UNIVERSITY OF STRATHCLYDE

Department of Design, Manufacture and Engineering Management

Development of Spine Design Process and Implementation of Axiomatic Design Theory for Cyber Physical System Design Analysis

by Xinyu Yang

A thesis presented in the fulfilment of the requirements of the degree of Master of Philosophy

October 2016

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which led to the award of a degree.

Signed:

Date:

Abstract

This thesis presents a Spine Design Process (SDP) combining Axiomatic Design Theory to provide a unified understanding of the design process for Cyber Physical System (CPS) early stage design analysis.

The thesis begins with an overview of CPS and recent literature relating to problem solving and creative thinking process, design principles and philosophies, and design process models in the areas of mechanical, software, and multidisciplinary. The literature found that there is lacking a unified view of process models in different engineering domains that could fully support the design and analysis of CPS. For the scope of CPS, it is necessary to form a model that capable of covering several process models to generate a unified design process.

Following the literature review, a SDP process model is proposed to serve as a unified approach for CPS system design, adapting concepts from problem solving process, axiomatic design, and engineering design process models. The SDP process consists of six main stages namely: analysis, interpretation, concept, elaboration, construction, and operation. Axioms, the essence of design described in Axiomatic Design Theory, are used in each design stage for design evaluation and validation. The process illustrating a sequential prescriptive design approach potentially can be used in general cases of CPS design. The unified approach in SDP provides a new way of combining design processes in different engineering domains by using Axiomatic Design Theory.

Two case studies were conducted to evaluate the use of the proposed process model. The first case study describes the use of the process in designing and analysing of an automation cake icing forming system. The system uses a four degrees of freedom robotic arm and a vision system to complete the task. The process was used to analysis the hierarchy structure of the system as well as design validation and evaluation. The second case study presented in the thesis analyses the architecture design of a virtual reality system that is used to simulate and model coal shearer. The use of the process took a top down approach in analysing the existing structure of the system.

In conclusion, it is found that a unified view of design process can be used to support design analysis and systematic thinking of a design project and Axiomatic Design Theory is useful for design and validation of CPS. In addition, the thesis extends the use of Axiomatic Design Theory to CPS systems for it provides strong analytical method for system evaluation which is useful in SDP process model.

Contents

List of Publications

1. **Xinyu Yang**, Xiutian Yan. The Development of Spine Design Process and Synthesis Reasoning Model for Early Stage Design Analysis and Validation for Cyber Physical System. Eleventh International Conference on Design Principles & Practices, Toronto, Canada, March 2017. Proposal accepted
2. Wang Xuewen, Qin Yi, **Yang Xinyu**. *Et al.* Study on Virtual Assembly Simulation System of Coal Shearer. *Journal of Graphics*. 2015, 36(2)
3. **Yang Xinyu**, Yan Xiutian. Spine Design Methodology (SDM) for Cyber-Physical System. *Six China-Scotland SIPRA Workshop*. Stirling, United Kingdom. 2015 June. Best poster award

List of Figures

List of Tables

Chapter 1 Introduction and Background

1.1 Introduction

Design of system is comprised of intensive human activities. Design process models have been proposed to manage design activities. The development of design process models provide guidance for a design team or an individual to understand a new design task or analysis an existing design. Different process models have been proposed based on various implementation areas and situations. Systems are consistently changing with the advent of new ideas and technologies. It is essential for designers to equip with a progressive view of the design process and the system. The process model needs to be changed or updated along with the progression of newly appeared systems to match the need of designers. This thesis introduces a design process model for design analysis of Cyber-Physical System (CPS). CPS is a newly emerged field in recent years that offers close interaction between cyber and physical components. CPS has been identified as a major role in the design and development of systems in the future. Advanced automations could be made by deploying large scale CPS that connects all physical components and interacts with them.

1.2 Motivation

The world has foreseen the benefits of developing CPS and there are hundreds of tools have been developed to ease the way of designing tasks. Tools ranging from architecture and modelling to final verifications are all available for designers to use. Despite the tools are there, designers tend to be more curious about the design process. What design approaches to take to accomplish design goals? How to choose tools that suitable for design task? How to manage multidisciplinary design work in a minimum use of resources?

To deal with those issues, a design process model is important for designers to understand the process in a holistic manner. CPS requires an integration of various domains of knowledge and it is critical for designers keep exploring knowledge from other disciplines in order to make better products or systems. The motivation for this research was to bridge the previously dispersed knowledge into a unified framework in order to:

- combine design process models into a unified one;
- help designers analyse their design early in the project to save time and costs, and
- encourage the study of multidisciplinary analysis and design of CPS.

1.3 Research Aim and Objectives

With previous introduction and discussion, the aim and objectives of this research are presented below. The aim of this research was to develop a unified process model that incorporates recognised mechanical, software and system engineering process models to support early stage design analysis for CPS.

The objectives achieved in this research are all contributed to the research aim. Through investigation, the following objectives were addressed in the thesis:

- Brief introduction and understanding of Cyber Physical System (O1).

- Review of problem solving and human creative thinking process that support system design (O2).
- Review of Axiomatic Design Theory design principles for design and decision making (O3).
- Review of system design process models in mechanical engineering process, software engineering process, and multidisciplinary process (O4).
- Develop process model to support early stage analysis and design of CPS (O5).
- Evaluate process model in case studies (O6).
- Review and proposals for further work (O7).

