geometric (attributes such as surface finish etc) information about the product. Users are able to perform engineering tasks directly from the single master model, avoiding duplication of data, eliminating the possibility of errors, saving time and costs and increasing productivity. This master model also paves the way for the integration of software applications. This is very important because the product passes through several engineering disciplines which use the data in several different ways and the master model enables each user to work on the data freely and concurrently without modification, translation or recompilation of data within the system. This module forms the geometric side of the DFD system, and will be used in stages 2, 3 and 4 of DFD.

5.6.5 Finite Element Analysis (FEA)

The engineering design process involves not only design synthesis, but also design analysis which aids the process of synthesis. Design analysis is primarily aimed at determining the performance of a particular trial design. In order to determine or predict the performance of a design, it is often necessary to calculate a field, which is defined as a quantity that varies with position within the device analyzed. There are several kinds of fields, and each of them has a different influence on the device performance. Closely associated with any of these fields, is a potential. The fields are related to the potentials as their derivatives with respect to position. Shown below are typical fields and their associated potentials.

Various Aspects of Performance

Field	**Potential**
Heat Flux	Temperature
Mechanical Stress	Displacement
Electric Field	Voltage
Magnetic Field	Magnetic Vector Potential
Fluid Velocity	Fluid Potential

The calculation of all the above fields and potentials can be done using Finite Element Analysis (FEA). FEA methods are predominantly used to perform computer based analysis of behaviours of structures, physical systems and components. Such behaviours include static, dynamic, thermal, electromagnetic and fluid analysis. The linear forms of these behaviours are commonly performed, and when necessary, non-linear analysis is also done. The FEA process involves the following steps :

(1) Creation of the Finite Element Model - Involves (i) Definition of geometry, nodes and choice of element types and (ii) Specification of material properties, loading and boundary conditions
(2) The Finite Element program then performs the required analysis - Involves (i)

Formulation of the necessary mathematical equations and (ii) Solving the equations (3) The Finite Element program then outputs desired results - This involves (i) Computation of node and element values of the field potentials (e.g. displacements, stresses, temperatures, reactions, forces, etc.) and (ii) Post-processing of the results in the form of (plots, code checks, etc.).

The modelling process in FEA involves a key activity of designing and generating the required finite elements, resulting in a particular mesh form. As a result of the large amount of computations involved in FEA, coarse mesh sizes are usually used for non-critical locations in the device, while for more critical areas, finer sizes are used. All proprietary FEA software packages usually contain various types of elements, ranging from beams, bar, shell, plates and brick (solid) elements to triangular, quadrilateral, axisymmetric, isoparametric, 2-D and 3-D element types. The designer then has to choose which of these elements are most suitable, and which would give more accurate results, for the particular application. This FE analysis module will be particularly useful in stages 3 and 4 of DFD.

5.6.6 Design Retrieval

Irrespective of the nature of the company all companies will have some of their past designs which are very good. These designs could be used as part of new products. For example a company manufacturing 'crawler tractors' could use the same design of operator seats in their new ranges if the original design was found to be very good. Even when a company is not involved in new products, it still has to refer to the old designs to serve the needs of the customer. Therefore it is necessary to create the following:

1. A general 3D format to store designs.
2. A method of using them as subsystems in new designs.
3. A method to transform the paper designs from the past to computer models in 3D.

The point 3 above itself may become a major project since this is one of the major problems confronting companies in moving to CAD.

The development of this module will be akin to the concept of case-based reasoning (design), which is a design approach involving the use of design experience in the form of episodes and employs analogical reasoning to select and transform specific solutions to previous design problems to be appropriate as solutions for a new design problem [220]. Nakatani et al [221] have also in their work developed the case-based reasoning approach to support retrieval of previous specifications, functions as well as components and parts of a design layout to address a new design problem.

This module when completed will have the basics for a database of past designs which will be of immense use to the designer, in stages 2, 3, 4 and 5 of DFD.

5.6.7 Sketching Input

Sketching is the natural way design engineers expand their thinking and brainstorm with themselves and sketches are the physical expressions of the thinking process. In the first place they have to be captured and processed to derive reasonably accurate geometric shapes. The existing methods of computer sketching do not provide full freedom to the designer. They require starting and finishing points, the types of curves and so on. But a good sketching input system should provide complete freedom to the designer whereby he could sketch naturally on a paper in a digitizer and the computer takes over from there. This will have two benefits as given below:

(1) Designing involves the use of standard parts and tailor made or designer made parts. A free hand sketching system with the 'Design Retrieval' facility would enhance the capability of the designer to design a compatible product.
(2) An old paper version of a design could be placed on top of a digitizer and the stylus could be traced over it to pretend sketching (or scanned), and then the computer would understand the old design which could then be transferred to the 'design database'. This module will find application in stages 2, 3, and 4 of DFD.

5.6.8 Robust Engineering Design

Robust Engineering Design (RED) is a process that can make a product/process performance insensitive to variations orignating from various sources, such as combinations of manufacturing variability, environmental variability and product degradation, during the life cycle of a product. The basic principles of RED can be described as follows [222].

(1) Products should be designed to be robust to downstream variation acquired in manufacture and use.
(2) Computer and physical experiments on products and processes should incorporate both design and noise factors.
(3) Following statistical modelling, design factors should be moved to nominal levels which achieve target output and minimize the effect of the noise factors and tolerances in the design factors themselves.
(4) Improving tolerances is expensive and should only be considered where (3) is ineffective.

In Jebb and Wynn [223] the process of understanding the full life-history of the product in order to isolate the key noise fators was described as *immersion* and often requires considerable imagination and engineering knowledge. The design process in DFD makes this immersion systematic and provides the necessary platform for the extraction of critical design and noise factors for RED experimentation and modelling. To achieve robustness Taguchi suggests the following sequence of events in the design process (i) system design (ii) parameter design and (iii) tolerance design. System design is the physical embodiment of the functional requirements of the product, where one applies special engineering and scientific knowledge and intution. Parameter design is the process identifying the optimal settings of various parameters under the control of the designer to limit variation. Tolerance design involves the to control of the variation in critical parameters when everything else has failed to control the variation of performance within the required limit. Robust engineering design deals with parameter design using controlled experiments. To cut down the number of trials considerably from the huge exhaustive list in the experiment it uses a mathematical technique called 'experimental design'. Taguchi has advocated the use of 'orthogonal arrays' for the design of experiments. There has been a lot of development to the methods suggested by Taguchi to make the design robust. This module deals with such issues, and it is particularly useful in stages 3, 4 and 5 of DFD.

5.6.9 Experimental Design

As mentioned earlier in 'RED', there has been lots of development to the methods suggested by Taguchi. One of the main limitation of orthogonal arrays suggested by Taguchi was the restriction on the number of interactions that could be investigated. This module can generate the design that would handle any given set of interactions. Also there are other ways of designing experiments which are also included in this module. This module will be useful in stages 3, 4 and 5 of DFD.

5.6.10 Design For Cost

Design for cost could be defined as the process of bringing back to the early stage of product development enough information on costs to enable the designer to use them in decision making. This ensures that cost is given equal status with other more functional aspects of design. Cost estimation of a design is one of the critical aspects of design and it is needed at various levels of resolution to make decisions at different stages of design management. For instance cost estimation of a ship for tendering purposes could not be detailed and could be based only on the specialist functions it has to provide. A display cabinet for a garment shop on the other hand could be detailed even at the tendering stage. Costing can be classified as function based, feature based and activity based according to the attribute which forms the basis of costing. Their effectiveness varies according to the resolution and stage of the design process. The design of a product goes through several stages during the transformation from an abstract brief to a physically realisable concrete product. Each stage of development picks up costs and certain activities will generate more costs or be more sensitive to design decisions. These are called cost drivers and there will be cost drivers in all the costing models irrespective of what attribute they are based on. Thus a comprehensive design for costing system should allow not only the various cost models but also the identification of cost drivers in these models. In addition to the provision of these fundamental requirements there should be a comprehensive design system which identifies clear milestones in the process of design so that appropriate attributes within the cost models can be used. Within the structure of DFD, optimisation of architectures is done using function costing, optimisation of layouts and parts is done using feature based costing and optimisation of the detailed design, including materials & manufacturing process, is done using activity based costing. This module will find application in stages 2, 3, 4 and 5 of DFD.

5.6.11 Multi-Criteria Optimization

During the design process, several design requirements are generated to satisfy the specifications derived fron the design brief. These requirements are usually in the form of design objectives/attributes and constraints, which can be qualitative or quantative in nature, and may contain uncertainties. The design team may also have certain preferences amongst these requirements, for the proposed design. It is in this situation, that an optimal solution has to be found. This will involve performing a multi-criteria optimization, on the generated design solution space. Optimization problems in engineering design, depending on the circumstance, can be grouped under two main classes, that is, (a) multi-attribute design optimization involving the selection of the most viable alternative amongst several alternatives and multi-objective design optimization, which involves finding the best design within the feasible range of the problem. In either of the two classes, dependent upon the nature of the design problem, number and types of objectives and constraints, the necessary optimization methods have to be selected and used in arriving at a solution. A majority of design optimization problems however come under multi-objective design optimization. Here, the process involves:

(i) Defining the design region - choosing the design variables, their range of values and interactions, and determining any constraints and their relationships.
(ii) Establishing the objectives - choosing the objective measures (functions), determining the relationships between the design variables and the objective measures, as well as desired or acceptable levels of objectives.
(iii) Finding the most appropriate multi-objective optimization technique(s), and then using them to find the best combination of objectives.

Design Function Deployment, with its target information, relationship matrix and interaction tables, facilitates these steps. The optimization module, will be used in stages 2, 3, 4 and 5 of DFD.

5.6.12 Materials and Manufacturing Process Selection

Materials selection is a key activity that is performed in the design of a product, and which necessitates due attention for it to be done in a rational way. The sheer volume of materials as well as data that has to be consulted today, invariably precludes a manual process and makes computerisation of materials databases imperative.

The main objective of the project is to develop a materials database system, which is suited to the terminology and taxonomy of the design specification process within DFD. In each of stages 1 to 4 of DFD, a significant number of generated design functions, will influence the selection of materials and manufacturing process. These materials and manufacturing process related design requirements (functions) will consist of both explicit and vague types. In order to rationally select the required materials, these requirements need to be fully elucidated. This requires the development of an adequate interface with the specification process in DFD and a materials and manufacturing process selection system. The system will be developed to interact dynamically with the DFD system in a concurrent manner at any of the DFD stages of the design process. Such interface will use similar charts to the DFD ones to translate the materials and manufacturing process related design functions into measurable materials properties and manufacturing process characteristics, which will then form the selection criteria profile for selecting the materials and corresponding manufacturing processes. Materials and manufacturing process selection can be performed in any of stages 2, 3 and 4 of DFD.

5.6.13 Design For Assembly

Design for assembly is aimed at reducing the number of separate parts of a component, to a minimum, and making the remaining parts as easy as possible to assemble. Design for Assembly often results in changes to the components of an assembly as well as in savings in component costs. A key feature of DFA, is the attempt to estimate assembly costs early in the design stage.

The DFA method as developed by Boothroyd and Dewhurst, can be divided into three steps of (a) selecting the most cost effective assembly method (from amongst manual methods, special purpose machine methods and programmable or automated machine assembly methods), (b) Analysing the product with respect to the selected assembly method, and (c) Improving the design and then performing re-analysis. The choice of a suitable assembly method, is usually influenced by factors, such as : (1) The market life of the product, (2) The proportion of defective parts produced, (3) The number of parts needed to build different product styles, (4) The number of parts that will be subjected to major re-design during the product market life, (5) The variety of different products and their similarity, (6) The company investment potential, (7) The annual production volume per shift and (8) The number of parts per assembly. This module can be used in stages 3, 4 and 5 of DFD.

5.6.14 Design For Manufacture

Design for manufacture focusses on designing individual components so that they are easy to manufacture. This involves the integration of the design and manufacturing operations and also the evaluation of part geometries as they relate to functionality and manufacturability. The aim here is to identify features of a design that can be altered to reduce manufacturing cost, without sacrificing other desirable factors such as reliability, functionality, durability, aesthetics, etc.

In order to assist the designer in making rational decisions as regards the manufacturability of a product, several guidelines have been proposed. These include : (1) Designing for a minimum number of parts, (2) Developing modular designs, (3) Minimizing part variation, (4) Designing parts to be multi-functional, (5) Designing parts for multi-use, (6) Designing parts for ease of fabrication (DFM), (7) Avoiding separate or different types of fasteners, (8) Minimizing assembly directions; designing for top-down assembly, (9) Maximizing compliance; design for ease of assembly, (10) Minimizing handling; designing for handling and presentation, (11) Evaluating assembly methods, (12) Eliminating or simplifying adjustments, (13) Avoiding flexible components and (14) Choosing adequate manufacturing methods; both primary and secondary methods. The adoption of DFM techniques usually results in reduction of parts, inventory, indirect overhead, storage, production line costs, assembly time and production risks. Product quality is also improved as well as increase in reliability and productivity.

5.6.15 Design For Testing

Within the product development cycle, Testing is usually a substantial percentage of the whole life cost of the product, and this percentage increases with increasing product complexity. It is hence imperative that designers, test engineers and ultimate users, recognise the importance of testing, and then participate in developing cost effective programmes for the achievement of testability. Such programmes must cover the activities involved right from the early stages of design, through manufacture, production, test and maintenance phases. Testability is considered as the ability of both the manufacturer (i.e. test engineers) and the user to establish and verify the defined characteristics of an engineering entity or product. It is also considered as a key feature of design, and hence the design of a product or process, should take account of how they would be tested, for compliance to expected performance. Some key aspects to be considered in the application of the concept of testability can be summarised as follows :

(a) Testability is the inherent property of an entity which facilitates the verification of its specified performance by defined techniques, (i.e. we need to be able to prove goodness);

(b) Testability should satisfy both performance and diagnostic test requirements and allow the failed area to be identified and isolated with a defined degree of confidence, (i.e. we need to be able to diagnose faults);
(c) Testability is an element of design, production methods and of quality management and achievement. It is therefore an essential element for management control, (i.e. it is a means of designing in the required quality).

The essential criteria for effective testability are :
(a) Each entity should have defined attributes which should be capable of verification to the depth required by the production and maintenance test philosophies;
(b) Test access to each entity should be sufficient to allow complete performance testing and adjudgement in line with the production and maintenance test philosophies with the minimum disturbance to the permanent structure of the entity;
(c) Each entity should be designed so that its performance can be verified by the simplest possible effective test methods which have a minimal effect on the operational life. This module will be particularly applicable to stages 3, 4 and 5 of DFD.

5.6.16 Design For Serviceability

Design for serviceability involves the deployment of serviceability guidelines during the early stages of design, in order to reduce service costs and ensure customers' satisfaction. Such guidelines, for various service modes, may include (a) provisions to detect servicing needs, (b) design features to enhance the ease of servicing and (c) estimated life-cycle service cost. Design for serviceability can lead to enhancement of the quality of a product by increasing availability, reducing service cost and ensuring performance after repairs. Serviceability can be regarded as a measure of the ease of performing all service related operations. Factors which can affect the serviceability of a product include : (1) Reliability of components and subsystems, (2) Labour cost, (3) Inventory cost, (4) Accessibility of components to be serviced, (5) Availability of necessary parts, tools, etc, (6) Quality of technical training, (7) Customer preferences, (8) Location of service and (9) Length of warranty

Design for serviceability can be considered under the following components :
(1) Diagnosability - This is the ease with which one can find the cause of a malfunction and the course of action for the associated repair.
(2) Maintainability - This is the ease with which one can perform regular or routine maintenance. The main issue here, is the accessibility of components.
(3) Crash Repairability - This is affected by the structural materials used for the product and the modularity of the components. The key issue is the optimum modular structure that minimizes the cost of crash.
(4) Repairability - This is the ease with which one can repair malfunctions. Usually results in the largest service cost. Major issues are repair time, replacement part cost and a tradeoff between the two.

In order to identify the measure of repairability of a product, the Service Mode Analysis (SMA) method has been used. SMA is a method of describing which service modes will impact a particular design, and in what manner. It also involves the study of how to repair the various malfunctions that a system may experience. This module will be used in stages 3, 4 and 5 of DFD.

5.6.17 Design For Environment

Design for the environment involves the development of guidelines for evaluating designs with respect to their environmental compatibility using environmental/green indicators, in assisting designers to make decisions satisfying environmental concerns, without compromising product quality and function. This involves the evaluation and design of product/process for environmental compatibility based on a life-cycle analysis of the various processes involved from product inception to disposal. A number of green indicators have been proposed for assessing the environmental impacts of a product/process based on the life-cycle analysis. These include the following :

(1) Percent Recycled - This is the percentage of recycled material in a product.
(2) Degradability- This is the ratio of the volume of degradable material in a product to the total volume of the product.
(3) Life - This is the time it takes for the degradable portion of a product to degrade. A curve showing the expected volume of reduction over time is used to determine the life of a discarded item.
(4) Junk Value - This is a measure of the total time a product will take to degrade into the environment. It is calculated as the area under the life curve (above). The units are cubic metres.years.
(5) Separability - This is a measure of what materials can be separated from a product. It is the ratio of the volume of separable materials to the total volume of the product. The notion of separability is different from disassembly. The aim here, is to separate out parts that are made of compatible materials. For example, copper and steel are incompatible in melt, while steel and aluminium may be charged into the same furnace.
(6) Life-Cycle Cost - This is the total cost incurred in the life of a product. This would include costs or purchasing, maintenance and disposal. This indicator can help surface hidden costs that were not being factored into product costing.
(7) Potential Recyclability - This is the ratio of the volume of recyclable materials to that of unrecyclable materials.
(8) Possible Recyclability - Composites and glued materials are potentially recyclable, but cannot be recycled because they are inseparable. This indicator has to be measured on a part by part basis and has to take into account the available recycling methods and their economic viabilities.

(9) Useful Life - This is defined as the time an item spends in the activity that it was designed for.
(10) Utilization - This is the ratio of the useful life of a material to the time it takes to "return" to the environment. For example, the utilization of a styrofoam cup is very low because it is used for 5 minutes and takes decades to degrade in the environment.
(11) Total and Net Emissions - These indicators take a sum total of solid, gaseous and waterborne emissions from the use of particular materials.
(12) Total Hazardous Fugitives - This is a measure of the weight of hazardous fugitives from the life-cycle. This is expressed as a ratio of the weight of hazardous chemicals emitted per unit weight of product.

5.6.18 Design For Reliability

Reliability is a characteristic of a product or system expressed by the probability that the system will perform without failures, its intended and required function under stated conditions and over a specified time period or amount of usage. Designing for reliability can be carried out at the Architecture, Subsystems, Layout, or Parts levels, within DFD, spanning over stages 2, 3, and 4. At the Architecture or Layout levels, reliability analysis is more concerned with the system time behaviour taking into account reliability, maintainability and other important factors. At the subsystems or part levels, reliability analysis is associated with performance of failure rates, failure mode and effect analysis and fault tree analysis, in order to check thefulfillment of reliability requirements and to detect and eliminate reliability weaknesses as early as possible in the design process.

5.6.19 Failure Mode and Effect Analysis (FMEA)

This is a method of studying the causes and effects of potential failures in a product, and using that information in the design process. The goal here is to increase the overall quality and reliability of a part, component or product, by anticipating failures and designing them out. By identifying modes of failure early in the design of a part, key components can be modified with minimal effect on the product cost. This involves firstly, examining all the ways in which a product can fail. The failure is then analysed statistically as to its effect on the total system, its seriousness, and its frequency of occurrence. Failure is then defined as the inability of component to perform its intended function. Failure modes include ways in which a component could fail such as fatigue, leakage, deformation, etc. The final action is then to alter the design to minimize or eliminate the effect of a given failure. This module will be particularly useful in stages 3 and 4 of DFD.

5.6.20 Fault Tree Analysis

Fault tree analysis or FTA is one of the principal methods of system safety analysis evolved in the aerospace industry in the early 60's. FTA is a valuable design tool to identify potential accidents in a system design and to help eliminate them. It can also be used to predict most likely causes of system failure. A fault tree is a model that graphically and logically represents the various combinations of events, both normal and fault, occurring in a system that lead to the undesired event under consideration which is called the 'top event'. Event in this context denotes a dynamic change of state that occurs to a system element such as hardware, software, human and environmental factors.

The goal of fault tree construction is to model the system conditions that can result in the undesired event. This needs a thorough understanding of the system before the construction of the fault tree. The events that lead to the undesired event or the 'top event' are logically denoted by OR and AND gates and are shown below the 'top event' in the fault tree. The inputs to some of the gates can be outputs from some other gates. The events trace to the origins which are called the 'basic events'.

Four symbols are in use to represent specific types of fault and normal events in fault tree analysis. (1) The **rectangle** defines an event that is the output of a logic gate and is dependant on the type of logic gate and the inputs to the logic gate. (2) The **circle** defines a basic inherant failure of a system element when operated within its specifications. It is a primary failure which is sometimes called a generic failure. (3) The **diamond** represents a failure that is purposely not developed further. (4) The **switch** event represents an event that is expected to occur or never to occur because of design and normal conditions. A cut set is a set of basic events whose occurrence cause the top event to occur. A cut-set is minimal if it cannot be reduced and still insure the occurrence of the top event. A listing of minimal cut sets is useful for design purposes in order to determine the 'weak links' in the system. For a fault tree with perhaps hundreds of gates and basic events it is clearly not easy, nor in general possible, to determine all minimal cut-sets by inspection. This module looks at computerisation to obtain the minimal cut-sets. This module will be particularly relevant at stages 3 and 4 of DFD.

5.6.21 Design Checklists

Design checklists are used to enable designers to use knowledge of requirements that have been found to be relevant in similar situations. The procedure involves preparing a list of questions that were found to be important in a similar situation, or in several similar situations. These questions are usually associated with needs to be met by the design and their relative importance, the various activities of the design process and fundamental questions such as what has to be done, why it has to be done, when it has to be done, where it has to be done, by what or whom is it to be done and how it is to be done. Other questions asked include those intended to generate a variety of design alternatives and those intended to ensure that all changes introduced are compatible with each other and with all the needs.

5.7 THE DESIGN MATRIX OF DESIGN FUNCTION DEPLOYMENT

Blessing [224] discusses the concept of the design matrix as representing the design process in the form of a structured set of issues and activities. This design matrix concept is based on the combination and extension of two approaches to design, that is, methodical design, which is a problem oriented approach to design [225] and IBIS, an Issue Based Information System based on the argumentative approach to design [226,227]. Methodical design was selected because it explicitly distinguishes between design stages and activities, and emphasizes the recurrent execution of the process on every level of complexity. IBIS on the other hand was selected because it supported the process of deliberation by capturing design rationale. In the resulting design matrix (Figure 5.9), the rows represented the different issues to be solved in the design process, based on the process stages distinguished by methodical design. The IBIS elements (issues, proposals and arguments) define the columns of the matrix and relate to the deliberation process of the designer/design team.

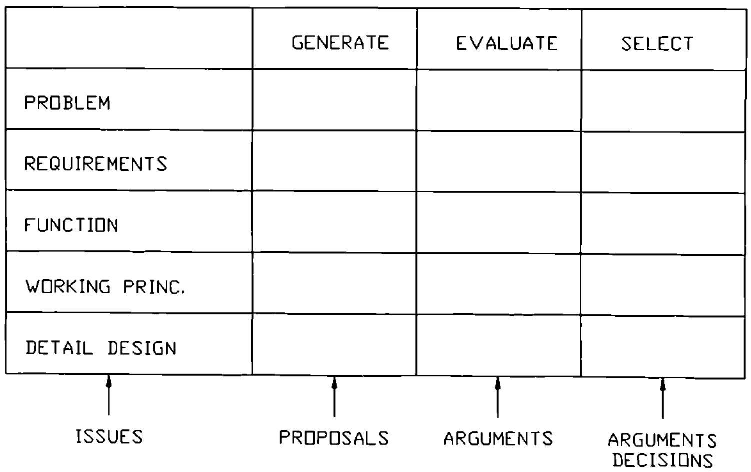


Figure 5.9 Two Dimensional Design Matrix - Blessing [224]

Simon [228] also discusses the design matrix by integrating two concepts referred to as (a) the morphology of the design process and (b) the anatomy of the design process. The morphology of the design process which was first proposed by Asimow [105] represents the different phases and stages of the life cycle of the product from conception to retirement. These phases include : Needs analysis, Feasibility study, Preliminary design, Detailed design, Production, Distribution, Consumption and Retirement

The anatomy of the design process on the other hand is attributed to the work of Rosenstein [229] and represents the actions of the designer/design team as progress is made in identifying and solving the design problems. These actions are usually repeated through each of the phases and stages of the morphology of the design process. These actions or activities have been classified into : Problem statement and needs formulation, Information collection, Modelling, Value statement, Synthesis of alternatives, Analysis and testing, Evaluation, Decision, Optimisation, Iteration and Communication. The resulting design matrix of the above considerations is shown in Figure 5.10. This matrix helps to give a panoramic view of engineering design, with the possibility of identifying any engineering action/activity corresponding to some position in the matrix.

Anatomy / Morphology	Problem Statement & Needs Formulation	Information Collection	Modelling	Value Statement	Synthesis of Alternatives	Analysis & Testing	Evaluation	Decision	Optimisation	Iteration	Communication	
Needs Analysis												
Feasibility Study												
Preliminary Design												
Detailed Design												
Production												
Distribution												
Consumption												
Retirement												

Figure 5.10 Two Dimensional Design Matrix - Simon [228]

The Design Function Deployment (DFD) system does encompass the two concepts of the morphology and the anatomy of the design process as discussed above, with the concept of the anatomy of the design process being more implicit. In applying the two concepts to Design Function Deployment, some extensions and enhancements became necessary, in order to accomodate additional issues represented within DFD. In DFD, the morphology of the design process represents the five stages shown in level 1 of the DFD structure (Figure 5.6), that is, the minimum prescribed path to be followed during the design process, which are : (1) Establishment of requirements and specifications, (2) Establishment of viable conceptual solutions (architectures), (3) Development of viable detailed designs (layouts), (4) Establishment of materials and manufacturing processes and (5) Generation of production plans. In the case of the anatomy of the design process within DFD, the generic design activities that can be performed include the following : (1) Requirements capture, generation and analysis, (2) Design synthesis, (3) Design evaluation and verification, (4) Engineering analysis, (5) Design optimisation and decision making, (6) Modelling, (7) Simulation, (8) Design for Quality, (9) Manufacturing process planning, (10) Drawing/graphics, (11) Production control, (12) Communication, (13) Rapid prototyping, (14) Design Information storage, and (15) Search and Selection. This list although not exhaustive represents the core activities performed within DFD. The above DFD design morphology and anatomy thus represent the 2-dimensional design matrix of DFD as shown in Figure 5.11.

DESIGN ACTIVITIES / DFD DESIGN STAGES	REQUIREMENTS CAPTURE	DESIGN SYNTHESIS	DESIGN EVALUATION	ENGINEERING ANALYSIS	DESIGN OPTIMIZATION	MODELLING	SIMULATION	QUALITY DESIGN	MANUFACTURING PROCESS PLANNING	GRAPHICS/ DRAWING	CASE BASED DESIGN	RAPID PROTOTYPING	PRODUCTION CONTROL	COMMUNICATION
ESTABLISH REQUIREMENTS AND SPECIFICATIONS														
ESTABLISH VIABLE ARCHITECTURES														
DEVELOP LAYOUTS FOR ARCHITECTURES														
ESTABLISH MANUFACTURING PROCESSES														
GENERATE PRODUCTION PLANS														

Figure 5.11 The Two Dimensional DFD Design Matrix

The design modules (methods) used during the stages of DFD are shown in level 2 of the DFD structure (Figure 5.6). These modules which are key to the DFD system, thus represent a third dimension to the design matrix within it, and can be called the (aids, enablers, tools) of the design process in DFD.Taking account of this third dimension, the result is a 3-dimensional design matrix of DFD, as shown in Figure 5.12.

This figure shows the global view of the design matrix in DFD, explicating the interactions of the three dimensions. A further elucidation of the matrix is provided by the other 2-dimensional views of the matrix relating the design process morphology and tools and the design process anatomy and tools represented by the elevation and plan views of Figure 5.12 respectively. It is worth noting that it is only the morphological dimension of the design process that represents a chronological ordering of distinct steps (i.e. a minimum prescribed path) and that the design activities representing the anatomy of the design process and the third dimension representing tools of the design process, do not represent any chronological ordering of distinct steps.

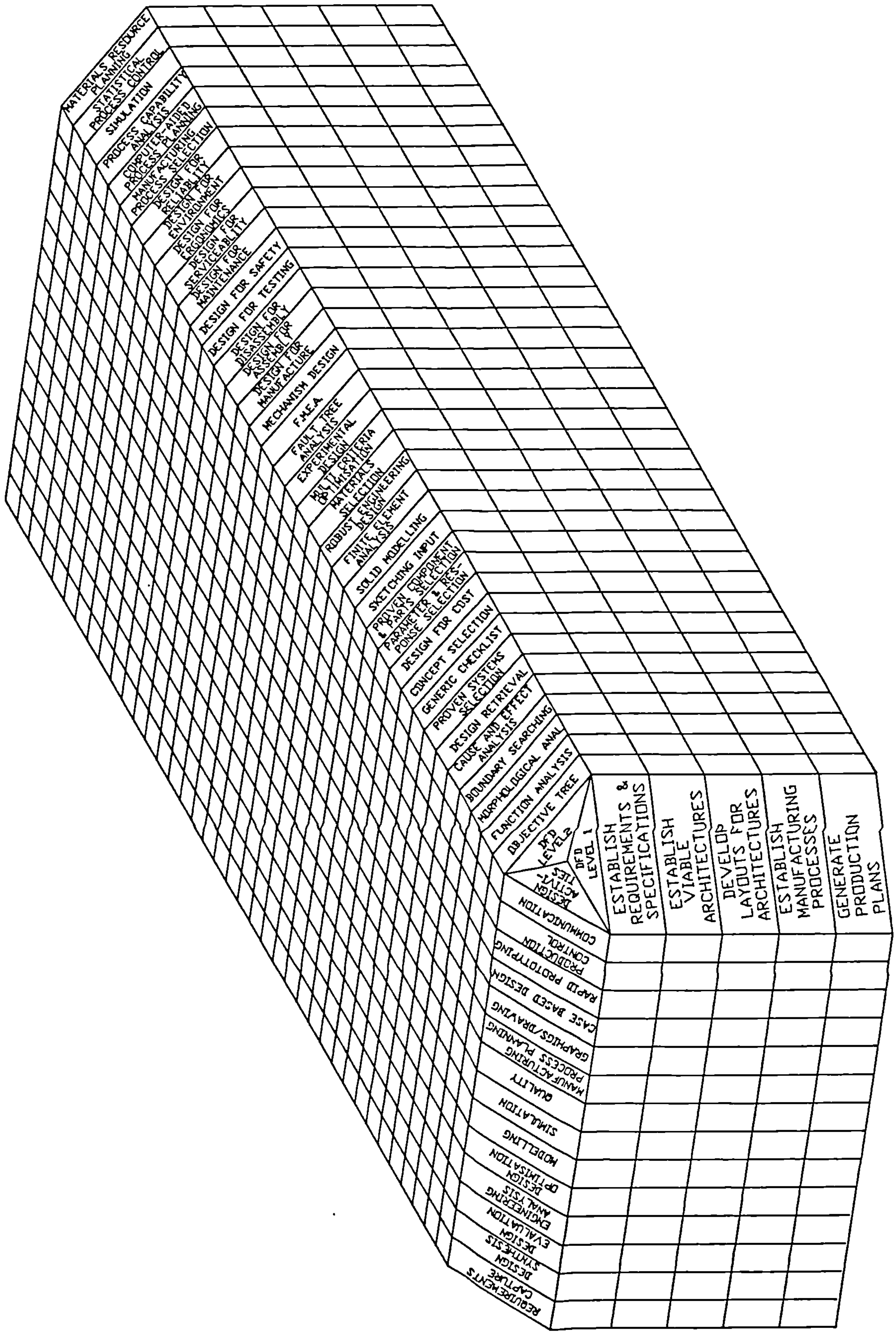


Figure 5.12 The Three-Dimensional DFD Design Matrix

Hence, there would be continual feedback, iteration and revision in performing any of the design activities while using any of the design tools, during the design process in DFD. Using the matrices, it is then possible to easily identify the design activities that should be performed at any of the DFD design stages, as well as the design tools that can be used for any of the design activities, at any stage of DFD, irrespective of the type of design or products. The DFD design matrix will also enable the design team to identify the necessary activities that need to be performed using the most suitable design tools when carrying out either original, adaptive or routine designs as well as when designing simple or complex, static or dynamic, overconstrained or underconstrained products. Furthermore, the design matrix by expressing clearly the different stages, activities as well as tools that constitute the DFD system, provides a rational and comprehensive basis for managing the design process in DFD, in a concurrent manner.

5.8 THE DESIGN SOLUTION SPACE

The design process can be regarded as a problem-solving activity and as such can be represented using a state space, where each state corresponds to a possible initial, intermediate or final design solution [85]. This state space can hence be referred to as a design solution space. Designing can therefore be described as the process of identifying a particular solution amongst a group of viable alternative solutions, to a design problem stipulated by a set of requirements, involving a navigation process from an initial state (requirements and specifications) to a final state (proposed solutions). This navigation process can also be viewed as a mapping process between the 'requirements space R' and the 'solution space S' as shown in Figure 5.13.

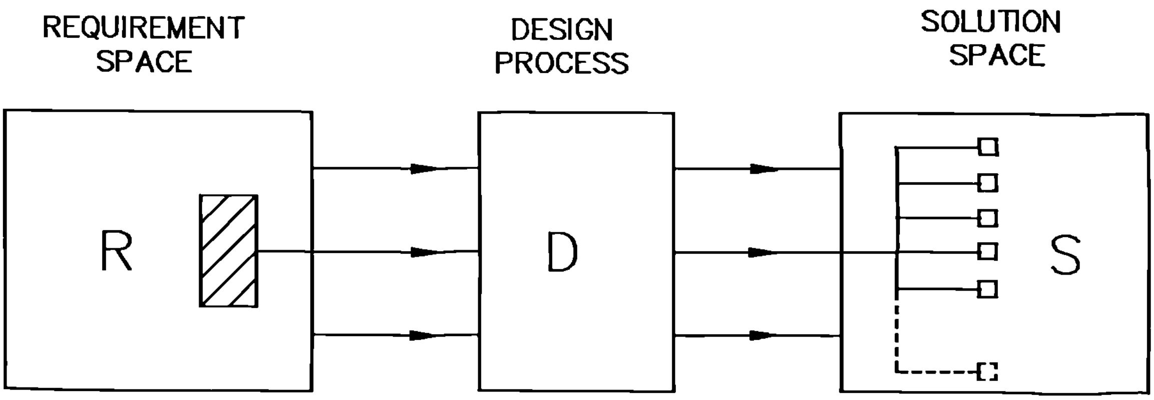


Figure 5.13 One to Several Mapping of Requirements to Design Solutions

The mapping D between the two spaces is the design process. The mapping D : R to S (Figure 5.13) is a one to several mapping and the process of designing should identify the solution that best suits a given set of requirements. In practice however, the requirements space is not static, but dynamic, as customers/users/clients tend to change and revise their original set of requirments. The requirements space is thus adjusted to a multidimensional one, also resulting in the expansion of the solution space, and the mapping D : R to S hence becomes a several to several mapping as shown in Figure 5.14.

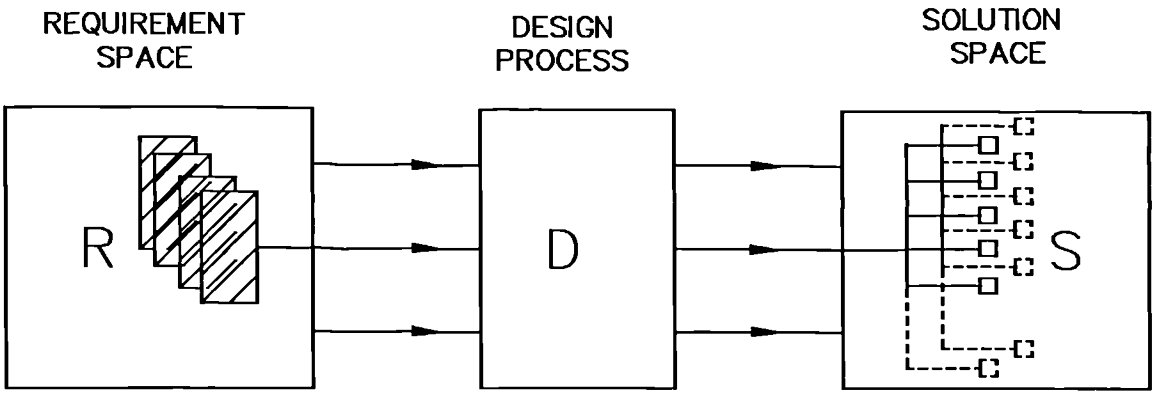


Figure 5.14 Several to Several Mapping of Requirements to Design Solutions

Banares-Alcantara [85] has also elucidated three different ways of using the information generated during the design process D, indicating that the choice of any of the strategies would determine how well it copes with the various complexities and facets of how the design process is carried out, as well as the resulting design solution space topology. These three forms of design solutions spaces are : (a) Linear design solution space topology, (b) Tree design solution space topology and (c) Network design solution space topology

Linear design solution space topology

In this design space topology, no history of the intermediate stages of design is kept, and every time a modification is made to the current design, the modififcation is made and the previous state is lost. Each intermediate state therefore has only one successor, and the design process is a line of nodes with the initial specification A_1 as the first node and the proposed final solution S_A as the final node (Figure 5.15). There is no mechanism for exploring an alternative in a previous node. This representation is a simple one to implement, but it is inadequate as it fails to provide a mechanism to explore alternatives, and hence not well suited to the philosophy of DFD.

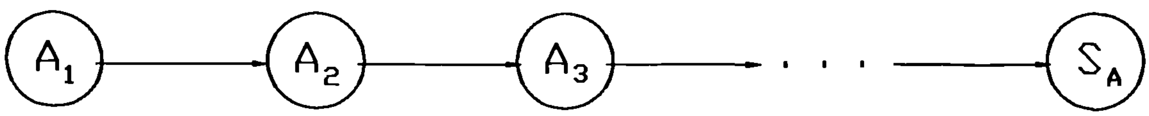


Figure 5.15 Linear Design Solution Space Topology

Tree design solution space topology

The tree design space topology is a more general case of the linear design sapce topology. Here a current design saved in memory prior to being modified, can at any point be examined by the designer at an intermediate stage and a different modification proposed. Each intermediate state can thus have more than one successor (each representing a different alternative originating a tree-like structure as ahown in Figure 5.16. This structure is a unidirectional graph taking into account the case of identical states reached by different exploration paths. In the same way as the previous case, a design decision made potentially affects only design states downstream of a single design path (i.e a change to a current design can only propagate to its successors). A new modification r_2 to A_2 can only affect designs on path C_1 --> S_C when they are created, but cannot affect designs on path A_3-->S_A. If the designer wanted to explore the application of both operators on the successors of the current node (i.e. propagate r_2 through A_3-->S_A), two options are possible. (a) Create a new (third) path where the next successor is the product of applying both operators simultaneously, and apply the rest of the operators of the old path to it, or (b) Have a demon-like facility which can visit each of the nodes in the old path and apply the new operator on them. The issues to address then are : (i) should the designs A_3-->S_A be modified by r_2 to A'_3-->S'_A (and the originals forgotten) ? or (ii) should a new path consisting of designs A'_3-->S'_A be created (reducing to the case above)?. The tree design space strategy is commonly used in most design support systems and although it is better than the linear design space case, it is not flexible enough to fully support the exploratory design process.

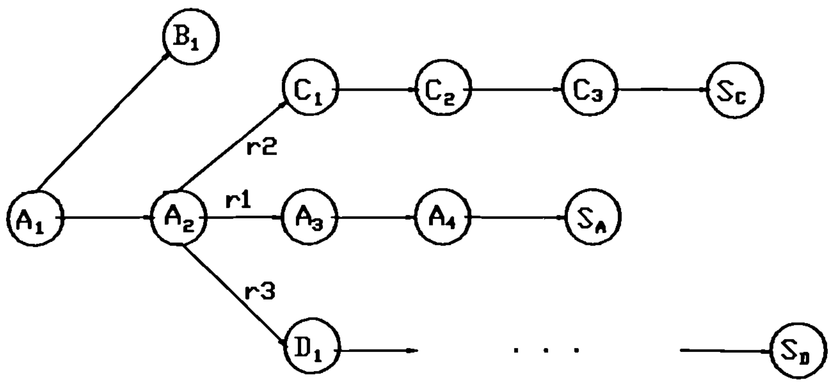


Figure 5.16 Tree Design Solution Space Topology

Network design solution space topology

The network design solution space strategy is a more general case than the tree design space topology. In this case, each one of the states is kept in memory and can be dynamically accessed at any point during the design process. A node can also have more than one successor and nodes from different branches can be linked through relations determined by the user/designer. Also, a decision in any particular node has the potential to affect its successors, predecessors or siblings and create new alternatives. The design solution space hence takes the form of a birectional/multi-directional graph (Figure 5.17).

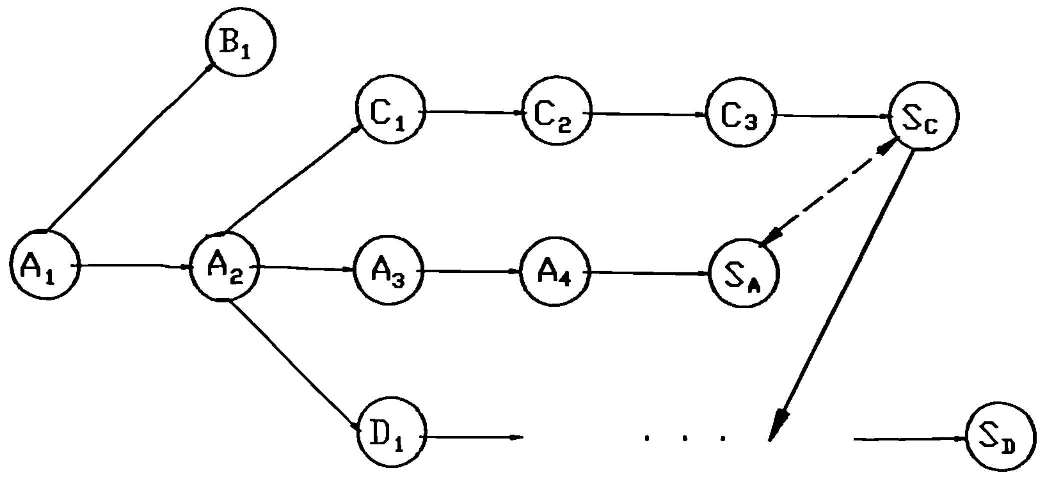


Figure 5.17 Network Design Solution Space Topology

This strategy is more amenable to most approaches to design. In the past, when competition was not rife, and technology was not very advanced, the designer often identified the first workable solution from the solution space S (Figure 5.13) as the design. This is akin to the adoption of the linear design solution space strategy. In the current CAD era, where man and computer work as the design team, one complementing the other to solve the design problem, this approach becomes inadequate. Furthermore, the evolution of the mechatronics approach [230] to design and the introduction of new materials and technologies, many more design solutions are added to the possible design solution space and the success of a product and its parent company, depends on the identification of the optimal or most desirable solution by the designer. The foregoing situation creates the necessity for a system that would allow the visualisation of the entire spectrum of viable solutions before homing in on a preferred solution. This will also require the adoption of a strategy somewhat in line with the network design solution space case, where the mapping process (Figures 5.13 & 5.14) includes all the activities throughout all the stages of Design Function Deployment, DFD.

5.9 THE DIMENSIONALITY AND MORPHOLOGY OF THE INCREASING COMPLEXITY OF DESIGN [231]

The design of a product commences with the identification of the need for the product, which is encapsulated in what is generally regarded as the **'customers requirements'**. Other factors that have to be taken into account include the primary functions of the product, manufacturing constraints, statutory requirements, cost, reliability, durability and so on. The above requirements have to be translated into measurable **'design requirements (functions)'** which would ensure that the desired product is properly conceived and developed. On deriving the appropriate and clearly defined design functions, it then becomes necessary for the designer to propose alternative conceptual product solutions that would satisfy the design functions. Traditionally, designers tend to home in quickly on what they figure the product should look like. A comprehensive proposal and evaluation of all possible options is usually not done. One of the principals of Design Function Deployment, is to provide the designer with the facility that enables the conceptualisation of all possible alternative solutions (architectures) at the early stages (stage 2) of DFD, from which the most promising and eventually the most viable ones are selected and considered in detail.

The process of proposing alternative solutions, involves the consideration of all possible **'architectures'** and associated subsystems. This means that in designing a product, it could be possible to propose a varying number of architectures. In practice however, this may not exceed a few number. At this level, each subsystems of each architecture, can also have a number of alternative types. Hence it is possible to derive a number of **'layouts'** using different combinations of the alternative subsystems, for each architecture. These layouts are variants of a particular architecture. For a particular layout and its subsystems combination, it is also possible to come up with alternatives of these layouts taking into account all the types and grades of the physical part that performs the function of a subsystem. For example, the prime mover in a mechanical system can be represented in two layouts by (a) an Internal combustion engine and (b) an electric motor. These two would have to be considered as resulting in 'alternative layouts' cum parts for a particular layout and subsystem combination.

In the design process, it can be seen that it is possible to propose several architectures, each of which can have several layouts and subsystems combinations, each of which can also have several layouts and parts. This shows an increase in the complexity of the design process taking into consideration all the possible architectures. For each of the **'parts'** of a derived layout, the **'form'** (shape and size), **'materials'** and associated **'manufacturing processes'** have to be derived. It is also possible that each part can be of varying alternative forms, made from one or more materials, using one or more manufacturing processes. This means a further increase in compelxity at this detailed level of design. The increase in complexity at the top level (DFD stages 2 and 3) can be represented in three dimensions of Architectures, Layouts and Parts as shown in Figure 5.18. Similarly, the increase in complexity at the lower level (DFD stages 3 and 4), can also be represented in three dimensions of Form, Material and Manufacturing Processes.