1.4 Research Methodology

The main structure of this research project was based on studying previously published literature, and analysing and conducting case studies. Conducting an appropriate literature review is a means of demonstrating knowledge in a particular study field [1]. The literature provides a vital role in delimiting the research problem, seeking new lines of inquiry, avoiding fruitless approaches, gaining methodological insights, identifying recommendations for further research, and seeking support for grounded theory [2]. It is important to conduct literature review in a systematic manner. Case studies, on the other hand, are empirical and practical study of real life content and it helps to define and evaluate research topics [3].

The main focus of literature review in this project was on the established design theories and design processes in order to identify gaps and clear research topics.

Major electronic databases were searched for related literature including Google Scholar, Science Direct, IEEE, ACM, etc. The literature was grouped by its disciplines and implementation areas. The following groups were identified from analysing literature: general knowledge of CPS, design processes in mechanical system, design processes in software system, design process in multidisciplinary system, design principles, and design thinking and problem solving process. The missing part in the literature was identified followed by developing a design process that fits the missing part.

Case studies were then performed to evaluate the proposed design process. Two case studies were conducted in this project for different purposes. The first case study was to evaluate the use of the proposed process in system design, validation and evaluation. The case study was in great details to conduct the actual design analysis and to evaluate the design process. The second case study was to analyse an existing design architecture by utilising the proposed process. Figure 1.1 shows the methodological relationships of different sections in this research.

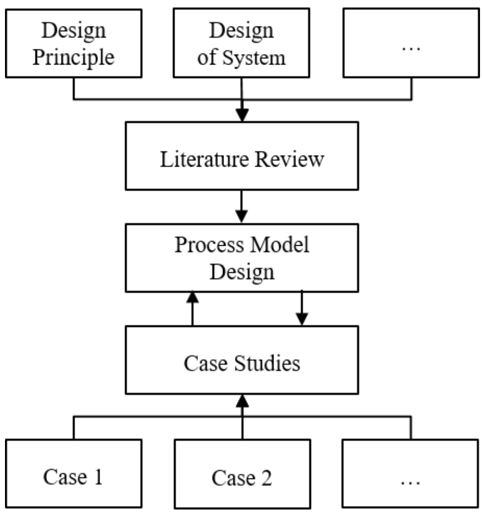


Figure 1.1 Research Process

1.5 Structure of Thesis

The thesis was constructed in a logical manner. Figure 1.2 provides an overview of the relationships of each chapter in the thesis.

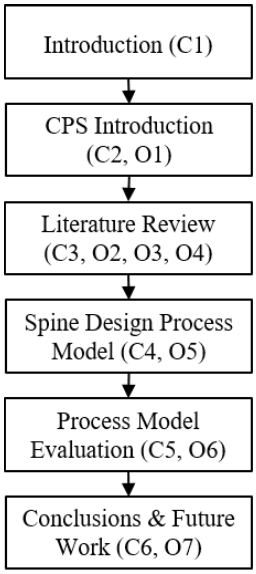


Figure 1.2 Thesis Structure

Chapter 1 provides an introduction to the thesis, including research background, aim and objectives, motivations, research methodology, and thesis structure.

Chapter 2 presents an overview of CPS by reviewing the background of the system, the characteristics of the system, and the application areas of the system.

Chapter 3 presents a review of current literature in related fields including, problem solving process, mechanical system design process, software design process, multidisciplinary design process, design theory and principle. The review of related domains of knowledge provides an in-depth view of the missing part in the literature that forms the model in Chapter 4.

In Chapter 4, Spine Design Process (SDP) is presented. SDP is a six stage process model. The aim of the SDP is to provide a holistic view of design process that addresses physical (mainly mechanical), cyber (mainly software), sensing, and actuating domains. By adopting Axiomatic Design Theory, the process model could support early stage design analysis, including requirement analysis, system decomposition, development of design matrix, and early stage design checking.

In Chapter 5, two case studies are presented to demonstrate and evaluate the use of the process model proposed in Chapter 4.

The thesis ends in Chapter 6 with conclusions, limitations and recommendations for future work.

Chapter 2 Cyber Physical System

2.1 Introduction

This chapter gives a brief introduction about Cyber Physical System to address objective 1,

- *Brief introduction and understanding of Cyber Physical System (O1)*

This chapter provides some general background about the development of CPS, the characteristics of CPS, and the applications of CPS.

2.2 Background and History of Cyber Physical System

CPS was first coined in 2006 by Helen Gill at National Science Foundations (NSF) in the US. Since that, numerous workshops, conferences have been held by research community. Figure 2.1 shows a timeline for CPS key events in recent years. Workshops and conferences pushed the research intensity for CPS, and it became a hot research topic since then.

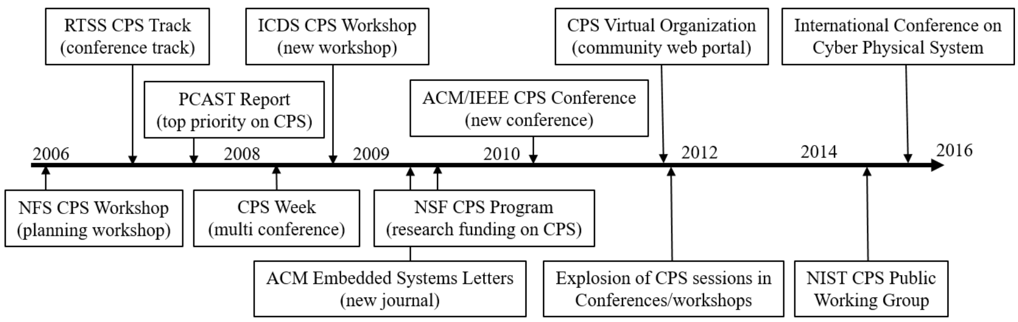


Figure 2.1 CPS Timeline

"CPSs are integration of computation with physical process. Embedded computers and networks monitor and control the physical process, usually with feedback loops where physical processes affect computers and vice versa." [4]

The term CPS has a broad and general meaning than Internet of Thing and embedded system or mechatronic system. Internet of Thing is a general term used to describe highly connected world in which systems are linked to other systems through network or other types of media. Internet of Things, in most cases, are open systems, because they share information with other systems and are affected by information from others. For example, a person can access to his/her cloud storage service anywhere on the planet where Internet is available, download or upload contents. In contrast to Internet of Thing, embedded system and mechatronic system are closed systems where information is only used locally or not in a large scale, but these types of systems have local precision control and robustness. However, CPS, on the other hand, is a combination of both the Internet of Things and embedded systems. CPS system has to provide high connectivity to current closed system and meanwhile maintain robustness of the original system. The connection of various systems generate a large amount of data that need to be captured and analysed via computation. The results of computed information is used for further decision making and control of current system.

"The economic and societal potential of such systems (CPS) is vastly greater than what has been realized, and major investments are being made worldwide to develop the technology." [5]

The potential economic and social impacts of CPS attract academia and industry pouring vast amount of funds to develop new technologies and tools to design such systems. The world is changing dramatically fast as new technologies are being implemented around us. To keep track of the changes, it is important for designers to understand CPS design from a systematic perspective in order to capture full abilities and potentials of future movements.

However, the design for CPS is a hard and long work. Due to interdisciplinary nature of CPS, it becomes very hard for people, especially inexperienced designers or student, to fully understand design information for CPS. Therefore, it is important to address design clearly at early stage and let designers fully understand the design process for CPS. The early stages of design analysis and planning would significantly reduce costs and time consuming later in the project.

2.3 Characteristics of Cyber Physical System

In general, a CPS is a highly interactive system connecting intangible cyber world and tangible physical world [4-8]. Breakthrough of design tools and technologies extend the ability of system engineers to develop more advanced, large scale, and multidisciplinary systems in a cost effective and time saving manner. The world is connected by communication networks, sensed by distributed sensor networks, monitored by computation systems, and controlled by physical actuator systems. CPS provides cyber capability to physical components monitoring and controlling physical components, and works in a highly automatic way [4, 5]. The potential impact of CPS is tremendous, either from technology development or from society economic improvement. The world has foreseen the benefits of developing such systems.

CPS focuses on four major domains, physical domain, cyber domain, sensing domain, and actuating domain. The four domains are intertwined together serving as a functional architecture that forms the core of CPS. A common CPS architecture is shown in Figure 2.2. Direction of data flow is illustrated in the architecture model using arrow lines. Sensing and actuating domain bridge the gap between cyber and physical domain. Through sensing domain, data can be captured and transmitted to cyber domain. On the other hand, physical domain receives data in the form of control signal from actuating domain. The paradigm is similar to close loop control system but in a more general manner and has more complexity.

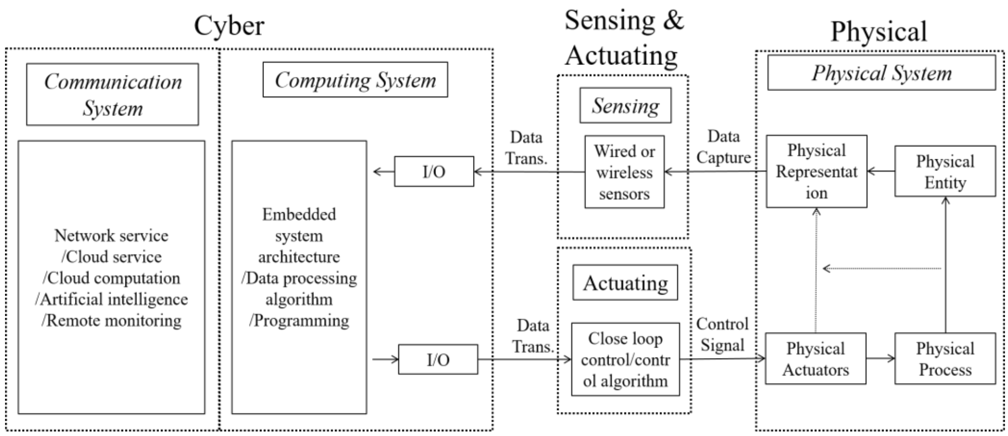


Figure 2.2 CPS Architecture Illustration

From the review of the literature, the CPS shows some key characteristics that a designer should be aware of when designing such systems. These characteristics are commonly shared by most CPSs. A summary of these characteristics are presented below,

Highly integrated [4-9]. Computational system and physical system are highly integrated in CPS. The physical system is controlled and monitored by computational system possibly in a large scale than traditional embedded and mechatronic systems.