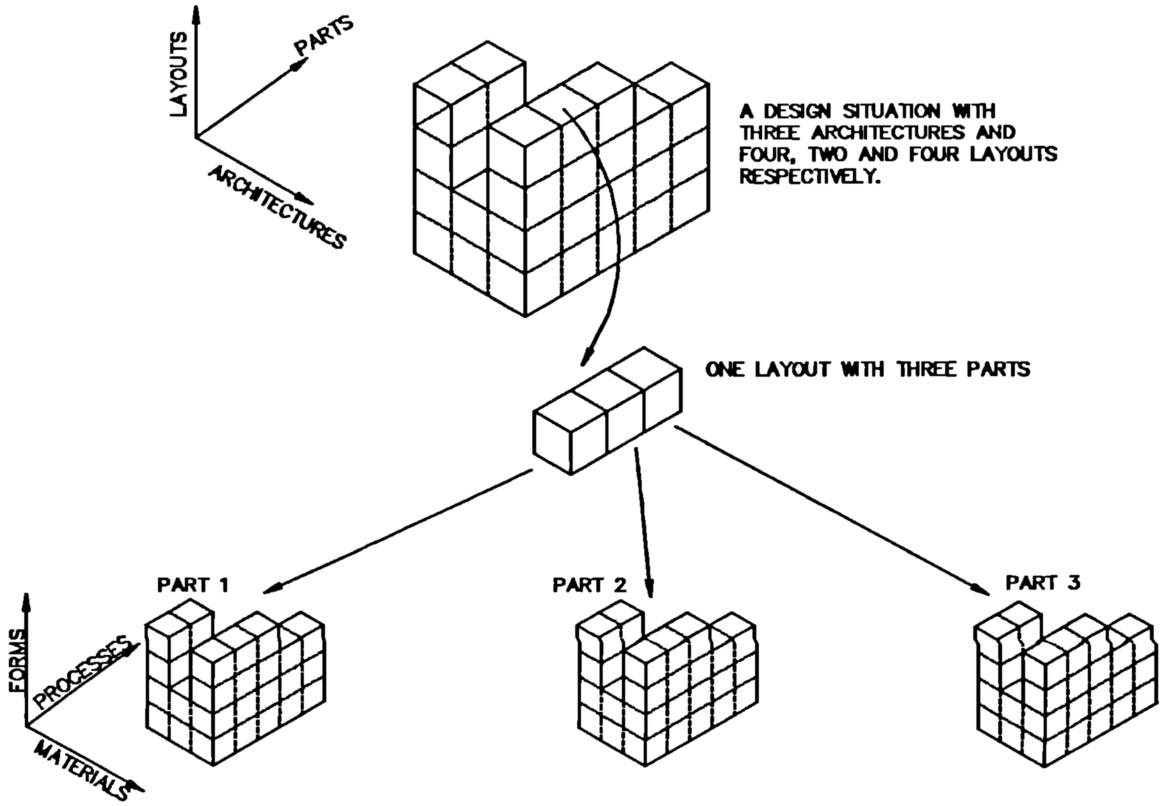


Figure 5.18 Morphology of the Complexity of Design in DFD

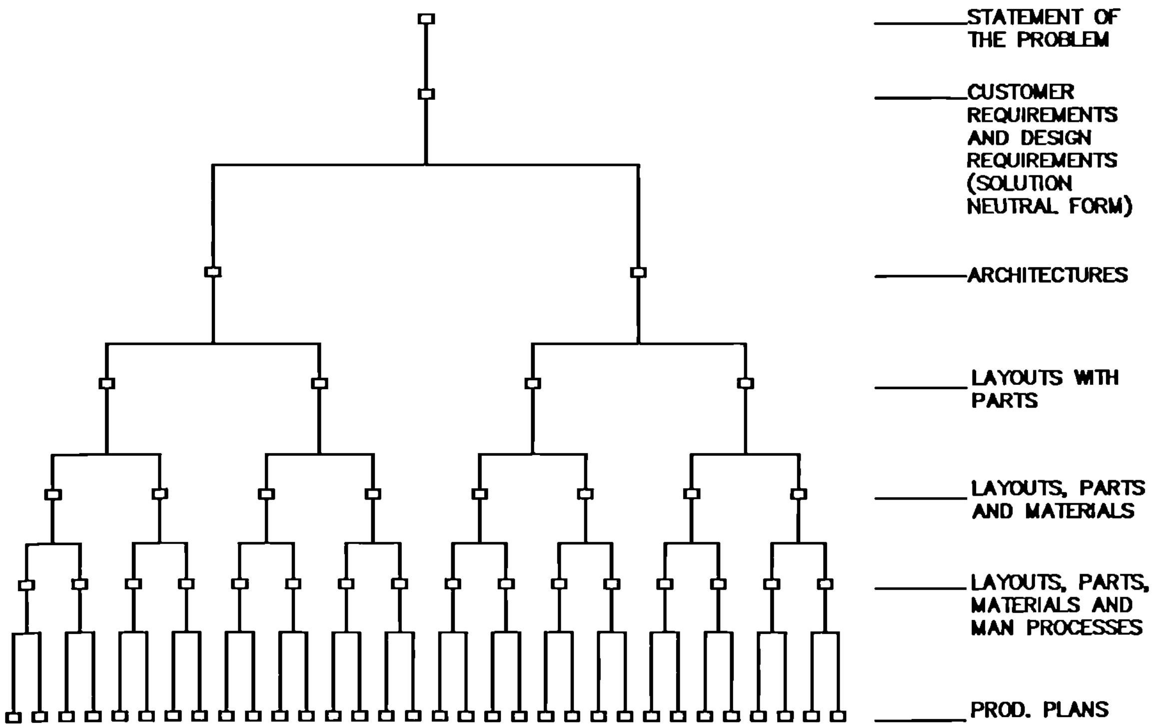


Figure 5.19 Increasing Complexity Through Design Stages in DFD

The complexity of product design right from the conceptual design stage to the detailed design stage, can hence be represented in a combined six dimensional space, with each dimension in theory varying from 1 to n. In practice, each dimension would be of a few order.

A major aspect of the design process, would be to contain the increasing divergence of possibilities of alternative design options in the form of architectures, layouts, parts, forms, materials and manufacturing processes (Figure 5.19). Situations would also arise during the design process that would necessitate the resolution of conflicts, deadlocks or bottle necks. In some instances, there would be need to tradeoff between competing customer or design requirements, as well as between subsystems and/or parts, thus resulting in the generation of new alternative architectures and layouts. The use of multi-criteria optimisation and sophisticated decision analysis techniques as well as the DFD charts, would be invaluable here. The morphological representation of the increasing complexity of product design as represented in Figure 5.18 helps to encapsulate the product design process within Design Function Deployment.

5.10 MANAGING THE DESIGN PROCESS WITHIN DESIGN FUNCTION DEPLOYMENT

5.10.1 Introduction

Two principal issues that constitute part of the ethos of concurrent engineering, and which are constantly highlighted are (i) the bringing upstream to the design stage, downstream issues which impinge on the design process and (ii) maximising the paralleling of the various design and other engineering activities. It is the second issue that this section seeks to address.

This section therefore discusses issues relating to paralleling the design process within the context of Design Function Deployment, DFD. Various techniques adopted within the systems and project management field are reviewed, particularly those aimed at reducing development time, such as PERT/CPM, Structured Analysis Design Technique and the Design Structure Matrix. Methods for decomposing design activites into groups of parallel and serial tasks will be discussed. The section then strongly enunciates a rational and systematic approach on how concurrent design can be performed in DFD, in relation to both the design process and the designed artifact or product. The benefits of this approach, in terms of separating the overall design into modular groups and activities, managing the complexity and interactions between the separate design tasks and subsequent reduction of the design cycle is also discussed.

5.10.2 Review of Design and Project Management Techniques

The adoption of the concurrent engineering approach to product development, implies that engineers, designers and product development managers, have to cope with a larger amount of design data and information. Engineers and designers in this climate, therefore need to adopt techniques for planning, organising and monitoring the large and complex network of design tasks arising in the product development process. Over the years, several techniques have evolved for managing complex and large projects as well as the design process. Such techniques include : Directed Graphs (DG), Project Evaluation and Review Technique/Critical Path Method (PERT/CPM), Structured Analysis and Design Technique (SADT) and Design Structure Matrix (DSM). These techniques are shown schematically in Figure 5.20 and are described below.

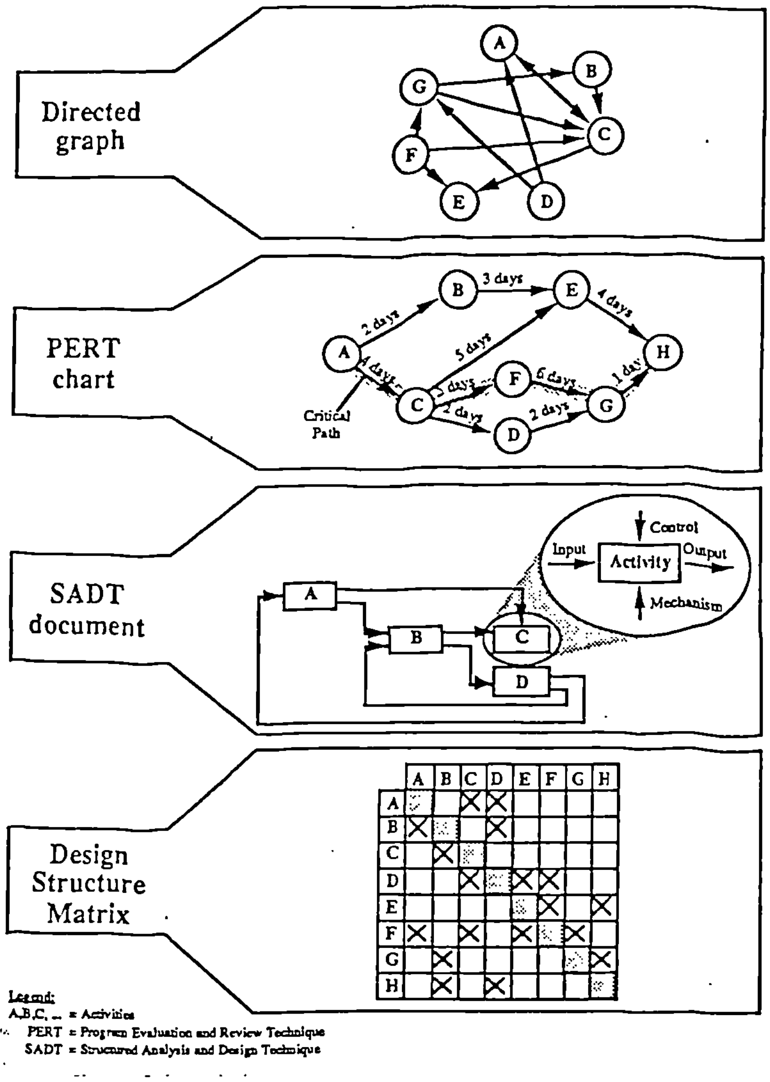


Figure 5.20 Existing Approaches to Design Process Modelling

Directed Graphs
In this technique, a graphical mapping involving nodes and arcs, are used to represent the entire design activity as a system of interconnected nodes. The nodes are used to represent individual sub-tasks, and the arcs which link the nodes are used to represent directed information flow. The link from one node to another, usually represent a required information transfer between the two nodes. The nodes are generally chosen arbitrarily, and as a result, it is difficult to discern any structure in the resulting digraph. The digraph also does not reflect the underlying structure of the design problem which it represents [181, 232].

Project Evaluation and Review Technique/Critical Path Method (PERT/CPM)
This technique, is perhaps the most popular of all the project management techniques. In developing a guide on its use, [233], use network diagrams to represent the precedence relationships among activities. Using the concept of nodes and arcs of the digraph technique, but with tasks placed along the arcs and the nodes representing task completion milestones, the resulting time line, represents a PERT chart. The length of the arc in this case, is then used to represent the activity's duration. In the PERT method, three time estimates are given to each task (optimistic, pessimistic, and a best guess). Probability of timely task completion can then be computed along with the associated start times for each activity. The critical path method (CPM), is then used to perform a linear time/cost tradeoff for tasks on the critical (longest lead time) path [182]. This technique however cannot be used to represent the circular information flows often encountered in design.

Structured Analysis and Design Technique (SADT)
This technique, which was developed by [234, 235], has been used widely for documenting project and design procedures in various disciplines. The technique uses a series of interconnected boxes and arrows (arcs) to represent information flow in the form of inputs or outputs from each activity. It essentially helps to give limited insight into the design process as well as the intra-task complexity.

The above three networking techniques (Directed Graphs, PERT/CPM and SADT) suffer from size limitations and an inability to explicitly represent circuits (coupling) of the design tasks. They are usually based on one way progression along paths, with no feedback or iteration and no-feedforward of information part-way through task [181, 182]. They tend to be used primarily for documenting design practices and do not handle interactions that occur within design tasks.

Design Structure Matrix (DSM)
The philosophy of the Design Struture Matrix (DSM) technique is that the design project can be divided into individual tasks, and the relationships and interactions among the tasks analysed to identify the underlying structure of the project. There is some belief that studying relationships between individual tasks can improve the

overall design process, as well as being an effective way of analysing alternative design strategies [236]. DSM with the aid of a graphical representation (matrix) uses the structure of design information flow to guide the decomposition of the design activity. In this matrix, the links and relations between the tasks are mapped out in such a way that makes their interdependence explicit. The design activity which consists of for instance **m** tasks, can be represented as an **m x m** matrix. Each of the tasks are labelled along the side of the matrix as row headings and across the top of the matrix as column headings, in an identical manner. The matrix element a_{ij} is non-zero if node **i** provides information to node **j**. A typical matrix is shown as the first one in Figure 5.21. Interpreting the task ordering as a time sequence helps to make the timing of information flow explicit. The marked elements within each row of the matrix identify the other tasks that must contribute information for proper completion of the design. In the matrix, marks below the diagonal represent information transferred to later tasks (i.e. task 2 must be completed before task 3); while marks above the diagonal depict information which are fed to earlier tasks (i.e. task 7 must be performed before task 3). The primary goal of design structure management is to find a sequence of these design tasks which allows this matrix to become lower triangular.

Once the design process has been established into a design structure matrix, the analysis proceeds in two separate stages, known as *Partitioning* and *Tearing* [181, 182, 237]. The process of partitioning aims at resequencing the design tasks, in order to maximise the availability of information required at each stage of the design process. Partioning helps to identify tasks which are coupled in a loop, and then clusters them as blocks on the diagonal of the design structure matrix (Figure 5.21). The strategies of the partitioning algorithm include: scheduling independent design tasks as early as possible and then simultaneously identifying tasks which are coupled for further analysis. Once partitioning has placed the design structure in a block-triangular form, the tearing analysis then begins. The goal of tearing is to resequence within the groups (blocks) of coupled tasks to find an initial ordering to start the iteration. The algorithms employed in tearing includes tearing with shunt diagrams and tearing with heuristics.The tearing process is exemplified in Figure 5.22. This form of matrix discussed above, represents the activity-activity incidence matrix. Other forms of incidence matrices have also been discussed by [184, 185]. These include the module-activity incidence matrix and the procedure (formula)-parameter (variable) incidence matrix. In these two cases, the analysis of the matrix, involves the use of suitable clustering algorithms [183, 238].

Decomposition of the design process and using the algorithms described above, has been considered to lead to the following advantages : (1) Separation of the overall design task into groups of design tasks and/or groups of modules and activities, (2) Potential activities that can be performed concurrently are detected, (3) The complexity of managing the design task is reduced and (4)The design cycle time is reduced.

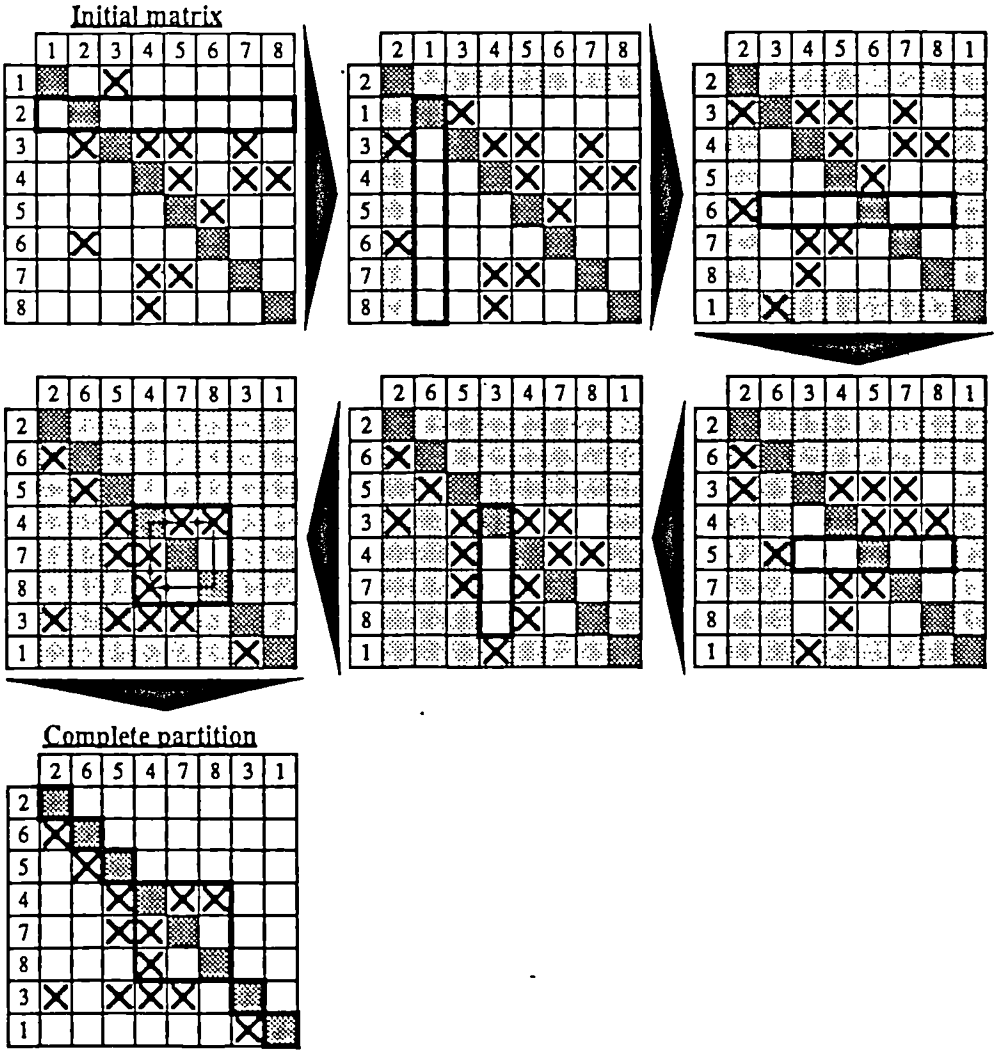


Figure 5.21 Partitioning Process in Design Structure Matrix

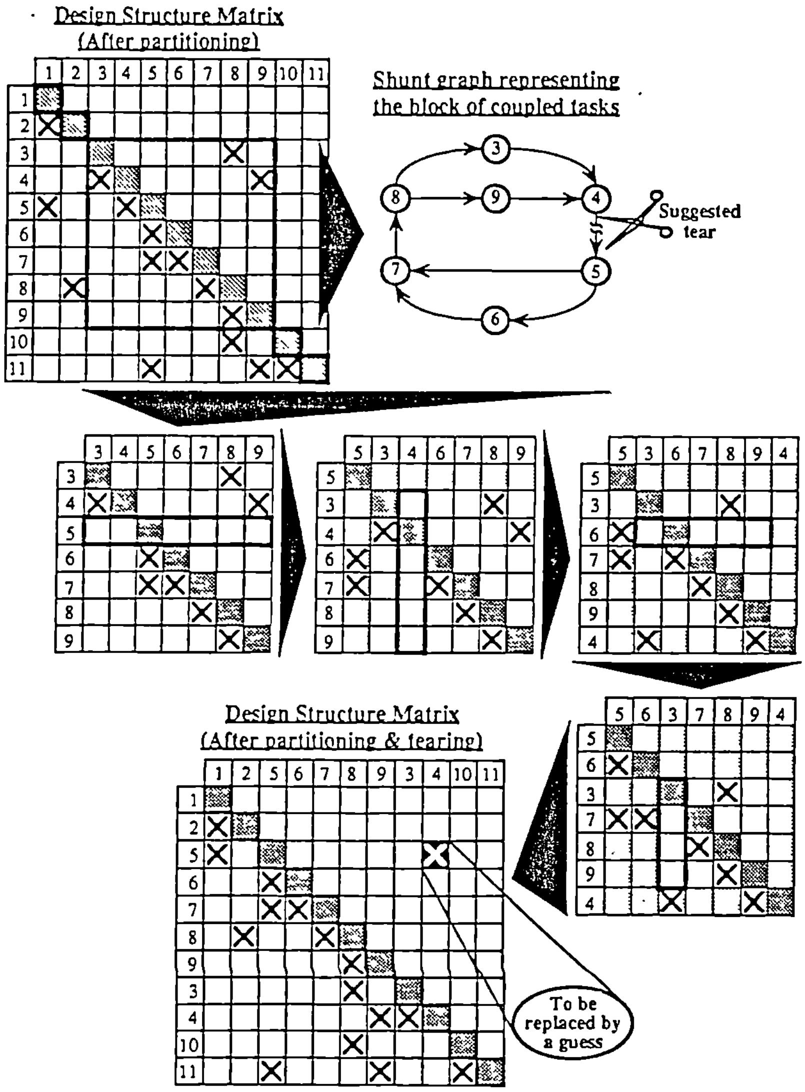


Figure 5.22 Tearing Process in Design Structure Matrix

5.10.3 Concurrent Design Within Design Function Deployment

The process of bringing design influencing downstream issues early upstream to the design stage, as part of fulfilling the ethos of concurrent engineering, does lead to an increase in the activities to be performed in the product development process. This factor coupled with the basic idea of shortening the product development cycle time, lends weight to the need to manage the design process and the associated interactions and complexities, more effectively. In Design Function Deployment, the process of concurrent (parallel) design, is considered from two main viewpoints, which are :(i) Concurrent design associated with the the design process (CDDP) though stages 1 to 5 and (ii) Concurrent design associated with the design artifact (CDDA). Considering the first case, that is, CDDP, it should be noted that the flow through the five stages of DFD, as shown in Figure 5.23, represent the minimum prescribed path (critical path) to be followed in the design process, and is thus not subject to any form of concurrency.

At each of the stages, several design activities are performed, with some of them somewhat repeated down the later stages, and in other situations somewhat overlapping. It is these activities that can be explored for possible concurrency, and to speed up the design process within each of the stages. Table 5.1 shows these activities for each of the DFD stages. Some of these activities (or design tasks) which take place in each of the DFD stages, would be inter-disciplinary, relating to for instance (marketing, design, manufacturing, finance, testing, etc.), while others will be intra-disciplinary (stress analysis, geometric modelling, feature-based design, thermal analysis, etc - all within for instance the design group). The above implies that the larger inter-disciplinary team would be involved in the inter-disciplinary activities, while the smaller design team would be responsible for the intra-disciplinary activities. Irrespective of whether these activities are inter- or intra-disciplinary, it is needful to decompose all the activities within each DFD stage and then to establish those that can be done in parallel and those to be carried out serially to precede other activities. In this concurrent design case, the applicable form of incidence matrix is the activity-activity matrix, which can be employed at each of the DFD stages. With this form of matrix, the triangularisation algorithm (Partitioning and Tearing) can then be used to decompose the activities into groups of activities, thus simplifying the entire design process and these activities can then be sequenced in such a way as to speed up the product development time. The result of the decomposition, would be the overlapping situation that occurs between the inter- and intra-disciplinary activities as they are being performed, with activities belonging to either of them being performed in parallel or in series. It is worth noting that the number and type of activities that would be carried out in parallel or in series would depend not only on the engineering domain of the design, but also on the nature, type or class of the product.

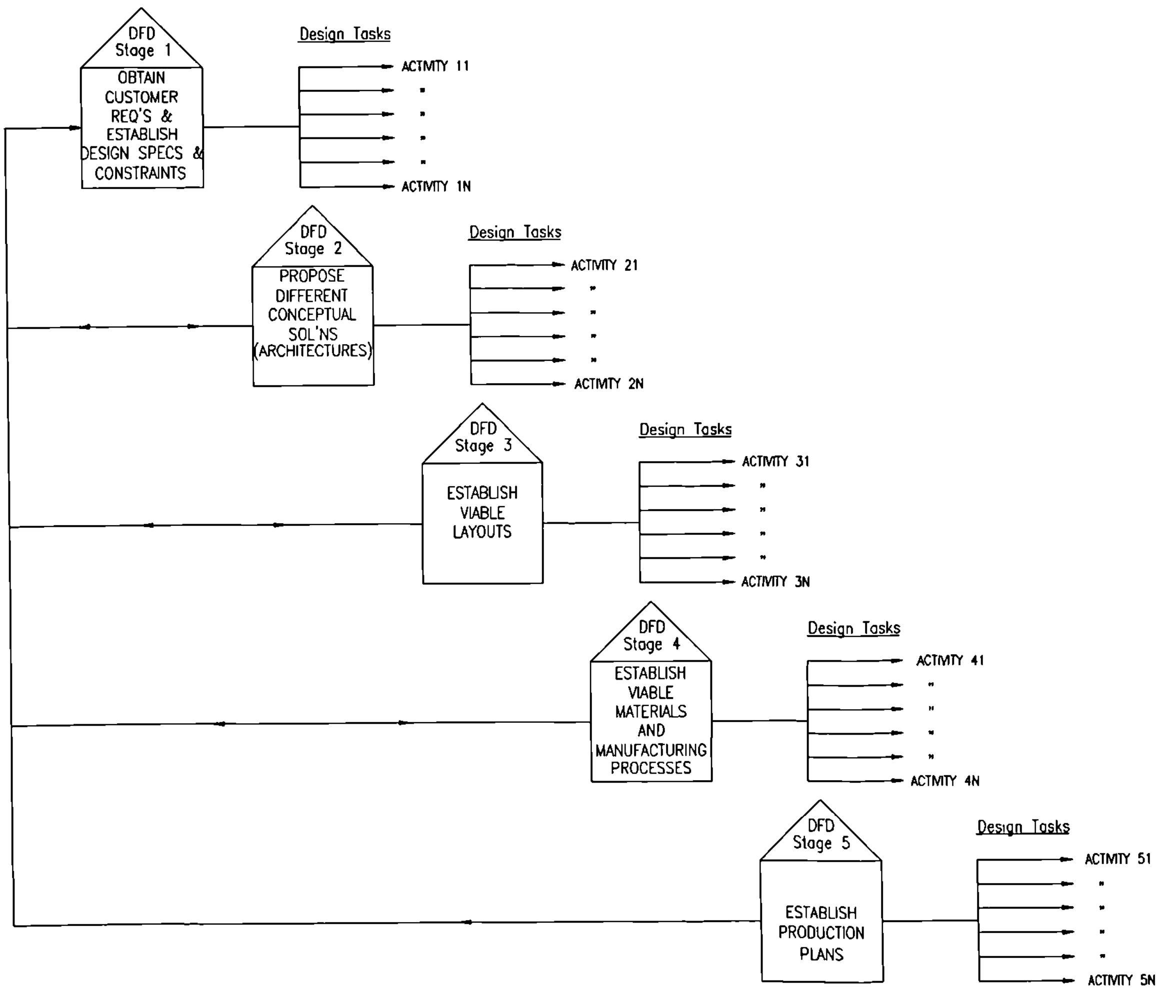


Figure 5.23 Flow of the Design Process in DFD

Table 5.2 Design Activities in DFD

Stage 1	Stage 2	Stage 3	Stage 4	Stage 5
Elicit Requirements	Establish subsystems from stage 1 Design Functions	Generate Layouts for each Architecture	Establish Viable Materials and Manufacturing Process	Job Scheduling
Analyse Requirements	Generate and Synthesise concepts (Architectures)	Derive Design Functions for each part of Layout	Derive Manufacturing Design Functions for each part of Layout.	Line Balancing
Categorise Requirements	Derive Design Functions	Analyse Design Functions	Analyse Design Functions	Batch Size Determination
Prepare Quality Plan	Analyse Design Functions	Group Design Functions	Group Design Functions	Routing Procedures
Derive Design Functions	Group Design Functions	Set Target Values for DFs	Set Target Values for DFs	Capacity Analysis
Analyse Design Functions	Set Target Values for DFs	Complete Correlation Matrix	Complete Correlation Matrix	Planning for Inspection
Group Design Functions	Complete Correlation Matrix	Complete Relationship Matrix	Complete Relationship Matrix	Simulation of Production Plans
Set Target Values for Design Functions	Complete Relationship Matrix	Rate Design Functions	Rate Design Functions	Layout, Machine and Tooling Procedures
Complete Correlation Matrix	Rate Design Functions	Materials Selection	Establish Critical Manufacturing Processes	
Complete Relationship Matrix	Evaluate each Architecture	Evaluate all Layouts	Manufacturing Process Plans	
Rate Design Functions	Explore Materials and Manuafacturing Processes	Select Viable Layouts	Optimise Manufacturing Process Plans	
	Select viable Architectures	Create Solid Models for Viable Layouts	Simulation of Manufacturing Processes	
		Design Retrieval	Mat. Resource Planning	
		Finite Element Analysis	Evaluate against DFM, DFA and DFDisassembly	
		Mechanism Analysis	Assembly Planning	
		FMEA/FTA Analysis	Rapid Prototyping	
		Robust Engineering Design		
		Optimisation		
		Evaluate against DF'Xs'		
		Rapid Prototyping		
		Select Optimal Layout		

In the second case, that is, CDDA, the performance of concurrent design depends to a large extent on the type and nature of the product. Products can generally be classified as static (fixed concept) or dynamic (short life cycle and requires new concepts), as well as overconstrained or underconstrained (could be ideas or skills based). For this case, four forms of concurrent design are considered. The first form involves the design of competing alternative architectures (at stage 2 of DFD) in parallel, as shown in Figure 5.24 and the second form involves the design of competing alternative layouts (at stage 3 of DFD) in parallel. The third form of concurrent design relates to designing in parallel the individual subsystems that constitute an architecture (between stages 2 and 3), while the fourth form is associated with the parallel design of the individual parts of a particular layout (between stages 3 and 4). Depending on the product type, either one or a combination or all of the four forms of CDDA, will be applicable. In the case of the third and fourth forms of concurrent design, the applicable incidence matrix would be the module-activity matrix, where module refers to either subsystems of an architecture or parts of a layout. The size of this matrix would depend on the scale and complexity of the product.

For static-simple(small scale)-underconstrained (SSU) products, the second form of CDDA would be more applicable. In the case where the SSU product has several and/or fairly sizeable parts that can be designed independently and if considerable effort is required, then there would be need to employ the fourth form of CDDA. In the case of static-simple(small scale)-overconstrained (SSO) products, no form of CDDA is applicable. The fourth form may however be applicable, if a similar condition to that of SSU occurs. For static-complex-underconstrained (SCU) and static-complex-overconstrained (SCO) products, the third and fourth forms of CDDA, are more applicable. The second form may also be applicable to SCU products, if sufficient resources are available, or if the parallel design can be done by a different arm or division of the enterprise. Dynamic products are to all intents and purposes underconstrained products. In the case of dynamic-simple-underconstrained (DSU) products, the first, second and fourth forms of CDDA are more applicable, while in the case of dynamic-complex-underconstrained (DCU) products, all four forms of CDDA are applicable. For both innovative and creative designs, the four forms of CDDA are also applicable, while in the case of adaptive and variant designs, only the second, third and fourth forms of CDDA are relevant. It should be noted that all the four forms of CDDA, go hand in hand with CDDP.

Dimensionality of Concurrent Design in Design Function Deployment

The employment of only CDDP, represents a one- dimensional case in design concurrency, irrespective of the DFD design stage, and this relates to SSO products. Here there is no occurrence of CDDA.

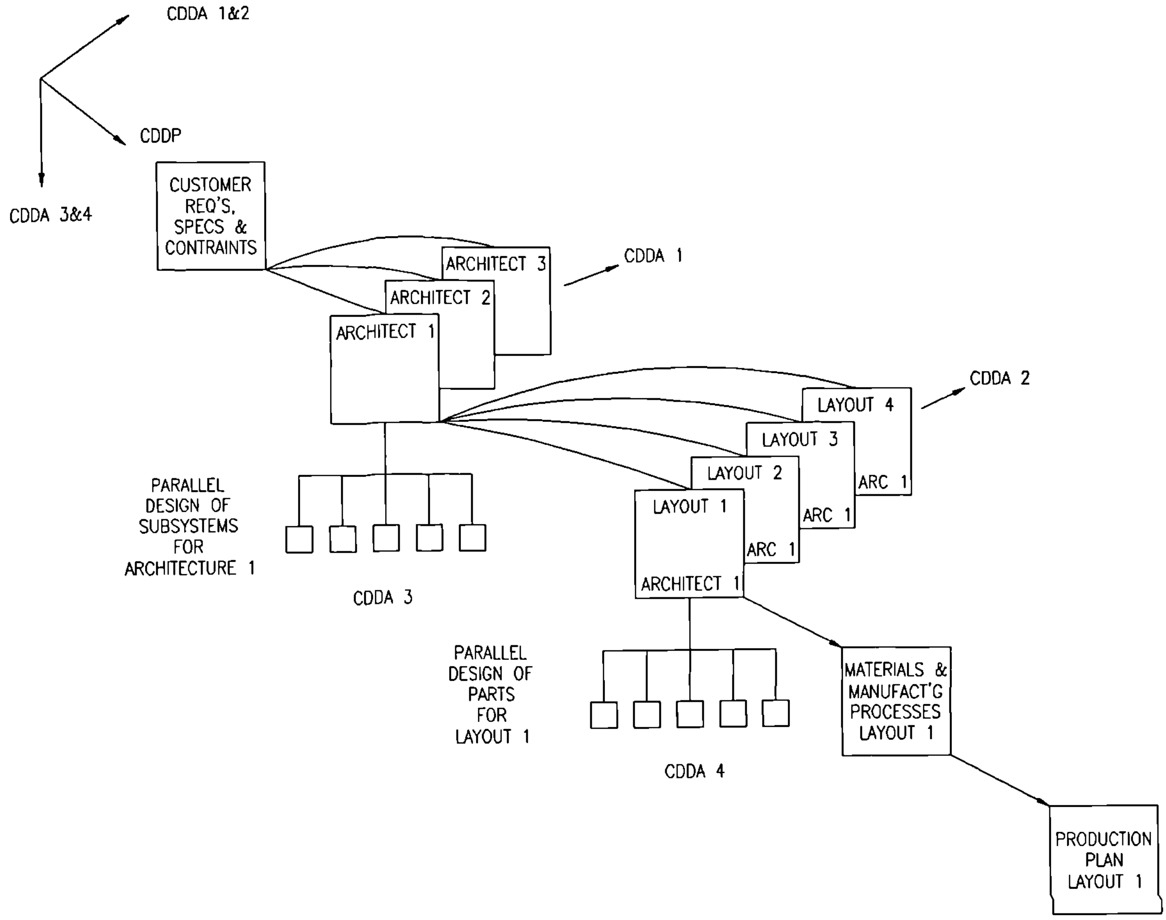


Figure 5.24 Concurrent Design in Design Function Deployment

When either of the four forms of CDDA are employed in addition to CDDP, then we have a two-dimensional case of design concurrency, occurring between stages 2 and 4. In the situation where either the third or fourth or both forms of CDDA are employed in addition to either the first or second or both forms of CDDA, in addition to CDDP, then we have a three-dimensional concurrency case, as in DSU and DCU products. For SSU, SCU and SCO products, the two-dimensional case predominates.

The Role DFD Charts in Concurrent Design

A principal activity performed at each of the DFD stages, is the creation of the DFD charts. These charts play a significant role in the concurrent design process associated with the design artifact (CDDA), as they are used amongst other things to store and capture alphanumeric (textual) data about the product under development. In both stages 2 and 3 of DFD, the charts apart from providing a single shot of all data about an a design, also provide, two main functionalities that support the concurrent design of the product. The first functionality relates to the use of the charts in categorising the subsystems and their characteristics for each architecture (stage 2) as well as parts and their characteristics for each layout (stage 3), and thus helping to provide insight into the number, nature, and scale of these subsystems or parts, as well as their interactions and dependencies. This helps to create the framework for embarking on paralleling the design of the subsystems and/or parts. In the case of the second functionality, the correlation matrix (that is, the roof of the charts) is used to explore the interactions, conflicts, dependencies and trade-off scenarios between either the subsystems or parts, as well as aiming to provide solutions for these bottlenecks. This is done through a process of elimination, combination, modification, transference, simplification and standardisation of the subsystems or parts.

Another functionality that is emerging from the correlation matrix, is its potential use as a form of module-module incidence matrix, where module refers to either subsystems or parts. The aim here would be to employ in the same way as before, the triangularisation algorithm in decomposing the subsystems or parts into groups of subsystems or groups of parts, and then to sequence their design, establishing which subsystems (stage 2) or parts (stage 3) should be designed in parallel and those to be designed in series.

This particular functionality helps to actualise the third and fourth forms of CDDA. Whereas the concurrent design of either the subsystems of an architecture or the parts of a layout, were described earlier, how this would be done was not obvious. The use of the correlation matrix in the way described above, appears to provide the solution to how the concurrent design of subsystems and parts can be implemented.

5.10.4 Summary

It is increasingly becoming apparent, that for the concurrent engineering paradigm to survive, and for the benefits of concurrent engineering to be realised in industry, efforts are needed to develop tools, techniques, as well as methodologies for achieving concurrent engineering in practice. The aim of this section has been to examine the principal issue of reducing product development cycle time, and how it can be achieved in practice. In this regard, several design, systems and project management techniques have been reviewed. Building on these available techniques, particularly the Design Structure Matrix, DSM, a rational approach to concurrent design within the taxonomy of Design Function Deployment, DFD, has been enunciated. This research work is on-going, and the next stage of the work will be focussed on how to implement the concepts developed in this paper, within the DFD software system. Other research efforts will also go on in parallel as regards modelling and simulating the design process within DFD using case studies, as well as testing the techniques with current practices in industry.

5.11 PRODUCT MODELLING WITHIN DESIGN FUNCTION DEPLOYMENT

5.11.1 Introduction

A major limitation of conventional CAD systems is the lack of integration of both textual and geometric design data. There is need to properly document textual data representing design ideas, intent, descriptions and specifications [239]. The integration of textual and geometric design data is one of the key issues the product modelling environment within DFD, will seek to address. Kimura [240] defines a product model as "a generic model used for representing all types of artifacts which appear in the course of manufacturing. It represents target products, their materials and intermediate products, tools and machines, and any other manufacturing resources and environmental objects". Another definition given by Anumba [241] states that "a product model can be defined as an abstract description of a product with ideally all relevant product data stored in the model and able to be abstracted to documents in various formats".

Within Design Function Deployment, Product Modelling can be defined as "the capturing and storing of data that is necessary for testing (by analysis, simulation, etc) and optimising the product, and to perform the downstream activities such as manufacturing, use, service and disposal" [242].

5.11.2 The Rationale For An Integrated Product Modelling Environment

The lack of utilisation of the full benefits of most modern CAD systems can be attributed to the large number of associated software programmes which exist as islands of automation. In industry, the general trend is to perform manual transcriptions of product design data between each of these software packages, resulting in time consuming activities which are costly and also prone to many errors. Efforts made in the recent past to address this, has involved the development of dedicated interfaces for data transfer between any two or more software packages. The total number of interfaces (T) required is given by n(n-1), where n is the number of software packages. T rises phenomenally, as n becomes a significant number. To address this a number of initiatives have emerged over the years, aimed at developing a core product model that is large enough to be a repository of all the information of the product. This has culminated into what is known as ISO 10303 standard, STEP (Standard for the Exchange of Product Model Data) [243]. The provision of a core product model will inevitably reduce the number of interfaces between the central storage region and the associated software packages. Each of these packages can then have their bi-directional interfaces to this core model. In this case, the total number of interfaces (T) is given by 2n (see Figure 5.25).

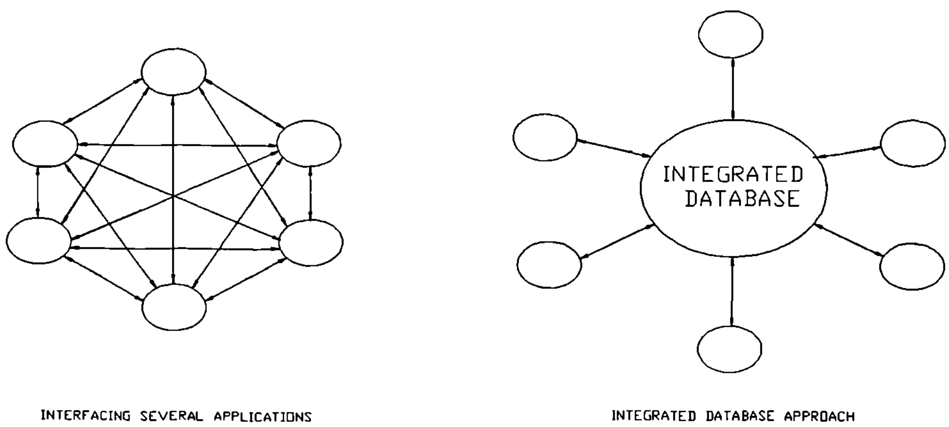


Figure 5.25 Multi-Application Integration

Furthermore, virtually all of the CAD systems currently available in industry, tend to represent only the geometric information about a product. They generally do not have any form of representing the textual information generated at the early conceptual stages of design such as customer/user/client requirements, specifications and constraints, design goals, the rationale for the design decisions made and the flow of design information. This condition creates fragmentation of product information, from product inception right down to retirement, as well as resulting in an insufficient support of the many engineering activities during product development [244]. Another disadvantage, is the prevention of effective retrieval of existing relationships between requirements and solutions as well as retrieval of completed designs.

It is becoming widely accepted that there is need for a more rational approach to the representation of an evolving design, besides the traditional 2D or 3D geometric model. Other issues that have contributed to this need include the following [245]:
(1) the lack of integration of product information makes data flow among systems difficult.

(2) difficulties arising from discrepancies between documents and the shear volume of drawings and revisions.
(3) computerisation of more activities in the design process which is making paper documents to become less convenient as a means of information transfer between activities and team members.
(4) the option to transfer design information in digital form would help to (i) avoid transcription and interpretation of errors, (ii) save effort and time and (iii) make it more amenable to revise and update control systems.
(5)There is an increasing need for advanced information technologies to integrate and coordinate various life cycle considerations during product development. The product modelling environment will generate an information reservoir of complete product data to support various activities at the different product development stages [246].

5.11.3 Requirements of An Integrated Product Modelling Environment

Having established the need for an integrated product modelling environment, it is needful to establish the requirements of such an environment within an integrated concurrent design system. Such requirements include the following [247-249]:

(1) The product modelling environment should support the complex and large volume of product data.
(2) It should be stable while supporting modifications or revisions to the product structure.
(3) It should support a complete representation of the product as much as possible, right from requirements to manufacturing and production planning.
(4) The modelling environment should be flexible to allow the addition of new types of product data as well as new views of the product.
(5) The product modelling environment should be implemented within an open system architecture, in order to take full advantage of available information technologies as well as minimise capital investments.
(6) The product modelling environment should be developed to support concurrent engineering principles and practices.
(7) It should take into account the complexity of product development tasks and their inter-dependencies.
(8) The product modelling environment should support (i) the presentation of actual product data, (ii) facilitate product documentation and (iii) offer decision alternatives both in terms of strategies and with respect to choosing between design solutions.
(9) It should support the preservation of alternative decisions made about the evolving product, at the early stages of design in order to take account of uncertainties downstream.

(10) It should support the representation and exploration of process and product alternatives in order to reduce costly iterations and increase product flexibility.
(11) It should have adequate means for capturing the textual and geometric data of the evolving design, and to store them in an organised manner.
(12) The implementation of a product modelling environment should incorporate (i) a complete representation of the physical design process, (ii) sufficient data models for representing technical objects and their associated textual and geometric information, (iii) adequate user friendly man-machine interfaces and (iv) an integrated system architecture with the capability to communicate between different design modules.
(13) The product modelling environment must capture the functional, behavioural and physical aspects of the evolving product.
(14) The product modelling environment must support the performance of consistent transitions between models in an integrated environment.

5.11.4 Product Modelling Within Design Function Deployment

The above discussions on the rationale as well as the requirements for an integrated product modelling environment forms the basis for the development and implementation of such an environment integrally within Design Function Deployment (DFD). DFD provides an integrated framework for capturing all product design information generated throughtout the design process. In general, DFD will enable the capture of both textual and geometric information of the evolving product. The textual information will be principally captured by the use of the charts through stages 1 to 5, while geometric data will be captured with the solid modeller in level 2 and the design database in level 3 of DFD. The product modelling environment in DFD is being envisaged to provide a comprehensive product model, which includes both textual and geometric product information, a database management system to store design and a design retrieval system to extract full or part details of existing designs in the database.

The scenario for the product modelling environment within DFD, is both dynamic and hierarchical in nature, represented by what can be regarded as the "Total Product Model". It is dynamic in the sense that the product model is somewhat open-ended, evolving and enlarging as one progresses from stage 1 of DFD down to stage 5, encompassing product data corresponding to the design process of level 1 and to individual design modules which are utilised in level 2. The hierarchical aspect of the product modelling environment is associated with the generated solution space, as product model abstractions will be represented for both top level conceptual solutions (architectures) and detailed solutions (layouts), as well as the lower level subsystems of architectures and parts of layouts.

Ensuing discussions will thus focus on the product modelling representations within each stage of the design process in DFD.

In stage 1 of DFD, two forms of product models are represented namely :

(1) Textual model (captured by the DFD chart) and (2) Cost model. The textual model captures two main sub-models, that is, the requirements model and the specifications and constraints model. The cost model is associated with the cost attributes applied to the requirements and the specifications and constraints model.

In stage 2, the models represented for each viable architecture are :

(1) Textual model - consisting of (a) the specifications and constraints model and (b) the subsystems and subsystems characteristics model
(2) Function model - representing the decomposition of the functions of the subsystems for each architecture
(3) Subsystems model - lists of each susbsytems within an architecture and associated models depending on their complexity. Each of these subsystems can also have the same type of models associated with their parent architecture, such as the textual model, geometric model, cost model, etc.
(4) Geometric model - geometric representation of the architecture
(5) Cost model - associated with the function model and the textual model
(6) Topological model - consisting of (a) the sketching model and (b) morphological analysis model, i.e. model of possible solutions and (c) 2D layout drawings.

DFD stage 3 involving the generation of viable detailed solutions (layouts) represents the stage where the design solution space is generated, and a larger number of models are represented here for each layout. These include :

(1) Textual model - consisting of (a) subsystems and subsystems characteristics model and (b) parts and parts characteristics model.
(2) Function model - representing the decomposition of the functions of the parts for each layout.
(3) Parts model - bill of parts within a layout and associated models depending on their complexity. Each of these parts can also have the same type of models associated with their parent layout, such as the textual model, geometric model, cost model, optimising model, analysis model, etc.
(4) Geometric model - geometric representation of the layout, which can consist of a wire-frame model, surface model, and solid model.
(5) Analysis model - consisting of FEA model, mechanism/kinematic model, etc.
(6) Optimisation model - consisting of robust engineering design model and multi-criteria optimisation model.
(7) Failure model - consisting of the FMEA and FTA models.
(8) Cost model - associated with the function model and the textual model.

(9) Topological model - consisting of (a) the sketching model and (b) morphological analysis model, i.e. model of possible solutions, (c) form element model and (d) 2D layout drawings.

In stage 4 of DFD, the models represented are related to the materials and manufacturing processes associated with the parts of a viable layout, and they include the following:

(1) Textual model - consisting of (a) parts and parts characteristics model and (b) manufacturing process and process parameters model.
(2) Parts model - bill of parts within a layout and associated models depending on their complexity. Each of these parts can also have the same type of models associated with their parent layout, such as the textual model, materials model, manufacturing model, assembly model, etc.
(3) Materials model - representing the materials for each part of a layout.
(4) Manufacturing planning model, (5) Assembly planning model and (6) Cost model - associated with materials and manufacturing process parameters.

In stage 5 of DFD, involving the generation of production plans, the models represented include the following :

(1) Textual model - consisting of (a) manufacturing process and process parameters model and (b) production planning operations and operations parameters model.
(2) Parts model - bill of parts within a layout and associated models depending on their complexity. Each of these parts can also have the same type of models associated with their parent layout, such as the textual model, manufacturing planning model, assembly planning model, production planning model, tooling model, etc.
(3) Materials model - representing the materials for each part of a layout.
(4) Manufacturing planning model, (5) Assembly planning model, (6) Production planning model, (7) Tooling model, (8) Inspection model and (9) Cost model - associated with materials and manufacturing process parameters.

The above model representations for each of the DFD stages are shown schematically in Figure 5.26. In each stage, the textual models are captured with the aid of the DFD charts.

5.11.5 The Master Modelling Concept

A key requirement of the product modelling environment is to support all the life cycle issues associated with the evolving product in a concurrent manner, by the provision of a central and common product database. This is to ensure that the design data and information used during design, jigs and fixture design, manufacture, packaging, commissioning, service, decommissioning, disposal, and other relevant areas, is the same without any discrepancies.