Distributed and networked [6, 8, 10, 11]. CPS can be connected by communication and sensing networks. The development in communication technologies enable large scale wireless sensing network to be deployed in remote places for controlling and monitoring the physical entities.

Complex and multidisciplinary [4, 7, 8, 12]. CPS not only involve one single engineering discipline. It is a complex and multidisciplinary system that needs knowledge from many different knowledge domains.

Reorganising and reconfiguring [13, 14]. CPS needs to be flexible. The system can be adapted to fit to new needs or add new functions. So that the system is capable of achieving more functions under different implementation scenarios.

Robustness and dependability [4, 7, 9]. CPS is designed to be reliable. Some application areas (for example avionics and medical devices) require highly dependable and robust systems to be developed, thus robustness and dependability are the key characteristics for CPS.

Safety, privacy and security [7, 9]. As the CPS always operates in a network environment. It is essential to ensure the security of the data as well as the privacy. Cyberattacks should be aware when designing large scale distributed systems.

2.4 Applications of Cyber Physical System

The application areas for CPS are tremendously large. Some important application areas including medical and healthcare, manufacture and transportation, power grid and energy system. This section gives some examples about CPS that have been already implemented or some will be implemented in the near future.

2.4.1 Medical and Healthcare

CPS gives new opportunities for intelligent healthcare and medical systems. Devices deployed in hospitals are life critical systems that provide high quality continuous care for patients. Patients are the objects for medical and healthcare systems, due to the target is human beings, the CPS medial systems need to achieve high reliability, security and privacy, and device verifiability [15]. A typical architecture diagram is shown in Figure 2.3, the interaction relationships between patients, medical devices, and caregivers.

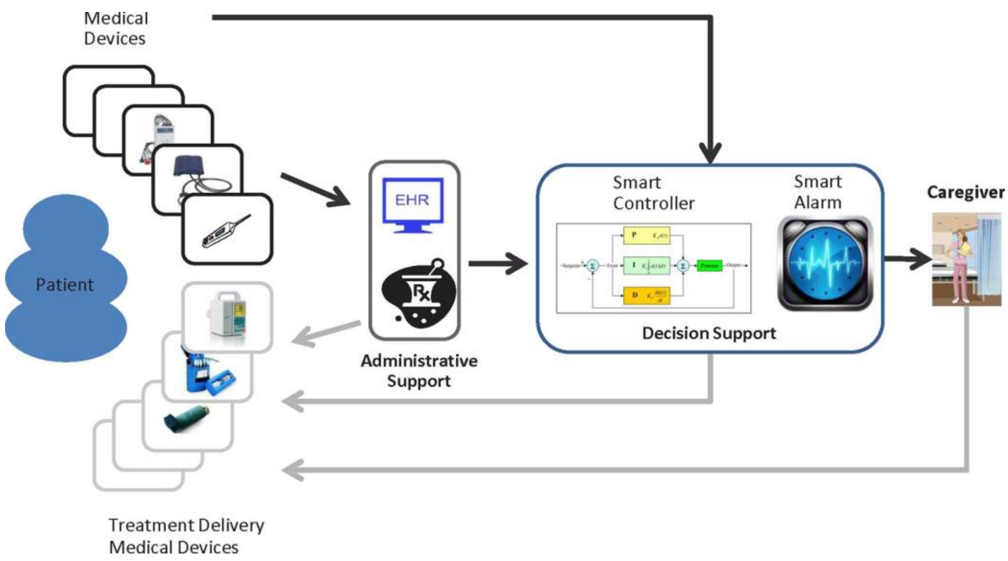


Figure 2.3 Conceptual CPS Structure for Medical System [15]

For example, Lim *et al* [16] propose a healthcare CPS for frail elderly people and people with disabilities. The proposed system is intended to capture people's daily activities and uses scenario based functional design to analyse people's actions in order to provide services at desired locations. The system reminds the user of important activities such as taking drags. The proposed system uses remote computing system and sensor system to collect and analyse data from private or home space and public space. Figure 2.4 show an architecture diagram used in the system in order to capture information in private space for a user.

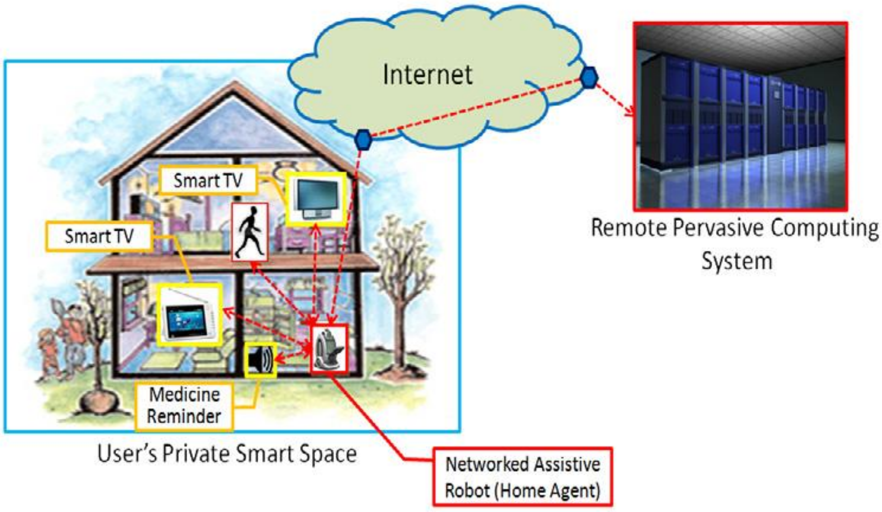


Figure 2.4 Architecture Diagram for CPS interactive system [16]

2.4.2 Manufacturing and Transportation

In addition to the medical and healthcare areas, manufacturing and transportation are other major areas that would benefit from the development of CPS. Modern vehicles are equipped with CPSs such as vehicle information display, in car entertainment system, adaptive cruise control system, etc. Vehicles have hundreds of electronic devices installed to monitor the important parameters in a vehicle. The system is highly integrated and communicated with each other to provide reliable services for the vehicle user.

Some of the transformation in transportation system may have long term effects on the societies. For example, the development of self-driving vehicle requires multidisciplinary knowledge from different engineering domains. The self-driving vehicle is highly interacted with physical space and requires high computation capabilities and advanced algorithm in achieving driving actions for a vehicle.

In the manufacture industry, automation has been made year after year to improve production efficiency and quality of products. The implementation of CPS would provide more advanced manufacture solutions. For example, a CPS manufacture system architecture is proposed in [17]. The proposed smart manufacture system has five abstraction layers named as configuration level, cognitive level, cyber level, data to information conversion level, and smart connect level, as shown in Figure 2.5. The advanced connectivity of physical parts ensures data can be captured in real-time and the intelligent analytical and computational capabilities provided in the cyber space can analyse those data and create value.

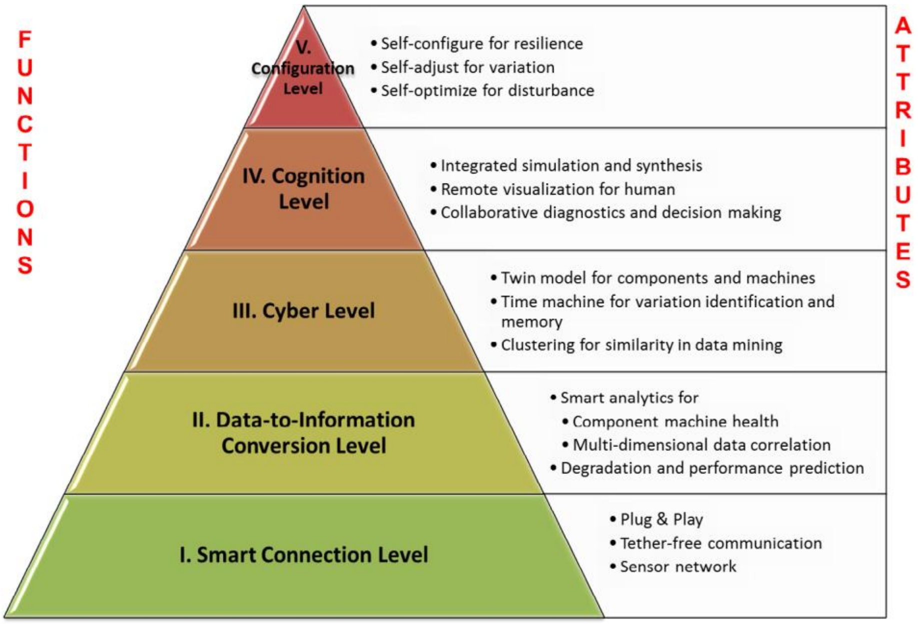


Figure 2.5 5C Architecture for Implementation of CPS [17]

2.4.3 Power Grid and Energy System

Another major implementation area for CPS is in the energy industry. The power supply systems are complicated and difficult to control, CPS provide large scale sensing and actuating capabilities that can be implemented to the design of smart grid systems. The power consumption and distribution in an area can be monitored and controlled by deploying these technologies to improve the performance of energy system [18].

2.5 Conclusions from Cyber Physical System

This chapter gives a brief introduction about the background of CPS to enable a basic understanding of what it is and how it is important for the future development. Some common characteristics of CPS are then summarised. These characteristics may not new in some disciplines, but the combination of them makes CPS special towards the traditional engineering systems. Some application areas are then reviewed with some examples to show the potentials of these systems.

Chapter 3 Literature Review

3.1 Introduction

Three objectives are addressed in this chapter to form the basis of the literature review.

- *Review of problem solving and human creative thinking process that support system design. (O2)*
- *Review of Axiomatic Design Theory design principles for design and decision making. (O3)*
- *Review of system design process models in mechanical engineering process, software engineering process, and multidisciplinary process. (O4)*

The goal of the literature review in this chapter was to identify the most commonly used design process models. By addressing the objectives in this chapter, the key findings of the review were identified and argued in accordance with the aim. The literature was focused on design process models and design theories.

Literature review was split into three main sections, problem solving and creative thinking process in engineering design, design theory and principles, and design process models. Literature was mainly acquired from peer reviewed journals, books, conference publications, and international standards, which provide a good coverage of current knowledge related to this research.