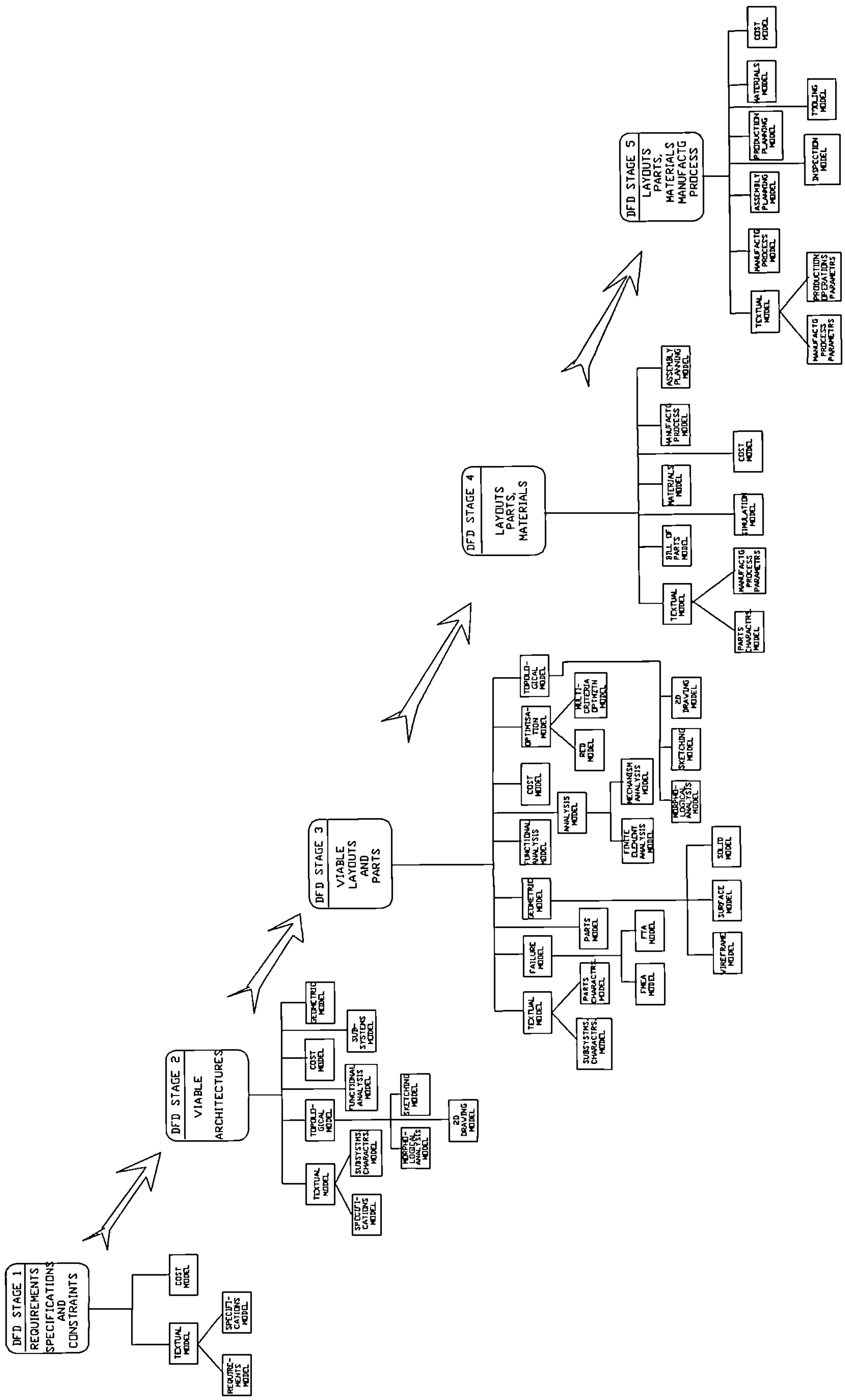


Figure 5.26 The Dynamic and Hierarchical Product Modelling Process in DFD

Thus enabling a common sharing of design information, removal of unnecessary interfaces between design groups, provision of more current and consistent information, facilitation of timely feedbacks on the design and subsequent reduction of delays. The product 'master model' is thus created and stored as early as posssible and then used for the necessary, analysis, evaluation and optimisation in a concurrent manner. The core representation of the master model is the geometric model, which is supported by other models created during the design, such as the requirements, specifications and function models as well as other downstream models such as the materials, manufacturing process, assembly and production planning models.

5.11.6 Transition Between Product Models in DFD

A key aspect of the product modelling environment within DFD, is the storage of the different models and associated sub-models within a common repository, either in the form of files within a directory or within a database. The common storage location will contain references and cross-references between the models either within a DFD stage or between models across the stages. The product modelling environment will administrate the information distributed in the different models. The necessity for integrating these models, is due to the fact that each model only represent a subset of all the properties of the evolving design. What is then required is to be able to perform consistent transitions between the models. Transitions must of necessity occur between models in the different DFD stages, with links established between the models in stage 1 right through to stage 5. This is necessary for traceability purposes, the maintenance of consistency in the design output, and design reviews. Other transitions will also occur between models within a particular stage, e.g. transition between the solid model and the FEA model, in stage 3 of DFD.

5.11.7 Summary

This subsection has discussed product modelling in DFD within the context of the rationale for an integrated product modelling environment and the requirements of such a system. The main objective of the product modelling environment within DFD is to capture and represent the partial product models associated with the product's life cycle in an integrated manner as well as integrating both the textual (alphanumeric) and geometric data associated with any of the models. Further discussions were focussed on the various types of product models that are represented within DFD. The product modelling environment within DFD would be superior to other existing models (mainly geometric) and would constitute the main information carrier in the integrated DFD system. It will also enable the preservation of connections between the different models right from the requirements and specifications stage, i.e. DFD stage 1 to the production planning stage (DFD stage 5). A key function of the modelling environment will be the support for associations, cross-references and transitions between the models.

5.12 CONCURRENT DESIGN KNOWLEDGE CAPTURE IN DESIGN FUNCTION DEPLOYMENT

5.12.1 Introduction

The act of designing within Design Function Deployment, is characterised by the many design activities performed during the design process. Consequently, many different forms of design information referred to here as 'Design knowledge' are generated. Such design knowledge which might be in the form of textual, graphic, geometric information, and which are associated with both the design process and the evolving design artifact (product), are usually not well documented, and tend to be stored in different forms and locations in an incoherent way. In the cases where there is some form of coherent storage, it is usually the information about the design artifact that is stored, with the design information associated with the design process usually stored in the designers head (memory).

The process of capturing design knowledge within the context of Design Function Deployment, will involve acquiring and storing design information generated throughout the five stages of the design process. This information will be stored and organised in both human and computer interpretable form in a way that meaning is associated with it.

This section thus discusses the issues that highlight the need to capture design knowledge in a rational way, the types of design knowledge generated within the DFD process, and how they are structured and stored within DFD. The possible approaches that can be used for intelligently retrieving the design knowledge within DFD, will also be enunciated.

5.12.2 Motivation and Goal of Design Knowledge Capture

In the past, the issue about capturing design knowledge within an enterprise, has not been high on the agenda. In many circumstances, the original design requirements, the different alternatives pursued and the underlying intent and logical support for each design choice are usually lost or at best represented in scattered collections of paper documents, personal notebooks as well as the recollections of the designers. Within the last decade however, the increasing complexity of products and associated manufacturing processes as well as the intense global competition not only in the production of quality products but also in their release on time to market, has made the need to capture necessary design knowledge in a highly reusable form very critical. Several authors [250-255] have also highlighted additional reasons why the capture of design knowledge is imperative. These are discussed below.

(1) The capture of design knowledge is crucial when such knowledge is likely to be needed in the future. The original designers might not be available in the future when such knowledge is required and the engineering expertise already retained can hence

become invaluable for future projects.

(2) Design knowledge capture is also crucial in situations where (a) design decisions change throughout the design life cycle. In this case, using the documented design information, modifications can be made in a more sound and confident manner, (b) there is likely to be some turnover of designers during the life cycle of the product. In this case, the departure of a designer will also mean the loss of the design knowledge associated with him or her, unless the knowledge has been captured. In which case, when a new person enters the design team, the available design knowledge can then be used to educate and train the person on the history and state of the design. A typical case of the above involved the development of the NASA space station [256, 257]. The sheer magnitude and complexity of such a design project coupled with the operational lifetime makes the capture of design knowledge indispensable.

(3) Design knowledge can also be captured in order to explicitly represent design intent and rationale as this can help individual designers clarify their own thinking as well as allowing design team members to critique and augment the reasons behind the decisions about the design [258], e.g. by keeping track of the relationships and differences between the design options explored [259, 260], ensuring that all relevant issues and requirements have been addressed [261], detecting flaws in one's reasoning [262], identifying and tracking the consequences of changes in requirements and design decisions as well as potential resolutions of conflicts between designers [263, 264].

(4) The capture of design knowledge can also ensure that the rationale for design decisions made by one group of designers can be used by other groups, and thus avoiding the redundancy of effort and incompatibility of design decisions.

(5) The capture of design knowledge can also support the generation of diagnostic procedures to help maintainability.

(6) The capture of design knowledge is also useful for : *Explanation* - to explain how and why a particular design decision was made, *Verification and Simulation* - to evaluate and determine if the characteristics of the final design are consistent with the intended characteristics as represented by the top level objectives (requirements and specifications), *Modification* - to predict the effect of making changes to the design and *Re-use* - to synthesise a design from previous designs with a similar specification.

(7) The capture of design knowledge also helps to represent the whole spectrum of design information generated throughout the design life cycle, including originating requirements and specifications, alternative design paths and solutions (both chosen and neglected), arguments for and against and rationales for decisions made, as well as textual and geometric design data. Thus capturing the *design history*.

5.12.3 Design Knowledge Capture in Design Function Deployment

The design process in DFD proceeds through five stages of : (1) Establishment of requirements and specifications, (2) Establishment of viable architectures, (3) Development of viable layouts, (4) Establishment of materials and associated manufacturing processes and (5) Generation of production plans

Throughout each of the above five stages, many design activities are performed in translating the originating requirements into a detailed design and a final constructable product or system. These activities also involve the use of many design tools and methods as represented in level 2 of the DFD structure (Figure 5.6). The consequence of these design activites is the generation of a host of design information (data) associated with both the design process and the design artifact, and which are principally in the form of textual (alphanumeric) and geometric data. The generated design information which is associated with both the evolving product and and the design process, needs to be fully captured and represented within the DFD design system. The approach to be adopted for this is to concurrently capture the various forms of design knowledge (information) at each of the design stages in DFD.

Types of Design Knowledge Within DFD

The design knowledge (information, data) generated within DFD can be categorised into two main groups, that is, (1) Design knowledge associated with the design artifact (product or object) and (2) Design knowledge associated with the design process. The design knowledge associated with the design artifact include :

(a) *Product functional knowledge* - This relates to the functionality of the product, and represents the purpose for which the product is designed, that is, "what the product or object does".
(b) *Product physical knowledge* - This is concerned with the static characteristics of the product. It describes the product in terms of the components from which it is made (decompositional knowledge), how these components are assembled together (structural knowledge) and other characteristics pertaining to the static nature of the product (physical and materials properties). It's representation describes "what the product or object is".
(c) *Product behavioural knowledge* - This relates to the dynamic characteristics of the product within its estimated or actual operating environment, that meet its functional requirements. Behavioural knowledge describe the way in which product functions are achieved, i.e "how the product or object does it".

The design knowledge associated with the design process include :

(a) *Originating (initial) requirements* and how they are translated into solution neutral design specifications and constraints.

(b) *Design specifications* and how they are deployed and satisfied in conceptual designs (architectures) and detailed designs (layouts).
(c) *Design actions and decisions* - These represent actual decisions made in the performance of any activity, during the design process.
(d) *Design options/alternatives* - This sort of knowledge represents the alternative solution options considered in the course of the design process. These alternative solutions within DFD can be associated with (architectures, layouts, parts, forms, materials, manufacturing processes and production plans).
(e) *Design intent and design rationale* - Design intent represents the purpose, aim, goal or objectives that motivate design decisions made. Design rationale which is somewhat synonymous to design intent, represents the justification (logical basis, fundamental reason) for design decisions made. It can also be regarded as a record of the reasoning process, that is, a record of design options considered, choices that were (and were not) made along with the reasons for the decisions as well as how designers satisfied themselves that their action would work as intended [252]. Mostow [96] discusses design rationale as representing : (i) why a design action was taken, (ii) why a design action/plan taken to achieve a goal ought to work (correctness rationale), (iii) why an alternative design action was taken instead of another one (appropriateness rationale) and (iv) why a particular set of criteria was used in evaluating designs.
(f) *Constraints on the design process and product* - This relates to restrictions imposed on the design process, and might involve lack of human and financial resources, as well as lack of time. In the case of the product, constraints might relate to cost, manufacturing capability and capacity, etc.
(g) *Post design evaluations (design reviews)* - This form of information is useful for documenting the goodness of a design, and can also be used to determine the validity and worth of the reasoning employed in the design process, for subsequent usage in the design of a new product.

Structuring Design Knowledge in DFD

The structuring and representation of the various types of design knowledge within Design Function Deployment, will be discussed under the five stages of the DFD process.

DFD Stage 1 - Establishment of requirements and specifications

In this stage, the applicable types of knowledge are : originating (initial) requirements, design specifications, design actions and decisions, design intent and rationale, constraints on the design process and product, product functional knowledge and product behavioural knowledge. The above forms of knowledge are captured in this stage within the stage 1 DFD chart. The rows in the chart represent the requirements which also contain constraints both on the product and on the design process, while the columns represent the design specifications and constraints. The design decisions made as well as the intent and rationale behind them are implicit within the chart. At this stage issues which relate to the functionality and behaviour of the product are included within the requirements and specifications in the chart, as well as in the form of how the specifications satisfy the requirements and the target values of the specifications.

DFD Stage 2 - Establishment of viable architectures

In stage 2, applicable types of knowledge include : design specifications, design actions and decisions, design alternatives, design intent and rationale, constraints on the design process and product, product functional knowledge, product physical knowledge, product behavioural knowledge and post design evaluations. Design specifications from stage 1 are now recorded in the rows of the stage 2. DFD stage 2 involves the development of alternative design concepts (called architectures). For each architecture separate DFD charts are developed, with the subsystems and subsystems characteristics of the architectures developed to satisfy the specifications and constraints. These subsystems are represented in the columns of the charts. Each of these charts also contain the information that relates to the functionality, behaviour and physical representation of each architecture that need to be satisfied in the design. For the architectures developed in this stage, necessary optimisation and evaluation can be performed against the design specifications and constraints, using the level 2 modules, in order to ascertain their viability and plausability. Depending on the complexity of the product being designed, some form of topological layout or 2D drawing and sketches might be developed at this stage to capture the physical knowledge of the proposed architectures. The designer/design team throughout the activities in this stage, document the various decisions as well as the intent and reasons behind them, noting when such design actions and decisions were made.

DFD Stage 3 - Development of viable layouts

In stage 3, applicable types of knowledge include : design specifications, design actions and decisions, design alternatives, design intent and rationale, constraints on the design process and product, product functional knowledge, product physical knowledge, product behavioural knowledge and post design evaluations. Stage 3 involves the development and expansion of the conceptual designs (architectures) from stage 2 into detailed designs called layouts. This results in a number of layouts for each conceptual solution. Here also, each layout represents alternative designs and are captured in the stage 3 DFD chart. Geometric and analysis models (e.g. Finite element analysis, kinematic/mechanism analysis) are also developed for the layouts. These models enable the capture of the physical and behavioural knowledge of each viable layout. These layouts are developed to satisfy the top level requirements, specifications and constraints arising from the earlier stages 1 and 2 of DFD. Necessary optimisation and evaluation are performed on the layouts to ascertain their viability and to choose the most satisfying one(s). The intent for the line of actions undertaken and the rationale for the choice of any particular layout are also recorded here.

DFD Stage 4 - Establishment of materials and associated manufacturing processes
In stage 4, applicable types of knowledge include : design actions and decisions, design alternatives, design intent and rationale, constraints on the design process and product, product functional knowledge, product physical knowledge, product behavioural knowledge and post design evaluations. The design actions and decisions made here relate primarily to the choice of materials and associated manufacturing processes for each part of a layout(s). In this case design alternatives relate to the utilisation of alternative materials and manufacturing processes to achieve the same function in the parts of a layout. Once the materials and manufacturing processes for each part of a layout are chosen, the DFD stage 4 chart is then completed for each layout. The choice of materials and manufacturing process, results in the complete definition of the physical knowledge (e.g. size, shape, material, weight, etc) of the product. It is also at this stage that the functional and behavioural properties of the layouts can be fully and realistically established. At this stage all the layouts under consideration are evaluated using the design tools in level 2 of DFD, against issues like manufacture, assembly, serviceability, reliability, etc, in order to establish the most viable layouts utilising the most promising materials and manufacturing processes. The design intent and rationale for the design actions and decisions taken at this stage are recorded by the designer/design team.

DFD Stage 5 - Generation of production plans
This stage involves primarily the generation of production plans for the alternative layouts for which materials and manufacturing processes have been established in stage 4. For each of the layouts alternative production plans based on the choice of materials and manufacturing process, can be developed and evaluated before choosing a particular one. The applicable design knowledge here include : design actions and decisions, design alternatives, design intent and rationale and post design evaluations. The other forms of design knowledge would already have been captured at the earlier stages. Design decisions made and rationales behind them will in this case be associated with the choice of a particular production plan for any alternative layout. Having established the production plan for a chosen layout, comprehensive post design evaluations can then be carried out taking into account the complete knowledge (both product and design process based) associated with the layout.

Capturing Design History

Design history is a record of the design process and provides (i) an organised history of the design decision-making process, (ii) an explicit representation of design objectives or solutions and their refinements, interactions and alternative decompositions, (iii) the ability to backtrack to explore different alternatives generated at previous decision points, (iv) the ability to determine all decisions that might be affected when changes are made and (v) the facility to replay portions of the recorded history [255].

The basic unit of design history is the design state, and the history is a time sequence of design states. At a higher level the design state is represented by the five stages of the DFD design model and at a lower level by the different activities performed at each of the DFD stages. In DFD, a single designer or a design team might be involved in the design process, and they might also be responsible for performing the design activities (tasks) either in series or in parallel. The record of design history within DFD, thus should accomodate the above scenario. In moving from one stage to another in DFD, many activities are performed while in moving from one activity to another within a stage, only a very limited number of activities might be performed. In the case of looking at each DFD stage as a design state, the sequence is linear and explicit while in the case of considering each activity within each stage as a state, the sequence is not as explicit and the sequence of each of these design activity states can involve both serial and concurrent sequences. In this case, the activities within each DFD stage has to be decomposed into those which are serial and those that are concurrent, before a proper time sequence can be established.

Irrespective of whether we are considering the DFD stages or activities within each stage as design states, the following information must be recorded for each state :

(1) The index for the state, (2) The preceding (previous) states, (3) the subsequent (next) states, (4) The record of activity or activities involved in the transformation from the previous state to the current and (5) Special states - the last state and the current state.

For each of these states, each applicable design knowledge type maintains its own history [255] over the course of the design process through the five stages of DFD. Necessary attributes to be maintained as functions of the design states with respect to the evolving product knowledge, include the following : **versions, status, parents, children and importance,** as well as *a record of the time these attributes were established* and *who was responsible for the activities performed* for the particular design state.

Intelligent Retrieval of Design Knowledge in DFD

The intent behind the capture of design knowledge is to be able to utilise it either during the design process or at a later date when developing a new design and in some other cases for design reviews and validation. Other reasons for retrieving captured design knowledge include [253] :

(a) to allow for criticism of the design by other persons directly or indirectly involved as well as those not actually involved in the dcesign process, (b) to introduce new members of the design team to the current status and history of the design, (c) to determine the effects of a design modification or change on the rest of the design, (d) to diagnose functional, behavioural and structural problems which might arise in the design, (e) to serve as a permanent reference for the design in the future, and (f) to assist in the design of other products which are similar in some respect.

The design knowledge that can be captured within DFD as discussed earlier are of various types and diverse in nature. Furthermore, some of the knowledge will be textual in nature while others will be geometric in nature. It is therefore of necessity that different types of indexing and retrieval methods would have to be employed. Such retrieval methods will include : (1) indexing mechanisms for handling requirements, specifications and constraints, (2) indexing mechanisms for handling and maintaining the decompositional links between design solutions and sub-solutions (e.g architectures and subsystems, layouts and parts) within an abstracted hierarchy, (3) indexing mechanisms for accessing materials, manufacturing process, parts, subsystems, layouts or architectures, etc) and (4) indexing mechanisms based on functionality and uses.

These indices will be implemented through either or a combination of direct, navigational, and query based retrieval methods. Direct index is useful for quick access but exact identifier must be known. Navigational index allows a user to browse through various levels of information in order to find what is desired. A query-based index provides a way to access information whose location may be unknown or spread across many areas of the design memory (database) [253].

5.12.4 Summary

The aim of this subsection is to explain a very useful capability of the Design Function Deployment (DFD) system, in relation to the capture of design knowledge (both product and design process based) generated throughout the design process within DFD. The rationales behind capturing design knowledge has been discussed as well as the various types of design knowledge that can be captured within DFD. Other discussions then focussed on how such knowledge is captured and represented within DFD, how the history of the design process can be recorded and possible ways of retrieving and utilising the captured design knowledge.

5.13 CONCURRENT MATERIALS SELECTION IN DESIGN FUNCTION DEPLOYMENT

5.13.1 Introduction

The aim of this section is to discuss the approach being adopted towards an integrated materials selection process within Design Function Deployment, DFD. It begins by examining the issues which are associated with the need to carry out materials selection in a rational and systematic manner within DFD. It then looks at the requirements for a computerised materials selection system, that will suit the materials selection objectives of DFD. The central theme of the section based on the concept of Materials and Manufacturing Process Function Deployment (MMPFD), is then fully expantiated along with preliminary discussions on the system architecture for the implementation of the proposed materials and manufacturing process selection system within DFD.

5.13.2 The Need for Rational Materials Selection

Traditionally, designers in selecting a material during the design process, tend to concentrate mainly on the materials properties that must be attained to satisfy the functional requirements of the product. In addition they also restrict themselves to 'comfort zone' materials, that is materials with which they are familiar. Experience has, however, shown that several other factors both quantitative and qualitative, impinge on the materials selection process. These include design constraints, manufacturing processes, form (shape and size) of the product,cost of raw materials, maintenance and surface finishes. Others are issues such as safety, durability, reliability, availability, socio-economic and political factors as well as assembly and disassembly. The need to select a material for a product can arise as a result of a number of factors Plevy [265]. These factors in addition to the issues which necessitate the need for rational materials selection, as well as the need for computerised materials databases, are discussed below.

(1) Materials selection is a key activity and one of the most important decisions made within the design process. (2) Proliferation of materials due to advancements in science and technology, indicates the need for a rational and systematic approach to materials selection. (3) Due to the sheer volume of materials type, the designer can no longer rely on manual methods of searching for materials. Hence the need for computerised materials databases. (4) To avoid restriction of materials selection to 'comfort zone' materials, i.e. materials that the designer is only familiar with. (5) To utilise new materials and processes to enable innovation in design. (6) To make materials information readily available to designers during the design process. (7) To achieve or to improve on a specified product performance. (8) To eliminate a material or service failure. (9) To accomodate a change in component function. (10) To solve processing difficulties and/or take advantage of new processing techniques. (11) To reduce material and/or production costs and rationalise on materials stockholding. (12) To anticipate or to exploit a change in the availability of a material. (13) To take advantage of the introduction of a new product, or adjust to a decline in the market. (14) To accomodate a change in design, fashion or legislation. (15) To cope with new and/or adverse environmental conditions.

5.13.3 Requirements for a Computerised Materials Selection System within DFD

The prime objective of this work, is to develop a materials database system that will suit the terminology and taxonomy of the design specification process within Design Function Deployment. The necessary requirements for such a system are discussed below [266].

(1) The system should be developed to integrate with the overarching Design Function Deployment system.
(2) The system should be developed to interact dynamically with the Design Function Deployment system in a concurrent manner at any of the stages of the design process within it.
(3) The system should aid designers in their problem solving and decision making process, when selecting materials during the design activity.
(4) The system should be easy to learn and easy to use.
(5) The system should be detailed enough to provide all possible materials information required by designers at different levels of needs.
(6) The system should be able to provide feedback and performance information from previous designs in relation to field experience in use and maintenance.
(7) The system should be capable of providing information on alternative materials, their available and manufacturable forms (shapes and sizes) and suitable manufacturing processes.
(8) The system should provide relevant materials information for product life cycle (Design, Manufacture and Use) as well as for the design life cycle (Concept, Embodiment and Detailed design stages).
(9) The system should be developed taking into consideration current standards related to data quality, terminology and representation as well as traceability of materials information to relevant sources.
(10) The system should be easily extendable and maintained.
(11) The system should be developed in the light of concurrent engineering, so as to be integrated with other design and analysis software, to enable the import and export of data.

5.13.4 Materials and Manufacturing Process Function Deployment [267]

Materials information needs of designers do vary during the design life cycle. At the conceptual stage, the designer is interested in information for the whole range of materials classes, from which to screen and select candidate materials or materials classes. The quality of materials properties data is not a priority at this stage. At the embodiment stage, materials selection is reduced to the examination of more detailed materials property data for potential materials selected during the conceptual stage. The materials information needs at the final, i.e. detailed design stage, is reduced to a much narrower number of candidate materials. Here the designer needs very detailed amount of accurate, reliable and traceable materials information and data, for the potential or selected material. The above considerations are shown schematically in Figure 5.27. For any of these stages of design, it is necessary that the materials and manufacturing process requirements are well established and defined. This section discusses the concept of Materials and Manufacturing Process Function Deployment (MMPFD) within DFD.

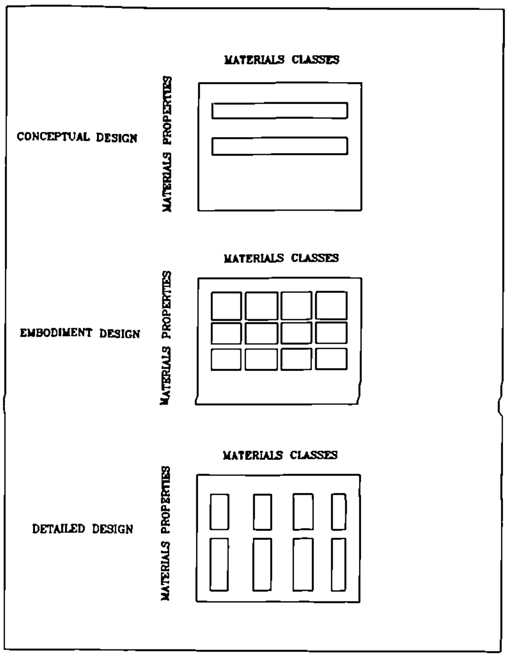


Figure 5.27 Materials Needs Versus Design Life Cycle

Materials and Manufacturing Process Function Deployment (MMPFD) can be defined as the concurrent process of materials and manufacturing process selection, that suits the terminology and taxonomy of the design specification process within the stages of Design Function Deployment, involving the extraction of materials and manufacturing process related design functions from stages 1, 2 and 3 of DFD, analysing, refining and categorising them into 'materials functions' and 'manufacturing process' respectively, and then translating them into quantitative and qualitative materials properties and manufacturing process characteristics. The process uses similar charts to those of DFD for the translation process.

The MMPFD process proceeds through five stages as shown in Figure 5.28. These stages are mutually dependent, and movement between them proceeds in both directions as shown by the feedback loop, indicating that the materials functions and the resulting materials property profiles interact and influence each other. The implication of this is that, if for instance the materials properties do not meet the materials functions, it is necessary to either improve the material performance or to change the requirements on the product [268].

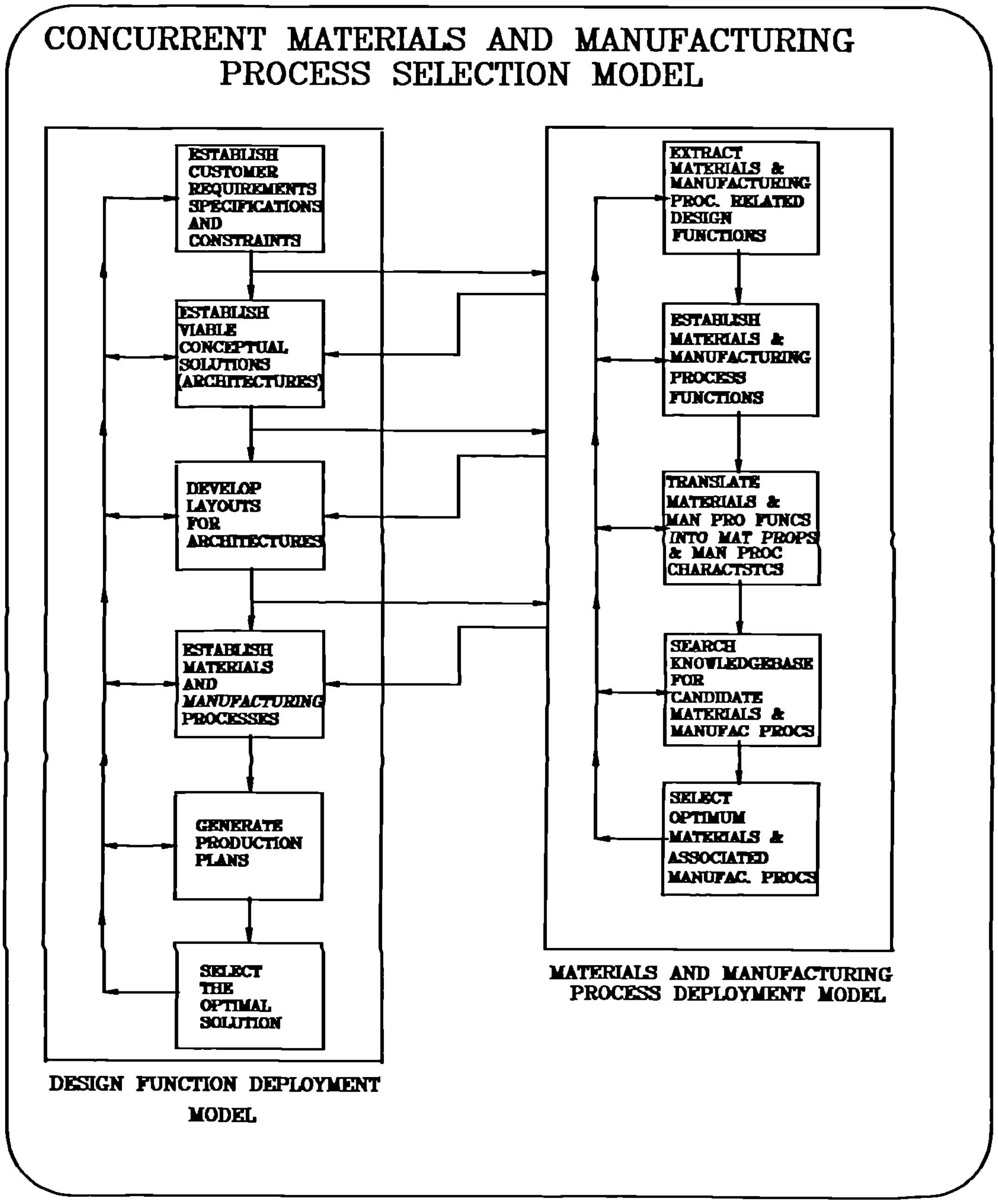


Figure 5.28 Concurrent Materials and Manufacturing Process Selection Model

DFD Stage 1

The first stage involves the identification and extraction of the materials and manufacturing process related design functions generated within the DFD process, along with their corresponding target values. These design functions may in some cases be fairly explicit, while in other cases, may be vague and imprecise. They will usually be related to functional, manufacturing process, cost, reliability requirements, etc, as well as impinge on the product being designed.

DFD Stage 2

In stage 2, these materials and manufacturing process related design functions, are then analysed, refined and categorised into primary, secondary and tertiary 'materials and manufacturing process functions'. The primary level represents the secondary design functions from stages 1, 2 and 3 of DFD, that is, the secondary level of specifications and constraints (DFD stage 1), subsystems of an architecture (DFD stage 2) and parts of a layout (DFD stage 3). The secondary materials and manufacturing process functions represent the tertiary design functions from stages 1, 2 and 3 of DFD, that is, specifications and constraints (DFD stage 1), subsystems characteristics (DFD stage 2), and parts characteristics (DFD stage 3). The tertiary materials and manufacturing process functions then represent the refined materials and manufacturing process related design functions, at a more detailed and precise level and with more realistic target values.

DFD Stage 3

In stage 3, the materials and manufacturing process functions are translated into quantitative, actionable and measurable specific materials properties and manufacturing process characteristics, along with their target values. These specific materials properties and manufacturing process characteristics, are then also categorised into primary, secondary and tertiary levels. The nature and composition of this categorisation, will depend on the particular stage of DFD, at which materials selection is being performed. When the selection is being performed between stages 1 and 2 of DFD, the primary level is associated with the specifications and constraints derived from stage 1 of DFD, the secondary level is represented by the generic materials properties and manufacturing processes, while the tertiary level represents the specific materials properties and manufacturing process characteristics. In the case of the selection process occurring between DFD stages 2 and 3, the primary level is associated with the particular subsytems of an architecture, while the secondary and tertiary levels represent the generic and specific materials properties and manufacturing process characteristics respectively. When the selection process is performed between stages 3 and 4 of DFD, the primary level is associated with the particular parts of a layout, while the secondary and tertiary levels represent generic and specific materials properties and manufacturing process characteristics respectively, as in the case above.

The last activity in this stage, is the determination of the importance ratings of the materials properties and manufacturing process characteristics, using the relationship values between the materials and manufacturing process functions and materials properties and manufacturing process characteristics and the importance ratings of the materials and manufacturing process functions, which are recorded in the Materials and Manufacturing Process Function Deployment chart. Other ranking methods such as the digital logic method [269] and the reciprocal pairwise comparison matrix method [270] can also be explored in the ranking process.

DFD Stage 4

In stage 4, the materials properties and manufacturing process characteristics, along with their importance ratings are then used to constitute the selection criteria profile. This profile is then used to search through the materials and manufacturing process knowledge/rulebase via a user friendly interface, to select candidate materials and manufacturing processes. The knowledge base accesses materials and manufacturing process information which are stored in the materials and manufacturing process database.

DFD Stage 5

In this final stage, the key properties and parameters in the selection criteria profile, are employed in a multi-criteria fashion to weight and rank the selected viable materials and manufacturing processes and then subsequently selecting the optimal or most suitable material and corresponding manufacturing process. This activity will be performed using the multi-criteria optimisation module based on many alternative selection algorithms.

5.13.5 Concurrent Materials and Manufacturing Process Selection Within Design Function Deployment

The Materials and Manufacturing Process Function Deployment process is performed concurrently with the progression through the DFD design stages, as shown in Figure 5.28. At each of the DFD stages, design functions are generated, a number of which relate to materials properties and manufacturing process characteristics. These materials and manufacturing process related design functions, may be explicit and in some cases vague, depending on the nature and complexity of the product being designed. The concurrent materials and manufacturing process selection process occurs 1/2 a step behind the DFD process, that is, it occurs between stages 1 and 2, 2 and 3, and 3 and 4, as shown in Figure 5.29. In the first selection process, that is, between stages 1 and 2, the materials and manufacturing process related design functions extracted are associated with specifications and constraints generated in stage 1 of DFD. At this stage, since no solution is being referred to, the subsequent materials properties and manufacturing process characteristics are likely to be few and also vague. The selection process is hence a screening process to eliminate non-viable materials classes and possibly manufacturing processes. Here, the whole spectrum of possible materials and manufacturing processes are explored. Between stages 2 and 3, the extracted materials and manufacturing process related design functions are associated with subsystems characteristics of the viable conceptual solutions (architectures) developed in stage 2 of DFD. In this case, there is a better insight into the expected product and hence a better understanding of the desired materials properties and manufacturing process characteristics. A reduced number of materials classes and corresponding manufacturing processes, are now explored using more materials properties and manufacturing process characteristics as selection criteria. In the detailed design stage between stages 3 and 4 of DFD, the extracted materials and manufacturing process related design functions are associated with the parts characteristics of the viable layouts established in stage 3 of DFD. The focus in this stage is on the materials and manufacturing process functions of the individual parts of a viable layout.

For each of these selection activities, the stages described in the materials and manufacturing process deployment model, as shown in Figure 5.28, are proceeded through. In the case of the first two selection processes between DFD stages 1 and 2 and between stages 2 and 3, only the first four stages of the model may be undergone, as these stages involve a screening process, and the aim is not necessarily to choose an optimum material. In the selection process between DFD stages 3 and 4 however, the whole process through the five stages of the model, are proceeded through. Here the detailed design of the layouts and corresponding parts have been done, and the need is for the selection of the most suitable materials and manufacturing processes for each of the parts of a layout.

5.13.6 System Architecture for The Materials and Manufacturing Process Selection System

The computerised materials and manufacturing process selection system, will be developed to be driven principally by the Function, Manufacture and Use requirements of a product under development. The system is planned to be an intelligent supporting tool, which uses flexible and comprehensive "design rules" to aid optimum selection of materials as well as manufacturing methods in the design process. The overall system will consist of the following modules : (1) User Interface Module, (2) Materials and Manufacturing Process Knowledge/rule base Module, (3) Materials and Manufacturing Process Database Module and (4) Multi-Criteria Optimiser Module. The system architecture integrating these modules with the whole concurrent materials selection process is shown in Figure 5.29.

<u>User Interface</u>

This is the interacting link between the designer/user and the materials selection system. It will be developed to be user- friendly, using a graphical user interface builder. It provides the access to the materials function deployment process of translating materials functions into specific materials properties and the knowledge/rulebase, multicriteria optimiser and materials database modules.

<u>Materials and Manufacturing Process Knowledge/rule base</u>

The knowledge base module which will be a rule based system, will contain design rules associated with the factors which influence or have a bearing on materials and manufacturing process selection. The rules will consist of both declarative (facts and relations) and procedural (functions, condition action If... then) rules. They will be used in conjunction with the selection criteria setup, in the selection of suitable materials and manufacturing process.

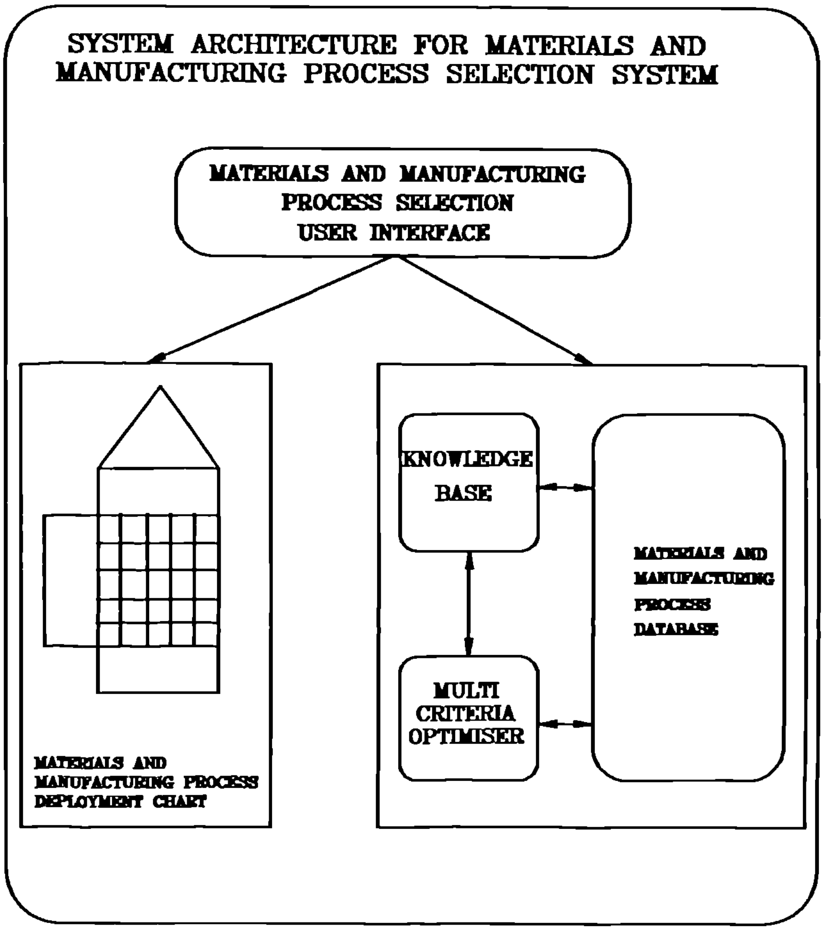


Figure 5.29 System Architecture For Materials and Manufacturing Process Selection System

Two strategies will be adopted in developing the design rules. The first strategy will take into account design rules which relate to all materials classes, which would be used in selecting materials across the whole range of materials irrespective of their classes. The second strategy will on the other hand take account of design rules which apply only to a particular class of materials and are hence used in the selection of a material within that class. This second strategy will be applicable in the situations where the designer has already decided on the class of potential materials. For each of the two strategies, two levels of design rules will be developed.

The first level will contain rules that are used to screen and select potential materials classes or materials, while the second level will consist of design rules that would take account of the various interacting factors affecting the selection process as well as being used in performing trade offs and resolving conflicts. A suitable knowledge base/expert system shell will be employed for the above development.

An analysis of the materials selection problem shows that three main factors, that is, Materials (type and properties), Manufacturing process and Form (shape and size) interact with each other in seeking to achieve the desired function of a component or product. These interactions, depending on whether one or two of the factors are fixed, unimportant, or varying, can be considered under the following cases [266]:

	Fixed or Unimportant	Variable
1.	Form	Materials, Manufacturing
2.	Manufacturing	Materials, Form
3.	Materials	Form, Manufacturing
4.	Materials, Form	Manufacturing
5.	Materials, Manufacturing	Form
6.	Form, Manufacturing	Materials
7.	____________	Materials, Form, Manufacturing

The above interactions can be such that the selection process will involve not only the choice of a suitable material, but also involve finding a suitable manufacturing route or choosing a shape that can be manufactured using the selected material, in order to achieve the functional requirements of the product. Harmony therefore has to be sought between these factors in systematically selecting a material, or deciding on the final product. It is planned that extensive work will be done to explore these interactions with the possibility of including the findings in the knowledge base. There would be a need therefore to elicit information in this regard from the design industry.

Materials and Manufacturing Process Database

The database module will store information about the materials classes and materials properties (mechanical, electrical, thermal, physical, environmental, etc). It will also include information on cost of materials, available forms (shapes and sizes) of materials, manufacturing processes, current applications and on manufacturers and suppliers. It is envisioned that an object-oriented database management system will be employed for the development of this module. The database module will be developed taking into account how to interface it intelligently with the knowledge base module.

Multi-Criteria Optimiser

The multi-criteria optimiser module will contain a number of multi-atrribute optimisation methods employing various optimising and selection algorithms. On completing the screening and selection of candidate materials in stage 4 of the model (Figure 4), this module will then be used to optimise and select an appropriate material from the list of candidate materials, using suitable selection algorithms. The designer will also have the option to use engineering judgement in choosing the desired material.

This module will be developed using the C/C+ + programming languages, which have been chosen for the Centre's work.

5.13.7 Summary

The emphasis on the importance of integrating the materials an manufacturing process selection within the design process in Design Function Deployment, has been the prime objective of this section. A rational and comprehensive approach to materials and manufacturing process selection within Design Function Deployment, DFD, has hence been enunciated. This was guided by the need to develop a computerised materials and manufacturing process selection system that suits the terminology and taxonomy of the design specification process within DFD. The section began by examining the need for rational materials selection in engineering design as well as the need for computerised materials database systems. The requirements of a comprehensive computerised materials and manufacturing process selection system that can be integrated within DFD, was then discussed in detail.

The new concept of Materials and Manufacturing Process Function Deployment (MMPFD) was then discussed showing its model and how it integrates with DFD in a concurrent manner. The section concludes with discussions on the system architecture for the implementation of the concurrent materials and manufacturing process selection system within Design Function Deployment, DFD.

5.14 DESIGN OPTIMISATION WITHIN *DESIGN FUNCTION DEPLOYMENT*

5.14.1 Introduction

Optimization in general is the selection of the best course of action from amongst available alternatives. In practice, designers tend to focus on a single design solution which is then refined continuously until a satisfactory design is arrived at. But for the design to be 'right first time' and for the optimization process to be effective, the designer should propose as many design solutions as possible before selecting any one of them. This generation of the 'solution of space' is one of the key features of Design Function Deployment, DFD.

A majority of the traditional design optimization activities are usually based on single objective measures such as cost, strength, shape, weight or ratios of these which do not truly represent the conditions that products are subjected to. Other criteria, against which product performances are usually evaluated include robustness, fitness for purpose, reliability, ease of manufacture and assembly, safety, environmental compatibility, maintenance, market appeal, life cycle costs and so on. Therefore, for 'Design Optimization' to be rational, it should be performed with respect to 'Multiple Design Criteria' which includes objectives, attributes and constraints, some of which may be conflicting [271].

This section aims to elucidate the issues related to multi-criteria design optimization in engineering design and how this fits into Design Function Deployment. A review of useful multi-criteria optimization methods is also given to facilitate the understanding of the rationale of the design optimization process within Design Function Deployment. The section then concludes by enunciating the approach to multi-criteria design optimisation within DFD.

5.14.2 Review of Multi-Criteria Optimization Methods

Multi-Criteria Decision Making (MCDM) refers to making decisions in the presence of multiple, usually conflicting criteria. Various researchers [272-276] have reviwed and classified the various multi-criteria optimization methods. The solutions to MCDM problems are either to design the best alternative or to select the best one among the previously specified finite alternatives. MCDM process involves designing/searching for an alternative that is most attractive in accordance over all criteria. Multi-criteria optimization problems in engineering design generally come under two classes viz: (a) multi-attribute design optimization and (b) multi-objective design optimization. A brief description of these are given below and the reader is referred to the cited texts for detailed descriptions.

Multi-Attribute Design Optimization

Multi-attribute design optimization is the process of selecting the preferred solution in a situation with a predetermined finite number of alternatives. Such alternatives have associated with them, a level of achievement of both quantitative and qualitative attributes. The final selection of an alternative is done by both inter- and intra-attribute comparisons which may involve either explicit or implicit tradeoffs. Several researchers have reported on multi-attribute optimization [277, 278] and Hwang &Yoon [273] have produced a classification of major Multi Attribute Decision Making (MADM) methods based on (a) the type of information to be supplied by the decision maker and (b) the salient features of the information, as shown in Fig. 5.30.

Multi attribute decision making methods are procedures that specify how attribute information is to be processed in order to arrive at a choice. There are two major categories of models associated with them, namely (a) non-compensatory models and (b) compensatory models. Non-compensatory models do not permit tradeoffs between attributes, that is, a disadvantage in one attribute cannot be offset by an advantage in some other attribute. Comparisons are made on an attribute by attribute basis. Methods associated with this model include Dominance, Maximin, Maximax, Conjunctive, Disjunctive and Lexicographic methods.

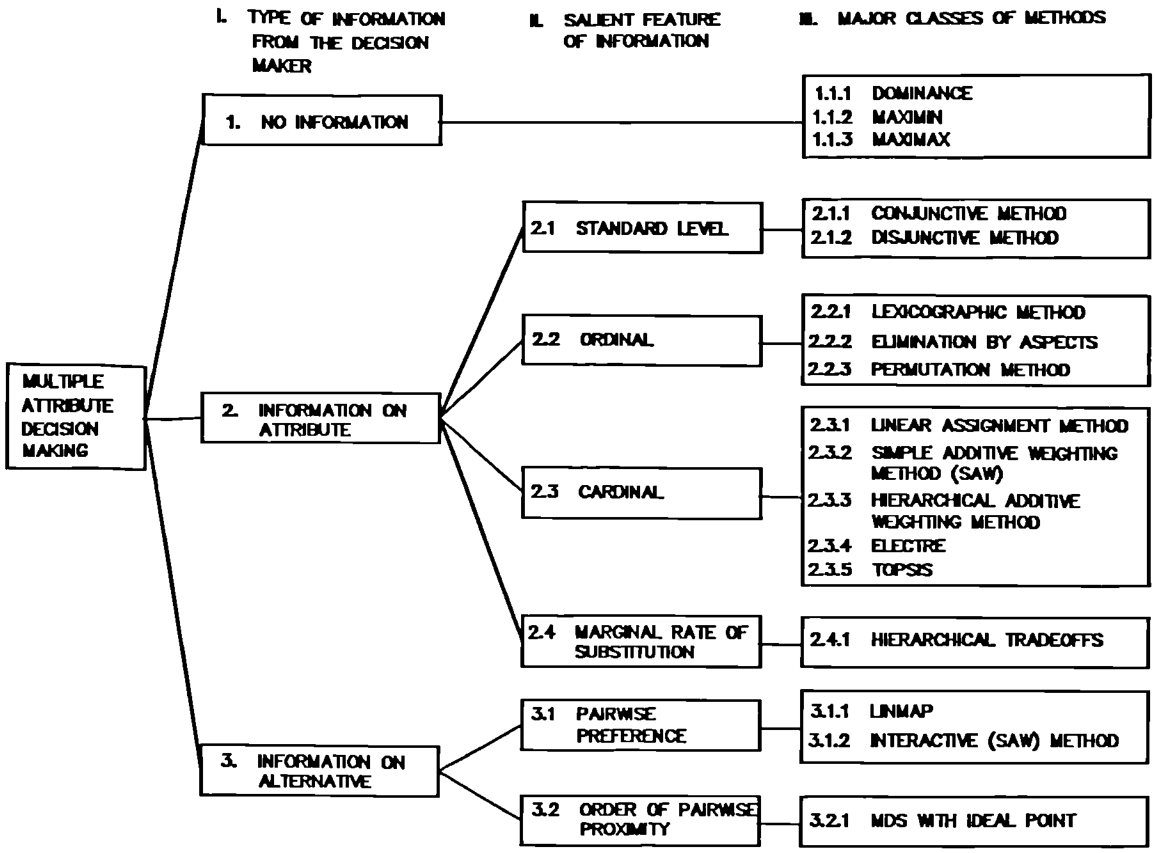


Figure 5.30 Taxonomy of Multi-Attribute Decision Making Methods

Compensatory models on the other hand permit tradeoffs between attributes, that is, a change in one attribute can be offset by opposing changes in any other attribute. A single number is usually assigned to each multi-dimensional characterisation representing an alternative. Compensatory models fall into three categories (a) scoring model (b) compromising model and (c) concordance model. In the scoring model, an alternative is selected which has the highest score or the maximum utility and hence the problem is how to assess the appropriate multi-attribute utility function for the relevant decision situation. Simple additive weighting, hierarchical additive weighting and interactive simple additive weighting methods belong to this model. Alternately, the Compromising model selects an alternative which is closest to the ideal solution. Technique for order of preference by similarity to ideal solution (TOPSIS), linear programming techniques for multidimensional analysis of performance (LINMAP), and nonmetric multidimensional scaling with ideal point (MDS) belong to this model. Finally the concordance model arranges a set of preference rankings which best satisfies a given concordance measure. Permutation, linear assignment and elimination of choice translating reality (ELECTRE) methods belong to this model.

Multi-Attribute Decision Making Methods and Applications

Following the classification shown in Fig. 5.30, these methods can be analysed in three groups namely (a) methods for which no preference information is given (b) methods for which information on attributes is given and (c) methods for which information on alternatives is given. The methods associated with each class are discussed below.

Dominance does not require any assumptions or transformations of attributes. The first two alternatives are compared and if one dominates the other the dominated one is discarded. The undiscarded one is then compared with the third alternative and any dominated alternative is discarded. This is continued until the non-dominated set is obtained. The dominance method is hence used in initial filtering. *Maximin* is used only when attributes have a higher degree of comparability and uses a specialized degenerate weighting. A weight of '1' is given to the worst attribute and all others are given '0' weights. The worst attribute values for all alternatives are found and the alternative with the least is selected. *Maximax* is similar to maximin except that instead of the worst attribute the best attribute is given the weight of '1'. The best attribute values for all alternatives are found and the alternative with the highest is selected. In *conjunctive* method minimum attribute values or the cut-off values acceptable for each of the attributes must be given. The method is used not to select the the best alternative but to dichotomize them into acceptable/non-acceptable categories. In the *disjunctive* method in contrast an alternative is evaluated on its greatest value of an attribute. The conjunctive and disjunctive methods described in this paragraph are grouped as ' methods for standard level of attributes'.

The *lexicographic method* requires the attributes to be ranked in the order of importance. Once the most important attribute is selected, the alternative having the highest on this attribute is chosen. If multiple alternatives have the highest value on the specified attribute, then the attribute ranked second is compared across all alternatives. The process continues until a single alternative emerges or until all attributes have been examined. In *elimination by aspects method* an attribute is selected and all alternatives not passing the cut-off on that attribute are eliminated. Another attribute is then selected and the process continues until all alternatives but one are eliminated. In short the alternatives that do not satisfy some standard level are eliminated and attributes are ordered in terms of their discriminating power in a probabilistic mode rather than in terms of their importance.The *permutation method* consists of testing each possible ranking of the alternatives against all others. With m alternatives $m!$ permutation rankings are available and the method identifies the best ordering of the alternative rankings, and then the dominating alternative.

The three methods described in this paragraph are called the 'methods for ordinal preference of attributes'.

Linear assignment method gives an overall preference ranking of the alternatives based on a set of attributewise ranking and a set of attribute weights. It features a linear compensating process for attribute interaction and combination. In the *simple additive weighting method* weights are assigned to each attribute and the marginal worth assesments within attributes are reflected by making numerical scaling of intra-attribute values. A total score for each alternative is then obtained by multiplying the scale rating for each attribute value by the importance weight assigned to the attribute and then summing these products over all attributes for each alternative. The alternative with the highest score is then chosen. In *hierarchical additive weighting method* values or preferences are assigned to higher level objectives and the instrumentality of each attribute attaining these higher level objectives are assessed. The inter attribute weights are hence inferred from the direct assessment of the higher level objectives. The *ELECTRE* method uses the concept of an outranking relationship and consists of a pairwise comparison of alternatives. The comparison is based on the degree to which evaluations of the alternatives and the preference weights confirm or contradict the pairwise dominance relationships between alternatives. It examines both the degree to which the preference weights are in agreement with pairwise dominance relationship and the degree to which weighted evaluations differ from each other. The *TOPSIS* method is based upon the concept that the chosen alternative should have the shortest distance from the ideal solution and the farthest from the negative ideal solution. Distances to both the ideal and the negative ideal solutions are considered simultaneously by taking the relative closeness to the ideal solution. The method assumes that each attribute takes either monotonically increasing or decreasing utility. The five methods described in this paragraph are grouped as 'methods for cardinal preference of attributes'.

In *hierarchical tradeoff method* tradeoff information is explicitly utilized. Marginal Rate of Substitution (MRS) and Indifference Curves are the two terms used to describe tradeoffs. MRS involves making tradeoffs between two attributes at a time and this process depends on the levels of the two attributes. Indifference curves represent the locus of all attribute values indifferent to a reference point. This method falls into the group 'methods for marginal rate of substitution of attributes'.

LINMAP method was developed for accessing weights as well as locating the ideal solution. Here m alternatives composed of n attributes are are represented as m points in the n dimensional space. An ideal point denoting the most prefered is assumed to exist. Once the location of the ideal solution is decided, an alternative with the shortest distance from the ideal solution is chosen. *Interactive simple additive weighting method* is used to rank alternatives subject to an (initially) unspecified linear utility function. These two methods are grouped as 'methods of pairwise preference'.

In *Multidimensional scaling with ideal point (MDS)* ordering of the proximities of pairs of alternatives are used to construct a multidimensional spatial representation. Alternatives are represented by points in the space. The points that are close together are assumed to be close together in terms of preference. An ideal alternative can then be located in this space and the distance from the ideal point is measured to rank the alternatives in terms of preference. This method falls into the group 'methods for pairwise proximity'.

Multi-Objective Design Optimization

Multi-objective design optimization is the process of finding the best solution within a feasible range of the design problem, subject to a set of measurable objectives, a set of well defined constraints and tradeoff information between the objectives. Haimes et. al. [275] highlight three approaches to the solution of multi-objective optimization problems, which represent different schools of thought. They are (i) finding the preferred solution directly (ii) first generating the non-inferior set and then finding the preferred solution from amongst these and (iii) developing the non-inferior solutions and then allowing the decision maker to choose which of the solutions to implement. Hwang and Masud [274] have also classified the methods under this class as shown in Figure 5.31.

Multi-Objective Decision Making Methods and Applications

The classification of the methods in Figure 5.31 can be discussed under four groups namely (a) methods for no articulation of preference information (b) methods for a priori articulation of preference information (c) methods for progressive articulation of preference information and (d) methods for a posteriori articulation of preference information.

The *global criterion method* follows three steps (i) obtain the ideal solution (ii) construct a payoff table and (iii) obtain the preferred solution. This falls into the group where no articulation of preference information is needed. *Utility function methods* require the knowledge of the utility function of the multiple objectives before solving the vector maximum problem. *Bounded objective methods* require that at least the minimum acceptable level of achievement for each objective function be given. The utility function methods and the bounded objective methods are called the methods for cardinal information. *Lexicographic method* requires that objectives be ranked in order of importance and the preferred solution obtained by this method is one which maximizes the objectives starting with the most important and proceeding according to the order of importance of the objectives. *Goal programming method* requires that goals be set for each objective to be attained. A preferred solution is then defined as the one which minimizes the deviations from the set goals. There is also a requirement for the ordinal ranking of the objectives. *Goal attainment method* is a variant of goal programming and requires that both a goal vector and a vector of weights relating to the relative under- and over-attainment of the desired goals be defined. The lexicographic, goal programming and goal attainment methods are called the methods for mixed ordinal and cardinal information. The five methods described in this paragraph form the group 'methods for a priori articulation preference information'.

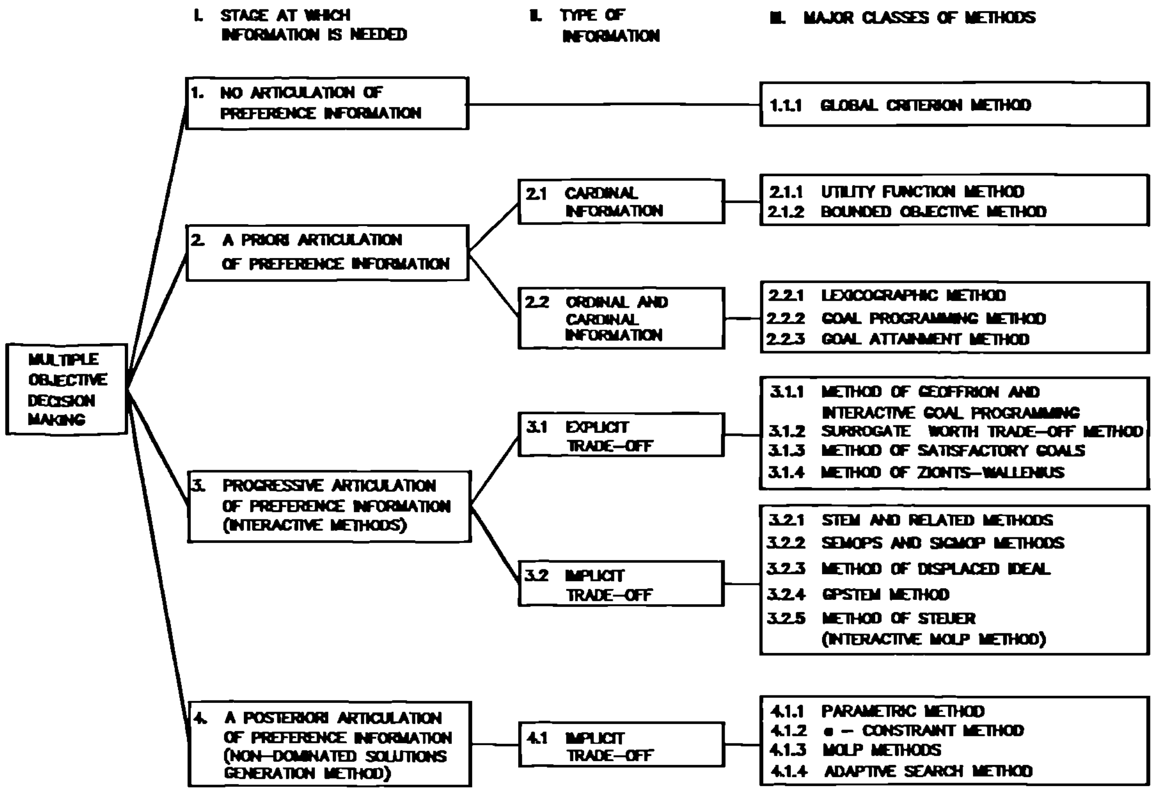


Figure 5.31 Taxonomy of Multi-Objective Decision Making Methods

Geoffrion and interactive goal programming method demonstrates that a large step gradient algorithm can be used for solving the vector maximum problem, if an overall utility function defined on the values of the objectives can be specified. This procedure is based on the Frank-Wolfe algorithm. *The surrogate worth tradeoff method* consists of two phases (i) identification and generation of non-dominated solutions which form the tradeoff functions in the objective surface and the search for a preferred solution in the non-dominated solutions. The preferred solution is obtained by interactively assessing the indifference band and by using the surrogate worth functions. The *method of satisfactory goals* uses the "bounded objective method" interactively to determine a satisfactory solution and it involves specifying a set of acceptable initial goal levels and then identifying objective functions whose goal levels are the least satisfactory. The least satisfactory objectives are iteratively and interactively removed until the most favourable solution is reached. The *method of Zionts-Wallenius* assumes that all the objective functions are concave (to be maximized) and the constraints form a convex set, while non-linear functions are linearized. It involves choosing an arbitary set of positive multipliers and generating a composite objective function using these multipliers. The composite objective function is then optimized to produce a non-dominated solution to the problem. The four methods described so far in this paragraph forms the sub-group 'methods for explicit tradeoff information'.

STEM and related methods are for solutions of multi-objective linear programming problems. The method allows the decision maker to learn to recognize good solutions and the relative importance of the objectives. In this method phases of computation alternate with phases of decision. *The Sequential MultiObjective Problem Solving technique (SEMOPS)* is an interactive programming technique that dynamically involves the decision maker in a search process to locate a satisfactory course of action. The method cyclically uses a surrogate objective function based on goals and the decision maker's aspirations towards achieving the objectives. *The Sequential Information Generator for Multi Objective Problems (SIGMOP)* embeds a nonlinear goal programming approach within the principal problem and the set of auxiliary problems of the optimization phase to replace the surrogate objective functions in SEMOPS. *GPSTEM method* is a link between goal programming and STEM. The method explores the non-dominated solutions that give the objective functional values close to goal levels imposed by the decision maker. *Interactive multi objective linear programming method by Steuer* presents to the decision maker $2k+1$ non dominated extreme points at each iteration where k is the number of objectives and the decision maker has to indicate the most preferred solution from this set. Once this solution is identified, the non-dominated extreme points in the neighbourhood are explored and a new set of $2k+1$ non-dominated solutions are identified and presented to the user. The process continues until a satisfactory solution is found. The nine methods presented in this paragraph fall into the group 'Methods for progressive articulation of preference information'.

Another group in the multi objective decision making methods is 'methods for a posteriori articulation of preference information'. These methods determine a subset of the complete set of non-dominated solutions to the vector maximum problem from which the decision maker chooses the most satisfactory solution while making implicit tradeoffs between objectives based upon some previously unindicated or non-quantifiable criteria. These methods usually generate a large number of non-dominated solutions and are generally incorporated into some of the interactive methods. Some methods under this group are the parametric, ε constraint, multi objective linear programming and adaptive methods.

The above review has involved the identification and discussion of seventeen multi attribute decision making methods and nineteen multi objective decision making methods. Their suitability and method of implementation and incorporation into the Design Function Deployment System, are discussed in the next sub-section.

5.14.3 Design Optimization Within Design Function Deployment

The design process within DFD starts with the establishment of requirements. These requirements are then deployed or brought to life by translating them into meaningful specifications which in the DFD terminology are called 'stage 1 design functions'. These stage 1 design functions are deployed through the subsystem characteristics of the different architectures or conceptual solutions proposed. These subsystem characteristics are the stage 2 design functions. At this level the subsystems can have different configurations. The combinations of these different configurations of subsystems are called layouts. This development of layouts or detailed designs is the activity in stage 3, which includes the definition of the parts characteristics of each layout. The entire collection of the viable layouts hence form the solution space from which the optimal solution has to be selected. In stage 4 each part belonging to every layout has to be identified with a suitable material, manufacturing process and the appropriate form or geometry. In stage 5, production plans which include processes, quantity economies, process parameters, quality control plans and others have to be proposed for each part in each layout. Optimization should be performed at several places during this design process within DFD. It has to handle situations with multiple attributes, objectives, uncertainty, conflicting requirements, deadlocks and tradeoffs.

In stage 2 when more than one architecture is generated the most promising one has to be selected. The optimization process here is a multi-attribute one, representing a selection or choice problem. The specifications from stage 1 along with their target information and relative importance weights, constitute the criteria (attributes) to be used in selecting the most promising architecture. In addition the information from the interaction matrix of the specifications also provides the basis for performing tradeoffs between the specifications in the optimization process. The most satisficing architecture is then further developed into alternative layouts and constituent parts. If there is more than one satisficing architecture they can be developed in parallel in the further stages of DFD depending on time constraints and available resources. In stage 3 of DFD where alternative layouts have been generated for promising architectures, the same multi-attribute optimization problem arises. There is then also a need to select the most viable layout amongst the alternatives in an architecture, and then selecting the optimal or most satisficing layout amongst the most viable layouts from all the architectures.

Choosing the optimal layout will however depend on the later stages of DFD, that is, stages 4 and 5, which involve the determination of the most appropriate materials, manufacturing processes and production plans for a layout. These later stages in DFD, that is, 3, 4 and 5, also involve the detailed design of the layouts and their constituent parts. The optimization problem in stage 3 hence involves both multi-attribute and multi-objective types; that in stage 4, principally involves selecting the best materials and associated manufacturing processes for the constituent parts of all the alternative layouts. In stage 5, the optimization problem is that of determining the best production plan, based on lowest cost, ease of manufacture and assembly objectives. In addition, DFD contains some auxiliary design methods such as Design for Manufacture, Assembly, Cost, Environment, Safety and Reliability as well as Failure Mode and Effect Analysis. These modules, wherever appropriate in stages 3, 4 or 5, will be used to evaluate each of the alternative layouts, which are then given merit indices, indicating their performance ratings with respect to the desired attributes of these design methods. These merit indices, will also form part of the criteria for eventual selection of a layout.

The optimization model within DFD, hence consists of both a top down (stages 1 --> 2 --> 3) and a bottom up (stages 5 --> 4--> 3) approach. This approach enables the detail exploration of the design solution space, before deciding on any particular solution. The multi-objective optimization process in stage 3 involves the principal objective of maximising the utility of a particular layout within a feasible boundary, subject to a set of measurable objectives, a set of well defined constraints and trade-off information between the objectives. This information is obtained from the DFD charts constructed for each layout. The measurable objectives and well defined constraints represent the most critical subsystems characteristics (including their target value information) from the parent architecture. These also have associated with them, relative importance ratings (weights) which have been derived from the translation process of the design specifications (from stage 1) into subsystems characteristics. The trade-off information arise from the interaction matrix of the subsystems characteristics. Using the above information , the designer can hence formulate the necessary optimisation problem including the mathematical model, and then choose the most appropriate multi-objective optimisation method to be used. This process is performed for each layout, after which the multi-attribute optimisation process is carried out to select the optimal or most satisficing layout. Performing the multi-objective optimisation before the multi-attribute optimization in this third stage, also ensures that the attributes used to select between the layouts are more definite, realistic and quantitative. It is pertinent that the optimization activities in stages 4 and 5 be resolved and completed before a rational and comprehensive optimization can be done in stage 3.

In the light of these considerations, the authors are convinced that the following multi-criteria decision making methods, form the essential minimum in the optimization process within Design Function Deployment. They are : (i) Simple additive weighting method, ELECTRE, TOPSIS, LINMAP, Goal Programming, Goal attainment method, Method of Geoffrion and interactive goal programming, The Surrogate worth trade-off method, Method of Zionts-Wallenius, STEM and related methods, SEMOPS and SIGMOP methods.

5.14.4 Summary

A review of multi-criteria design optimization problems in engineering design has been carried out. The scenarios for both multi-attribute and multi-objective design optimization as well as the proposed philosophy and implementation of the design optimization process in DFD, has been discussed. It has been highlighted that at the early stages of the design process in DFD, multi-attribute and fuzzy optimization methods will be more applicable, while multi-objective optimization methods will be more useful at the later stages of the design process.

5.15 DESIGNING WITH DESIGN FUNCTION DEPLOYMENT [279, 280]

The process of design in DFD involves proceeding through the five stages of the design model. The activities in each of these stages are : (1) Stage 1 - Requirements analysis and determination of solution neutral specifications and constraints, (2) Stage 2 - Establishment of viable architectures, (3) Stage 3 - Establishment of viable layouts, (4) Stage 4 - Selection of materials and associated manufacturing processes and (5) Stage 5 - Development of production plans.

5.15.1 DFD Stage 1 - Requirements Analysis and Establishment of Specifications

In this stage, the first step is to establish who the possible customers are. There can be several customers associated with a product, with varying levels of involvement. Such customers can include : (i) the customers who will use the product, (ii) the proprietor who owns the product, (iii) the company that manufactures the product, (iv) the service engineers who service and maintain the product, (v) the government body who formulates legal and statutory regulations, etc. The more effort and care one takes to create an exhaustive list of customers (arbiters) early in the design sequence, and then to incorporate their views during the design process, the less the likelihood that an apparently satisfactory solution might be declared unacceptable at the end [217].

The next step is then to elicit and record the requirements from each group of customers. Customer requirements will include both explicit and implicit requirements; Manufacturing constraints might include requirements relating to total volume, manufacturing rate, manufacturing capability and capacity, etc.; Use Requirements might include those of maintenance, use environment, recyclability, disposal, et.c.; Company requirements might include : profit margin, capital expenditure etc.; Legal requirements might relate to safety in production, safety in operation and safety in maintenance. This is the process of defining the problem clearly.

These elicited requirements may be imprecise, vague and ill-defined. The next step then involves refining and classifying them. To avoid unnecessary effort in refining vague requirements, in DFD, it is advisable to limit the number of refinements to three levels of : Primary, Secondary and Tertiary requirements as shown in Figure 5.32. Tertiary customer requirements are then treated as the customer requirements for design purposes.

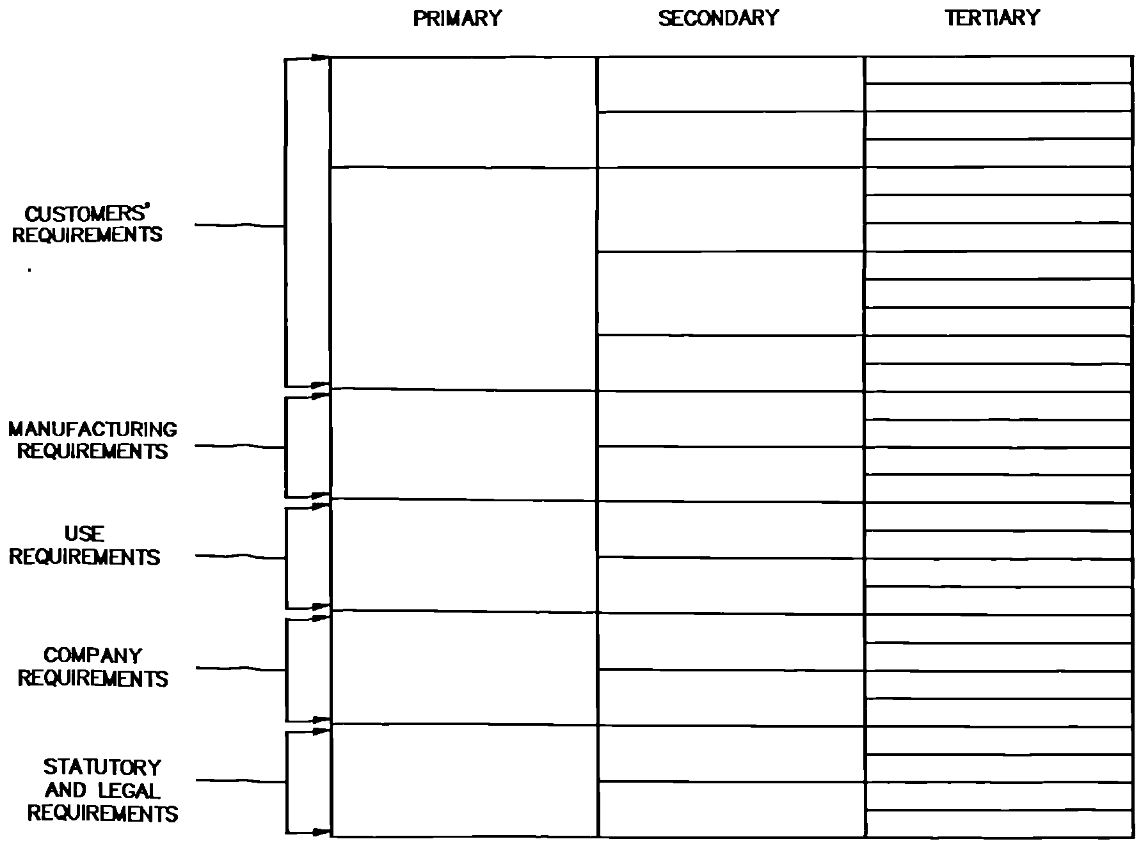


Figure 5.32 Classified List of all Requirements

After classifying the requirements, the next step involves constructing the quality plan. This is performed by the design team, who then rate the tertiary customer requirements, by combining : the degree of importance to the customer (measured on a scale from 1 to 5 representing no influence to very strong influence), the degree of improvement according to the quality plan (i.e. the ratio of the targeted rating by the customers to the current rating of the company's products), the sales advantage (using 1.5 for very important, 1.2 for important and 1.0 for non-relevance). Other contributory factors such as the environment, may also be considered. From these weighting values, the absolute and relative importance ratings of each tertiary requirement are then computed. The absolute importance for each requirement is the product of all the contributory factors mentioned above. The relative importance is then calculated by normalising them on a scale of 1 to 9.

An Example : If for instance a tertiary customer requirement described as - **easy to hold,** has the following weighting values :

- ¤ Degree of importance to the customer - (3)
- ¤ Improvement required = targeted rating according to quality plan (5) divided by current rating by the customer (4) = 5/4
- ¤ Sales advantage : very important (1.5)
- ¤ Environmental consideration : not relevant (1.0)
- ¤ The absolute importance rating is then $= 3 \times \frac{5}{4} \times 1.5 \times 1 = 5.63$

This procedure is followed for all the tertiary requirements. The normalisation process can then be carried out either by (i) taking the sum of all absolute weights and expressing individual ratings as percentage of the sum or (ii) expressing the largest as 9 and working out the others on a relative 1 to 9 scale.

The next step in this stage, involves translating or deploying the tertiary requirements into design elements. These design elements are referred to as design functions in DFD.

Design Function

Design function is the identifiable and actionable design requirement translated from customer and other requirements (DFD Stage 1) or design functions (DFD Stages 2-4) which ensures the satisfaction of the initiating customer requirements and the quality performance of the resulting physical product.

The translation process involves taking the tertiary customer requirements one by one and checking whether any of the already generated design functions deploy them. If they are not fully deployed, additional design functions are generated to fully deploy them. In this stage, that is, DFD stage 1, the design functions (specifications and constraints) should be established in a solution neutral form (i.e. with no reference to any particular solution). The translation process is continued until all tertiary requirements have been deployed. These specifications and constraints which deploy the requirements, are the tertiary design functions. They are then re-arranged, grouped and classified into secondary and then primary design functions as shown in Figure 5.33.

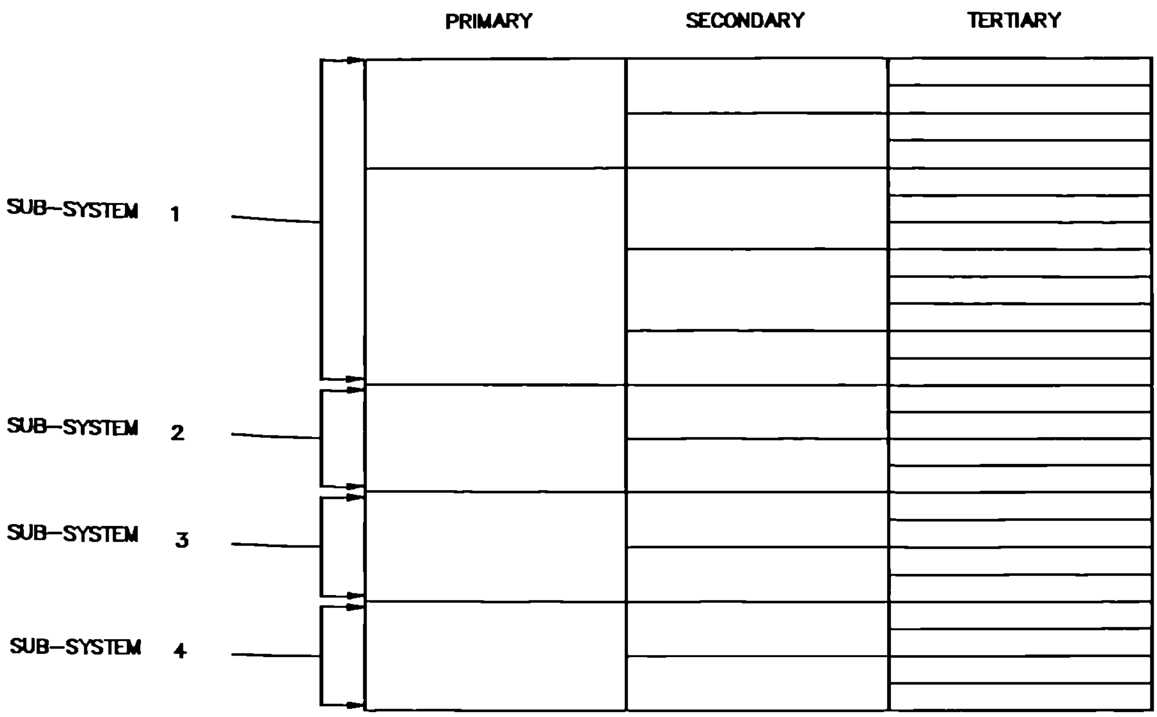


Figure 5.33 Classified Design Functions

The next step involves establishing the target information for the tertiary design functions. The target information consists of : (i) the upper limit of the target value, (ii) the lower limit of the target value, (iii) the target value (nominal), (iv) the confidence level of the target value and (v) the improvement direction of the target value, expressed as :

+ - *for the higher the better,*
0 - *for the target is best* and
- - *for the lower the better .*

The target values are known as the "How Much" of each Design Function. The Target information : (i) provides an objective assurance that the originating requirements including those of the customer, are met, (ii) provides targets for future improvements, and (iii) provides specific objectives which guide subsequent design and a progress assessement checks.

After establishing the target information for the tertiary design functions, the next activity involves analysing the design functions for interactions. The interactions are examined form two perspectives. The first one is problem-focussed, while the second one is solution-focussed.

In the **problem-focussed approach**, the process involves establishing supporting and/or conflicting design functions, in the form of effects, demands and restrictions

they place on each other. Values used to represent the interactions in DFD are :
+2 for Strong positive, **+1** for Positive, **- 1** for Negative, and **- 2** for Strong negative.

These form of interactions can identify at a glance which of the 'hows' support one another and which are in conflict. The assignment of + ive or -ive interactions are based on influence of 'hows' in achieving other hows. The *+ ive interactions* can result in efficiency gains by not duplicating efforts, to achieve the same results, while the *-ive interactions* represent conditions where trade-offs are suggested. One should look at the design with a view to designing out the trade-off (i.e. eliminate it), before performing any trade-off. In the **solution-focussed approach,** the focus here is on designing out conflicts and/or designing in benefits. Depending on the nature, type and level of interactions between the design functions from the problem focussed analysis, actions then have to be taken to : (i) resolve all conflicts and (ii) perform trade-offs between design functions. This might necessitate the need to **eliminate, combine, transfer or modify** any of the interacting design functions. In some cases, **standardization** and **simplification** of particular design functions might become necessary. This process is also useful for subsequent generation of alternative design concepts. The interaction matrix that is employed for both approaches is shown in Figure 5.34.

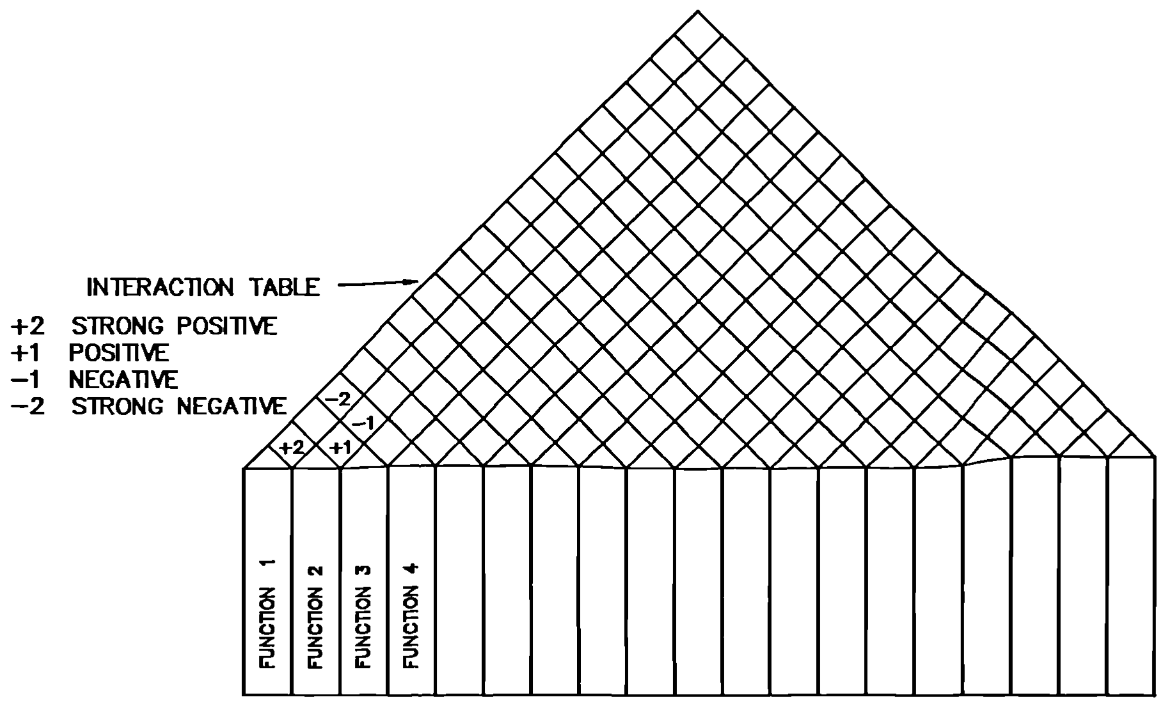


Figure 5.34 The DFD Interaction Matrix

On completing the interaction analysis, the next step involves constructing the relationship matrix, between the tertiary customer requirements and the tertiary design functions. The strength of relationship values used are 9, 3, 1 and 0, which represent strong, medium, weak and no relationship respectively. They are placed at the intersections of the tertiary customer requirements, 'whats' and tertiary design functions, 'hows', which are related. The process is continued until all the relationship values have been inserted. The completed matrix then helps in the interpretation of relationships, allowing cross-checking of blank rows or columns to indicate inadequate translations. After completing the relationship matrix, the final activity involves computing the importance rating for the stage 1 design functions (specifications and constraints). The importance ratings are the column sums of products of the importance rating of the customer requirement and the corresponding value in the relationship matrix. The sums of the products belonging to each column (hence to an individual design function) forms the absolute importance rating. The absolute ratings are then calculated on a relative 1 to 9 scale to form the relative importance ratings. The relative importance ratings give an indication of the care required in translating that design function into a realisable design. This is seen as the method of designing quality or 'fitness for purpose' into the design. The end product is a set of specifications including the necessary constraints with their relative importance ratings and target values. The completion of the above activities results in the DFD stage 1 chart, consisting of seven blocks or sections, that is, the requirements (Whats) along with their importance ratings, the specifications and constraints (Hows), the relationships, the interactions, the target information relating to 'Hows', the absolute importance ratings of 'Hows', the relative importance ratings of 'Hows'. The complete DFD stage 1 chart is shown in Figure 5.35.

5.15.2 DFD Stage 2 - Establishment of Viable Architectures

In stage 2, the main parameters from the specifications derived in stage 1, are recorded in a single block of rows, with their target information recorded in the columns to the right. For each of these parameters, realizable generic subsystems are then generated in the form of alternative solutions to the parameters. The next step then involves synthesising and generating combinations of the generic subsystems, using the morphological analysis, functional analysis and other analytic methods to arrive at alternative design concepts (architectures). For each architecture and associated subsystems, the following activities are performed : (i) establishment of the subsystems characteristics, (ii) establishment of the target information for the subsystems characteristics, (iii) construction of the interaction matrix, resolution of conflicts and performance of tradeoffs, (iv) construction of the relationship matrix, (v) computation of the absolute and relative importance ratings and then (vi) completion of the main chart for each architecture, of which a typical one is shown in Figure 5.36.

STAGE 1 DFD CHART

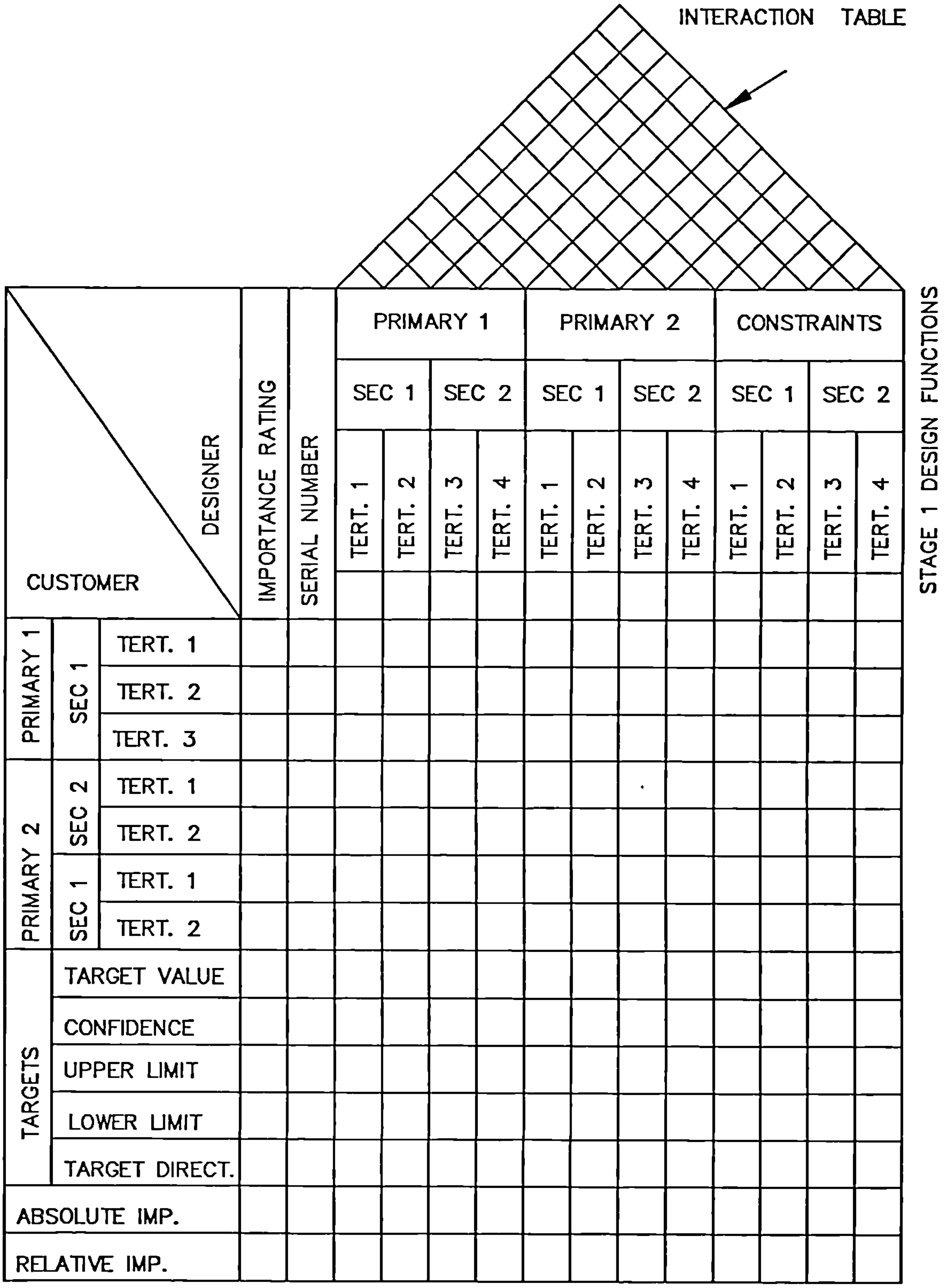


Figure 5.35 The DFD Stage 1 Chart

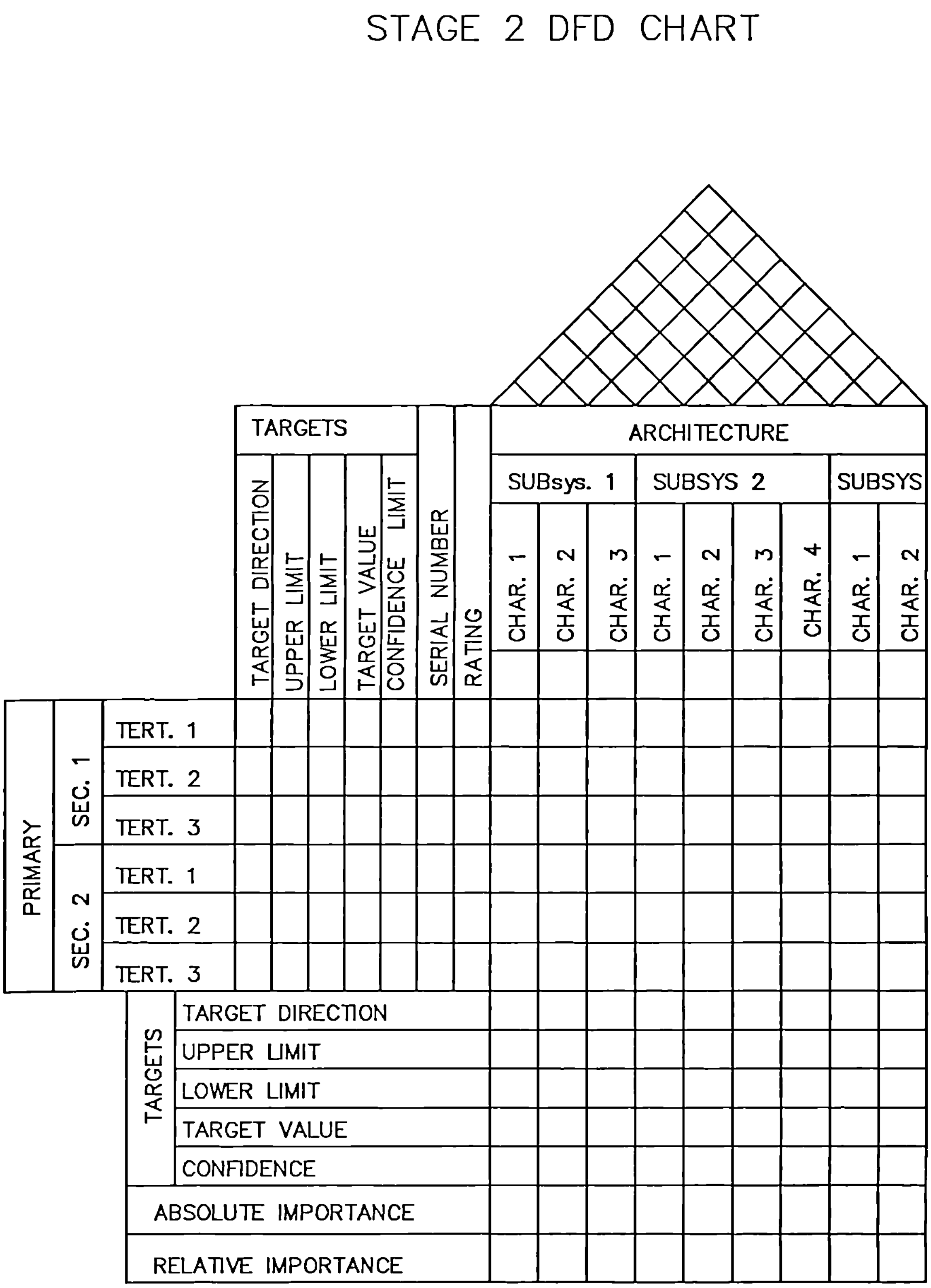


Figure 5.36 The DFD Stage 2 Chart

The final step in this stage, involves evaluating the specifications and constraints and using them to select the optimal architecture for further development. At this stage, topological and preliminary 3D models might be created for viable architectures, depending on the nature of the design and the complexity of the evolving product.

5.15.3 DFD Stage 3 - Establishment of Viable Layouts For Each Architecture

In this third stage, a similar procedure to stage 2 is followed. The first activity here involves recording the main subsystems characteristics that were established in stage 2 for an architecture in a single block of rows, with their target information recorded in the columns to the right of their row records. Then for each subsystem characteristic, alternative designs (either in textual form or in the form of sketches) are used to describe different alternative parts that will deploy or bring to life the subsystem characteristic. After completing this activity, viable combinations of the parts are then selected to form the individual subsystems. The viable subsystems are then synthesised and combinations generated, using the morphological analysis, functional analysis and other analytic methods including geometric modelling to arrive at viable layouts. For each layout, subsequent activities performed involve : (i) analysing the parts and establishing their characteristics, (ii) establishing the target information for the parts characteristics, (iii) constructing the interaction matrix, resolving conflicts and performing tradeoffs, (iv) constructing the relationship matrix, (v) computing the absolute and relative importance ratings and then finally (vi) completing the main chart for each viable layout, as shown in Figure 5.37.

At this stage, solid models of parts and assemblies can be built and necessary Finite element and mechanism analysis can be performed. Preliminary information about the materials for the parts can also be selected.

5.15.4 DFD Stage 4 - Selection of Materials and Establishment of Manufacturing Processes

In this stage, the parts characteristics that were established in stage 3 for the layout, are first of all recorded in a single block of rows, with their target information recorded in the columns to the right of their row records. Then for each part, using the materials and manufacturing process functions (translated from design functions) suitable material(s) and corresponding manufacturing processes are then identified and selected. The selected material and manufacturing process must of necessity deploy the part characteristics. The selection of the appropriate material and manufacturing process will influence the final geometry of the part and if modifications are necessary, these are performed. The above process will involve the use of materials and manufacturing process knowledgebases and databases. After establishing the manufacturing processes for each part of a viable layout, the subsequent activities involve : (i) identifying the process that will suit the constraints identified in stage 1, (ii) establishing the target information for the process parameters, (iii) constructing the interaction matrix, resolving conflicts and performing tradeoffs, (iv) constructing the relationship matrix, (v) computing the absolute and relative importance ratings and then finally (vi) completing the main chart as shown in Figure 5.38. At this stage, the geometries of the parts and layouts are finalised, together with their materials and manufacturing processes.