3.2 Problem Solving and Creative Processes

Design is originated from human actions and it is a complex creative thinking process, the study of human thinking process is vitally important to understand how designer's think when designing an artefact. When the complexity of design work increases, it becomes more difficult to manage a design or to be creative, and when the design is a multidisciplinary task, the complexity increases even more. This is partially due to that people have limited working memory that temporarily store information as they performing complex cognitive tasks [19]. When solving a problem, information is being processed in the working memory. Because of the limited memory spaces, the efficiency of problem solving process might decrease when problems become complicated. By understanding the rationale behind human thinking, thinking skills can be trained for individual designers.

Design a CPS is a multidisciplinary task that requires knowledge from different domains, and the system itself includes a relatively large different types of subsystems that integrated together to fulfil the final design objectives. This multidisciplinary design raises the complexity for designers to couple with both quantitative computations and qualitative decision makings. It also increases the difficulties for designers jumping through multiple domains and thinking through various perspectives for a system. A system can be viewed by different professionals in various ways, this also increases the difficulties of understanding and communicating a design. For example, Figure 3.1 shows an architecture diagram of a multidisciplinary system. Depending on a person's knowledge and field of study, the multidisciplinary system is a combination of different subsystems including electromechanical system, mechanical system, electrical system, control system, and software system, etc., various views can be made to the same architecture diagram. Critical reasoning, quantitative computations are needed for each individual domains in a design, the complexity increases when a system encompasses numerous domains.

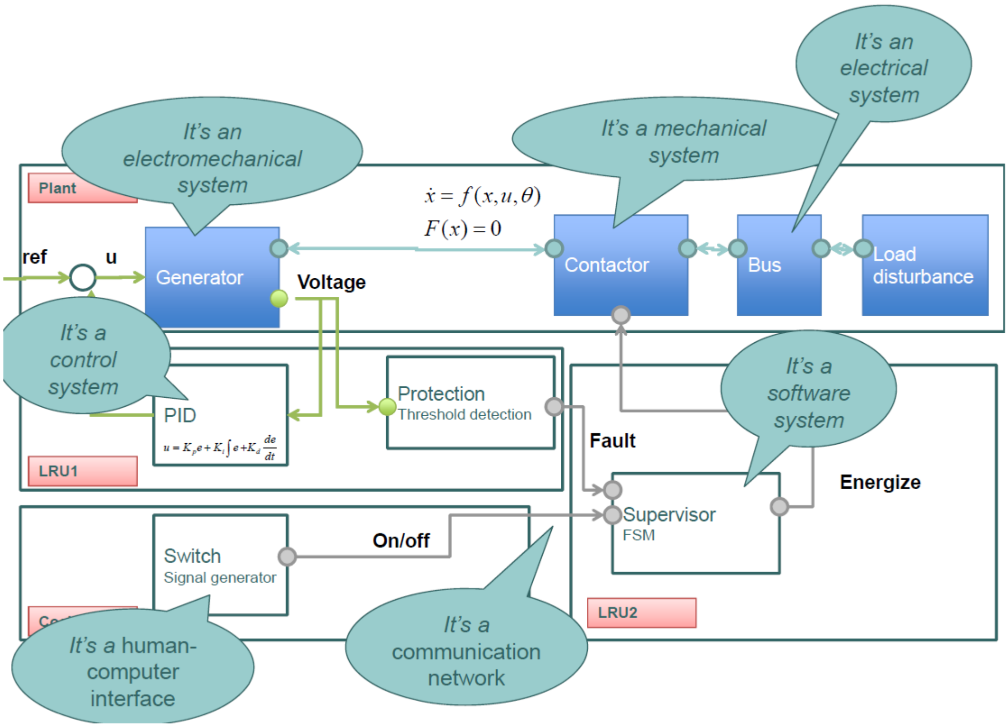


Figure 3.1 Views and Domains in a System Architecture [20]

Kahney [21] argues that the real world problems can be classified into two groups: well-defined problem and ill-defined problem. A well-defined problem has a well-structure that the solver is provided with all the information needed in order to solve the problem. On the other hand, an ill-defined problem has an ill-structure no information or only part of the information is given at the beginning of the problem. For example, Towers of Hanoi is a well-defined problem with definite solutions. On the other hand, passing an exam in university is an ill-defined problem because the task is only vaguely defined.

Despite the problem has different variations, there exists some similarities can be extracted from those problems. Problems share similar definition in dictionary. It is defined as, *"a matter or situation regarded as unwelcome or harmful and needing to be dealt with and overcome" [22].* In addition, problems also have two common states, the initial state and the goal state [21]. The initial state describes the information that a problem has at the beginning of a problem. In contrast to initial state, goal state is final status of a problem. A problem also need legal operators which define a set of tools that can be used for a problem and operator restrictions which define and govern operators and set the boundary for a problem [21].

The creative process in engineering is intended to help designers think out of box in order to get more innovative ideas of problem solving. One of the first creative process model was created in 1920s by Wallas [23]. Wallas identifies four stages for the creative process: preparation, investigating the problem thoroughly and comprehensively; incubation, thinking about the problem; illumination, the moment when the ideas come out; verification, validating and testing of ideas. Dewey [24] proposes a three stages problem solving model including define the problem, identify alternatives, and select best alternatives. Osborn [25] creates a brainstorming method for creative thinking and the model of creative thinking consists of three stages: fact finding, idea finding, and solution finding. Later, the model was extended by adding problem finding and acceptance finding stages in the overall process. Furthermore, Basadur [26] proposes a three stages two steps model that creative problem solving process may take. As shown in Figure 3.2, the stages are comprised of problem finding activity, problem solving activity, and solution implementation activity. Those activities are governed by the outside environment as constraint for thinking process.