STAGE 3 DFD CHART

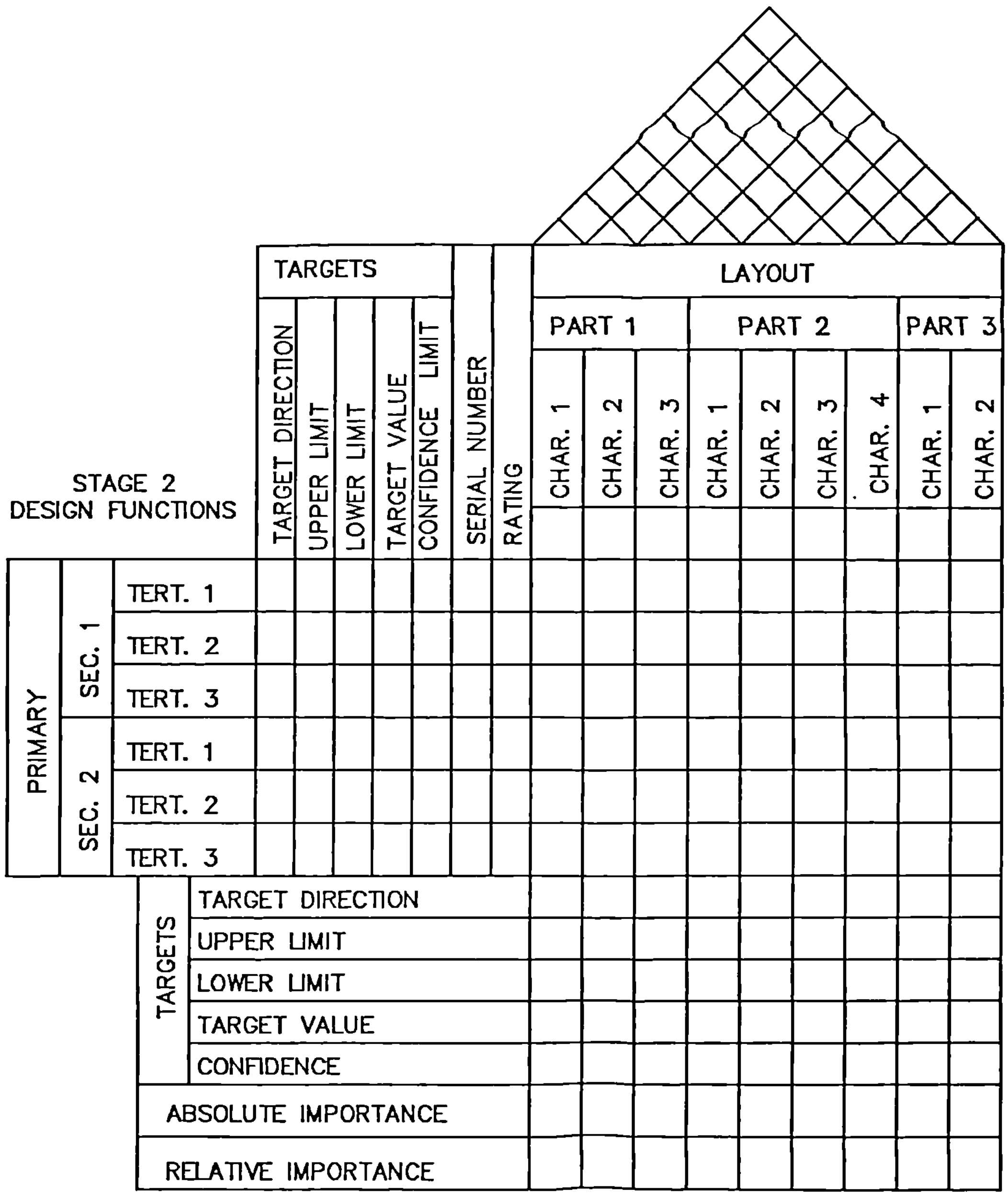


Figure 5.37 The DFD Stage 3 Chart

5.15.5 DFD Stage 5 - Development of Production Plans for The Parts of a Layout

In this stage, for each viable layouts, the process characteristics that were established in stage 4 for the parts of a layout, are recorded in a single block of rows, with their target information recorded in the columns to the right of their row records. For each part, the master production schedule can then be identified and established. This include information about operation planning, quantity economics and quality control details. Each viable layout with different materials, manufacturing process combination in stage 4 can have several production plans, varying with the volume of production. Hence they will have different charts. It is needful to also determine at this stage, critical control requirements, production maintenance requirements and mistake proofing requirements. A typical chart for a production plan in this stage, is shown in Figure 5.39.

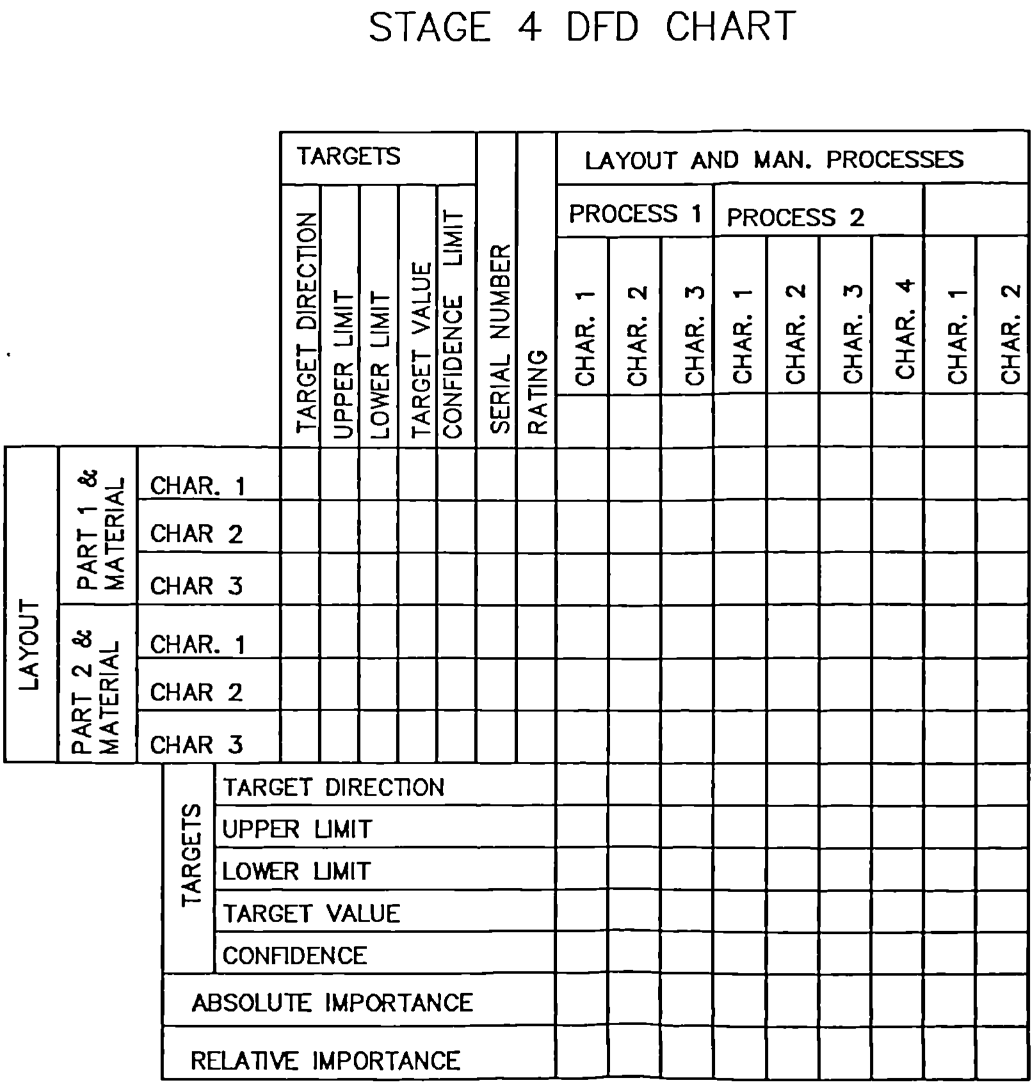
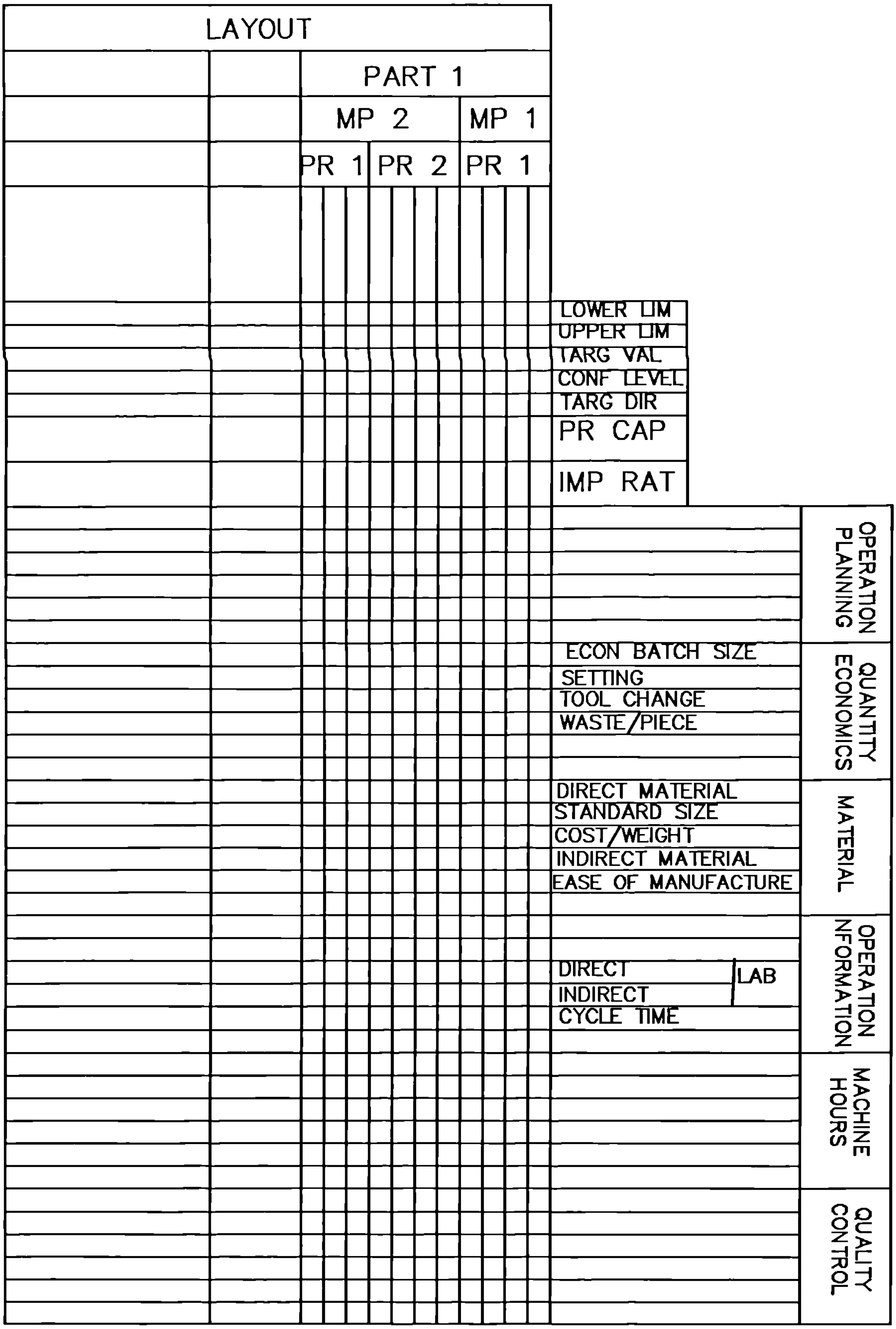


Figure 5.38 The DFD Stage 4 Chart

Figure 5.39 The DFD Stage 5 Chart



5.16 SUMMARY

There is no doubt about the need for a comprehensive design system in a modern product development and manufacturing company. Such a system is needed for capturing the product development process right from inception through manufacture to retirement and disposal. DFD which has been developed as a comprehensive concurrent engineering design system, has been the central focus of this chapter. The discussions throughout the chapter have explicated in detail, what DFD is and what it stands for, as well as its goals and objectives. Further discussions have also covered the DFD system structure, describing the three main components, that is, the DFD design model (level 1), the auxialliary design modules (level 2) and associated knowledgebases and databases (level 3), as well as expantiating the different stages of the design model. In section 5.6, the design modules within DFD were summarily described. In looking at the design matrix within DFD, the three fundamental aspects viz : the design process, design activities and design methods (modules), were further elucidated showing their interrelationships. A key ethos of DFD is the generation of the design solution space. The approach to generating the design solution space in DFD and the resulting morphology of the increasing complexity of the evolving product, were also discussed. Further discussions were aimed at demonstrating the capabilities and potentials of DFD as regards the concurrent management of the design process, the dynamic and hierarchical product modelling environment and the process of concurrent capture of design knowledge within the DFD system.

Materials and manufacturing process selection and establishment, is an integral part of the design process within DFD, In this chapter, the approach adopted within DFD in concurrently selecting materials and manufacturing processes, was enunciated as well as the proposed system architecture for the materials and manufacturing process selection system to be integrated into DFD. Another key module within DFD is multi-criteria design optimisation. The approach used within DFD in carrying out multi-criteria design optimisation was also strongly enunciated. The chapter then concluded with a general description of how to design using DFD, in a manual form.

CHAPTER 6

DESIGN FUNCTION DEPLOYMENT AS A CONCURRENT ENGINEERING DESIGN SYSTEM

6.1 INTRODUCTION

Having discussed the requirements for a concurrent engineering design system in chapter 4 as well as describing Design Function Deployment and explicating its potentials and capabilities in chapter 5, this chapter focusses on the relationship between Design Function Deployment, DFD and Concurrent Engineering and how DFD satisfies the highlighted requirements in chapter 4 as well as providing the platform for the realisation of concurrent engineering in practice. The proceeding sections, will show in detail, how DFD meets the requirements for a concurrent engineering design system.

Discussions will also focus on how the conceptual development and thinking within DFD as well as how the DFD software implementation satisfies the requirements of a concurrent engineering design system.

6.2 DFD AND GLOBAL GOALS OF CONCURRENT ENGINEERING

The DFD design system has been developed to support product development teams, by enabling the integration of all life cycle and functional issues under one umbrella. The provision and development of a generic central design database within a product modelling environment, enables early as well as concurrent access to design and manufacturing information. Thus enabling access to the same design information by each functional discipline of the product development team. The DFD charts also help to capture and display necessary design information, and thereby enabling the design team to focus on both external and internal customer requirements.

The activities in stage 1 of the design model in DFD, enable the designer/design team to respond flexibly and proactively to customer requirements, and thus be able to adjust the design solution space whenever there are additional new requirements or when there is a complete revision of the requirements' list.

DFD also represents a design for quality system - enabling focus on quality associated with external customers as well as quality associated with internal customers, in addition to focussing on the quality and robustness of the product. By supporting the various functional aspects of the product development team, DFD enables the availability of engineering and design knowledge about product performance upfront in the design stage. Thus ensuring that design decisions can be more realistically and accurately made. The level 2 modules DFD, represent concurrent engineering tools that can be used to rapidly assess and evaluate design alternatives. These tools also ensure the integration of all lifecycle issues in the product development process. The DFD design system framework, with the utilisation of the hierarchical decomposition of the design solution space and necessary project management techniques, enable the management of the design process in a parallel manner. The structured and systematic format of the design process within DFD provides a basis for evaluating and reviewing its usefulness in improving the efficiency and effectiveness of the designer/design team. Thus providing feedback information to continuously improve the product development process.

Design Function Deployment (DFD) in comprehensively satisfying the principles, goals and requirements of a concurrent engineering design system, presents itself as:

A platform for world class concurrent engineering

6.3 DFD AND GENERAL REQUIREMENTS OF A CONCURRENT ENGINEERING DESIGN SYSTEM

The DFD system at level 1, provides a structured format for carrying out the design process in a planned and controlled way. This ensures that all the necessary information generated, analysis performed, evaluations done and decisions made, are recorded and documented accordingly. The first stage of the design process in level 1 ensures that elicited requirements (customer, user, manufacturing, company, marketing, finance, sales, statutory, suppliers, etc) are all considered in an integrated and concurrent manner, before translation into design requirements. Hence the stage 1 of DFD, provides the basis for teamwork by the integration of the various functional and other interest groups influencing product development, at the early stages of design.

Level 2 of DFD contains the several design modules (methods and techniques) which can be used in a parallel manner, as design proceeds from conceptual to detailed design stages, and subsequent release to production planning. These modules are used in the generation, development, analysis, evaluation, optimisation and selection of design concepts (architectures and subsystems), layouts and constituent parts, materials and associated manufacturing and production processes. These modules, in their use, ensure that design concepts are not only functional, but can be manufactured, assembled, sold and used to the satisfaction of customers/users/owners.

Level 3 of the DFD system contains databases, rule and knowledge bases, some of which are associated with the level 2 modules. These modules store various design knowledge and information in the form of design rules, materials and manufacturing data and previous design information that can be retrieved and used for new designs.

This level will also contain a central generic product database management system, to capture the evolving total product model as the design process proceeds from stages 1 to 5. The databases and rulebases will be designed to enable concurrent access to both past and present design information by all interest groups (design, analysis, manufacturing, production, testing, etc). This will help to remove unnecessary interfaces between the groups, and hence reducing delays in the product development process. The link of the level 3 modules with levels 1 and 2 in the DFD structure, also helps in the implementation of concurrent engineering, by the integration of conceptual design, geometric modelling, materials and manufacturing process selection, as well as permitting the increase of product performance knowledge in the early stages of the design process, when costs are committed.

The structure, organisation and use of the DFD charts during the design process, enables the capture of the textual (alphanumeric) information associated with the evolving design. Such information includes the requirements, specifications and constraints, the design intent and rationale for decisions made as well as the history of the design process. The DFD design process along with the use of the charts, also enable the development of alternative design concepts for architectures (stage 2), layouts and parts (stage 3), in a parallel manner, thus reducing the development time. This involves removing individual or groups of design activities in the critical path, and performing them in parallel in a non-critical path, using traditional as well as other innovative project management techniques such as the Design Structure Matrix [180].

The DFD software will be developed to :

(1) have an intuitive user interface that allows the designer to progress in the design process in a natural - i.e. developed to be designer led.
(2) be generic and adaptable to different engineering design disciplines (mechanical, aerospace, civil/structural, electrical, etc).
(3) accomodate all relevant lifecycle issues of the design process as well as the product.
(4) support paralleling of the design process and providing a framework for its implementation.
(5) provide an interactive design environment enabling the performance of necessary design revisions, what if analysis and tradeoffs, through the use of the charts, and the systematic stage by stage progression and the support of multiple views and facets of the design.

The constituents of the overall system will be developed in a modular fashion in order to support an open architecture framework enabling the control, modification and extension of the system, as well as the integration of additional necessary concurrent engineering tools.

Some of the level 2 modules will also support the use of existing knowledge in the form of knowledge/rulebases.

6.4 DFD AND DESIGNERS/USER S/SYSTEMS REQUIREMENTS

6.4.1 DFD and User Interaction Requirements

The DFD design stages in level 1 helps to guide the designer in moving from one state to another.

The DFD system software user interface will be developed to :

(1) support the movement from one tool to another during usage in an intuitive manner.
(2) support multiple viewing windows of the DFD charts as well as other outputs for the different representations of the design.

6.4.2 DFD and Design Process Management Requirements

Design process control

The DFD design stages (level 1) including the charts represent a formalisation of the design process enabling a flexible control of both design goals (intent) and objectives.

The DFD system software system will implement a tool to record the design activities in the form of a time sequence (using dates and time) to support a control and monitoring mechanism for the design evolution process.

An Open architecture Object-Oriented database management system will constitute the main database engine of DFD. It will support the production of status, version, notification, product inheritance, and design rationale reports, as well as supporting communication facilities to enable both concurrent and common access to design information.

A design management module will be developed within the DFD software based on the Design Structure Matrix technique. This will support the identification of the core activities at any stage of the design process, guide the decomposition of these activities as well as the evolving product and enable the scheduling of the design process into both serial and parallel tasks supporting the concurrent design process in DFD.

The DFD system software with the aid of the charts and the design tools, enable the capture of design issues such as the goals and intent of design actions, the alternative course of actions undertaken during the design process as well as the logical basis (reasoning process) behind the design decisions made. Thus enabling traceability to design decisions made, their justifications and the impact and influence of such decisions on the evolving design and the design process.

The DFD design stages as shown in level 1 of the structure, represents the minimum prescribed path to be followed and thus acting as a systematic guide for the design team during the design and product development process.

Design Iteration

DFD supports a flexible approach to design, and in complement with the systematic stages of the design process, enables the revision of prior design decisions as well as previous solutions. This is shown in the DFD design model. The software implementation of the DFD system in addition to the development of the charts, will provide the feedback process and thus enabling traceability and design iteration.

Design Communication

The structured approach in DFD ensures that there is commonality of views and ideas as well as the various representations (textual and geometric) of the design throughout the five stages. The representations of the evolving design provides the necessary basis for communication between the various interests groups (marketing, design, analysis, manufacturing, testing, service, etc) involved in the product development process.

The DFD system based on the use of the design modules shown in level 2 of the structure diagram, charts, filecards and design notebooks, will enable the capture of the design information, both textual and geometric, which is required for the necessary communication, conflict negotiation and resolution as well as constraint management between each of the design disciplines.

The DFD software will support design data exchange between the design modules (level 2) based on the product modelling process within DFD and necessary product modelling standards (e.g. IGES/STEP) as well as the input/output parameters of each module.

The DFD software system will also provide notification, status, version and change control mechanisms to support communications between designers/design team members (design agents). The communication between designers and the level 2 design tools when required will be supported by intelligent rulebased interface routines.

Communication between various design agents (or engineering disciplines) will also be supported by the integration of their contribution at stage 1 of DFD. The provision of a central common database system based on the master modelling concept together with the interactive DFD charts, also ensure that the various disciplines have a common and concurrent access and view of the evolving design.

The software implementation of the level 1 of DFD including the charts and its integration with the necessary database management system will enable the sharing of design information among design team members.

Partitioning of Design Tasks

The DFD design stages as shown in level 1 of the structure provides a platform for structuring and partitioning the design process in perceivable tasks. The prescriptive approach to design in DFD is mainly based on a top-down approach, as demonstrated in the discussions on the design solution space generation and increasing complexity of the morphology of the design process within DFD (see section 5.9 of chapter 5). The DFD system however, also supports both a bottom-up and middle-out approach (as in QFD). This is further supported by the use of the level 2 tools as well as the design structure matrix technique in decomposing the design activites (tasks) into serial and parallel tasks and the design of the product or system into manageable modules and subsystems.

Feedback

The interactive links of the DFD charts as well as the systematic stages of the model, enable the performance of necessary feedbacks to earlier design decisions. The software implementation of DFD will ensure that this feedback is both dynamic and interactive, as the design evolves from requirements to detailed designs and production planning.

Design solution space generation

A key ethos of Design Function Deployment, DFD, is the necessity to generate the design solution space. DFD hence through its key components (levels 1, 2 and 3) as shown in Figure 5.6, provides the necessary procedures and techniques (tools) to enable the generation and exploration of the design solution space.

The use of level 2 modules such as the morphological analysis, functional analysis, geometric modelling, multi-criteria optimisation, etc, help to achieve the rapid generation and evaluation of the design solution space in a rational, cohesive and comprehensive manner.

The five stages of DFD provide the necessary strategy for the exploration of the design solution space and for ordering design decisions in the subsequent choice of a particular design solution

Abstraction of Design Stages

The DFD design stages represent the prescriptive model of the design process as well as providing the representation of the intermediate states in the design process. The product modelling environment within DFD will principally consist of both textual and geometric models. The resulting product models will represent all the life cycle aspects of the DFD design model (stages 1 to 5)

The DFD design stages along with the product modelling environment provide the basis for the abstraction of both the design process and the product.

Multiple Design Models

The DFD system (as in levels 1, 2 and 3) supports both the prescriptive and descriptive models (approaches) to design. Depending on the design domain and the design modules being used, the DFD system supports either a top-down (architectures to layouts to parts), bottom-up (design retrieval/case-based design) and middle-out (adaptive designs/QFD style development) approach to design. The product modelling environment within DFD will also provide the necessary abstractions of the different functional and geometric views and representations of the product.

Design Data Management

The dynamic product modelling environment in DFD together with the implementation of the integrated common product/design database provide the capabilities for design information (data) transfer between different design disciplines (agents), design tools, and other CAD/CAM tools. The use of data exchange standards, i.e. IGES/STEP within the DFD software system, will also support the exchange of design data between the different modules in level 2.

6.4.3 DFD and Design Artifact (Product) Requirements

Focus on Quality

The focus on the quality of the resulting product, is looked at from two main viewpoints, that is, (a) product quality based on the perception and satisfaction of the customer/user, (b) product quality associated with robust engineering design (designing the product to be insensitive to variability effects arising from manufacture and use) The use of the QFD style charts in level 1 of the DFD addresses the first viewpoint, as they are used to capture, refine, categorise and prioritize all the requirements affecting the product (including that of the customer/user). The prioritisation process uses a chart known as the quality plan chart, taking account of the various weightings from customer, competitors (improvement plan) and sales. These requirements are then translated into measurable design functions (specifications and constraints) in such a way as to ensure that the product quality characteristics as specified by the customer/user, are designed into the product. On the second aspect of product quality, a number of modules are being developed within DFD level 2 to support the design of the product for robustness. These include, Robust engineering design, experimental design, and parameter selection modules.

Decomposition of products

The DFD model in level 1 of the structure diagram including the use of the charts, help to provide a systematic means of decomposing large scale and complex products during the design process. The charts in particular can besides being employed in the development of the overall conceptual (architectures) and detailed (layouts) designs, be used to also capture the decomposed subsystems of architectures, or parts of a layout, when a large scale or complex product is being developed. Furthermore, the design management module will also support the decomposition process and design of such components (subsystems and parts) employing the necessary clustering and triangularisation algorithms (see section 5.10).

Abstraction of product design

The product modelling process including the product database as well as the level 2 modules, will enable the representation of the evolving product from multiple perspectives and abstractions (e.g. functional, structure and behaviour of the product). These representations will span through the five stages of the DFD process as well as supporting product abstraction at the general level of architectures or layouts to the specific and detailed levels of parts and components. Some of the level 2 modules such as the solid/geometric modelling module and the computer aided process planning module will provide the basis for supporting feature based design approaches within DFD.

Multiple design views

It is needful that within DFD, the designer will be able to visualise the evolving design from the various multiple representations of the product, in order to for instance be able to relate not only both the textual and geometric views, but also the functional, behavioural and structural views of the design. In DFD the level 2 modules such as the FE analysis, solid modelling, functional analysis modules as well as the associated charts will provide the basis to support this requirement.

The integration of the textual and geometric design data generated during the design process, is the main goal of the product modelling environment within DFD. Thus the DFD product modelling framework will be developed to support this integration in a dynamic way right from DFD stage 1 to stage 5.

6.4.4 DFD and Design Team Support Requirements

The overall philosophy of Design Function Deployment, DFD, is to support the collaborative design process amongst different design disciplines that contribute to the product development process. This is further exemplified by the wide range of design modules in level 2, covering a wide range of functional design disciplines from marketing, product planners, designers, analysts, to materials, manufacturing and production specialists. These level 2 modules also provide the framework for the different transactions (computations, communications, negotiations, decision making, etc,) carried out by members of the design team. Level 1 of DFD also acts as a guide to the design team in ensuring that the design process progresses in a systematic, logical and controlled manner, recording all the different design views, representations and decisions. The charts also help in capturing the different views from designers and experts in the design team, as well as enabling the performance of tradeoffs and resolution of any conflicting situations.

The capability to support multiple designers and experts so as to work in a co-operative and collaborative way, will be provided eventually by the software implementation of DFD. This will involve using necessary software tools based on the UNIX operating system and X-windows/Motif environment, as well as using the C and C++ programming languages. The use of multimedia/groupware systems which support collaborative work involving graphical group interaction, will also be investigated within the DFD software.

The knowledge bases in level 3 of the DFD structure, will take account of all the necessary knowledge domains that relate to the product being developed.

The database management system within DFD will be developed to support sharing of design information while also employing suitable UNIX/X-windows tools available.

The product modelling environment within DFD, is being developed to be integrated, dynamic and generic (domain independent). It would be linked with the database management system in such a way as to support sharing of the evolving product information amongst the design team members.

The integration of the product modelling activity with the database management environment, will also ensure the capture of design rationale, goals and constraints, design changes as well as support design change notifications, status reports and version control.

It is also the long term intention of DFD to support design teams irrespective of their geographical locations, and whose interactions can be centralised or distributed, asynchronous or synchronous. This will involve the use of tools like multimedia/groupware systems, UNIX/X-windows toolkits, in conjunction with networking facilities.

6.4.5 DFD and Manufacturing Requirements

Stages 4 and 5 of the DFD model primarily addresses the issues of manufacturing process planning and production planning respectively. Level 2 modules such as design for manufacture, assembly and computer aided process planning are key modules that support these activities. The solid/geometric modelling module will also provide the basis for extracting product design and manufacturing features to be used in computer aided process plaanning. The materials and manufacturing process selection modules together with the associated level 3 corresponding databases, also support the manufacturing and production planning processes.

6.4.6 DFD and Product Life Cycle Requirements

A majority of the modules in level 2 of the DFD structure represent and support various life cycles of the product right from design, through manufacture to use and disposal. These modules backed by the design process in level 1 and the rulebases and databases of level 3, provide the platform to support and consider all necessary life cycle issues while designing the product.

6.4.7 DFD and Design Life Cycle Requirements

The DFD structure represents the overall design system that can support the different design phases, that is, conceptual, embodiment and detailed stages. Level 1 represents the design model in the form of a prescribed minimum path for the design phases. It is supported by levels 2 and 3 which contain auxiliary design modules (methods) and rulebases/databases respectively to augment the activities throughout the design phases.

6.4.8 DFD and Computer Software Requirements

The primary objective of the DFD design system, is provide a computer hardware/software framework to support designers/design teams and engineers in an integrated, efficient and effective manner, during the product development process, by providing enabling tools, techniques and intelligent aids.

Thus the computer hardware/software requirements of a concurrent engineering design system, are key issues to be addressed in the software implementation of DFD.

The DFD software will hence be developed to :

(1) Provide an open and extensible information architecture that :

(a) allows the integration of existing CAD/CAM/CAE tools to provide a variety of functions and services to designers. This will be achieved using necessary graphical interface builders for the global system as well as locally for each design tool (module).
(b) supports a large population of tools as well as design and manufacturing data models. This is particularly satisfied by the design modules in level 2 of DFD.
(c) accomodates a distributed computing environment with appropriate levels of security based on open networking architecture to allow teams to work transparently.

The use of a database management system based on the master modelling concept within DFD would satisfy this requirement.

(2) Support an infrastructure that enables both procedural and object-oriented software development approaches as well as necessary knowledgebase and database management systems. The implementation of the DFD software will progress to take account of the above, as the overarching structure of level 1 and some of the level 2 modules can be developed using a combination of a procedural langauage like C with the more object-oriented C++. The development of other level 2 modules and the modules in level 3, will of necessity involve the use of knowledgebase/expert system shells and database management systems (both relational and object-oriented).

(3) Support a common interface structure that will enable designers/users to move from one design tool to another with minimal effort and learning, and thus concentrating on the tool rather than on the interface. This will be developed in DFD using the necessary Graphical user interface (GUI) and X-windows toolkits.

(4) Accomodate knowledgebases to support the design process. The provision of a suitable knowledgebase/expert system shell within the DFD system will address this requirement. The knowledgebase system will enable the generation, representation, capture and efficient use of existing design knowledge for the creation of new designs as well as supporting the decision making process based heuristic knowledge.

(5) Accomodate design databases for information storage and retrieval. The issues related to databases within DFD are particularly addressed by level 3 of the DFD structure diagram. The database development within DFD will employ suitable database management systems (both relational and object-oriented) for the development of databases for corporate design knowledge, proven and previous designs (systems, components and parts), materials and manufacturing processes. To integrate the textual and geometric data of the product model for the evolving design, a central intelligent database will be developed within DFD to capture the product information and to also provide a flexible way of accessing, displaying, interpreting and distributing (sharing) the information to all the design and product development team members.

(6) Accomodate an intelligent materials selection system. This system will be developed to interact dynamically with the design evolution process within DFD, such that materials selection can be performed at any of the five stages within DFD. It will also be linked with the necessary materials and manufacturing databases in level 3 of DFD.

6.4.9 DFD and Requirements Capture Requirements

Requirements Analysis and Establishment

A principal activity within the DFD system is the capture, analysis, and establishment of customers/users requirements. The techniques and procedures for doing this involves the development of the DFD stage 1 chart as well as the use of level 2 tools such as the objective tree, functional analysis and generic checklists.

The use of the charts to record the rationale for the customers requirements and the quality plan chart used for prioritising them help in giving more understanding of the customer requirements as well as enabling their implementation in a rational and objective way.

The DFD charts are developed to link dynamically with each other through the five stages of the design process. This will enable traceability through each stage and it will be possible to examine the implications of previous decisions made as well as enabling the design team to focus on customers' requirements throughout the development process.

The activities in stage 1 of the DFD design model including the use of the charts provide a systematic way to (i) record requirements from various groups (including the customers/users) and (ii) prioritise and focus on the key requirements that really matter. This ensures that customers' requirements and needs can be more rationally responded to and thus avoiding delays in the product development process.

Summarily, the DFD design process with the aid of the design tools and techniques in levels 2 and 3 of the structure, provide *a platform for translating customers needs,* requirements and perceptions (whether they be vague, subjective and qualitative) into physically realisable products or systems.

Requirements Representation

The DFD charts and the underlying data structures provide the means for representing requirements, specifications and constraints of the design. The use of level 2 modules like functional analysis and objective tree will also support this endeavour.

The software development of the DFD charts will provide input and output formats, supported by the underlying data structures and files. This will enable the creation, revision and change of the requirements and specifications/constraints when required.

6.4.10 DFD and Design Tools and Techniques Requirements

Design tools and techniques that support concurrent engineering are contained within levels 2 and 3 of the DFD system. These tools are used to perform the many activities that occur during the design process, such as design synthesis, analysis, modelling, evaluation, optimisation, communication, manufacturing process planning, etc.

In conjunction with the overarching level 1 DFD design model, they provide a basis for satisfying the requirements for tools and techniques within the DFD system.

Some of the level 2 modules represent techniques for performing algorithmic and symbolic computations. These modules will be supported by suitable user interfaces that will enable the selection of appropriate modules when needed.

To support creativity, DFD provides tools such as the morphological analysis and functional analysis that help in the generation and exploration of the design solution space. Necessary Design for 'X' tools as well as multi-criteria optimisation techniques help to support the designer/design team in making value judgements.

A group of modules in level 2 of DFD ensure that necessary analyses of the evolving design are performed at appropriate stages of the design process. These include tools like FMEA, FTA, FEA, process capability analysis, etc.

In analysing multiple design alternatives, the optimisation tools and evaluation tools such as the DF'X' tools are employed whenever necessary, throughout each of the DFD stages. The DFD level 2 modules also support the generation, evaluation and choice of design alternatives, whether they be conceptual solutions (architectures), detailed designs (layouts), or subsystems and parts. Robust Engineering Design tools are particularly useful in designing for product quality and optimisation.

The provision of necessary databases and rulebases/knowledgebases to support the modules in level 2, also help to provide a variety of evaluation and tradeoff assessments and design iteration. The level 2 modules generally represent the various life cycle issues of a product and thus support various functional groups in the product development team, in ensuring that sufficient design knowledge is brought upfront early in the design process. These tools thus support concurrent engineering by the incorporation of the various perspectives and participants in the design process and ensuring that the evolving product is optimised against the constraints from the various perspectives. The design management module in level 2, will also support the concurrent management of the design process in the utilisation of the tools.

6.4.11 DFD and Design Knowledge Capture Requirements

<u>DFD and Capture of Design History</u>

The constituents of design history include [227, 255]: (1) an organised record of the sequence of design decisions , (2) an explicit representation of design objectives (requirements, specifications and constraints) and their alternative decompositions as well as the resulting product structure and model, (3) alternative designs considered for each parameter during the design process, (4) the arguments for and against the alternative proposals and (5) the rationales (logical basis) for choosing the final design and rejecting other alternatives.

The above issues are captured in the following manner within DFD. Level 1 of DFD represents the stages of the design process, within which several activities are performed resulting in necessary design decisions. The DFD stages along with the use of the charts and the level 2 modules, enable the capture of design decisions made at any stage of the design process, whether they are decisions relating to establishing requirements or generating alternative solutions or developing manufacturing process and production plans for viable design solutions.

The DFD charts are used to represent the design objectives in the form of :

(i) design functions (specifications and constraints - in stage 1
(ii) architectures/subsystems characteristics - in stage 2
(iii) layouts/parts characteristics - in stage 3
(iv) layouts/parts/materials/manufacturing process characteristics - in stage 4
(v) layouts/parts/materials/manufacturing process/production operations parameters - in stage 5.

The activities within the DFD design process, particularly in stages 2, 3, 4 and 5 leans heavily on the generation of alternative design solutions for the derived design functions, in the form of conceptual solutions (architectures), detailed designs (layouts), alternate parts, materials, manufacturing processes and production plans. This is a key feature of the DFD process.

The outputs and results of the qualitative and quantitative analysis and evaluation of the design alternatives, using the level 2 modules provide the arguments for and against as well as the logical basis for the decisions made. These outputs show the response, behaviour and rating of the proposed designs against set criteria of the designer/design team.

In addition to the foregoing, the DFD design model, charts as well as the product modelling environment within it, also provide a platform for capturing, representing, documenting and recording, tracing and reviewing issues associated with the evolving product's design history.

<u>DFD and Capture of Design Intent and Rationale</u>

Design intent represents the goal/aim (what the designer wants to do) and purpose (why he/she wants to do it) for a particular design action taken, while design rationale represents the fundamental reason (why you want to do) behind a design decision and record of the reasoning process/logical basis (how to do and how it was done), that is, how design intent (goal and purpose) are achieved. Thus both design intent and rationale although not synonymous, tend to overlap in meaning. Design rationale in general tend to be associated with the arguments for and against the choice of any or an alternative course of action.

The DFD design process provides a logical record of design intent as design progresses from stage 1 to stage 5. Decisions taken in subsequent stages of the design process are and should always be dependent on earlier actions taken in preceding stages. The linking of the DFD charts provides a traceability mechanism to prior design decisions and their impact at later stages. The rationale for design decisions made are also recorded within the charts explicating the effect of interactions between the design functions at each stage and the reasons for choosing, discriminating or progressing with a particular design alternative or action. The integration and association of the charts (textual information) and the geometric model and features within the product modelling environment in DFD, also provides a platform to support (i) the capture of issues relating design intent and rationale (justifications) with respect to both the evolving product and the design process, (ii) effective data management, (iii) preservation of design data relationships, (iv) capture of dependencies and key features of the design, (v) continuous access to design information and (vi) the integrity of the product information .

6.5 SUMMARY

The aim of this chapter has been to show how Design Function Deployment satisfies the requirements of a concurrent engineering design system, which were enumerated in detail in chapter 4. The discussions in this chapter have hence demonstrated how Design Function Deployment taking account of its potentials, capabilities, and constituents, satisfies the requirements of a concurrent engineering design system as enumerated in chapter 4. DFD has been discussed in the light of the various types of requirements, showing how it robustly meets and caters for them, both in the manual form and in the software implementation.

CHAPTER 7

CONCEPTUAL FRAMEWORK FOR THE SOFTWARE IMPLEMENTATION OF DESIGN FUNCTION DEPLOYMENT

7.1 INTRODUCTION

The focus of this chapter, is to establish a conceptual framework for the software implementation of Design Function Deployment (DFD) by examining first of all, the four key dimensions of the DFD system. These are : (1) the design process within DFD (stages 1 to 5) represented by level 1 of the DFD structure, (2) the design activities implicit within DFD, (3) the design tools (modules) represented in level 2 of DFD and (4) the design output (resulting product artifact descriptions). The various components of this framework representing the activities and tools within DFD, as well as the output of the design process, will be discussed in terms of their functions, constituents and their mode of use and application within DFD. The discussions on each of the modules of the DFD software framework, will also include preliminary functional specifications for the DFD software system architecture.

7.2 KEY DIMENSIONS OF DESIGN FUNCTION DEPLOYMENT

In developing the conceptual framework for the software implementation of DFD, it is needful to re-examine the various key dimensions of the DFD system. These dimensions are discussed under the following :

(1) The design process within DFD,
(2) The design activities performed within DFD,
(3) The design tools (methods/modules) used within DFD and
(4) The design output (resulting product artifact descriptions) of the design process within DFD.

The design process in DFD, is characterised by five distinct design stages, which also exhibit certain attributes of : (i) sequential/serial processes, (ii) concurrent processes and (iii) a sequence of time events. These stages are : (1) Establishment of requirements and specifications, (2) Establishment of viable architectures, (3) Development of layouts for architectures, (4) Establishment of manufacturing processes and (5) Generation of production plans. Design activities are implicit within the DFD system, but generally relate to the following : customer requirements capture, design synthesis, engineering analysis, design evaluation and verification, design optimisation, modelling, manufacturing process planning, communication, etc. Design tools which are represented in level 2 of the DFD system structure, are associated with the design activities enumerated above.

The performance of the design activities at any of the stages of the DFD design process, utilising the design tools, ultimately result in a design output. This design output is characterised by a set of solution neutral specifications, an explosion of many solutions represented by the design solution space, an integrated, dynamic and hierarchical product model (incorporating both textual and geometric data) and the eventual optimal design resulting from the multi-criteria evaluation and optimisation performed throughout the DFD process. If the DFD software is to meet the needs of designers/design teams, it is imperative that it should accomodate all of the issues pertaining to the design process, activities, tools and design output, elucidated above.

7.3 PROPOSED DFD SOFTWARE SYSTEM ARCHITECTURE

The proposed system architecture of the DFD software is shown in Figure 7.1. It represents a modular open information architecture consisting of individual and independent modules supporting various functions of the design process within the DFD system. Each of these modules when required will be able to communicate with each other, as well as with the user of the system. This architecture must be extensible by enabling the addition of new design tools and accomodate new types of information, as well as support a flexible approach to design, enabling the designer/design team to commence design at any stage. The various components of the modular architecture are :

(1) The Graphical User Interface development environment (GUIDE)

(2) The Design process control module (DPCM)

(3) The Communications module (CM)

(4) The Design management module (DMM)

(5) The Design tools module (DTM)

(6) The Product modelling environment (PME)

(7) The Knowledge-base management environment (KBME) and

(8) The Database management environment (DBME).

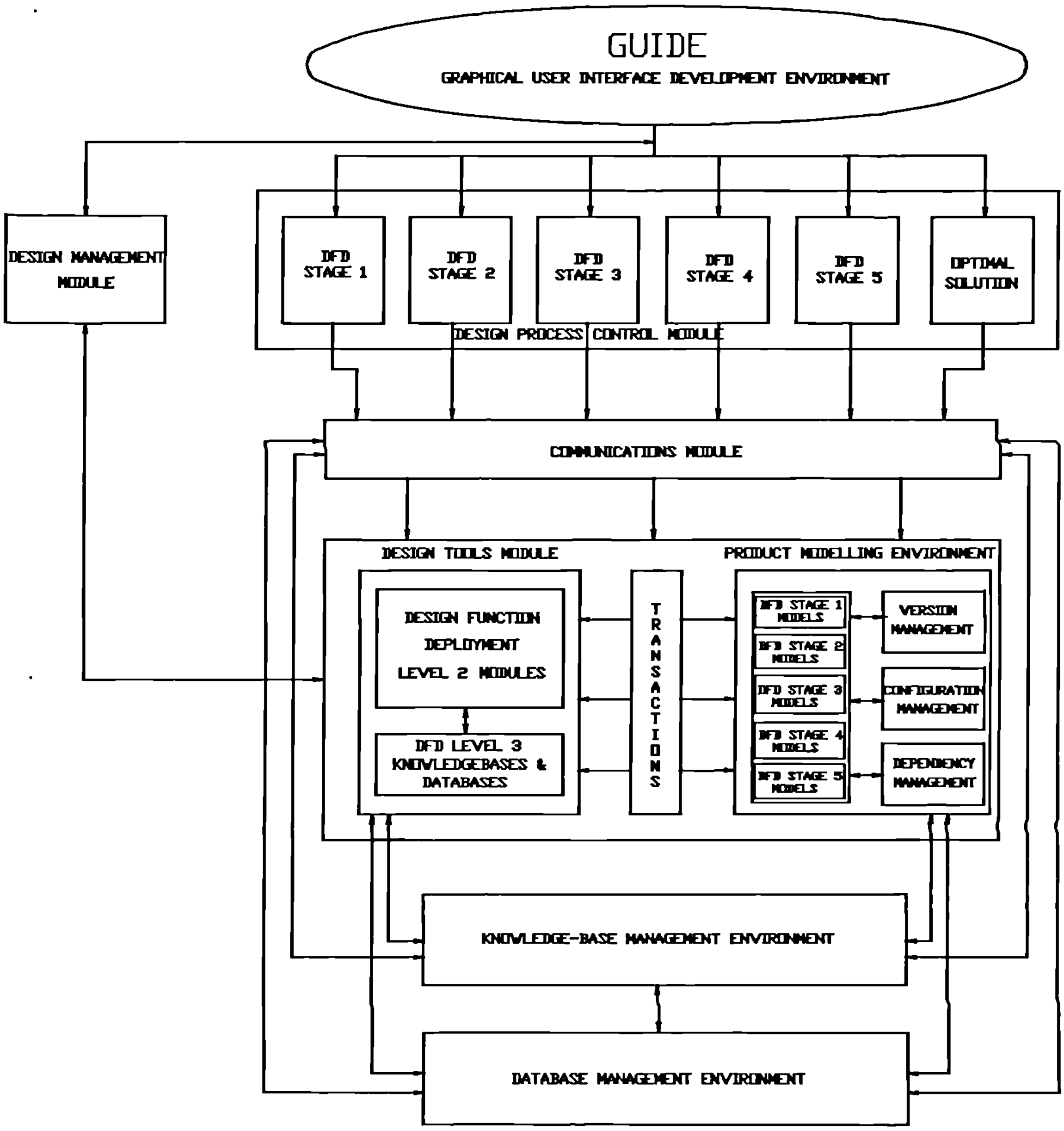


Figure 7.1 The Design Function Deployment System Software Architecture

7.3.1 The Graphical User Interface Development Environment

The graphical user interface environment will support the rapid creation and development of user interfaces for both the overall global system and for the individual design modules within the system. Such interfaces must be consistent and generally homogeneous across the different modules and should support easy navigation between the various modules and tools within the system. Both the global and local user interfaces will support the following functions : (a) File inputs and (b) interactive inputs. The interactive inputs will include the use of form techniques, menu techniques for selecting modules, command techniques as well as graphic and mapping based techniques [215]. The user interfaces will support the performance of the various activities in each of the DFD stages (level 1) as well as those performed when using the design tools in level 2 of the DFD system. The user interfaces will be developed to be flexible and adaptable as well as to support the displays of the design outputs, whether they be textual, graphic or geometric design information.

7.3.2 The Design Process Control Module

This module represents a subsidiary level to the global user interface and acts as a guide through the five stages of the design process in DFD. It consists of the DFD charts for each stage acting as graphical displays of the textual design information (in the form of requirements, specifications, design characteristics and parameters) generated through each of the DFD stages. It links to the design tools module via either the communication module or the design management module. This module controls the design activities performed during the design process as well as handling necessary design reviews and post design evaluations performed. This will be enabled by the dynamic links between the five design stages.

7.3.3 The Communications Module

This module will support the various communication activities performed within the DFD system. Such communications include those between the user and the various modules of the software system as well as necessary communication between the different design tools/modules. This module supports the communication and hence sharing of design information between participating design team members in a collaborative environment. It also provides the necessary access to both the knowledge-base and database management environments. The module will employ necessary communication protocols, which would define the rules for instance which the design tools have to obey in sending and receiving messages to and from other applications as well as the product modelling environment.

7.3.4 The Design Management Module

This module will act as the concurrency manager for the DFD system, supporting optimum planning, monitoring and scheduling of the various activities performed during the design process within DFD. This module will enable the management of the various activities performed in each of the DFD stages, the use of the various design tools constituted within the design tools module as well as the design of subsystems of an evolving design. Such management process will involve determining which activities can be performed either in series or in parallel, which design tools can be used preceding another or used concurrently and which subsystems of an evolving design can be developed in series or in parallel. The module will also provide information on the use of the design tools, as regards what they should be used for, when and how they should be used. The module will employ useful project management techniques such as the Design Structure Matrix [180, 237] to manage the relationships and dependencies between design activities and design tools, as well as the constraints on the design process. Since the prime objective of this module is to support the management of the design process, it will be controlled from the top level graphical user interface development environment as well as providing a link between the design stages and the design tools module.

7.3.5 The Design Tools Module

This module will contain the various application tools that are employed for the various design activities performed during the design process in DFD. These tools represent the modules in levels 2 and 3 within the DFD structure diagram, which account for the various life cycle perspectives of the product. They include those used for capturing and structuring requirements and specifications including constraints, synthesising conceptual solutions, developing detailed solutions, choosing materials and manufacturing processes and developing manufacturing process and production operations plans. They also support the concurrent analysis, evaluation and optimisation of the generated solutions, prior to choosing the optimal or most satisficing design. These tools are also used to create the various models of the evolving design which are constituted within the *total product model*. Some of the modules such as the one for materials selection will of necessity access their local databases.The application tools constituted within this module, are classified under the following headings :

(1) Requirements capture and analysis tools - objective tree, generic checklists and boundary searching.
(2) Design synthesis tools - functional analysis, morphological analysis, design retrieval, proven systems selection and proven components and parts selection.
(3) Design analysis tools - Finite element analysis and mechanism analysis.
(4) Design evaluation - design for cost, design for manufacture, design for assembly, design for environment, design for reliability, etc.
(5) Modelling tools - sketching input, feature-based design and solid modelling.
(6) Optimisation tools - concept selection, multi-criteria optimisation, mathematical

and symbolic computations.

(7) Robust engineering design tools - Taguchi methods, experimental design, parameter and response selection.

(8) Failure analysis tools - FMEA, FTA and cause and effect (fishbone) diagrams.

(9) Manufacturing process planning tools - materials selection, manufacturing process selection, computer aided process planning, process capability analysis and simulation.

(10) Production planning tools - job scheduling, line balancing, economic batch sizing, routing. materials resource planning and statistical process control.

This module is accessed by the design management module and also has a transaction link with the product modelling environment. This link enables the exchange of design data between any design tool and the product modelling environment, as well as between any two design tools, e.g. 3D modelling information transferred from the geometric modeller to the finite element analysis program. For the purpose of developing the necessary knowledge-bases and databases for some of the design tools, this module also links up with the knowledge-base and database management environments.

7.3.6 The Product Modelling Environment

This module is used to capture the total product model representations from stage 1 to stage 5 of the design process. These representations include the textual information generated within the DFD charts, the geometric information (models) for each architecture or layout as well as the results that arise from the use of any of the design tools. The product modelling environment captures the evolving product's knowledge dynamically as the design progresses from DFD stage 1 to stage 5. The core of the geometric information within the environment, is based on the 'master model' concept, that is, the geometric models created will be stored in a common database, which will then be accessible to any of the tools in the design tools module that need to work on the model, e.g. FEA, Mechanism analysis, computer aided process planning (CAPP), etc. In order to aid the utilisation of the models created in this module, other necessary modules are constituted within it. These submodules help to support the management of the changes that occur within the product models. They provide a set of consistent techniques to aid designers and users in the product model abstraction process, enable the record of design history, help to store different configurations of either architectures, layouts or parts and the exploration of alternative designs. They also support navigation through hierarchical designs and the management of multiple representations of changing design information. These submodules are [281]:

(1) a version management submodule - helps in managing the linear or branching derivations of design solutions. This submodule helps to support the dynamic nature

of the evolving product models and enables concurrent access by tools working on the same shared design data. It will also store temporal design versions and ensure that design tools access the most recent up to date version of the evolving design.

(2) a configuration management submodule - helps to govern the design object decomposition, such as the decomposition of conceptual solutions (architectures) into subsystems or detailed design solutions (layouts) into associated parts. This module supports the hierarchical aspects of the evolving dynamic product model; and

(3) a dependency management submodule - helps to manage the relationships between the various associated representations of the evolving design. This module will support the management of the transitions between the various product models created either within each DFD stage, or between the stages.

This module can also access both the knowledge-base and database management environments. The knowledge-base environment can be used to develop algorithms to act on the generated product models, while the database management system is used to store whatever product model information (data) is generated. This module will be developed to take cognisance of the STEP protocols for the exchange of design data.

7.3.7 The Knowledge-Base Management Environment

This module is the repository of the knowledge-based shell/toolkit that will be employed for the development of any knowledge-based/expert system software within the DFD software system. The knowledge-based shell will support both frame and rule-based representations. With the aid of this module it will be possible to develop for any application that requires a knowledge-based environment, a suitable inference engine, a knowledge base, the working memory, a user interface and an explanation facility. This module can be accessed by both the design tools and product modelling environment modules, when required for any development. It will also be possible to have direct access to this module through the communications module.

7.3.8 The Database Management Environment

The database management environment will be based on the object-oriented database management system. Object-oriented database systems provide a more user friendly data model involving actual objects the designer is concerned with. Such objects can be represented in the way designers think about them, e.g. architectures, layouts, assemblies, subsystems, parts, materials, etc. This hence obviates the need for translation to other data structures. In object-oriented database systems, there is no tradeoff between performance and non-redundant data storage, unlike relational databases systems, since the same object can be contained within other different objects. They are also more compact and efficient in storage. The object-oriented database system will be employed in the development of the design information repository that supports the product modelling environment. The database environment will also augment the development activities of the knowledge-base management system, particularly for applications requiring large information storage, such as materials and manufacturing process databases. The database management environment can also be accessed directly through the communications module. This module together with the knowledge-based management module will also support the development of the version, configuration and dependency management submodules within the product modelling environment.

7.4 SYSTEM DEVELOPMENT TOOLS

The software implementation of the conceptual framework discussed so far, will require the use of modern tools that are able to support such an engineering design system. The system can be developed within the workstation hardware environment as well as within the PC environment. In the workstation environment, it will be developed based on the UNIX operating system, under an X-Window/MOTIF window manager (for user interface development). In the PC environment, the appropriate window manager can be based on Microsoft Windows development toolkit. The core datastructures can be developed using a combination of both the C and C++ programming language environment. For the development of knowledge bases, a suitable knowledge-base shell can be employed.

7.5 SUMMARY

The logical extension of this research is the software implementation of Design Function Deployment (DFD) as a comprehensive computer based design system. This chapter therefore is aimed at establishing a bridge between the development of DFD as reported in this thesis and the necessary further work of software implementation. In this regard, a comprehensive framework has been established for the proposed DFD software implementation, containing the fundamental modules that must be constituted within the DFD software, in order to fully satisfy all the requirements of a concurrent engineering design system. Such modules as have been described earlier in this chapter, include: (i) a graphical user interface development environment, (ii) a design process control module, (iii) a communications module, (iv) a design management module, (v) a design tools module, (vi) a product modelling environment, (vii) a knowledgebase management environment and (viii) a database management environment.

CHAPTER 8

CONCLUSIONS

8.1 INTRODUCTION

This chapter which represents the last chapter of the thesis, will discuss the conclusions arrived at from the research programme. It details the experience, observations, challenges as well as exciting developments of the work and applications using Design Function Deployment. The discussions will summarise the issues discussed in chapters 1 to 7.

The research programme having come to a conclusion, has also opened up very many exciting and diverse areas for future research in design theory and methodology as well as aspects related to the software implementation of concurrent engineering design systems. These issues will also be discussed.

8.2 GENERAL CONCLUSIONS

This research was motivated by the long standing belief that in a modern product development environment, there is a need to integrate downstream manufacturing and use considerations into the early stages of design. Concurrently, the manufacturing industry has been witnessing an unprecedented fierce competition in the market place, with customers/users becoming more sophisticated and demanding in their requirements. In order to respond to the competition, it increasingly became incumbent on manufacturers not only to produce high quality products (with high technological content) that satisfied customers, but which also can be released early to the market place and at competitive prices.

The issue of product quality is particularly influential on the performance of a product in the market place, being primarily determined from the customers/users perception. With the knowledge that in the product development process, about 70-80% of product costs as well as quality characteristics are usually committed at the early stage of design, it also became obvious that design was pivotal in the eventual performance and success of a product.

The integration of the quality function into engineering design and the development of a comprehensive design system that provides an integrated platform for the implementation of concurrent engineering, hence became the mission of the research work, with the theme:

"Design It Right First Time"

If design has to be done 'right first time', it needs to be done in a planned, controlled and systematic way and should take account of all the main life cycle issues (manufacture, use, service, disposal, etc) of the product as well as the various stages (conceptual, embodiment and detailed) of the design process.

In respect of the above issues, the primary objective of this research was to develop a generic comprehensive design system that provides the basis and platform for the concurrent engineering approach to design, enabling the designer/design team to proactively respond to customer requirements and to develop well optimised products that can be released on time to the market place. Such a system was also required to provide the design team with a recipe of design tools to support the various design activities covering all the life cycle aspects of the product. The proceeding discussions below focusses on how the research work was carried out to satisfy the above objectives.

Chapter 1 of the thesis sets the scene for the research work, establishing the main objectives, rationale and significance of the research. It then concludes by describing the scope of the research programme and the outline of the thesis.

In addressing the research objectives, the research work commenced with an investigation into Quality Function Deployment (QFD), a technique commonly attributed to the Japanese Professor Yoji Akao [16]. It is a technique employed in capturing customer requirements (usually referred to as the 'Whats'), prioritising them, and translating them into measurable quality characteristics, or engineering characteristics or design requirements (also usually referred to as the 'Hows'). The above process constitutes the product planning stage of the QFD process. The process then proceeds from this stage to the parts planning stage involving the deployment of the quality characteristics (engineering characteristics or design requirements) from the first stage into parts charcateristics. In the next stage, that is, the process planning stage, the parts characteristics are deployed into process planning parameters. These process planning parameters are then in the fourth stage of production operations planning, deployed into production operations parameters. The deployment through these four stages represents the basic form of QFD.

Since gaining some popularity, efforts have been made to enhance and extend both the QFD technique and its usage and capability. Inspite of the enhancements made to QFD, it became apparent that it lacked certain design capabilities and did not adequately provide a platform for the generic design process. It was found to be conceptually static, that is, mainly able to cope with existing products having fixed design concepts, and incapable of handling very complex products as well as not supporting design tools and techniques for synthesis, analysis, evaluation and optimisation.

In the light of the above, further research was then embarked on to investigate and examine issues related to design theory and methodology. This research involved a detailed survey, review and evaluation of various design philosophies, models and methods. Additional work was also done in examining current on-going work in the development of computer based design systems. This work led to the observation that design models aggregated into four main categories of :

Prescriptive models based on the design process
Prescriptive models based on product attributes
Descriptive models and
Computational models

In the case of design methods, they were found to support the various design activities during the design process and can be classified into methods for setting objectives, synthesis, analysis, evaluation, optimisation, decision making, etc. The research activities on computer based design systems were found to be in their infancy and a number of researchers in the UK, Europe and the USA are getting involved in the development of such systems.

Concurrent with the above research into design philosophies, models, methods and systems, as well as the need to fully address the principal objective of the overall research project, research *into concurrent (simultaneous) engineering was embarked* on. This involved a comprehensive investigation of the principles, goals and benefits of concurrent engineering. The issues that need to be addressed in the realisation and implementation of concurrent engineering were also investigated along with current work being done to provide a software framework to support concurrent engineering activities in practice. The above issues cover the extent of the literature review and survey done in the research work, which has been comprehensively discussed in chapter 2.

Chapter 3 of the thesis then focussed on the evolution of the Design Function Deployment system. This involved discussions on the key features of Quality Function Deployment, QFD, Design models, methods and systems as well as key features of concurrent engineering, which contributed to the development of Design Function Deployment.

Since the main objective of the research was the developement of a comprehensive design system that provides a platform for concurrent engineering, the logical thing to do was to examine in great detail the requirements of such a system that would satisfy the needs of a generic design system and concurrent engineering in practice. This was the main focus of chapter 4. This chapter was developed based on information obtained from discussions had with engineers and designers from various industries as well as from research publications that focussed on the problem and which were aimed at addressing the same problem in one way or another. These requirements were discussed under general requirements and designers/users/system requirements. The designers/users/system requirements were further expantiated under the following : (1) user interaction requirements, (2) design process management requirements, (3) design artifact (product) requirements, (4) design team support requirements, (5) manufacturing requirements, (6) product life cycle requirements, (7) design life cycle requirements, (8) computer software and hardware requirements, (9) requirements capture requirements, (10) design tools and techniques requirements and (11) design knowledge capture requirements.

Having discussed the DFD evolution process in chapter 3 and the requirements for a concurrent engineering design system in chapter 4, the aim of chapter 5 was to fully elucidate the whole body of issues that constitute Design Function Deployment. These issues were related to the general philosophy, principles, goals and the structure of the DFD system. The design model as well as the auxiliary design tools (methods) which support the activities in each stage of the design process, were also discussed. The latter sections of chapter 5 focussed on the potentials and capabilities of the overall DFD system. Such potentials and capabilities include : (1) an explication of the design matrix in DFD, (2) the design solution space generation in DFD, (3) the product modelling environment in DFD, (4) the concurrent management of the design process in DFD, (5) the concurrent materials and manufacturing process selection in DFD, (6) the concurrent capture of design knowledge, rationale and history in DFD, and (7) the design optimisation process in DFD. This chapter concludes with discussions on designing using DFD.

With the full discussions on Design Function Deployment completed in chapter 5, chapter 6, then sets out to show how Design Function Deployment along with its software implementation, satisfies the needs and requirements of a concurrent engineering design system that were enumerated in chapter 4. This involved taking each of the requirements and discussing how DFD meets the requirements and in some cases to what degree it satisfied them. The discussions in chapters 5 and 6 demonstrated how DFD meets the requirements set out in chapter 4 as well as providing a platform for the realisation of concurrent engineering in practice.

In Chapter 7, the links between the developmental aspects of Design Function Deployment and the software implementation are discussed. This related essentially to the conceptual framework and system design of the DFD software as well as establishing the basis for the functional specification of the DFD software system.

8.3 SUMMARY

The prime objective of this research was to develop a comprehensive concurrent engineering design system, that captures the whole life cycle of the design and product development process, right from elicitation of customer requirements up to production planning and manufacture. The system was also expected to integrate available concurrent engineering design tools and techniques, as well as CAD/CAM/CAE tools which currently exist as islands of automation. Thus providing the designer or design team with a recipe of design tools that can be used at any stage of the design process. A key component of the overall research objective, was that the resulting system should enable and support the integration of the quality function into engineering design.

To address these objectives, three principal research areas were investigated, that is, Quality Function Deployment (QFD), Design philosophies, models, methods and systems and Concurrent Engineering. From the research into these three areas, useful features in relation to the research objectives, were identified and the most important ones extracted and considered in the development of DFD. The requirements for the desired concurrent engineering design system, was a key development in the whole research programme, as they provided the necessary guide and focus in the development of Design Function Deployment (DFD.

The thesis has clearly demonstrated that DFD has emerged as an innovative design paradigm and system for today's modern manufacturing industry. A detailed discussion on what it is, its goals, structure, components as well as the exciting capabilities and potentials, has been done. DFD is currently being applied to a number of in-house and industrial design case studies. These include (i) the design of a new generation wheelchair, (ii) the design of a riveting machine, (iii) the design of a log splitter, (iv) the design of a portable loo, (v) the development of specifications for a an invented electric bulb remover and (vi) the design of a collapsible bicycle. The experience as well as the feedback to date from the case studies, have been very encouraging. A number of research areas have also emerged from this research work. This is related to aspects of (i) integrating both textual and geometric design information (data), (ii) concurrently designing the product and the associated manufacturing and production processes, (iii) linking the design process concurrently with the manufacturing process planning, modelling and simulation, (iv) exploring the use of DFD in capturing designer's intent, knowledge, rationale for decisions as well as the history of the evolving design artifact, and (v) modelling and reengineering the whole manufacturing enterprise incorporating business, design, manufacture and other support processes in the organisation.

The objectives of this research have been fully satisfied, with the conceptual developement of DFD reaching a mature stage. Subsequent efforts are now geared towards the software implementation of the DFD system. In this regard, chapter 7 of the thesis has described a system architecture for the software implementation of DFD. This acts as the preliminary functional specifications for the eventual DFD design system software.

8.4 RECOMMENDATIONS FOR FUTURE WORK

(1) A logical progression of this work will be the software implementation and development of the DFD system. The use of modern approaches to software development particularly to support collaborative design teams will need to be explored.

(2) Another area for further research is the employment of the DFD system for the development of design case studies, both in-house and industry based. This is expected to give rise to other aspects for further research.

(3) There would also be need to establish and develop the theoretical foundations for the individual supporting design modules, shown in level 2 of the DFD structure diagram.

(4) Subsequent to establishing and developing the theoretical foundations for the DFD level 2 modules, the software implementation within the DFD system, particularly for those for which there are no available third party software, represents further work to be carried out.

(5) Work will also be required to develop more robust approaches to handling the subjective linguistic and alphanumeric (textual) information captured within the DFD charts. The employment of fuzzy sets theory and techniques would be useful in handling the uncertainty and subjectivity of the customers requirements and designers information.

(6) The DFD philosophy and principles have advanced to the stage where its concepts can be extended to other areas of design and engineering. The transfer of the knowledge and technology of the DFD system to other disciplines is considered to require further development. An example is shown in Figure 8.1, relating to life cycle design and construction which represents an adaptation of the concepts of DFD to the civil/structural engineering domain.

(7) The software implementation of the DFD system will also represent a major research and software development challenge.

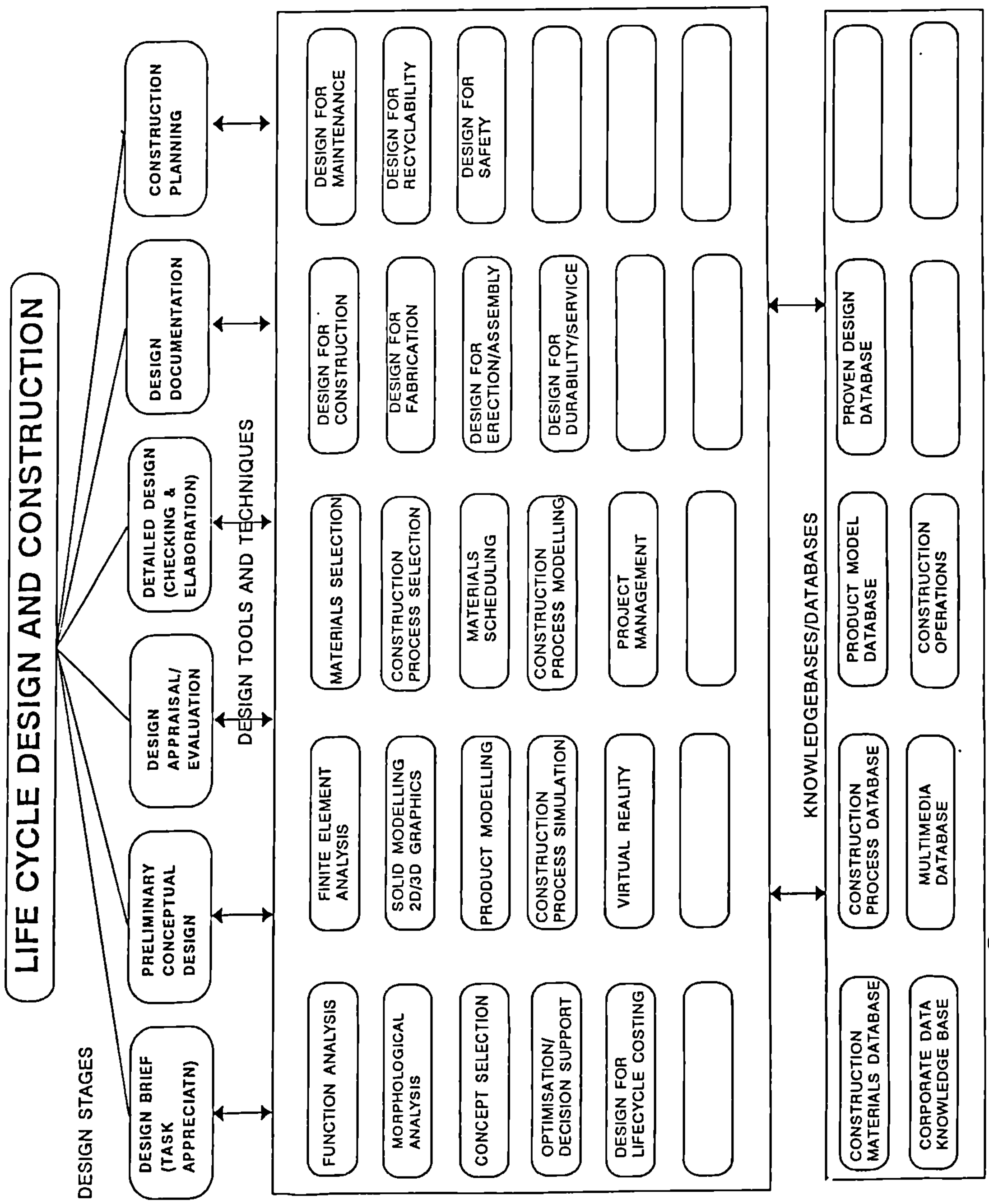


Figure 8.1 The Structure of Life Cycle Design and Construction System

REFERENCES

[1] Edinburgh, Duke of, " The Engineer in Commonwealth Development ", (Seventh Graham Clark Lecture), Institution of Mechanical Engineers, London, 1961.

[2] The Design Council, " Engineering Design Newsletter ", Issue No. 2, January, 1991.

[3] Vesey, J. T., "The New Competitors Think in Terms of 'Speed-to-Market'", Advanced Management Journal, Autumn 1991, pp 26 - 33.

[4] Winner, R. I., et al, "The Role of Concurrent Engineering in Weapons System Acquisition, IDA Report R-338, Institute of Defense Analysis, Alexandria, Virginia, USA, 1988.

[5] Cleetus, K. J., "Definition of Concurrent Engineering", CERC Technical Report Series, Research Note, CERC-TR-RN-92-003, Concurrent Engineering Research Centre, West Virginia University, Morgantown, USA, 1992, pp 1 - 5.

[6] Sivaloganathan, S & Evbuomwan, N. F. O., "A Survey on Design Philosophies, Models and Methods", City University Engineering Design Centre Technical Report No. 63, 1992.

[7] Besant, C. B and Lui, C. W. K., "Computer-Aided Design and Manufacture", 3rd Edition, Published by Ellis Horwood Ltd., West Sussex, England, 1986.

[8] Sullivan, L. P., "Quality Function Deployment", Quality Progress, June, 1998, pp 39 - 50.

[9] Hubka, V., "Design for Quality and Design Methodology", Journal of Engineering Design, Vol. 3, No. 1, 1992, pp 5 - 15.

[10] Hales, C., "Analysis of The Engineering Design Process in an Industrial Context", Ph.D Thesis, Department of Engineering, University of Cambridge, Published by Gants Hill Publications, Hampshire, England, 1987.

[11] Miles, L. D., "Techniques of Value Analysis and Engineering, 2nd Edition, Published by McGraw-Hill Co., New York, USA, 1972.

[12] Vinson, W. D. & Heany, D. F., "Is Quality Out of Control", Harvard Business Review, November-December, 1977.

[13] Kepner, C. H. & Tregoe, B. B., "The Rational Manager : A Systematic Approach to Problem Solving and Decision Making", Kepner-Tregoe Inc., Princeton, New Jersey, USA, 1965.

[14] Warfield, J. N. & Hill, J. D., "A Unified Systems Engineering Concept", Battelle Monograph No. 1, Battelle Memorial Institute, June, 1972.

[15] Hauser, J. R. & Clausing, D. P., "The House of Quality", Harvard Business Review, May - June 1988, pp 63 - 73.

[16] Akao, Y., "Quality Function Deployment - Integrating Customer Requirements", Productivity Press, Massachusetts, USA, 1990.

[17] King, R., "Better Designs in Half the Time", Published by GOAL/QPC, 3rd Edition, Methuen, Massachusetts, USA, 1989.

[18] Hales, R. F.,"Quality Function Deployment in Concurrent Product/Process Development", 6th Annual IEEE Symposium on Computer-Based Medical Systems, IEEE Computer Society Press, Piscataway, NJ, USA, 1993, pp 28 - 33.

[19] Mazur, G., "Quality Function Deployment for a Medical Device", 6th Annual IEEE Symposium on Computer-Based Medical Systems, IEEE Computer Society Press, Piscataway, NJ, USA, 1993, pp 22 - 27.

[20] Clausing, D. P. & Pugh, S., "Enhanced Quality Function Deployment", Proceedings of the Design and Productivity International Conference, Honolulu, Hawai, February 6 - 8, 1991, pp 15 - 25.

[21] Steiner, M. W., "In Search of QFD", 3rd International Conference on Design Theory and Methodology, The American Society of Mechanical Engineers, DE-Vol. 31, September 22 - 25, 1991, pp 191 - 194.

[22] Sontow, K. & Clausing, D. P., "Integration of Quality Function Deployment with Further Methods of Quality Planning", Working Paper No. LMP-93-005, Laboratory for Manufacturing and Productivity, MIT, Cambridge, Massachusetts, April, 1993.

[23] Mallon. J. C. & Mulligan, D. E., "Quality Function Deployment - A System for Meeting Customers' Needs", Journal of Construction Engineering and Management, Vol. 119, No. 3, September 1993, pp 516 - 531.

[24] Clausing, D. P., "Total Quality Development - The Development of Competitive New Products", Published by The American Society of Mechanical Engineers Press, New York, USA, March, 1994.

[25] Hauser, J. R., "How Puritan-Bennett Used the House of Quality", Sloan Management Review, Spring 1993, No. 34, pp 61 - 70.

[26] Quality Function Deployment, Implementation Manual for Three Day QFD Workshop, American Supplier Institute Inc., Dearborn, MI, USA, 1987.

[27] Pugh, S., "Concept Selection - A Method that Works", Proceedings of the International Conference on Engineering Design, Rome, March 9 - 13, WKD 5 Paper M3/16, 1981, pp 497 - 506.

[28] Clausing, D. P., "Concurrent Engineering", Proceedings of the Design and Productivity International Conference, Honolulu, Hawai, February 6-8, 1991.

[29] Pugh, S., "Total Design - Integrated Methods for Successful Product Engineering", Published by Addison-Wesley, Publishers Ltd, UK, 1991.

[30] Akiyama, K., "Functional Analysis", Productivity Press, Cambridge, Massachusetts, USA, 1991.

[31] Clausing, D. P., "Quality Function Deployment : Applied Systems Engineering", Quality and Productivity Research Conference, University of Waterloo, June 6, 1989.

[32] Pandey, A., "Quality Function Deployment : A Study of Implementation and Enhancements", M.Sc Dissertation, Sloan School of Management, MIT, Cambridge, Massachusetts, USA, July 1992.

[33] Krinninger, A. & Clausing, D. P., "Quality Function Deployment for Production", Working Paper, Laboratory for Manufacturing and Productivity, MIT, Cambridge, Massachusetts, August 1991.

[34] Warfield, J. N., "Structuring Complex Problems", Battelle Monograph, April, 1974.

[35] ter Haar, S., Clausing, D. P. & Eppinger, S. D., "Integration of Quality Function Deployment and the Design Structure Matrix", Working Paper No. LMP-93-004, Laboratory for Manufacturing and Productivity, MIT, Cambridge, Massachusetts, March, 1993.

[36] Locascio, A. & Thurston, D. L., "Multiattribute Design Optimisation with Quality Function Deployment", Proceedings of the 2nd Industrial Engineering Research Conference, Mitta, D. A. et al (eds), Industrial Engineering and Management Press, Norcross, GA, USA, 1993, pp 82 - 86.

[37] Masud, A. S. M. & Dean, E. B., "Using Fuzzy Sets in Quality Function Deployment", Proceedings of the 2nd Industrial Engineering Research Conference, Mitta, D. A. et al (eds), Industrial Engineering and Management Press, Norcross, GA, USA, 1993, pp 270 - 274.

[38] Dean, E. B., "Quality Function Deployment for Large Systems", International Engineering Management Conference, Management In a Global Environment, Sheraton Hotel & Conference Centre, Eatontown, New Jersey, October 25 - 28, 1992, pp 317 - 321.

[39] Maddux, G. A. et al, "Organizations Can Apply Quality Function Deployment As Strategic Planning Tool", Industrial Engineering, September 1991, pp 33-37.

[40] Maduri, O., "Design Planning of an Off-Highway Dump Truck - A QFD Approach", Golden Conference on Quality Through Engineering Design, Kuo, W. (ed), Elsevier Publishers, BV Holland, 1993, pp 97 - 104.

[41] Hochman, S. D., & O' Connell, P. A., "Quality Function Deployment - Using the Customer to Outperform the Competition on Environmental Design", Proceedings of the 1993 IEEE International Symposium on Electronics and the Environment, Published by IEEE NY, USA, 1993, pp 165 - 172.

[42] Halbleib, L et al, "Application of Quality Function Deployment to the Design of a Lithium Battery", Proceedings of the 43rd Electronic Components & Technology Conference, Published by IEEE NY, USA, 1993, pp 150 -154.

[43] Quality Function Deployment - Awareness Seminar, ASI Quality Systems, Challenge House, Sherwood Drive, Bletchley, Milton Keynes MK3 6DR,1992.

[44] Davison, S. and Jones, D., "QFD and Taguchi Methods in a Process Environment", European Taguchi Methods and QFD Symposium, ASI, May, 1992.

[45] Nichols, K., "The Best Designs Start With The Customer", 2nd European Symposium (Taguchi Methods), ASI, December, 1990.

[46] De Vera, D. et al, "An Automotive Case Study", Quality Progress, June, 1988, pp 35 - 38.

[47] Sullivan, L. P., "Policy Management Through Quality Function Deployment", Quality Progress, June, 1988, pp 18 - 20.

[48] Tokunaga, I., "TQC at Reports of Statistical Application Research", JUSE, Vol. 35, No. 1, 1988, pp 11 - 23.

[49] Snodgrass, T. J., "Quality Function Deployment Versus Value Information Techniques Using Analytical Language", Proceedings of the Society of American Value Engineers, Vol. 24, 1989, pp 99 - 100.

[50] Kogure, M. and Akao, Y., "Quality Function Deployment and CWQC in Japan", Quality Progress, October, 1983, pp 25 - 29.

[51] Jones, J. C. & Thornley, D. G., "Conference on Design Methods", Pergamon Press Ltd, London, September 1962.

[52] Gregory, S. A., "The Design Method", Butterworth & Co Ltd, London, 1966.

[53] Broadbent, G. H. & Ward, A, "Design Methods in Architecture", Lund Humphries, London, 1969.

[54] Design : Science : Method, Proceedings of the 1980 Design Research Society Conference, Jacques, R & Powell, J. A. (eds), Published by Westbury House, Guildford, Surrey, England, 1981.

[55] Proceedings of the International Conference on Engineering Design, ICED 89, Harrogate, 1989, WDK 18, Published by IMechE, UK, Vols 1 & 2.

[56] Proceedings of the International Conference on Engineering Design, ICED 81, Rome, 1981, WDK 5.

[57] Proceedings of the International Conference on Engineering Design, ICED 83, Kobenhavn, 1983, WDK 10.

[58] Proceedings of the International Conference on Engineering Design, ICED 85, Hamburg, 1985, WDK 12.

[59] Proceedings of the International Conference on Engineering Design, ICED 91, Zurich, 1991, WDK 20.

[60] Proceedings of the International Conference on Engineering Design, ICED 93, August 17 - 19, 1993, The Hague, 1993, WDK 22, Roozenburg N. F. M. (ed.), Vols 1, 2 & 3.

[61] Proceedings of the International Conference on Engineering Design, ICED 87, Boston, 1987, WDK 13, Eder E. W. (ed.), Vols 1 & 2.

[62] Proceedings of the 1988 NSF Grantee Workshop on Design Theory and Methodology, Design Theory '88, Newsome, S. L., Spillers, W. R. & Finger, S., (eds), Springer-Verlag, New York, 1989.

[63] Proceedings of the 1st International Conference on Design Theory and Methodology, Montreal, Canada, ASME, DE Vol. 17, September, 1989.

[64] Proceedings of the 2nd International Conference on Design Theory and Methodology, Chicago, Illinois, USA, Rinderle, J. (ed.), ASME, DE Vol. 27, September 16 - 19, 1990.

[65] Proceedings of the 3rd International Conference on Design Theory and Methodology, Miami, Florida, USA, Stauffer, L. A. (ed.), ASME, DE Vol. 31, September 22 - 25, 1991.

[66] Proceedings of the 4th International Conference on Design Theory and Methodology, Scottsdale, Arizona, USA, Taylor, D. L. & Stauffer, L. A. (eds.), ASME, DE Vol. 42, September 13 - 16, 1992.

[67] Proceedings of the 5th International Conference on Design Theory and Methodology, Albuquerque, New Mexico, USA, Stauffer, L. A. (ed.), ASME, DE Vol. 53, September 19 - 22, 1993.

[68] Proceedings of the 6th International Conference on Design Theory and Methodology, Minneapolis, Minnesota, USA, Stauffer, L. A. (ed.), ASME, September 11 - 14, 1994.

[69] Feilden, G. B. R. et al, " Engineering Design ", (' The Feilden's Report '), Her Majesty's Stationery Office, London, 1963.

[70] Feilden, G. B. R. " The 'Leverage' of Design ", The Practice of and Education forEngineering Design ", Proceedings of The Institution of Mechanical Engineers, Vol. 178, Part 3B, 1963-64, pp 7 - 10.

[71] Wallace, K. M., "Engineering Tripos Notes", Cambridge University, 1986.

[72] Hillier, W., Musgrove, J., & O' Sullivan, P., " Knowledge and Design ", in Mitchell, W. J. (ed) " Environmental Design : Research and Practice ", University of California, 1972.

[73] Darke, J., " The Primary Generator and The Design Process ", in Cross, N. (ed) " Developments in Design Methodolgy ", John Wiley & Sons Ltd., London, 1984, pp 175 - 188.

[74] Finkelstein, L & Finkelstein, A. C. W., " Review of Design Methodology", IEE Proceedings, Vol. 130, Part A, No. 4, June 1983.

[75] Luckman, J., " An Approach to the Management of Design ", in Cross, N.(ed) "Developments in Design Methodolgy ", John Wiley & Sons Ltd., London, 1984, pp 83 - 97.

[76] Archer, L. B., " Systematic Method for Designers ", in Cross, N. (ed) "Developments in Design Methodolgy ", John Wiley & Sons Ltd., London, 1984, pp 57 - 82.

[77] El Maraghy, H. A. & Chen, S., " A General Model for Mechanical Design ", Annals of the CIRP, Vol. 39 No. 1, 1990, pp 111 - 116.

[78] Gasson, P. C., "Theory of Design ", Published by Batsford Ltd., London, 1st Edition, 1974.

[79] Rabins, M. et al, "Design Theory and Methodology - A New Discipline", Mechanical Engineering, August 1986, pp 23 - 27.

[80] Caldecote, V., "The Design Team in Relation to The Individual Designer", The Practice of and Education for Engineering Design ", Proceedings of The Institution of Mechanical Engineers, Vol. 178, Part 3B, 1963-64, pp 16 - 19.

[81] La Rota, J. L. et al, "A Model-Based Approach to Structural Design", Applications of Artificial Intelligence in Engineering V, Vol. 1 : Design, Gero, J. S. (ed), Springer-Verlag, UK, pp 3 - 22.

[82] Chandrasekaran, B. (Gero, J. S. (ed)), "A Framework for Design Problem Solving", Workshop on Research Directions for Artificial Intelligence in Design", Standford University, March 1989, (Organised by the Design Computing Unit, University of Sydney, Australia).

[83] Suh, N., " The Principles of Design " Oxford University Press, N.Y, USA,1988.

[84] Rzevski, G., "On the Design of a Design Methodology", Design : Science : Method, Proceedings of the 1980 Design Research Society Conference, Jacques, R & Powell, J. A. (eds), Published by Westbury House,Guildford, Surrey, England, 1981, pp 6 - 17.

[85] Banares-Alcantara, R., "Representing the Engineering Design Process: Two Hypotheses", Artificial Intelligence in Design Conference, Gero (ed), Butterworth-Heinemann, Oxford, 1991, pp 3 - 22.

[86] Gero, J. S., "A System for Computer-Aided Design in Architecture", in Computer Aided Design, Viletstra, J and Wielinga, R.F. (eds), North Holland Co., Amsterdam, 1973.

[87] Smithers, T., "AI-Based Design versus Geometry-Based Design, or Why Design Cannot be Supported by Geometry Alone", Computer Aided Design, Vol. 21, No. 3, 1989, pp 141 - 150.

[88] Goals and Priorities for Research in Engineering Design, American Society of Mechanical Engineers, New York, USA, 1986.

[89] Cross, N., " Developments in Design Methodolgy ", John Wiley & Sons Ltd., London, 1984.

[90] Eder, W. E., " Definitions and Methodoligies ", in "The Design Method", Gregory, S. A. (ed) , Butterworth & Co Ltd, London, 1966, pp 19 - 31.

[91] Juster, N. P., "The Design Process and Design Methodologies", Technical Report, University of Leeds, May, 1985.

[92] Cagan, J. and Agogino, A. M., "Dimensional Variable Expansion - A Formal Approach to Innovative Design", Research in Engineering Design, Vol. 3, 1991, pp 75 - 85.

[93] Sriram, D., et al, "Knowledge-Based System Applications in Engineering Design: Research at MIT", AI Magazine, Vol. 10, No. 3, Fall, 1989, pp 79-96.

[94] Pahl, G. & Beitz, W., " Engineering Design ", Edited by Wallace, K., 1st Edition, The Design Council, London, 1984.

[95] Medland, A. J., "The Computer Based Design Process", Kogan Page, London, 1986.

[96] Mostow, J., "Toward Better Models of the Design Process", The AI Magazine, Spring, 1985, pp 44 - 57.

[97] Mittal, S. and Araya, A., "A Knowledge-Based Framework for Design", 5th National Conference on Artificial Intelligence, AAAI-86, August 11 - 15, 1986, Philadelphia, PA, USA, Vol. 2 : Engineering, pp 856 - 865.

[98] Broadbent, G. H., " Design Methods - 13 Years after - A Review ", in Jacques, R & Powell, J. A. (eds), " Design : Science : Method ", Westbury House, Guildford, U.K., 1980, pp 3 - 5.

[99] Finger, S. & Dixon, J. R., " A Review of Research in Mechanical Engineering Design. Part 1 : Descriptive, Prescriptive, and Computer-Based Models of Design Processes ", Research in Engineering Design, Vol. 1, 1989, pp 51 - 67.

[100] Cross, N., " Engineering Design Methods ", John Wiley & Sons Ltd., London 1991.

[101] Lawson, B. R., " Cognitive Strategies in Architechtural Design ", in Cross, N. (ed) " Developments in Design Methodolgy ", John Wiley & Sons Ltd., London, 1984, pp 209 - 220.

[102] Yoshikawa, H., " Design Philosophy : The State of the Art ", Annals of the CIRP, Vol. 38, No. 2, 1989, pp 579 - 586.

[103] Rodenacker, W. G., " Methodisches Konstruieren ", Springer - Verlag, 1970.

[104] Wallace, K. M., "Better By Design", Manufacturing Engineer, May, 1989, pp 35 - 36.

[105] Asimow, M., " Introduction to Design ", Prentice Hall Inc., USA, 1962.

[106] Watts, R. D., " The Elements of Design ", in "The Design Method", Gregory, S. A. (ed), Butterworth & Co Ltd, London, 1966., pp 85 - 95.

[107] Marples, D. L., " The Decisions of Engineering Design ", The Institution of Engineering Designers, London, 1960, pp 1 - 16.

[108] Krick, E. V., " Engineering & Engineering Design ", 2nd Edition, John Wiley & Sons Ltd., USA, 1969.

[109] Hubka, V., " Design for Quality and Design Methodology ", Journal of Engineering Design, Vol. 3, No. 1, 1992, pp 5 - 15.

[110] Seireg, A., " Methodology of Mechanical Systems Design ", International Conference on Engineering Design, (ICED), Rome, Italy, March, 1981, pp 309 - 322.

[111] French, M. J., " Conceptual Design for Engineers ", 2nd Edition, The Design Council, London, 1985.

[112] Harris, A. J., " Can Design be Taught ", Proceedings of The Institution of Civil Engineers, Part 1, No. 68, August, 1980, pp 409 - 416.

[113] Pugh, S., "Total Design - Integrated Methods for Successful Product Engineering", First Edition, Addison-Wesley Publishers Ltd., England, 1990.

[114] Taguchi, G., " Introduction to Quality Engineering : Designing Quality into Products and Processes, Asian Productivity Organisation, Tokyo, Japan, 1986.

[115] Vasseur, H. et al, "Quality Innovation in Design and Manufacturing : An Economic Model", Manufacturing Review, December, 1993.

[116] Sanchez, S. M. et al, "Quality by Design", Chapter 10, in "Concurrent Engineering : Automation, Tools and Techniques", Kusiak, A. (ed), Published by John Wiley and Sons Inc, NY, USA, 1993, pp 235 - 286.

[117] Ertas, A. and Jones, J. C., "The Engineering Design Process", Published by John Wiley and Sons, Inc.,NY, USA, 1993.

[118] Ullman, D. G. & Diettrich, T. A., " Mechanical Design Methodology ", Computers in Engineering, American Society of Mechanical Engineers, New York, 1986, pp 173 - 180.

[119] Ullman, D. G. Stauffer, L. A., & Diettrich, T. A., " Preliminary Results of an Experimental Study of the Mechanical Design Process ", Proceedings from the NSF Workshop on the Design Process, Waldron, M. B., ed., Ohio State University, Oakland, CA, 1987, pp 145 - 188.

[120] March, L., " The Logic of Design ", in Cross, N. (ed) " Developments in Design Methodolgy ", John Wiley & Sons Ltd., London, 1984, pp 265 - 276.

[121] Matchett, E. and Briggs, A. H., "Practical Design Based on Method (Fundamental Design Method), in "The Design Method", Gregory, S. A.,(ed) Butterworth & Co Ltd, London, 1966, pp 183 - 199.

[122] Jones, J. C., " Design Methods - Seeds of Human Futures ", John Wiley & Sons Ltd., UK, 1970.

[123] Hybs, I. and Gero, J. S., "An Evolutionary Process Model of Design", Design Studies, Vol. 13, No. 3, July, 1992, pp 273 - 290.

[124] Neville, G. E., "Computational Models of Design Processes", in Proceedings of the 1988 NSF Grantee Workshop on Deisgn Theory and Methodology, Design Theory '88, Newsome, S. L., Spillers, W. R. & Finger, S., (eds), Springer-Verlag, New York, USA, 1989, pp 82 - 96.

[125] Agogino, A. M. et al, "AI/OR Computational Model for Integrating Qualitative and Quantitative Design Methods", Proceedings : NSF Engineering in Design Conference, University of Massachusetts, Amherst, MA, USA, June 11- 14, 1989, pp 97 - 112.

[126] Cagan, J. and Agogino, A. M., "Innovative Design of Mechanical Structures from First Principles", AI EDAM, Vol. 1, No. 3, pp 169 - 189.

[127] Mostow, J. et al, " A Domain-Independent Model of Knowledge-Based Design", in Proceedings of the 1988 NSF Grantee Workshop on Deisgn Theory and Methodology, Design Theory '88, Newsome, S. L., Spillers, W. R. & Finger, S., (eds), Springer-Verlag, New York, USA, 1989, pp 97 - 99.

[128] Dixon, J. R., "Iterative Redesign and Respecification : Research on Computational Models of Design Processes", in Proceedings of the 1988 NSF Grantee Workshop on Deisgn Theory and Methodology, Design Theory '88, Newsome, S. L., Spillers, W. R. & Finger, S., (eds), Springer-Verlag, New York, USA, 1989, pp 104 - 107.

[129] Gero, J. S., "AI and New Computational Models of Design", International Association for Bridge and Structural Engineering (IABSE) Colloquium, on Knowledge-Based Systems in Civil Engineering, Beijing, China, 1993, pp 25-32.

[130] Hubka, V., "DesignTactics = Methods + Working Principles for Design Engineers", International Conference on Engineering Design, ICED, Kobenhavn, 15 - 18 August, 1983, pp 16 - 34.

[131] Chawdhry, P. K., "Integration and Management of Concurrent Design Activities", Proceedings of the International Conference on Engineering Design, ICED 93, August 17 - 19, 1993, The Hague, 1993, WDK 22, Roozenburg N. F. M. (ed.), Vol. 3, pp 1481 - 1485.

[132] de Kleer, J., "An Assumption-Based Truth Maintenance System", Artificial Intelligence, Vol. 28, 1986, pp 127 - 162.

[133] Ross, P., "A Simple ATMS", Dept. of Artificial Intelligence Internal Report, University of Edinburgh, May, 1989.

[134] Bauert, F. and Ball, N. R., "The Integrated Design Framework : Blackboards as a Communication Medium between Designers and Software Medium", Proceedings of the International Conference on Engineering Design, ICED 93, August 17 - 19, 1993, The Hague, 1993, WDK 22, Roozenburg N. F. M. (ed.), Vol. 3, pp 1351 - 1358.

[135] U.S. Air Force, "Integrated Computer-Aided Manufacturing (ICAM) Architecture Part II, Volume IV - Function Modelling Manual (IDEFO)", Air Force Materials Laboratory, Wright-Patterson AFB, Ohio 45433, AFWAL-TR-81-4023, June, 1981.

[136] Bracewell, R. H. et al, "Schemebuilder, A Design Aid for the Conceptual Stages of Product Design", Proceedings of the International Conference on Engineering Design, ICED 93, August 17 - 19, 1993, The Hague, 1993, WDK 22, Roozenburg N. F. M. (ed.), Vol. 3, pp 1311 - 1318.

[137] Oh, V. K. and Sharpe, J. E. E., "Integrated Computer Support for Interdisciplinary System Design", in "Methodologies, Techniques, and Tools for Design Development", Vol. 5, PD-Vol. 64-5, ASME, Tanik, M. M. et al (eds), Proceedings of the 2nd Biennial European Joint Conference on Engineering Systems Design and Analysis, July 4 - 7, 1994, London, pp 107 - 114.

[138] Gillam, A., "Vehicle Knowledge-Based Design Environment", Journal of Spacecraft and Rockets, Vol. 30, No. 3, May-June, 1993, pp 342 - 347.

[139] Beitz, W. and Feldhusen, J., "Management Systems and Program Concepts for an Integrated CAD Process", Research in Engineering Design, Vol. 3, 1991, pp 61 - 74.

[140] Meerkamm, H., "Design System MFK - An Important Step Towards an Engineering Workbench", Journal of Engineering Manufacture, Part B, Vol. 207, Proceedings of the Institution of Mechanical Engineers, 1993, pp 105 - 116.

[141] Akman, V. et al, "A Fundamental and Theoretical Framework for an Intelligent CAD System", Computer Aided Design, Vol. 22, No. 6, July/August, 1990, pp 352 - 367.

[142] Finger, S. et al, "Design Support Systems for Concurrent Engineering : A Case Study in Large Power Transformer Design", Proceedings of the International Conference on Engineering Design, ICED 93, August 17 - 19, 1993, The Hague,WDK 22, Roozenburg N. F. M. (ed.), Vol. 3, pp 1433-1440.