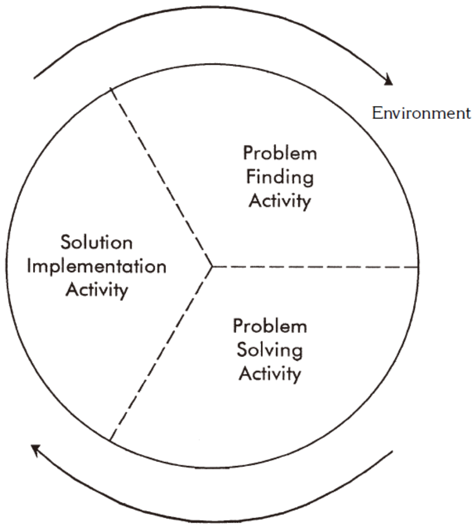


Figure 3.2 Basadur Creative Problem Solving Process [26]

In summary, different process might be taken to conduct a problem solving and creative process, but those processes are sharing some similar elements: the designers, the creative process, the product/artefact/system, and the environment [27].

3.3 Axiomatic Design Theory

Design theory and principle are looking for the similarities in design and design activities from a high abstraction level. Unlike design process, the guidance provided in the theory and principle has a more general usage in support of design. Similarities in design activities are observed from design practice which form the ontology of design process, and further design principles are extracted from analysing existing design solutions from experienced designers and successful designs. This section presents general knowledge of design theory and principle that support the understanding of design process.

Design principles have been widely investigated by researchers. One of the most significant findings of design principle is Axiomatic Design Theory developed by Suh [28-30]. Suh describes that there exists a certain set of natural principles that govern the physical world (the design of artefacts). The principles has two basic axioms named as, Axiom 1, the independent axiom, and Axiom 2, the information axiom. Axiom 1 requires maintaining the independence of the functional requirements and Axiom 2 requires minimising the information content of a design. The theory has been widely used in requirement engineering [31], design of mechanical systems [29], design and management of software system [32, 33], and for enterprise design and management [34]. The methods of classifying systems and design mapping process provided by axiomatic theory are essential for designers to understand the design process. The key idea of this theory is to reduce information entropy contained in a design, in another word to reduce the complexity of design but keep all other functions of the system. Moreover, individual theorems and corollaries are generated based on the two basic axioms. The axiomatic theory also provides a design process which split into four main domains: customer domain, functional domain, physical domain, and process domain.

To illustrate independence axiom, design equation is being developed to define the mapping from functional domain to physical domain. Functional requirements are the elements in functional domain and design parameters are elements in physical domain. A design equation is expressed as [28],

$$\{FR\} = [A]\{DP\} \tag{3.1}$$

where {FR} is the functional requirement vector, {DP} is the design parameter vector, and [A] is the design matrix. Each line of the design matrix can be written as [28],

$$FR_i = \sum_j A_{ij} DP_j \tag{3.2}$$

The design matrix [A] is of form [28],

$$[A] = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix} \tag{3.3}$$

In general, the element A_{ij} may be expressed in a differentiate form as shown below [28],

$$A_{ij} = \frac{\partial FR_i}{\partial DP_j} \tag{3.4}$$

When A_{ij} is not a constant, it must be evaluated at a specific design point in physical domain. A square design matrix occurs when m equals to n, the square matrix is essential in axiomatic design since in an ideal situation of a design, each functional requirement in functional domain can be related to a certain design parameter in physical domain. A typical design matrix has three types of form: uncoupled design, decoupled design, and coupled design. Further definitions can be found in Appendix A. When $m>n$, a couple design results or the functional requirements cannot be satisfied. When $m<n$ the design is redundant or a coupled design. In both of these two cases, new functional requirements or design parameters need to be found. An ideal design occurs

when the number of functional design parameters is equal to the number of functional requirements.

The information in a design is in a form of drawings, equations, material specifications, operational instructions, software, etc. Depending on the types of design, the forms of information may vary. Axiom 2 provides a way of estimating information content in a design. Similar to the definition of information in information theory [35], the information in axiomatic design is defined in terms of probability [28],

$$Information = I = \log_2(\frac{1}{p}) \tag{3.5}$$

where p is the possibility of achieving a certain task. The definition of information is in logarithm form with base 2. In a more general form, the information can be written as [36],

$$I = \log_2(\frac{range}{tolerance}) \tag{3.6}$$

The information axiom is used to choose design solutions when all solutions for a given task are satisfying Axiom 1. A solution has less information is a better solution.

3.4 Design Process Models Overview

Definitions of design and design process model are reviewed first, followed by a review of design process model in mechanical engineering, software engineering, and multidisciplinary engineering. The natural of design process models can be viewed as generalization and abstraction of empirical research through direct or indirect observation. There is a very large amount of literature in the field of design process, some of them have particular application situations, and some are for more general purposes. This section covers some of the major process models developed and used for last decades.