[143] Finger, S., Fox, M. S., Prinz, F. B. and Rinderle, J. R., "Concurrent Design", Applied Artificial Intelligence, Vol. 6, No. 3, 1992, pp 257 - 283.

[144] Finger, S. & Dixon, J. R., " A Review of Research in Mechanical Engineering Design. Part 2 : Representations, Analysis, and Design for the Life Cycle", Research in Engineering Design, Vol. 1, 1989, pp 121 - 137.

[145] Hartley J. R., "Concurrent Engineering", Productivity Press, Cambridge, Massachusetts, 1990.

[146] Bedworth D. D.,Henderson M. R., Wolfe M. P., "Computer - Integrated Design and Manufacturing", McGraw Hill, International Edition, Singapore, 1991.

[147] Broughton T., "Simultaneous Engineering in Aero Gas Turbine Design and Manufacture", Proceedings of the First International Conference on Simultaneous Engineering, 4-5 December 1990, London UK, pp 25 - 36.

[148] Eversheim W., "Trends and experience in applying Simultaneous Engineering in Germany", Proceedings of the First International Conference on Simultaneous Engineering, 4-5 December 1990, London UK, pp 5 - 23.

[149] Brookes and Gatehouse Limited, Seminar on Simultaneous Engineering, Organised by the Management and Design Division, Institution of Electrical Engineers, Savoy Place, London, 8th October, 1992.

[150] Stephanon S. E. and Spiegl F., "The Manufacturing Challenge", Van Nostrand Reinhold, New York, 1992.

[151] Ellis, L. W., "Practices in Concurrent Engineering", International Engineering Management Conference - Managing in a Global Environment, Eatontown, New Jersey, October 25 - 28, 1992, pp 115 - 118.

[152] Lake, J. G., "A Program Design for Concurrent Engineering Education", International Engineering Management Conference - Managing in a Global Environment, Eatontown, New Jersey, October 25 - 28, 1992, pp 71 - 76.

[153] Ramana Reddy, Y. V. et al, "Computer Support for Concurrent Engineering", IEEE Computer, January, 1993, pp 12 - 16.

[154] Kannapan, S. M. & Marshek, K. M., "A Schema for Negotiation Between Intelligent Design Agents in Concurrent Engineering", Intelligent Computer Aided Design, Brown, D. C. et al (eds), Vol. 4, Chap 36, Elsevier Science Publihsers BV (North Holland), 1992, pp 1-25.

[155] Knight, S and Jackson, J., "Simultaneous Engineering Saves Manufacturer's Lead Time, Costs and Frustration", Industrial Engineering, Vol. 21, No. 8, August 1989, pp 25 - 27.

[156] Walklet, R., "Continuous Improvement and Simultaneous Engineering", Automotive Engineering, Vol. 97, No. 10, October 1989, pp 59 - 63.

[157] Krishnaswamy, G. M. and Elshennawy, A. K., "Concurrent Engineering Deployment : An Enhanced 'Customer Product Approach'", Computers and Industrial Engineering, Vol. 23, Nos. 1-4, 1992, pp 503 - 506.

[158] Nevins, J. L. and Whitney, D. E., "Concurrent Design of Products and Processes", Defazio, T. L. et al (eds), McGraw-Hill, New York, USA, 1989.

[159] Gould P., "Concurrent Engineering for Survival", Professional Engineering, July/August 1992.

[160] Mital A. and Morse E. I., "The Role of Ergonomics in Designing for Manufacturability", in the book "Design for Manufacturability", Halende M. and Nagamachi M. (Eds), The International Ergonomics Association, Taylor & Francis 1992.

[161] Garret, R. W., "Eight Steps to Simultaneous Engineering", Manufacturing Engineering, November 1990, pp 41 - 47.

[162] Brazier D. and Leonard M., "Concurrent Engineering: Participating in Better Designs", Mechanical Engineering, January 1990, pp 52 - 53.

[163] Yeh, R. T., "Notes on Concurrent Engineering", IEEE Transactions on Knowledge and Data Engineering, Vol. 4, No. 5, October 1992, pp 407 - 414.

[164] Smith, D., "Simultaneous/Concurrent Engineering, the Systems Thinking Approach to Total Life Cycle Design", Concurrent Engineering & CALS Washington '92 Conference and Expo., June 1 - 4, 1992, Society of Computer Aided Engineering, pp 613 - 625.

[165] Shina S. G., "New Rules for World Class Companies", Concurrent Engineering - Special Report, IEEE Spectrum, July 1991, pp 23 - 26.

[166] Medhat, Sa'ad, "Simultaneous Engineering - 10 Steps for Implementation", Simultaneous Engineering Newsletter, Management and Design Division, Institution of Electrical Engineers, Savoy Place, London, April 1993, pp 3-4.

[167] Jebb, A, Sivaloganathan, S and Evbuomwan, N. F. O., "Simultaneous Engineering", City University Engineering Design Centre, Technical Report No. 66, 1992, pp 1- 15.

[168] McDonnel Douglas Corporation, "Concurrent Engineering Handbook", 2nd Edition, Published by McDonnel Douglas Missile Systems Company, 1992.

[169] Neave, H. R., "The Deming Dimension", SPC Press Inc, Knoxville, Tennessee, USA, 1990.

[170] Mizuno, S., "Management for Quality Improvement", Productivity Press, Cambridge 1988.

[171] Goldstein G., "Integrating Product and Process Design", In Design for Manufacture, Corbett et.al (eds), Addison-Wesley, 1991, pp 92 - 97.

[172] Whitney D. E., "Manufacturing by design", In Design for Manufacture, Corbett et. al (eds), Addison-Wesley, 1991, pp 37 - 48.

[173] Nadler G., "Design Teams: Breakthroughs for Effectiveness", Design Management Journal, Spring 1991, pp 10 - 19.

[174] Fotta, M. E. & Daley, R. A., "Improving Interpersonal Communications on Multifunctional Teams", *Handbook of Concurrent Engineering", Parsei, H. R.* & Sullivan, W. G. (eds), Chapman & Hall, London, 1993.

[175] Mortimer J. and Hartley J., "Simultaneous Engineering", The DTI, Enterprise Initiative, 1990.

[176] Hartley J. and Mortimer J., "Simultaneous Engineering", Industrial Newsletters Ltd, Dunstable, Beds LU5 6BS, UK, 1990.

[177] Clealand, D. I. and Bursic, K. M., "Strategic Technology Management - Systems for Products and Processes", American Management Association, New York, USA, 1992.

[178] Yang, H., Lin, A. & Lu, W., "Intelligent Manufacturing Planning with Learning Capability in a Concurrent Engineering Environment", Concurrent Engineering & CALS Washington '92 Conference and Expo., June 1 - 4, 1992, Society of Computer Aided Engineering, pp 279 - 292.

[179] Place, D. L., "General Application Tools for Concurrent Engineering", International Engineering Management Conference - Managing in a Global Environment, Eatontown, New Jersey, October 25 - 28, 1992, pp 215 - 218.

[180] Evbuomwan, N. F. O., Sivaloganathan, S & Jebb, A., "Concurrent Design within Design Function Deployment", Proceedings of the 2nd International Conference on Concurrent Engineering & Electronic Design Automation, Bournemouth, UK, April 7-8, 1994, Medhat, S. (ed), pp 167 - 175.

[181] Gebala, D. A. & Eppinger, S. D., "Methods for Analysing Design Procedures", DE-Vol. 31, Design Theory and Methodology, ASME, 1991, pp 227-233.

[182] Eppinger, S. D. et al, "Organising Tasks in Complex Design Projects", 2nd International Conference on Design Theory and Methodology, September 16 - 19, 1990, ASME, Chicago, Illinois, USA, pp 39-46.

[183] Kusiak A. and Chow W. S., An Algorithm for Clustering Identification, IEEE Transactions on Systems, Man. and Cybernetics, SMC-17(4), 1987, pp 696 - 699.

[184] Kusiak A. and Park K., Concurrent Engineering : Decomposition and Scheduling of design activities, International Journal of Production Research, Vol 28, No 10, 1990, pp 1883 - 1900.

[185] Kusiak, A. and Wang, J., "Concurrent Engineering : Simplification of the Design Process", Computer Aaplications in Production and Engineering : Integration Aspects, Edited by Doumeingts, G et al, Elsevier Science Publishers, B. V., North Holland, IFIP 1991, pp 297 - 304.

[186] Wade, D, "Database Management for Concurrent Engineering", High Performance Systems, April 1990, pp 49 - 56.

[187] Lewis, J, "An Approach to Applications Integration for Concurrent Engineering", Proceedings of the Second National Symposium on Concurrent Engineering, 7 - 9 February, 1990, West Virginia University, Morgantown, USA, pp 141 - 153.

[188] Sriram, R., Perkins, W. A. & Livezey, B. K., "A Distributed Shared Database System for Concurrent Engineering", Concurrent Engineering & CALS Washington '92 Conference and Expo., June 1 - 4, 1992, Society of Computer Aided Engineering.

[189] Hills, P., "Simultaneous Engineering", Seminar on Simultaneous Engineering, Organised by the Management and Design Division, Institution of Electrical Engineers, Savoy Place, London, 8th October, 1992.

[190] Sharon, A., "The Biggest Bottlenecks in Concurrent Engineering", IFIP TC5/WG 5.3/WG 5.2 Working Conference on Manufacturing in the Era of Concurrent Engineering, Herzlya, Israel, 13 - 15 April, 1992, Halevi, G & Weill R. D. (eds), Elsevier Science Publishers, BV (North-Holland),pp 5 - 13.

[191] Kalai, E. and Smorodinsky, M, "Other Solutions to Nash's Bargaining Problem", Econometrica, Vol. 43, No. 3, 1975, pp 513 - 518.

[192] Sobolewski, M. et al, "Knowledge-Based Integration in Concurrent Engineering", in "Concurrent Engineering Approach to Materials Processing", Dwivedi, S. N. et al (eds), Published by Minerals, Metals and Materials Society, Warrendale, PA, USA, pp 39 - 55.

[193] Sriram, D. and Logcher, R., "The MIT Dice Project", IEEE Computer, January, 1993, pp 64 - 65.

[194] Cutkosky, M. R. et al, "PACT : An Experiment in Integrated Concurrent Engineering Systems", IEEE Computer, January, 1993, pp 28 - 37.

[195] Huyn, P. N. et al, "Automated Concurrent Engineering in Designworld", IEEE Computer, January, 1993, pp 74-76.

[196] Bowen, J. & Bahler, D., "Constraint-Based Software for Concurrent Engineering", IEEE Computer, January, 1993, pp 66-68.

[197] Tong, C & Gomory, A., "A Knowledge-Based Computer Environment for the Conceptual Design of Small Electromechanical Appliances",IEEE Computer, January, 1993, pp 69-71.

[198] Gupta, S. K. et al, "Concurrent Evaluation of Machinability During Product Design", IEEE Computer, January, 1993, pp 61-63.

[199] Dewan, P. & Riedl, J., "Toward Computer - Supported Concurrent Software Engineering", IEEE Computer, January, 1993, pp 17-27.

[200] Hutchison, K. & Amundsen, A., "Concurrent Engineering Environment for Electronic Circuit Design", Proceedings of the Second National Symposium on Concurrent Engineering, 7 - 9 February, 1990, West Virginia University, Morgantown, USA, pp 299 - 314.

[201] Quality Function Deployment, Implementation Manual for Three Day QFD Workshop, American Suppliers Institute Inc., Dearborn, Michigan, USA, 1987.

[202] Bossert, J. L., "Quality Function Deployment - A practitioner's Approach", American Society of Quality Control Quality Press, Wisconsin and Marcel Dekker, Inc., NY, USA, 1991.

[203] Jo, H. H. et al, "Design Frameworks for Concurrent Engineering", Computers and Industrial Engineering, Vol. 23, Nos. 1 - 4, 1992, pp 11 - 14.

[204] Graham, A., "The CAD Framework Initiative", Transactions of IEEE, Design and Test of Computers, September, 1991, pp 12 - 15.

[205] Rzevski, G., "The Integrated Product Development Process : Issues and Methods", Proceedings of the International Conference on Engineering Design, ICED '93, The Hague, Netherlands, August 17 - 19, 1993, pp 493-498.

[206] Sohlenius, G., "Concurrent Engineering", Annals of the CIRP, 1992, Vol. 41, No. 2, pp 645 - 655

[207] Pham, D. T. and Yang, Y., "A Genetic Algorithm Based Preliminary Design System", Journal of Automobile Engineering, Part D, Vol. 207, Proceedings of the Institution of Mechanical Engineers, 1993, pp 127 - 133.

[208] Sakthivel, T. S. and Kalyanaraman, V., " A KBES for Integrated Engineering", Engineering with Computers, Vol. 9, 1993, pp 1 - 16.

[209] Jebb, A., Sivaloganathan, S. & Evbuomwan, N. F. O., "Design Function Deployment - A Paradigm for Concurrent Engineering", Proceedings of the International Conference on Design to Manufacture in Modern Industry, Bled, Slovenia, June 7 - 9, 1993, Part 1, pp 78 - 85.

[210] Kimura, F. and Suzuki, H., "A CAD System for Efficient Product Design Based on Design Intent", Annals of the CIRP, 1989, Vol. 38, No. 1, pp 149 - 152.

[211] The First CIRP International Workshop on Concurrent Engineering for Product Realisation, Tokyo, Japan, June 27-28, Annals of the CIRP, , 1992, Vol. 41, No. 2, pp 743-746.

[212] Cutkosky, M. R., Tenenbaum, J. M. and Brown, D. R., "Working with Multiple Representations in a Concurrent Design System", Journal of Mechanical Design, Transactions of the ASME, September 1992, Vol. 114, pp 515 - 524.

[213] Sinclair, M. A., Siemieniuch, C. E. and John, P. A., "A User-Centered Approach to Define High-Level Requirements for Next-Generation CAD Systems for Mechanical Engineering", IEEE Transactions on Engineering Management, Vol. 36, No. 4, November 1989, pp 262 - 270.

[214] Dwivedi, S. N. and Sobolewski, M., "Concurrent Engineering - An Introduction", CERC Technical Report Series, CERC-TR-TM-90-006, Concurrent Engineering Research Centre, West Virginia University, Morgantown, USA, 1990.

[215] Jebb, A. et al., "Design Function Deployment", Proceedings of the 1st Joint European Conference on Engineering Systems Design and Analysis, Istanbul, Turkey, ASME, PD-Vol. 47-1, Vol. 1, 1992, pp 251 - 256.

[216] Jebb, A. et al, "Design Function Deployment - A Platform for Cross Disciplinary Product Development", Mechatronics - The Integration of Engineering Design, IMechE Conference on Mechatronics, University of Dundee, September, 1992, pp 23 - 28.

[217 Smith, G. L., "A Model for Designing", Industrial Engineering, November, 1989, pp 35 - 38.

[218] Grant, D.P., "How to Construct a Morphological Box", Journal of Design Methods and Theories, Vol. 11, No. 3, 1977, pp 129 - 158.

[219] Grant, D.P., "How to Construct a Morphological Tree", Journal of Design Methods and Theories, Vol. 11, No. 3, 1977, pp 159 - 184.

[220] Maher, M. L. and Zhang, D. M., "CADSYN : Using Case and Decomposition Knowledge for Design Synthesis", Proceedings of AI in Design, Gero, J (ed) 1991, pp 137 - 150.

[221] Nakatani, Y et al, "Case-based Reasoning and Decision Aid for Engineering Design", Proceedings of the 1st World Congress on Expert Systems, Orlando, Florida, USA, December 1991, Vol. 1-4, Pergamon Press, pp 369 - 376.

[222] Wynn, H. P., et al, "Robust Engineering Design within Design Function Deployment", Proceedings of the International Conference on Engineering Design, ICED 93, August 17 - 19, 1993, The Hague, 1993, WDK 22, Roozenburg N. F. M. (ed.), Vol. 2, pp 890 - 897.

[223] Jebb, A. and Wynn, H. P., "Robust Engineering Design Post - Taguchi", Transactions of the Royal Society, London, A327, 1989, pp 605 - 616.

[224] Blessing, L. T. M., "A Process-Based Approach to Computer Supported Engineering Design", Proceedings of the 9th International Conference on Engineering Design, ICED '93, The Hague, Netherlands, August 17 - 19, 1993, pp 1393 - 1400.

[225] Boer, S. J.de, "Decisions Methods and Techniques in Methodical Engineering Design", Dissertation, University of Twente, Netherlands, 1989.

[226] Kunz, W. and Rittel, H., "Issues as Elements of Information Systems", Centre for Planning and Development Research, University of California, Berkeley, USA, 1970.

[227] Nagy, R. L., Ullman, D. G. and Dietterich, T. G., "A Knowledge Base Data Representation for Collaborative Mechanical Design", Proc. of ASME/DTM, September 1991.

[228] Simon, H. A., "A Student's Introduction to Engineering Design", Pergamon Press Inc, NY, USA, 1975.

[229] Rosenstein, A. B., "The Modern View of the Design Process", Presented at the National Congress of the Society of Automotive Engineers, Chicago, Ill, USA, January 1967.

[230] Bradley, A. et al, "Mechatronics - Electronics in Product and Processes", 1st Edition, Published by Chapman & Hall, UK, 1991.

[231] Jebb, A., Sivaloganathan, S and Evbuomwan, N. F. O., "Evolution of Design Function Deployment", Technical Report No. 64, Engineering Design Centre, City University, London, 1992.

[232] Harary, F. "Graph Theory", Addison-Wesley, Reading, Massachusetts, USA, 1969.

[233] Wiest, J. D. and Levy, F. K. 1977. "A Management Guide to PERT/CPM, 2nd Edition, Prentice-Hall, Englewood Cliffs, New Jersey, USA.

[234] Ross, D. T. 1977. "Structured Analysis (SA) : A Language for Communicating Ideas", IEEE Transactions on Software Engineering, Vol. SE-3, No. 1, January, pp 16 - 34.

[235] Ross, D. T. 1985. "Applications and Extensions of SADT", IEEE Computer Magazine, April, pp 25 - 34

[236] von Hippel, E. 1990. "Task Partitioning: An Innovation Process Variable", Research Policy, Vol. 19, pp 407 - 418.

[237] Steward, D. V. 1981. "The Design Structure System: A Method for Managing the Design of Complex Systems", IEEE Transactions on Engineering Management, Vol. EM-28, No. 3, August, pp 71 - 74.

[238] Kusiak, A. 1990. "Intelligent Manufacturing Systems", Prentice-Hall, Englewood Cliffs, NJ, USA.

[239] Lenau, T. and Alting, L., "Intelligent Support Systems for Product Design", Annals of the CIRP, Vol. 38, No. 1, 1989, pp 153 - 156.

[240] Kimura, F., "Product and Process Modelling as a Kernel for Virtual Manufacturing Environment", Annals of the CIRP, Vol. 42, No. 1, 1993, pp 147 - 150.

[241] Anumba, C. J., "Functional Integration in CAD Systems", Civil-Comp '93, Edinburgh, 1993, and Computing Systems in Engineering (in press).

[242] Sivaloganathan, S. et al, "Modelling the Product", Proceedings of the International Congress on Modelling and Simulation, Vol. 1, McAleer, M & Jakeman, A (eds), December 6 - 10, 1993, The University of Western Australia, Perth, Australia, pp 133 - 138.

[243] King, B. J. and Norman, P. W., "A Step in the Right Direction", Professional Engineering, November, 1992, pp 12 - 14.

[244] Shaw, N. K., et al, "Product Data Models", Research in Engineering Design, Vol. 1, 1989, pp 43 - 50.

[245] Watson, A. S. and Boyle, A. D., "The Creation and Use of Product Models Within The Construction Industry : The CI-PM Project", Informing Technologies for Construction, Civil Engineering and Transport, Powell, J.A. and Day, R. (eds), Published by Brunel University/SERC, 1993, pp 236 - 244.

[246] Krause, F. L. et al, "Product Modelling", Annals of the CIRP, Vol. 42, No. 2, 1992, pp 695 - 706.

[247] Lindberg, L. and Agerman, E., "The Product Structure - The Backbone of CIM", Annals of the CIRP, Vol. 41, No. 1, 1992, pp 165 - 168.

[248] Bjork, B-C., "Basic Structure of a Proposed Building Product Model", Computer Aided Design, Vol. 21, No. 2, March, 1989, pp 71 - 78.

[249] Jakobsen, K., "Transition Between Product Models", Proceedings of the International Conference on Engineering Design, ICED '85, Hamburg, 26 - 28 August, 1985.

[250] Klein, M., "DRCS : An Integrated System for Capture of Designs and Their Rationale", Artificial Intelligence in Design '92, Gero, J. S. (ed), kluwer Academic Publishers, Netherlands, 1992, pp 393 - 412.

[251] Klein, M., "Capturing Design Rationale in Concurrent Engineering Teams", IEEE Computer, January, 1993, pp 39 - 47.

[252] Chandrasekaran, B. et al, "Functional Representation as Design Rationale", IEEE Computer, January, 1993, pp 48 - 56.

[253] Babin, B. A. and Loganantharaj, R., "Designer's Workbench : A Tool to Assist in The Capture and Utilisation of Design Knowledge", Proceedings of AI in Design, Gero, J (ed), 1991, pp 251 - 267.

[254] Ganeshan, R. et al, "Representing and Reasoning with Design Intent", Proceedings of AI in Design, Gero, J (ed), 1991, pp 737 - 755.

[255] Ganeshan, R. et al, "A Framework for Representing Design Intent", Design Studies, Vol. 15, No. 1, January, 1994, pp 59 - 84.

[256] Freeman, M. S., "The Elements of Design Knowledge Capture", in Proceedings of the 4th Conference on AI for Space Applications, Huntsville, AL, USA, November, 1988.

[257] Wechsler, D. B. and Crouse, K. R., "An Approach to Design Knowledge Capture for The Space Station", In Proceedings of Space Station Automation II, Vol. 729, SPIE, 1986, pp 106 - 113.

[258] Yakemovic, K. C. B. and Conklin, E. J., "Report on a Development Project Use of An Issue-Based Information System, CSCW 90 Proceedings, 1990, pp 105 - 118.

[259] Kleer, J. D., "Choices Without Backtracking", AAAI-85, 1985, pp 79 - 85.

[260] Lee, J. and Lai, K. Y., "What's in Design Rationale ?", Human-Computer Interaction, 1991.

[261] McCall, R., "PHIBIS : Procedurally Hierarchical Issue-Based Information Systems", Proceedings of Conference on Planning and Design in Architecture, 1987.

[262] MacLean, A. et al, "Design Rationale : The Argument Behind the Artifact", Proceedings of CHI, Austin, Texas, USA, 1989.

[263] Klein, M. "Supporting Conflict Resolution in Cooperative Design Systems", IEEE Systems, Man and Cybernetics, Vol 21, No. 6, 1991.

[264] MacLean, A. et al, "Questions, Options and Criteria : Elements of a Design Rationale for User Interfaces", Journal of Human Computer Interaction : Special Issue on Design Rationale, 1991.

[265] Plevy T.A.H., "A Systems Approach to Materials and Process Selection", The Metallurgist and Materials Technologist, September 1981, Vol. 13, 469-474.

[266] Evbuomwan, N. F. O., "New Materials and Technologies in Engineering Design", Technical Report No. 48, Engineering Design Centre, City University, London, 1991.

[267] Evbuomwan, N. F. O., "Materials Selection within Design Function Deployment", Technical Report No. 73, Engineering Design Centre, City University, London, 1994.

[268] Bengtson, U. et al, "Design Requirements - Materials Properties : A Basis for Materials Selection", Computerisation and Networking of Materials Databases, Kaufman, J. G. & Glazman, J. S. (eds), ASTM STP 1106, 1991.

[269] Farag, M. M., "Selection of Materials and Manufacturing Processes for Engineering Design", 1st Edition, Published by Prentice Hall, UK, 1989.

[270] Saaty, T. L., "A Scaling Method for Priorities in Hierarchical Structures", Journal of Mathematical Psychology, Vol. 15, pp 234 - 281.

[271] Evbuomwan, N. F. O. et al, "Design Optimisation within Design Function Deployment", 9th International CAD/CAM, Robotics and Factories of the Future Conference, August 18 - 20, 1993, Newark, New Jersey, USA.

[272] MacCrimmon, K. R., "An Overview of Multiple Objective Decision Making", in "Multiple Criteria Decision Making", Edited by Cochrane, J. L. and Zeleny, M., University of South Carolina Press, Columbia, South Carolina, 1973, pp. 18 - 44.

[273] Hwang, C. L. and Yoon, K., "Multiple Attribute Decision Making - Methods and Applications. A State of the Art Survey", Springer-Verlag, NY, USA, 1981.

[274] Hwang, C. L. and Masud, A. S. M., "Multiple Objective Decision Making - Methods and Applications. A State of the Art Survey", Springer-Verlag, NY, USA, 1979.

[275] Haimes, Y. Y. et al, "Multiobjective Optimization in Water Resources Systems", Elsevier Scientific Publishing, Amsterdam, The Netherlands, 1975.

[276] Dlesk, D. C. and Liebman, J. S., "Multiple Objective Engineering Design", Engineering Optimization, Vol. 6, 1983, pp 161 - 175.

[277] Pugh, S., "Concept Selection - A Method that Works", Proceedings of the International Conference on Engineering Design, Rome, March 9 - 13, WKD 5 Paper M3/16, 1981, pp 497 - 506.

[278] Kuppuraju, N., Ittimakin, P and Mistree, F., "Design Through Selection: A Method that Works", Design Studies, Vol. 6, No. 2, 1985, pp 91 - 106.

[279] Jebb, A., Sivaloganathan, S and Evbuomwan, N. F. O., "Design Function Deployment - A Comprehensive Design System", 9th International CAD/CAM, Robotics and Factories of the Future Conference, August 18-20, 1993, Newark, New Jersey, USA.

[280] Jebb, A., Sivaloganathan, S and Evbuomwan, N. F. O., "Design Function Deployment User Guide", Engineering Design Centre, City University, June, 1993.

[281] Wells, D. L. et al, "Architecture of an Open Object-Oriented Database Management System", IEEE, Computer, October, 1992, pp 74-82.

[282] Pareti, J., "CAE Tools to Reduce Time to Market", Proceedings of the IMechE, Effective CADCAM '91, European Conference, November 6 - 7, 1991, Paper C429/049, pp 27 - 34.

[283] Ashby, M. F., "Materials Selection in Mechanical Design", 1st Edition, Pergamon Press, UK, 1992.

BIBLIOGRAPHY

[1] Bakerjian, R., " Design for Manufacturability ", " Tool and Manufacturing Engineers Handbook - Vol. 6 ", SME, 1992.

[2] Vance, J. M., " Design for Manufacture and Assembly Methods ", International Journal of Mechanical Engineering Education, Vol. 20, No. 3, 1992, pp 187-201.

[3] Boothroyd, G. & Dewhurst, P., " Product Design for Assembly Handbook ", Boothroyd Dewhurst, Inc., Wakefield, RI.

[4] Boothroyd, G., Knight, W., & Dewhurst, P., " Estimating the Costs of Printed Circuit Assemblies ", Printed Circuit Design, Vol. 6, No. 6, June 1989.

[5] Dewhurst, P., " Cutting Assembly Costs with Molded Parts ", Machine Design, July 21, 1988.

[6] Dewhurst, P. & Boothroyd, G., " Design for Assembly in Action ", Assembly Engineering, January 1987.

[7] Navinchandra, D., " Design for Environmentability ", 3rd International Conference on Design Theory and Methodology, DE-Vol. 31, ASME, September 1991, Miami, Florida, pp 119 - 125.

[8] Bevan, N., Jebb, A. & Wynn, H.P., " Design to Cost ", Proceedings of The Institution of Mechanical Engineers, Vol. 1, International Conference on Engineering Design, Harrogate, UK, 1989, pp 415 - 424.

[9] Jebb, A. & Sivaloganathan, S., " Design to Cost ", Technical Report No. 47, Engineering Design Centre, City University, 1991.

[10] Gershenson, J. & Ishii, K., " Life-Cycle Serviceability Design ", 3rd International Conference on Design Theory and Methodology, DE-Vol. 31, ASME, September 1991, Miami, Florida, pp 127 - 134.

[11] Passey, R. D. C., " FMEA Preparation - How to Perform a Failure Mode and Effect Analysis and Save Money ", Engineering Designer, Vol. 17, No. 1, January/February 1991, pp 22 - 29.

[12] Jebb, A. & Wynn, H. P., " Robust Engineering Design Post-Taguchi ", Philosophical Transactions of the Royal Society, London, Vol. 327 A, pp 605 - 611

[13] Mantyla, M., " An Introduction to Solid Modelling ", Computer Science Press, Maryland, USA, 1988.

[14] Sivaloganathan, S., " Sketching Input for Computer Aided Engineering ", PhD Thesis, City University, London, 1991.

[15] Jovane, F. et al, "A Key Issue in Product Life Cycle : Disassembly", Annals of the CIRP, Vol. 42, No. 2, 1993, pp 651 - 658.

[16] Boothroyd, G and Alting, L., "Design for Assembly and Disassembly", Annals of the CIRP, Vol. 41, No. 2, 1992, pp 625 - 636.

[17] Borsboom, T., "The Environment's Influence on Design", Design Management Journal, Fall, 1991, pp 42 - 47.

[18] Pitts G., 'Techniques in Engineering Design' by Butterworths &co, 1973.

[19] What every engineer should know about Finite Element Analysis - John R Brauer (ed), Marcel Dekker Inc., USA, 1988.

[20] Baran, N. M., "Finite Element analysis on Microcomputers", McGraw-Hill Book Co., USA, 1988.

[21] Bevan, N. et al, "Robust Design Methods in Action", Proceedings of IMechE, ICED '89, August 22 - 25, 1989, Vol. 1, Harrogate, WDK 18, pp 273 - 281.

[22] Vance, J. M., "Design for Manufacture and Assembly Methods", International Journal of Mechanical Engineering Education, Vol. 20 N0 3, 1992.

[23] Computer Integrated Testing - Edited by Allen Buckroyd, BSP Professional Books, Oxford, UK, 1989.

[24] Gershenson, J. & Ishii, K., "Life-cycle serviceability design", Design Theory and Methodology, DE-Vol. 31, ASME, 1991.

[25] Navinchandra, D., "Design for Environmentability", Design Theory and Methodology, DE-Vol. 31, ASME, 1991.

[26] Birolini, A., "Reliability Engineering In The Design Of Technical Systems", ICED '87, Boston, USA, 1987.

[27] R.E. Barlow and H.E. Lambert "Introduction to Fault Tree Analysis" in Reliability and Fault Tree Analysis R.E. Barlow et al (eds), Published by Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

APPENDIX

Publications Arising from This Research & Thesis

[1] Jebb, A. et al., "Design Function Deployment", Proceedings of the 1st Joint European Conference on Engineering Systems Design and Analysis, Istanbul, Turkey, ASME, PD-Vol. 47-1, Vol. 1, 1992, pp 251 - 256.

[2] Jebb, A. et al, "Design Function Deployment - A Platform for Cross Disciplinary Product Development", Mechatronics - The Integration of Engineering Design, IMechE Conference on Mechatronics, University of Dundee, September, 1992, pp 23 - 28.

[3] Jebb, A., Sivaloganathan, S. & Evbuomwan, N. F. O., "Design Function Deployment - A Paradigm for Concurrent Engineering", Proceedings of the International Conference on Design to Manufacture in Modern Industry, Bled, Slovenia, June 7 - 9, 1993, Part 1, pp 78 - 85.

[4] Jebb, A., Sivaloganathan, S and Evbuomwan, N. F. O., "Design Function Deployment - A Comprehensive Design System", 9th International CAD/CAM, Robotics and Factories of the Future Conference, August 18-20, 1993, Newark, New Jersey, USA, in Advances in Manufacturing Systems : Design, Modelling and Analysis, Sodhi, R. S. (ed), Manufacturing Research & Technology, Elsevier Science B. V., The Netherlands, Vol. 22, pp 303-308.

[5] Evbuomwan, N. F. O. et al, "Design Optimisation within Design Function Deployment", 9th International CAD/CAM, Robotics and Factories of the Future Conference, August 18 - 20, 1993, Newark, New Jersey, USA.

[6] Sivaloganathan, S. et al, "Modelling the Product", Proceedings of the International Congress on Modelling and Simulation, Vol. 1, McAleer, M & Jakeman, A (eds), December 6 - 10, 1993, The University of Western Australia, Perth, Australia, pp 133 - 138.

[7] Evbuomwan, N. F. O., Sivaloganathan, S & Jebb, A., "Concurrent Design within Design Function Deployment", Proceedings of the 2nd International Conference on Concurrent Engineering & Electronic Design Automation, Bournemouth, UK, April 7-8, 1994, Medhat, S. (ed), pp 167 - 175.

[8] Evbuomwan, N. F.O., Sivaloganathan, S, & Jebb, A, "Design Function Deployment - A Design for Quality System", Colloquium on "Customer Driven Quality in Product Design", 6th May, 1994, The Institution of Electrical Engineers, Savoy Place, London, Digest No. 1994/086, pp 7/1-7/3.

[9] Evbuomwan, N. F.O. & Sivaloganathan, S, "Generation of Conceptual Designs within Design Function Deployment", in "Methodologies, Techniques, and Tools for Design Development", Vol. 5, PD-Vol. 64-5, ASME, Tanik, M. M. et al (eds), Proceedings of the 2nd Biennial European Joint Conference on Engineering Systems Design and Analysis, July 4 - 7, 1994, London, pp 61 - 67.

[10] Sivaloganathan, S, & Evbuomwan, N. F.O., "Generation of Design Specifications within Design Function Deployment", in "Methodologies, Techniques, and Tools for Design Development", Vol. 5, PD-Vol. 64-5, ASME, Tanik, M. M. et al (eds), Proceedings of the 2nd Biennial European Joint Conference on Engineering Systems Design and Analysis, July 4 - 7, 1994, London, pp. 53 - 60.

[11] Evbuomwan, N. F.O., Sivaloganathan, S, Jebb, A, & Stickley, A. S. G., "Quality Function Deployment- Past, Present and Future", Proceedings of the 10th ISPE/IFAC International Conference on CAD/CAM, Robotics and Factories of the Future, August 21-24, 1994, Ottawa, Canada, pp 596 - 601.

[12] Odusote, V. A. O., Evbuomwan, N. F.O. & Sivaloganathan, S, "Software Implementation of Design Function Deployment", Proceedings of the 10th ISPE/IFAC International Conference on CAD/CAM, Robotics and Factories of the Future, August 21-24, 1994, Ottawa, Canada, pp 143 - 148.

[13] Evbuomwan, N. F.O., Sivaloganathan, S & Jebb, A, "Towards Concurrent Materials and Manufacturing Process Selection in Design Function Deployment", Proceedings of the 10th ISPE/IFAC International Conference on CAD/CAM, Robotics and Factories of the Future, August 21-24, 1994, Ottawa, Canada, pp 448 - 453.

[14] Evbuomwan, N. F.O., Sivaloganathan, S, & Jebb, A, "A State of the Art Report on Concurrent Engineering", 1st International Conference on Concurrent Engineering, Research and Applications, August 29-31, 1994, Sheraton Hotel, Pittsburgh, PA, USA, Paul, A. J & Sobolewski, M (eds), pp 35 - 44.

[15] Evbuomwan, N. F.O. & Sivaloganathan, S, "The Nature, Classification and Management of Tools and Resources for Concurrent Engineering", 1st International Conference on Concurrent Engineering, Research and Applications, August 29-31, 1994, Sheraton Hotel, Pittsburgh, PA, USA, Paul, A. J & Sobolewski, M (eds), pp 119 - 126.

APPENDIX B

DESIGNING WITH DESIGN FUNCTION DEPLOYMENT

A CASE STUDY

THE DESIGN OF A CAR LIFTING DEVICE

1. INTRODUCTION

Design Function Deployment (DFD) is a comprehensive design system developed to provide a platform for the realisation of concurrent engineering in product development. The process of design in DFD involves proceeding through five stages of :

(1) Establishment of customer requirements and determination of design specifications and constraints in a solution neutral form.

(2) Synthesis and development of different conceptual design solutions, called **Architectures.**

(3) Development of combinations of alternative subsystems within each promising architecture, called **Layouts.**

(4) Establishment of different materials and manufacturing process combinations for each part in each viable layout.

(5) Development of production plans for each viable layout, part, materials and manufacturing process combinations.

This illustrative case study will hence be developed, following the methodology outlined above. The case study is based on 'The Design of a Car Lifting Device' for domestic cars, and which is to be mass produced at a cheap price for customers.

2. STAGE 1 OF THE DFD DESIGN PROCESS

The customers for the proposed car lifting device were first of all established. They included :

(1) The car owner and/or driver of the car
(2) The car manufacturer
(3) The service engineers (mechanics & garage owners).

Based on the customers listed above, the necessary customer requirements were then elicited. In addition, manufacturing requirements and constraints as well as in-house company requirements were elicited. The details of the raw requirements are shown in column 2 of Table 1.

These requirements were then refined and categorised into primary, secondary and tertiary requirements, as shown in columns 3, 4 and 6 of Table 1 respectively. The classification of the requirements was done to ensure that they can be compared and prioritised at the same level of generality and basis. Having classified the requirements, the next step involved constructing the 'Quality Plan' (i.e. prioritising the tertiary requirements).

This was done by combining : the degree of importance to the customer (on a scale of 1 - 5), the degree of improvement according to the quality plan (the ratio of the targeted rating by the customers to the current rating of the company's products), and the sales advantage of each tertiary requirement. The absolute importance rating of each tertiary requirement was then obtained by multiplying these factors. If we take the tertiary requirement : **low effort to lift car** as an example, we have

Customer rating = 5

Degree of improvement required = 1.2

Sales advantage = 1.2

Therefore Absolute importance rating = $5 \times 1.2 \times 1.2 = 7.2$.

This process is performed for each tertiary requirement to obtain their absolute importance ratings. The relative importance rating of each tertiary requirement, is then calculated by normalising all the absolute values (using the highest as a datum) on a scale of 1 - 9.

Table 1 Classification of Customer Requirements

S/Nos	Raw Customer Requirements	Primary	Secondary	S/Nos	Tertiary Requirements
1	Should be Portable	Easy Handling	In Operation	1.1.1	Portable
2	Withstand Tough Handling			1.1.2	Easy to Operate
3	Easy Operation			1.1.3	Low Effort to Lift Car
4	Maintenance Free			1.1.4	No Extra Tools
5	Reliable			1.1.5	Easy to Set Up
6	Cater for all Cars and Loads			1.1.6	Operate in Standing Position
7	No Extra Tools			1.1.7	Quick Release of Load
8	Safe and Stable			1.1.8	Easy Anchorage to Most Cars
9	Easy to Use				
10	Use in both Wet & Dry Ground		Outdoor Use	1.2.1	Withstand Tough Handling
11	Low Effort to Lift Car			1.2.2	Maintenance Free
12	200,000 Units per year			1.2.3	Corrosion Resistant
13	Cost per Piece = £5			1.2.4	Use in both Wet & Dry Ground
14	Available Capital = £250,000			1.2.5	Cope with Uneven Ground
15	Quick Release of Load				
16	Corrosion Resistant	Versatile	In Use	2.1.1	Cater for all Cars and Loads
17	Cope with Uneven Ground			2.1.2	Stable while Supporting Load
18	Easy Anchorage to Many Cars			2.1.3	Safe in Use
19	Safety against Sudden Collapse			2.1.4	Support Car in Lifted Position
				2.1.5	Safety against Sudden Collapse
			Reliable	2.2.1	Works all the Time
				2.2.2	Must not Jam in Operation
		Constraints	Manufacturing	3.1.1	200,000 Units Per Year
			Cost	3.2.1	Cost per Piece around £5
			Capital	3.3.1	£250,000 Available

The complete quality plan (prioritisation) of the tertiary requirements, is shown in Table 2.

The next activity in this stage, involved translating (deploying) the tertiary requirements into design elements. These design elements are called design functions within DFD.

Design Functions are the identifiable and actionable design requirements translated from customer and other requirements (DFD Stage 1) or design functions (DFD Stages 2 - 4) which ensures the satisfaction of the initiating customer requirements and the quality performance of the resulting physical product/process.

The translation process involved taking each tertiary customer requirements and then generating design elements (design functions) which satisfy them. The design functions were established in a solution neutral form (i.e. without reference to any particular design solution. This process was continued until all the tertiary customer requirements, were deployed. The resulting design functions were then classified as specifications and constraints into primary, secondary and tertiary levels. The classified design functions are shown in Table 3.

Table 2 Quality Plan

S/Nos	Tertiary Requirements	Customer Rating	Degree of Improvemnt Required	Sales Advantage	Absolute Importance Rating	Relative Importance Rating
1.1.1	Portable	4	1	1.2	4.8	6
1.1.2	Easy to Operate	5	1	1.2	6	7
1.1.3	Low Effort to Lift Car	5	1.2	1.2	7.2	9
1.1.4	No Extra Tools	3	1	1	3	4
1.1.5	Easy to Set Up	4	1	1	4	5
1.1.6	Operate in Standing Position	3	1	1.2	3.6	4
1.1.7	Quick Release of Load	4	1	1.2	4.8	6
1.1.8	Easy Anchorage to Most Cars	4	1	1.2	4.8	6
1.2.1	Withstand Tough Handling	5	1	1	5	6
1.2.2	Maintenance Free	3	1	1	3	4
1.2.3	Corrosion Resistant	4	1	1	4	5
1.2.4	Use in both Wet and Dry Ground	4	1.2	1	4.8	6
1.2.5	Cope with Uneven Ground	5	1	1	5	6
2.1.1	Cater for all Cars and Loads	4	1	1.5	6	7
2.1.2	Stable while Supporting Load	5	1.2	1	6	7
2.1.3	Safe in Use	5	1	1	5	6
2.1.4	Support Car in Lifted Position	5	1	1	5	6
2.1.5	Safety against Sudden Collapse	5	1	1	5	6
2.2.1	Works all the Time	5	1	1	5	6
2.2.2	Must not Jam in Operation	5	1	1	5	6
3.1.1	200,000 Units Per Year	4	1	1	4	5
3.2.1	Cost Per Piece Around £5	5	1	1.5	7.5	9
3.3.1	£250,000 Available	4	1	1	4	5

Subsequent activities involved setting target values for some of the design functions, examining interactions between the design functions and then completing the DFD stage 1 chart. These details are shown in Figure 1.

Table 3 Classification of Design Functions

Serial Nos.	Tetiary Design Functions		Secondary Design Functions		Primary Design Functions
1.1.1	Weight of effort application device	1.1	Effort Application System	1	GENERAL SPECIFICATIONS FOR CAR LIFTING DEVICE
1.1.2	Length/Breadth of effort application device				
1.1.3	Position of effort application device				
1.1.4	Height of effort application device				
1.1.5	Low operating effort force to lift car				
1.1.6	High Efficiency				
1.1.7	No failure of effort application device				
1.1.8	Smooth effort application device motions				
1.1.9	Strong, durable and tough material for effort application device				
1.1.10	Good corrosion resistance of effort application device material				
1.2.1	Non-impairment of force transmission device motions	1.2	Force Transmission System		
1.2.2	No failure of force transmission device				
1.3.1	Minimum car load to be lifted	1.3	Load lifting and supporting system		
1.3.2	No failure of load lifting and supporting device				
1.3.3	Non-impairment of load lifting device motions				
1.3.4	Compressive/Buckling strength of load lifting and supporting device				
1.3.5	Weight of load lifting and supporting device				
1.3.6	Dimensions and shape of load lifting device				
1.3.7	Lift height				
1.3.8	Dead height of load lifting device				
1.3.9	Strong, durable and tough material for load lifting and supporting device				
1.3.10	Good corrosion resistance of load lifting and supporting device material				
1.4.1	Length/Breadth of supporting base system	1.4	Supporting Base System		
1.4.2	Weight of supporting base system				
1.4.3	Dead height of supporting base system				
1.4.4	Shape of supporting base system				
1.4.5	Good load distribution action on all surfaces				
1.4.6	Low centre of gravity for supporting base				

1.4.7	Strong, durable and tough material for supporting base system				
1.4.8	Good corrosion resistance of supporting base material				
1.5.1	load lowering speed	1.5	Load lowering system		
1.5.2	No failure of load lowering device				
1.5.3	Non-impairment of load lowering action				
1.5.4	Dimensions and shape of load lowering device				
1.6.1	Operate at a distance away from car	1.6	General operating properties		
1.6.2	Set up time and Minimal set up actions				
1.6.3	Set up height				
1.6.4	Set up position				
1.6.5	No sharp edges				
1.6.6	Safety lock to prevent sudden collapse				
1.6.7	Built-in tools and Sealed units				
1.6.8	Tight and rigid fittings				
2.1.1	Mass Production -> 800 units per week	2.1	Manufacturing Constraints	2	CONSTRAINTS
2.1.2	Easy to manufacture				
2.1.3	Quick manufacturing method(s)				
2.2.1	Easy to assemble	2.2	Assembly Constraints		
2.2.2	Quick assembly				
2.3.1	Low cost material(s)	2.3	Cost Constraints		
2.3.2	Low cost manufacturing method(s)				
2.3.3	Low cost assembly methods				
2.3.4	Low inventory costs				

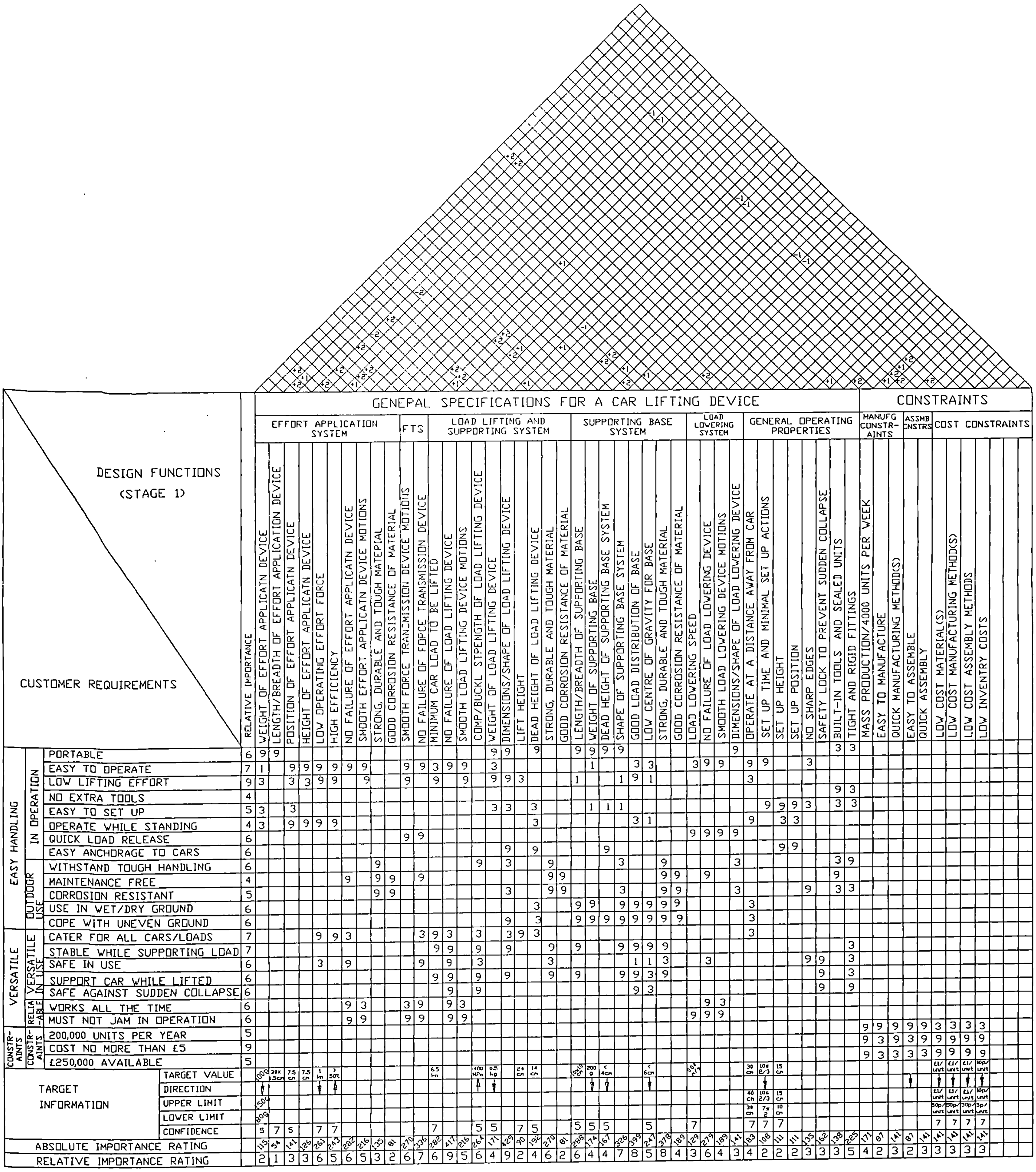

GENERAL SPECIFICATIONS FOR A CAR LIFTING DEVICE
CONSTRAINTS
EFFORT APPLICATION SYSTEM
FTS
LOAD LIFTING AND SUPPORTING SYSTEM
SUPPORTING BASE SYSTEM
LOAD LOWERING SYSTEM
GENERAL OPERATING PROPERTIES
MANUFG CONSTR-AINTS
ASSMB CNSTRS
COST CONSTRAINTS
DESIGN FUNCTIONS (STAGE 1)
CUSTOMER REQUIREMENTS
RELATIVE IMPORTANCE
EASY HANDLING
IN OPERATION
OUTDOOR USE
VERSATILE
RELIABLE
CONSTR-AINTS
PORTABLE
EASY TO OPERATE
LOW LIFTING EFFORT
NO EXTRA TOOLS
EASY TO SET UP
OPERATE WHILE STANDING
QUICK LOAD RELEASE
EASY ANCHORAGE TO CARS
WITHSTAND TOUGH HANDLING
MAINTENANCE FREE
CORROSION RESISTANT
USE IN WET/DRY GROUND
COPE WITH UNEVEN GROUND
CATER FOR ALL CARS/LOADS
STABLE WHILE SUPPORTING LOAD
SAFE IN USE
SUPPORT CAR WHILE LIFTED
SAFE AGAINST SUDDEN COLLAPSE
WORKS ALL THE TIME
MUST NOT JAM IN OPERATION
200,000 UNITS PER YEAR
COST NO MORE THAN £5
£250,000 AVAILABLE
TARGET INFORMATION
TARGET VALUE
DIRECTION
UPPER LIMIT
LOWER LIMIT
CONFIDENCE
ABSOLUTE IMPORTANCE RATING
RELATIVE IMPORTANCE RATING

Figure 1

Analysis of the DFD Stage 1 Chart - Figure 1

This figure encapsulates all the information generated in stage 1 of DFD for this case study. It shows the classified customer requirements (Whats), the design functions (i.e. specifications and constraints - Hows) and the relationships between them as shown in the matrix. The chart also shows the interactions between the design functions, their target values as well as the derived absolute and relative importance ratings.

Looking at the roof of the chart, which shows the interactions between the design functions, one can observe that they are generally positive interactions. These positive interactions (i.e. complementary actions) tended to occur within each of the potential subsystems (secondary design functions). A case in point, is the load lifting and supporting system as shown in the figure. This complimentary interactions between design functions of the same potential subsystems, do seem to indicate the possibility of combining functions within each subsystem. A close look at the relative importance ratings of the design functions, also reveal some pointers towards the critical design functions. If one considers those with a rating of 5 and above, it can be seen that they relate to the main functions and performance of the intended car lifting device. In proceeding to the next stage, all the design functions are however considered, as some of them are however critical, even if they have low ratings, as can be observed with the constraints. A close look at the target information, shows that the confidence levels of the target values were set at either 5 or 7 (on a scale of 1 - 10), as they were tentative, since the design process was in its early stages. One can also observe from the figure that, the secondary design functions, give an indication of the likely subsystems that will constitute the car lifting device.

3. STAGE 2 OF THE DFD DESIGN PROCESS

In this stage, conceptual solutions (Architectures) were developed, based on the derived specifications from stage 1. This was done using the morphological analysis method (see level 2 of the DFD structure). The procedure followed involved listing in a single block of rows, the main parameters (i.e. in this case, secondary design functions) from stage 1. These parameters represented the main subsystems of potential conceptual design solutions. For each of these main parameters, realisable generic subsystems were generated in the form of alternative solutions to the parameters. The details of this activity are shown in Table 4.

Table 4 Generation of Architectures

	PARAMETERS	VARIATIONS				
		1	2	3	4	5
A	Effort Application System	Lever & Piston	Air Pump	Lever & Gears	Lever	
B	Force Transmision System	Hydraulic	Air	Gears	Screw & Nut /Collar	Screw & Threaded Shaft
C	Load Lifting and Supporting System	Hydraulic Cylinder & Piston	Bellows	Screw & Nut /Collar (Non Self Locking)	Linkage Mechanism (Scissors)	Screw & Threaded Shaft
D	Supporting Base System	Hydraulic Barrel/ Cylinder	Bellows	Flat Circular Base	Flat Base	Threaded Barrel
E	Load Lowering System	Pressure Release Outlet	Air Release Outlet	Centrifugal Brake (Governor)	Linkage Mechanism (Scissors)	Screw & Threaded Shaft

This table is usually referred to as the *morphological box.* The next step then involved inspecting each of the alternative solutions and then selecting useful morphologies (i.e. synthesising and generating combinations of the generic subsystems).

Possible combinations of useful morphologies in the form of conceptual design solutions (Architectures) from Table 4, are :

(1) A1B1C1D1E1

(2) A2B2C2D2E2

(3) A3B3C3D3E3 [1]

(4) A4B4C4D4E4

(5) A5B5C5D5E5

It can be observed that the promising architectures are mainly combinations of subsystems based on the same principle and concept, e.g. A1B1C1D1E1. This might be due to the fact that fundamental principles upon which each alternative subsystem solution for each parameter is based, are distinctly independent and incompatible. So that for instance A3B2C3D4E5 and A1B2C3D4E4, would both be unfeasible.

Having established the potential conceptual design solutions (architectures), it is necessary to choose the most promising concept for further development. In doing this, a multi-criteria optimization method was used, (i.e. the simple additive weighting (SAW) method). The result of this simple selection process is shown in Table 5.

Table 5

	SELECTION CRITERIA	Rating	ARCHITECTURES				
			Arch 1	Arch 2	Arch 3	Arch 4	Arch 5
1	Low operating Effort	6	8	5	7	6	5
2	High Efficiency	5	8	4	6	5	4
3	Smooth Effort Applicatn	6	7	5	7	5	6
4	Smooth Force Transm	6	8	8	6	7	6
5	No Failure of Liftg. Dev.	9	8	6	7	8	7
6	Good Load Distribution	8	7	8	7	6	8
7	Good Corrosion Resist.	4	7	8	6	6	6
8	Low Centre of Gravity	5	7	7	7	6	7
9	Load Lowering Speed	3	8	7	7	5	5
10	Operate at a Distance	4	6	7	6	7	6
11	Setup Time	2	7	5	7	6	6
12	Safety Lock	3	6	6	8	6	6
13	Mass Production	4	7	5	5	6	7
14	Low Cost Materials	3	7	6	5	8	6
15	Easy to Manufacture	2	6	6	5	6	6
16	Quick to Manufacture	3	6	5	5	7	6
17	Easy to Assemble	2	7	7	5	7	7
18	Quick to Assemble	3	7	6	5	7	7
SAW Method (Total)			563	487	496	496	488
Ranking			1	4	2	2	3
Total Weighting (Ignoring Rating)			127	111	111	114	111
Ranking			1	3	3	2	3

The selection criteria used were the critical specifications and constraints (design functions) derived in stage 1, taking account of their relative importance ratings. In performing some form of sensitivity analysis, the importance ratings of the selection criteria were ignored. The resulting weight of each architecture is as shown in the last row of Table 5, showing some switch in weightings compared to when the SAW method was used. The concept (architecture) based on the hydraulic principle however maintained the best position. This concept for all intents and purposes, was then chosen for subsequent development in stage 3. The architecture A4B4C4D4E4 based on the linkage mechanism (scissors) although rated below the hydraulic jack, might in the long run be cheaper to manufacture. For the purpose of this exercise, the hydraulic jack architecture was developed further.

For this chosen architecture and the associated subsystems, the following activities were performed (i) establishment of the subsystem characteristics (DFD stage 2 design functions) which satisfy the specifications and constraints from stage 1, (ii) establishment of the target values for some of the subsystems characteristics, (iii) construction of the relationship matrix, (iv) computation of the absolute and relative importance ratings and (v) completion of the main chart for the architecture. This completed chart is shown in Figure 2.

4. STAGE 3 OF THE DFD DESIGN PROCESS

In this stage, a similar procedure to stage 2 was followed. However, in developing alternative layouts for the Hydraulic jack architecture, it was necessary to resort to developing 2D drawings as shown in Figure 3. Existing previous designs were also examined in developing these layouts [2, 3]. Based on the subsystem characteristics (stage 2 design functions) of the Hydraulic jack architecture, alternative layouts (L_1 - L_5), were developed. The layouts (L_1 - L_3) were based on a direct vertical load lifting motion while (L_4 & L_5) were based on an angular motion which is translated into a vertical load lifting motion. The (L_4 & L_5) types are also known as Trolley Jacks. The (L_1 - L_3) types are usually known as Bottle Jacks, with L_2 being the double lift variant of this type.

The Trolley types do have more load capacity and involve the use of lesser effort than the Bottle types. The Bottle types are however cheaper, and more portable, as the Jack is being considered for domestic use. They would also be cheaper to manufacture and assemble than the Trolley type. For the purpose of this case study, layout L_3 was chosen for further development. For this layout, the parts characteristics (stage 3 design functions), their target information and the relationship matrix were established. The complete chart for this layout is shown in Figure 4. A preliminary 2D drawing of the layout is shown in Figure 5.

5. STAGE 4 OF THE DFD DESIGN PROCESS

From Figure 4, the materials related design functions were extracted for each part of the layout. These were then refined into representative materials properties, as shown in Table 6. In addition, the following constraints were considered :

(1) The car lifting device is to be mass produced at a rate of about 200,000 per annum.

(2) From Figure 5, it can be seen that the geometry of the layout points to a monolithic construction.

(3) Other criteria include those based on Design for Assembly and Manufacture rules, i.e.

designing for minimum number of parts
designing parts to be multifunctional
minimising assembly directions
designing for top down assembly

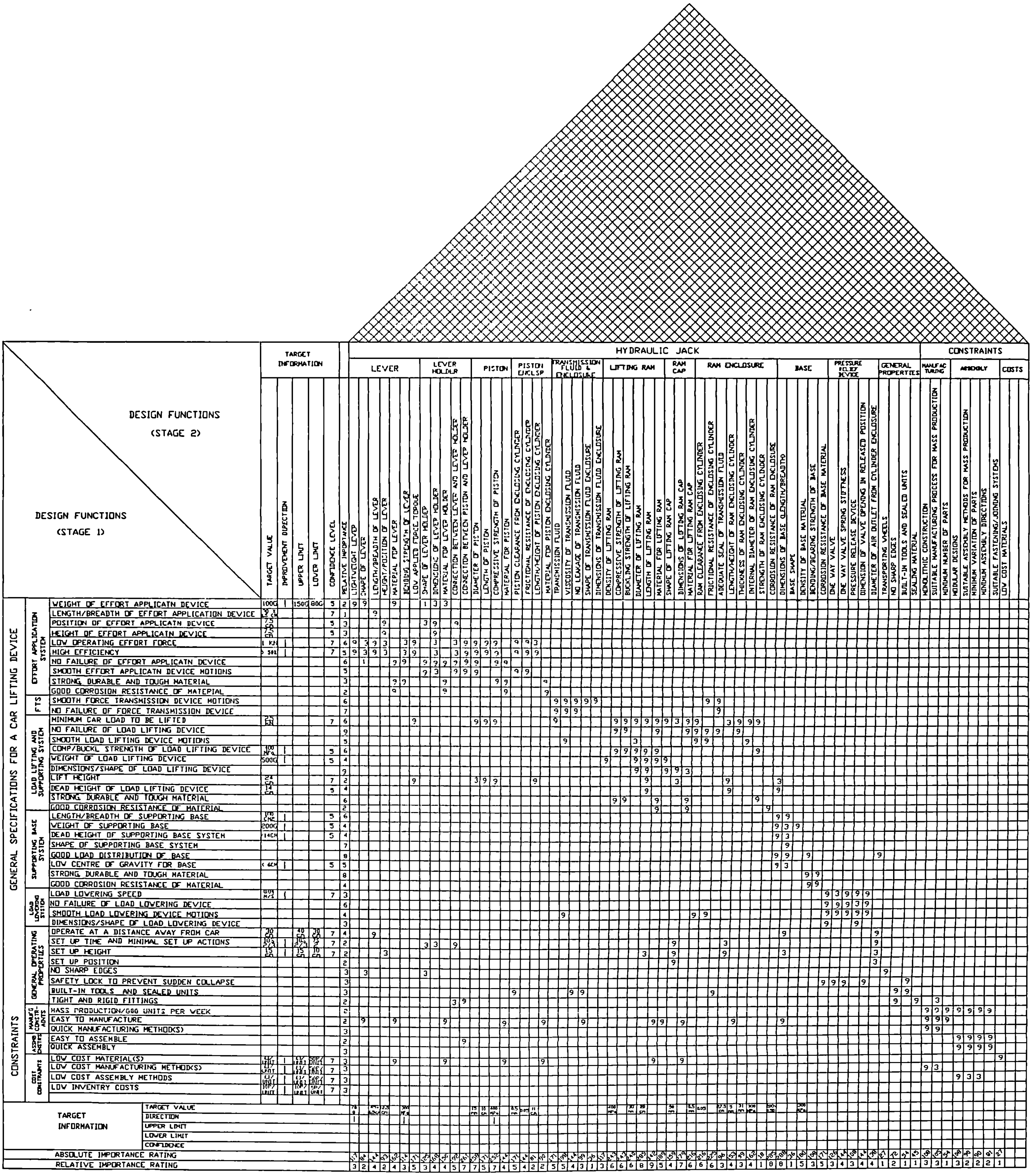

DESIGN FUNCTIONS (STAGE 2)
DESIGN FUNCTIONS (STAGE 1)
TARGET INFORMATION
HYDRAULIC JACK
CONSTRAINTS
LEVER
LEVER HOLDER
PISTON
PISTON ENCLSP
TRANSMISSION FLUID & ENCLOSURE
LIFTING RAM
RAM CAP
RAM ENCLOSURE
BASE
PRESSURE RELIEF DEVICE
GENERAL PROPERTIES
MANUFACTURING
ASSEMBLY
COSTS
GENERAL SPECIFICATIONS FOR A CAR LIFTING DEVICE
EFFORT APPLICATION SYSTEM
FTS
LOAD LIFTING AND SUPPORTING SYSTEM
SUPPORTING BASE SYSTEM
LOAD LOWERING SYSTEM
GENERAL OPERATING PROPERTIES
WEIGHT OF EFFORT APPLICATN DEVICE
LENGTH/BREADTH OF EFFORT APPLICATION DEVICE
POSITION OF EFFORT APPLICATN DEVICE
HEIGHT OF EFFORT APPLICATN DEVICE
LOW OPERATING EFFORT FORCE
HIGH EFFICIENCY
NO FAILURE OF EFFORT APPLICATN DEVICE
SMOOTH EFFORT APPLICATN DEVICE MOTIONS
STRONG, DURABLE AND TOUGH MATERIAL
GOOD CORROSION RESISTANCE OF MATERIAL
SMOOTH FORCE TRANSMISSION DEVICE MOTIONS
NO FAILURE OF FORCE TRANSMISSION DEVICE
MINIMUM CAR LOAD TO BE LIFTED
NO FAILURE OF LOAD LIFTING DEVICE
SMOOTH LOAD LIFTING DEVICE MOTIONS
COMP/BUCKL STRENGTH OF LOAD LIFTING DEVICE
WEIGHT OF LOAD LIFTING DEVICE
DIMENSIONS/SHAPE OF LOAD LIFTING DEVICE
LIFT HEIGHT
DEAD HEIGHT OF LOAD LIFTING DEVICE
STRONG, DURABLE AND TOUGH MATERIAL
GOOD CORROSION RESISTANCE OF MATERIAL
LENGTH/BREADTH OF SUPPORTING BASE
WEIGHT OF SUPPORTING BASE
DEAD HEIGHT OF SUPPORTING BASE SYSTEM
SHAPE OF SUPPORTING BASE SYSTEM
GOOD LOAD DISTRIBUTION OF BASE
LOW CENTRE OF GRAVITY FOR BASE
STRONG, DURABLE AND TOUGH MATERIAL
GOOD CORROSION RESISTANCE OF MATERIAL
LOAD LOWERING SPEED
NO FAILURE OF LOAD LOWERING DEVICE
SMOOTH LOAD LOWERING DEVICE MOTIONS
DIMENSIONS/SHAPE OF LOAD LOWERING DEVICE
OPERATE AT A DISTANCE AWAY FROM CAR
SET UP TIME AND MINIMAL SET UP ACTIONS
SET UP HEIGHT
SET UP POSITION
NO SHARP EDGES
SAFETY LOCK TO PREVENT SUDDEN COLLAPSE
BUILT-IN TOOLS AND SEALED UNITS
TIGHT AND RIGID FITTINGS
MASS PRODUCTION/600 UNITS PER WEEK
EASY TO MANUFACTURE
QUICK MANUFACTURING METHOD(S)
EASY TO ASSEMBLE
QUICK ASSEMBLY
LOW COST MATERIAL(S)
LOW COST MANUFACTURING METHOD(S)
LOW COST ASSEMBLY METHODS
LOW INVENTRY COSTS
TARGET INFORMATION
TARGET VALUE
DIRECTION
UPPER LIMIT
LOWER LIMIT
CONFIDENCE
ABSOLUTE IMPORTANCE RATING
RELATIVE IMPORTANCE RATING

Figure 2

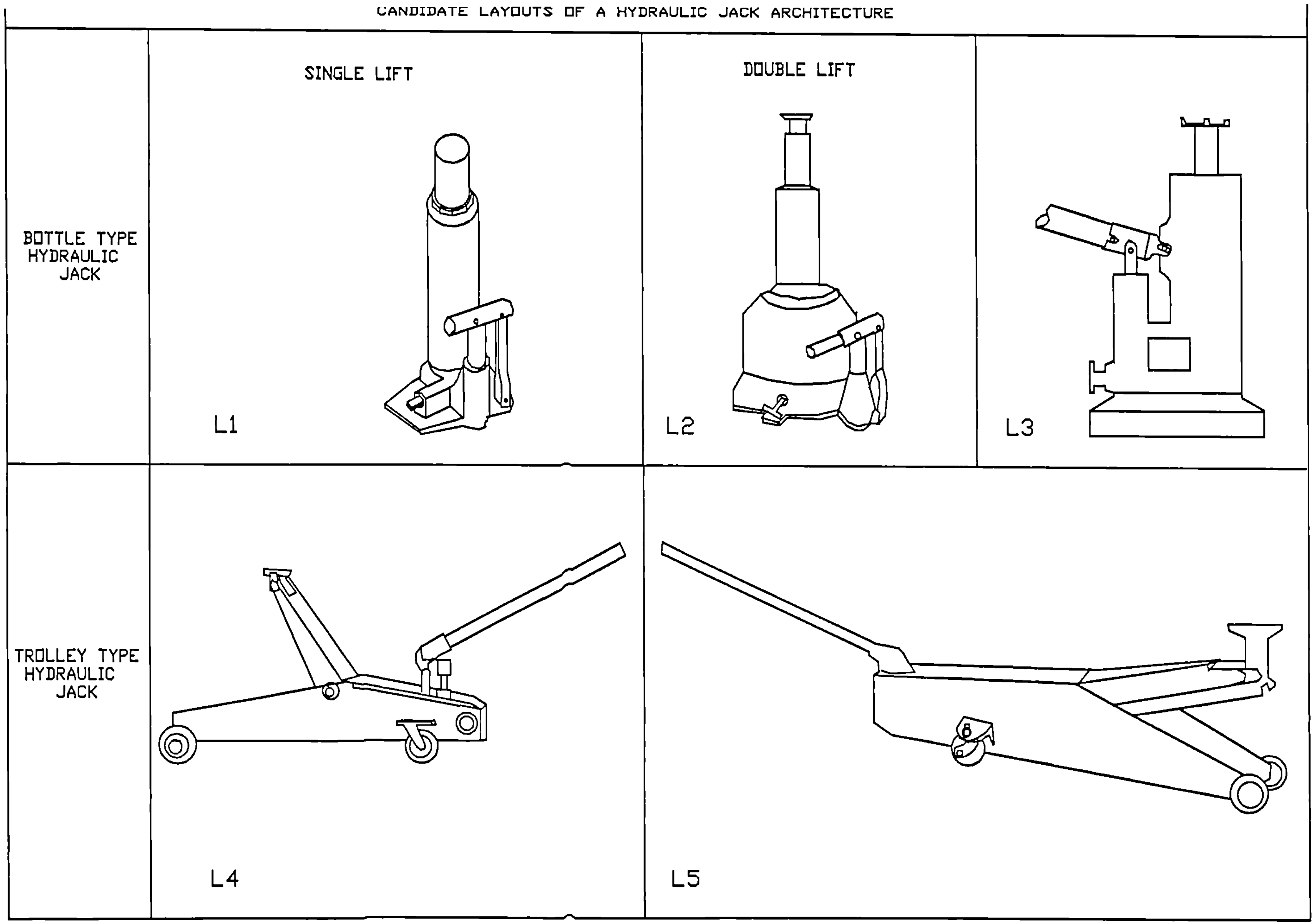

CANDIDATE LAYOUTS OF A HYDRAULIC JACK ARCHITECTURE
BOTTLE TYPE HYDRAULIC JACK
SINGLE LIFT
DOUBLE LIFT
L1
L2
L3
TROLLEY TYPE HYDRAULIC JACK
L4
L5

Figure 3

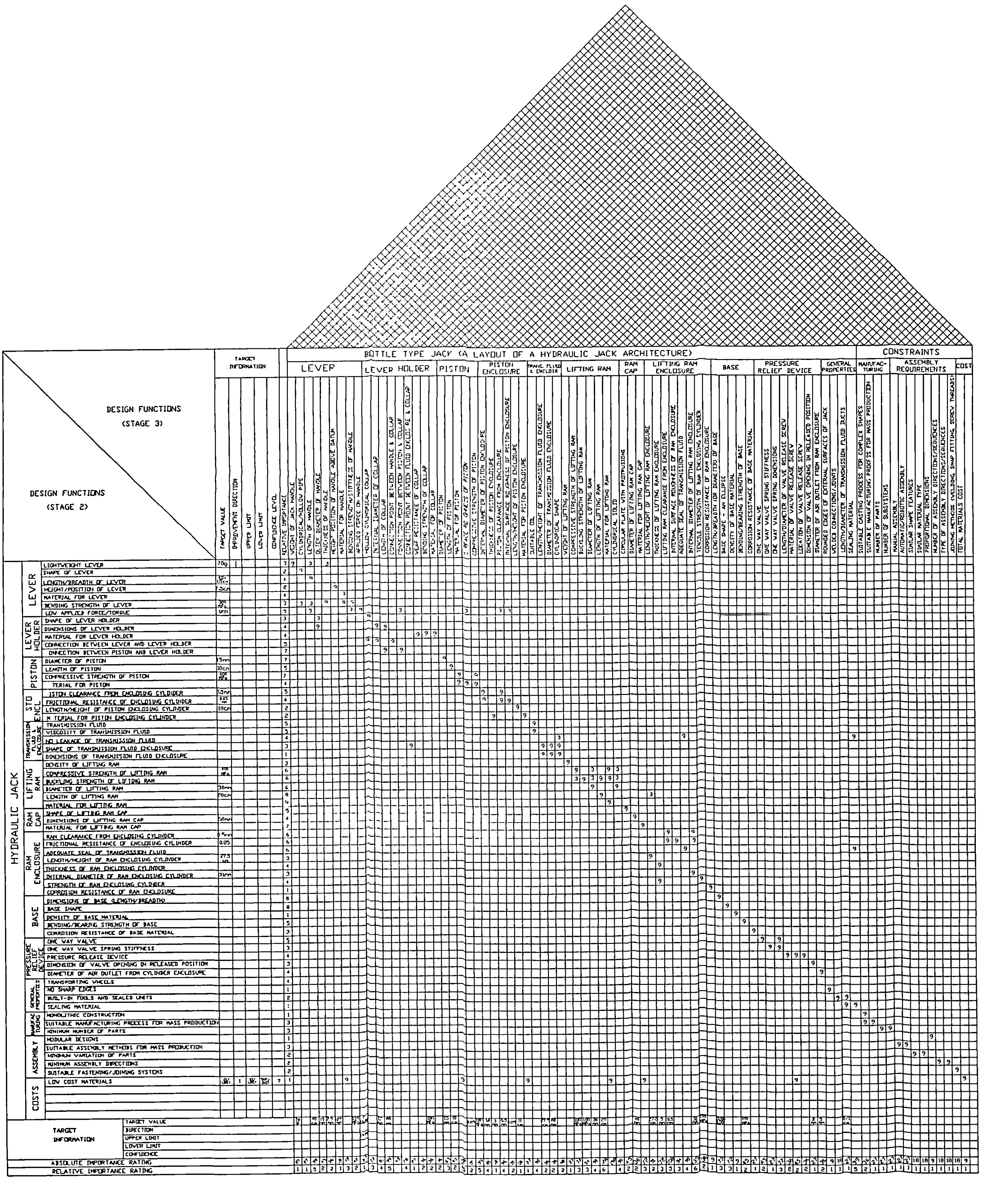

BOTTLE TYPE JACK (A LAYOUT OF A HYDRAULIC JACK ARCHITECTURE)
CONSTRAINTS
DESIGN FUNCTIONS (STAGE 3)
DESIGN FUNCTIONS (STAGE 2)
HYDRAULIC JACK
TARGET INFORMATION
ABSOLUTE IMPORTANCE RATING
RELATIVE IMPORTANCE RATING

Figure 4

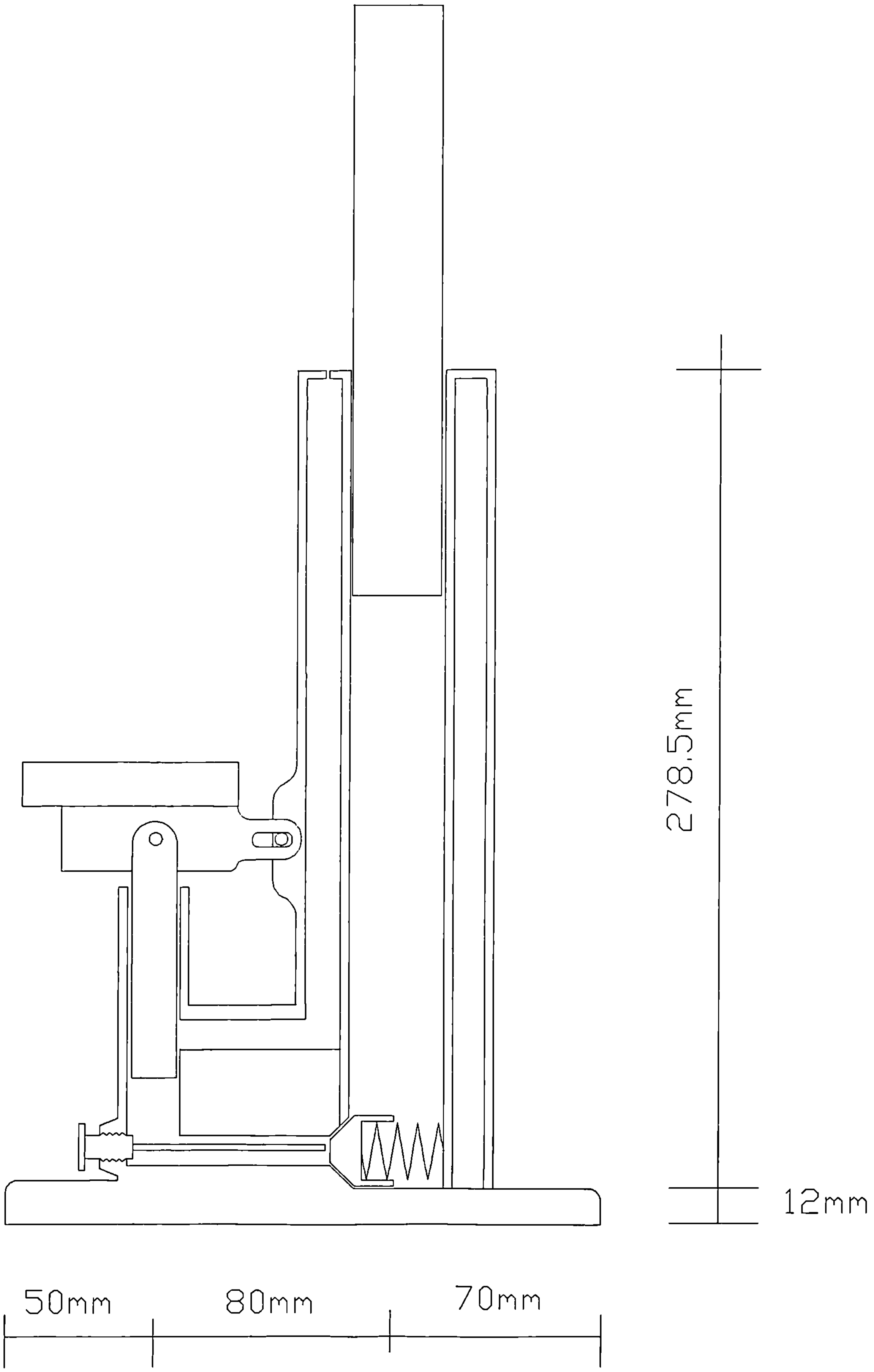


Figure 5

Table 6

	Secondary Design Functions (DFD Stage 3)	Tertiary Design Functions (Materials Related)	Materials Functions (Properties, etc)
	Lever	Weight of Handle	Density of Material
			Diameter/Thickness of lever
		Cylindrical (Hollow) Pipe	Thickness/Length of handle
		Bending Strength/Stiffness	Flexural Strength, Shape Factor
			Flexural Modulus
		Material for handle	
	Lever Holder	Hollow Cylindrical Collar	Length, Diameter & Thickness
		Wear Resistance	Hardness
		Bending Strength of Collar	Flexural Modulus, Shape Factor
		Material for Collar	
	Piston	Surface Roughness	Coefficient of Friction
		Compressive Strength	Compressive Strength
		Material for Piston	Length/Diameter
	Piston Enclosure	Surface Roughness	Coefficient of Friction
		Material for Enclosure	Thickness
	Fluid Enclosure	Cylindrical Shape	Length, Diameter, Thickness
	Lifting Ram	Compressive Strength	Compressive Strength
		Buckling Strength	Buckling Strength, Length
		Weight of Ram	Density of Material
			Diameter, Length
		Cylindrical Solid	Diameter, Length
		Material for Lifting Ram	
	Lifting Ram Cap	Circular Plate	Diameter/Thickness
		Material for Lifting Ram Cap	
	Lifting Ram Enclosure	Internal Surface Roughness	Coefficient of Friction
		Tensile Strength	Tensile Strength, Thickness
		Corrosion Resistance	Corrosion Resistance
	Base	Elliptical Shape	2 Diameters, Thickness
		Density	Density of Material
		Bending Strength	Bending Strength, Thickness
		Bearing Strength	Bearing/Compressive Strength
		Corrosion Resistance	Corrosion Resistance
	Pressure Relief Device	Valve Spring Stiffness	Stiffness, Diameter, Thickness
		Material for Spring	
		Material for Release Screw	Compatible with Enclosure

Based on these constraints, it was decided that the casting process would be most suitable. The particular type of casting method was however not decided upon. Some possible types included :

Sand casting
Shell casting
Investment casting
Pressure die casting (hot chamber or cold chamber)

In choosing the particular type of casting process to be used, certain criteria were considered. These included :

(1) suitability for high production rates
(2) suitability for good strength properties
(3) suitability for casting complex shapes
(4) suitability for good dimensional accuracy and finishing
(5) suitability for high melting temperature alloys

Taking into account the above factors, the Pressure die casting process comes out as the most suitable process. For the production volume desired (about 200,000), it is the most economical process. It is also particularly good for complex shapes.

Considering the materials properties, three types of materials emerged as potentials, after searching the Cambridge Materials Selector database. These were : Aluminium alloys, Cast Iron (nodular) and Steel. Amongst the three, Steel would be the most difficult to cast. Aluminium alloys and Cast Iron now become the most likely materials to be used. The pressure die casting process consists of both the hot-chamber type and the cold-chamber type. The hot-chamber type is more suited for low melting alloys such as Zinc, Tin, Lead, etc. The cold-chamber process on the other hand is more suited for high melting alloys like Aluminium, Magnesium and Copper. Ferrous alloys, e.g Cast Iron can also be cast using the cold-chamber process, but with some difficulty. The fact that the potential materials under consideration, i.e. Aluminium and Cast Iron were high melting alloys, the need for hot-chamber process is obviated. In the light of the above, the cold-chamber pressure die casting process was then chosen.

Taking into account the chosen manufacturing process, i.e. the cold-chamber pressure die casting process, the Aluminium alloy would be the most likely material to be used, as it is well suited to the process. Apart from the lower corrosion resistance of Cast Iron compared to Aluminium, its high melting temperature would mean reduced life spans for the casting dies. Based on the geometry - Figure 5, it would also be easier to cast the thin enclosure walls of the Jack, using the Aluminium alloy. If a smaller number of Jacks were however required, the Sand casting process, might then be chosen, as it would be more economical, and Cast Iron would then be more ideal as the material. For the purpose of this case study, it was decided that the Aluminium alloy (which most satisfies the materials properties), would be used.

Having decided on the materials and the manufacturing process, the necessary process parameters that would ensure that each part can be manufactured to meet the requirements of the parts characteristics, were then established. In this process, a number of sources [4-9], were consulted. These acted as knowledge bases for the cold-chamber pressure die casting process. Based on the established process parameters, the relationship matrix was then developed, with target values also set for some of the process parameters. The complete stage 4 chart is shown in Figure 6.

From this figure, it can be observed that some of the process parameters that are likely to be critical in the casting process, showed high importance ratings. These were selected for further development in stage 5. It is worth noting that the process parameters generally relate to the parts characteristics of each part of the layout. From the chart, some interactions can also be observed. Positive interactions can be observed between : (1) the melting temperature of the Aluminium alloy and the injection pressure, (2) the taper (draft angle) and/or die lubrication and the ejection speed and ejection force. These interactions need to be considered when designing the casting dies and when establishing the process parameters.

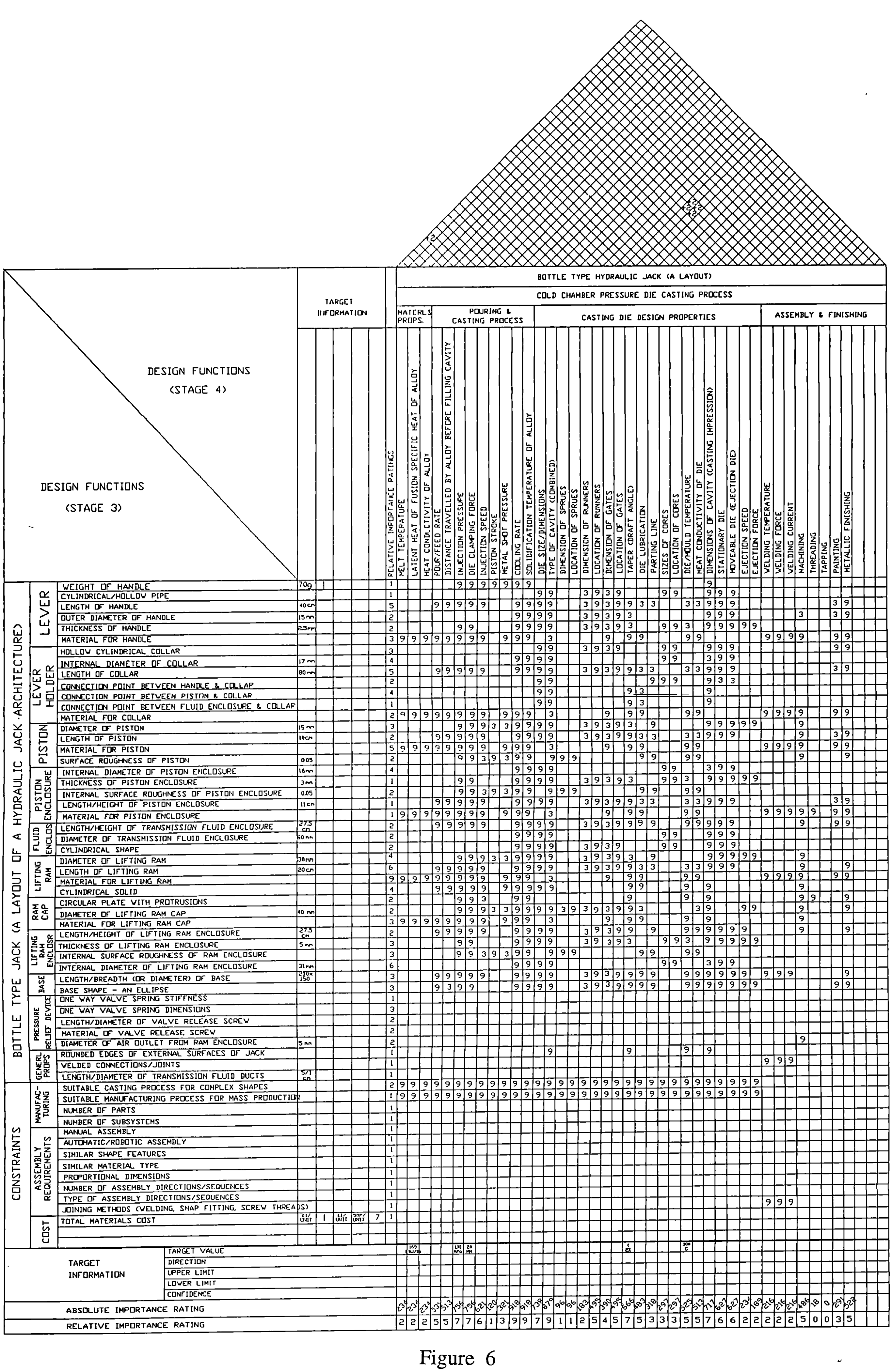

BOTTLE TYPE HYDRAULIC JACK (A LAYOUT)
COLD CHAMBER PRESSURE DIE CASTING PROCESS
MATERLS PROPS.
POURING & CASTING PROCESS
CASTING DIE DESIGN PROPERTIES
ASSEMBLY & FINISHING
TARGET INFORMATION
DESIGN FUNCTIONS (STAGE 4)
DESIGN FUNCTIONS (STAGE 3)
BOTTLE TYPE JACK (A LAYOUT OF A HYDRAULIC JACK ARCHITECTURE)
CONSTRAINTS
LEVER
LEVER HOLDER
PISTON
PISTON ENCLOSURE
FLUID ENCLOS
LIFTING RAM
RAM CAP
LIFTING RAM ENCLOSR
BASE
PRESSURE RELIEF DEVICE
GENERL PROPS
MANUFAC-TURING
ASSEMBLY REQUIREMENTS
COST
TARGET VALUE
DIRECTION
UPPER LIMIT
LOWER LIMIT
CONFIDENCE
ABSOLUTE IMPORTANCE RATING
RELATIVE IMPORTANCE RATING

Figure 6

Another approach to developing the charts at this stage, might be to create a chart for each part showing the relationships between its characteristics and the process parameters.

It was also decided that the casting process steps would involve the following :

(1) Casting the lever, lever holder and piston in one die.

(2) Casting the lifting ram and lifting ram cap in another die.

These two casting processes can be carried out using a combination die.

(3) Casting the piston, fluid and lifting ram enclosures as well as the base in one casting process using a single cavity die.

It may however be necessary to explore other possibilities by using a combination of the single, multiple and/or combination dies.

6. STAGE 5 OF THE DFD DESIGN PROCESS

This stage essentially illustrated how the production operations activity can be carried out, and how the production operations parameters can be established for the critical process parameters identified in stage 4. The activity here involved establishing for each critical process parameter, production operations parameters under the following groupings : (a) Operation Planning, (b) Quantity Economics, (c) Materials, (d) Operation Information, (e) Quality Control and (f) Machine Hours.

If for example one considers the critical process parameter :

<u>Injection Pressure</u>,

The operation planning parameters are as shown below :

Difficulty of controlling parameters	= 2
Frequency of expected problems	= 2
Severity of problems if encountered	= 3
Ability to detect problems if they occur	= 3
The importance rating of Injection Pressure	= 7

The Total Points value is calculated as

$$7 \times 2 \times 2 \times 3 \times 3 = 252$$

This is performed for all the critical process parameters. The ones which exhibit high total points, further represent critical process parameters that must of necessity be focussed on during the production operation planning activity.

For the other production operation parameters, necessary relationships were established. The representative chart is shown in Figure 7. It represents the DFD stage 5 chart showing the relationships between the critical manufacturing process parameters and the production operations parameters. The chart also shows the quality control actions to be taken for the critical process paramaters.

PRODUCTION OPERATIONS PLANNING FOR COLD CHAMBER PRESSURE DIE CASTING PROCESS

BOTTLE TYPE HYDRAULIC JACK (A LAYOUT)

COLD CHAMBER PRESSURE DIE CASTING PROCESS

DESIGN FUNCTIONS (STAGE 4)	DESIGN FUNCTIONS (STAGE 5)	TARGET INFORMATION	PROCESS CAPABILITY	RELATIVE IMPORTANCE RATINGS	OPERATION PLANNING: DIFFICULTY OF CONTROLLING PARAMETERS	FREQUENCY OF EXPECTED PROBLEMS	SEVERITY OF PROBLEMS IF ENCOUNTERED	ABILITY TO DETECT PROBLEMS IF THEY OCCUR	TOTAL POINTS	QUANTITY ECONOMICS: ECONOMIC BATCH SIZE	MACHINE SETTINGS	TOOL CHANGE	WASTE PER PIECE	MATERIALS: DIRECT MATERIAL	STANDARD SIZE	COST PER WEIGHT	INDIRECT MATERIAL	EASE OF MANUFACTURE	OPERATION INFORMATION: JOB INSTRUCTIONS	LABOUR: DIRECT	LABOUR: INDIRECT	CYCLE TIME	QUALITY CONTROL: CONTROL DOCUMENTS	CONTROL METHOD	FREQUENCY	SAMPLE SIZE	MACHINE HOURS	PREVENTATIVE MAINTENANCE TASKS	OPERATOR EDUCATION/TRAINING	REMARKS
MATERLS PROPS.	MELT TEMPERATURE		○	2	2	1	2	3	24		O				O								O	O						
	LATENT HEAT OF FUSION/SPECIFIC HEAT OF ALLOY	169 BTU/lb	○	2	3	1	3	3	54														O	O						
POURING & CASTING PROCESS	POUR/FEED RATE		○	5	1	1	3	3	45	O			O			O							O		O				O	
	DISTANCE TRAVELLED BY ALLOY BEFORE FILLING CAVITY		○	5	3	1	3	3	135																					
	INJECTION PRESSURE	100 MPa		7	2	2	3	3	252		O												O	O	O			O		
	DIE CLAMPING FORCE	20 MN	○	7	3	1	3	3	189														O	O	O			O		
	INJECTION SPEED			6	3	1	3	3	162	O	O												O	O	O			O		
	COOLING RATE		○	9	3	1	2	3	162	O													O	O	O					
	SOLIDIFICATION TEMPERATURE OF ALLOY			9	3	1	3	2	162	O													O	O	O					
CASTING DIE DESIGN PROPERTIES	DIE SIZE/DIMENSIONS			7	1	1	3	2	42	O																				
	TYPE OF CAVITY (COMBINED)			9	1	1	3	2	54	O																				
	LOCATION OF RUNNERS			5	1	1	3	3	45				O																	
	DIMENSION OF GATES			4	1	1	3	3	36				O																	
	LOCATION OF GATES			5	1	1	3	3	45				O																	
	TAPER (DRAFT ANGLE)	< 2%		7	2	1	3	3	126																					
	DIE LUBRICATION		○	5	3	2	3	2	180	O													O	O	O			O		
	DIE/MOULD TEMPERATURE	300 C		5	2	1	3	3	90														O	O	O					
	HEAT CONDUCTIVITY OF DIE			5	2	1	2	2	40														O	O						
	DIMENSIONS OF CAVITY (CASTING IMPRESSION)			7	1	1	2	2	28																					
	STATIONARY DIE			6	1	1	3	2	36	O		O			O													O		
	MOVEABLE DIE (EJECTION DIE)		○	6	1	1	3	2	36	O		O			O													O		
ASSEMBLY & FINISHING	MACHINING			5	1	1	1	1	5	O	O		O															O	O	
	METALLIC FINISHING			5	1	1	1	1	5																				O	

3 - STRONG PROBABILITY

2 - MEDIUM PROBABILITY

1 - SMALL PROBABILITY

O - RELATIONSHIP

○ - PROCESS CAPABILITY

Figure 7

7. CONCLUSIONS

The essence of this case study has been to illustrate the use of DFD in the systematic development of a product. The product considered in this case, was a car lifting device. The case study involved first of all eliciting the customer requirements, refining, classifying and prioritising them, after which they were translated (deployed) into solution neutral specifications and constraints (DFD stage 1 design functions). Based on the specifications and constraints, conceptual solutions were then developed in stage 2. This involved using some of the design modules shown in level 2 of the DFD structure, such as functional analysis and morphological analysis methods.

From amongst the conceptual solutions developed, a solution based on the hydraulic principle, was chosen for subsequent development in stage 3. In stage 3, the chosen concept (architecture), was further developed into five alternative layouts, representing different variants of the hydraulic concept architecture. Here again, one of the layouts was selected for subsequent development in stage 4, after evaluating against necessary criteria (stage 2 design functions). In stage 4, potential materials and manufacturing process combinations, were explored and the adequate material and manufacturing process was selected. Based on the selected material and manufacturing process in stage 4, production operation parameters were then established in stage 5. Throughout each stage, the textual information about the evolving product, was captured using the DFD charts.

8. REFERENCES

[1] Phelan, R. M., "Fundamentals of Mechanical Design", 3rd Edition, McGraw-Hill Co., NY, USA, 1970, pp 229 - 230.

[2] Car Care, "Be Your Own Motor Mechanic", 1st Edition, Eaglemoss Publications Ltd., UK, 1985.

[3] Garratt, J., "Design and Technology", 1st Edition, Cambridge University Press, Cambridge, UK, 1991.

[4] Kalpakjian, S., "Manufacturing Engineering and Technology", Addison-Wesley Publishing Co., USA, 1989.

[5] Beadle, J. D., (ed), "Castings", Production Engineering Series, The Macmillian Press Ltd., UK, 1971.

[6] Barton, H. K. & Barton, L. C., "Diecasting Die Design", The Machinery Publishing Co. Ltd., London, 1956

[7] Doehler, H. H., "Die Casting", McGraw-Hill Co. Inc., NY, USA, 1951.

[8] Beeley, P. R., "Foundry Technology", 1st Edition, Newnes-Butterworths, London, UK, 1979.

[9] Scholes, J. P., "The Selection and Use of Cast Irons", Engineering Design Guides 31, Published by the Design Council, BSI and Council of Engineering Institutions, Oxford University Press, UK, 1979.