3.4.1 Definition of Design and Design Process Model

All systems, no matter what they may use for power sources and building materials are designed by intensive human activities. It is important that human played an inevitable role in system development and evolvement in which, at present, enabled us to form systematic tools to view, control, and change surrounding physical world in a methodical way. To help the improvement of human society, more complex and advanced systems will be designed and former legacy systems will be upgraded or eliminated. However, the roads to new or upgraded systems are unforeseeable. Therefore, it is important to build a systematic understanding of design and the road that leads to successful designs.

Design, from engineering perspective, can be viewed as a process of accomplishing something and it is a process in which requires tight integrations of many engineering discipline. Laptops, machines, airplanes, cars, buildings, paintings, and many other products which are created for certain purposes can be categorized under the phrase of engineering design. The formal definition of engineering design may vary from different disciplines. However, attempts have been made on to achieve well-accepted and acknowledged definitions. The following expresses some viewpoints on the definition of design.

The word design, defined in dictionary is a noun, refers to "an underlying scheme that governs functioning, developing, or unfolding a plan or protocol for carrying out or accomplishing something." [22] The design scheme can be divided into certain logical processes which represent basic functions, plans, and procedures. By encapsulating and extracting similar attributes of design projects, one can have a better understanding of the relationships among different procedures and functions in order to achieve design objectives.

From engineering perspective, Dym and Little [37] defined engineering design as "a systematic, intelligent process in which designers generate, evaluate and specify designs for devices, systems or processes whose form(s) and function(s) achieve clients' objectives and users' needs while satisfying a specified set of constraints". Dym addes that design as a process should be systematic and intelligent. In other words, human based activities that require high level of reasoning and thinking are important towards a successful design. It also mentions that a design should comply with a group of constraints in which designers can develop boundaries of a design task and obey boundary settings when conducting design work.

Finkelstein [38] defines design as "a creative process which starts from a requirement and defines a contrivance or system and the methods of its realization or implementation, so as to satisfy the requirement. It is a primary human activity and is central to engineering and the applied arts". Design is a human creative activity involving engineering and arts to produce concept static or dynamic design products. It is the new design that generates more profits for an organisation and provides more functionalities for its users.

Evbuomwan and Sivaloganathan [39] describe design as "the process of establishing requirements based on human needs, transforming them into performance specification 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". The definition is a summary of prevailing descriptions of design including the key elements that a design is required: needs, requirements, solutions, specifications, creativity, constraints, scientific principles, technical information, functions, mapping, transformation, manufacture and economics.

The design process, on the other hand, is the illustration of a rationale process that aid the final design solution and in most cases, it is an iterative process. It is the study that help designers work and think [40].

3.4.2 Components of Design Activities

Design is a complex activity with some features and characteristics in nature. Design can take top down or bottom up approaches, may involve evolutionary process, knowledge based exploration, and deep investigation [41]. Design is also a creative, rational, and interactive process that requires designers with various abilities including good memory, pattern recognition abilities, logical reasoning and critical thinking skills, mathematical analysis, computer simulation, etc. [42, 43]. It is essential, for design process research, to understand basic elements that form design activities and reason for ordering the design activities into a structured design process.

Design activities are comprised of a series of components in a logical order. Sim and Duffy [44] summarise design activities into three main groups: design definition, design evaluation, and design management. In each group, detailed activities are being defined and summarised in Table 3.1. The activities are a collection of general approaches that are used in most design processes. A design process may not contain all of the design activities depending on the usage of design process, it may have one or several design activities. This summary is useful in understanding general design activity elements taken in different design tasks. The elements presented in the table are generalised from literature studies and show the usefulness to link general design activity with design ontology.

Table 3.1 Design Activities Composition [44]

Design Definition	**Design Evaluation**	**Design Management**
Synthesizing	Decision making	Constructing
Abstracting	Evaluating	Exploring
Generating	Selecting	Identifying
Decomposing	Analysing	Information gathering
Associating	Modelling	Resolving
Composing	Simulating	Searching
Structuring	Testing	Decomposing
Detailing		Prioritizing
Defining		Planning
Standardizing		Scheduling

3.4.3 Mechanical System Design Process Models

Pugh [45] presents a total design process model, as shown in Figure 3.3, which addresses a systematic approach of performing a design task. From market and user need identification, to satisfying that need through a successful product. The main view provided in total design is that a design should not be treated as a single design that only fulfil the required functionality, but a design that has vertical impacts to all business sectors related to the market, and most importantly generate economic benefits. Total design described that design is driven by the market where customer needs generated. A good design does not only satisfy the product requirements but also generates profits for the design team or the organization. Total design process model gives a comprehensive overview of a design from high level of abstraction that has been used for education and guidelines for designers. Another point that Pugh addressed is the static and dynamic design [46]. A static design means the concept used in a design is not new, only the evolution of existing design. On the contrary, dynamic design means a totally new concept that has not been used by others. For example, Ford manufactured first commercial four wheels car for the market. The four wheels car was a new concept at the time Ford designed the car and it can be treated as a dynamic design. Later on, all cars are designed with four wheels. The concept of four wheels was a static concept after invention, so all cars after that were treated as static design. Suspensions and engines might be different from the original design, but the concept is fixed. All cars that are used now are based on four wheels.

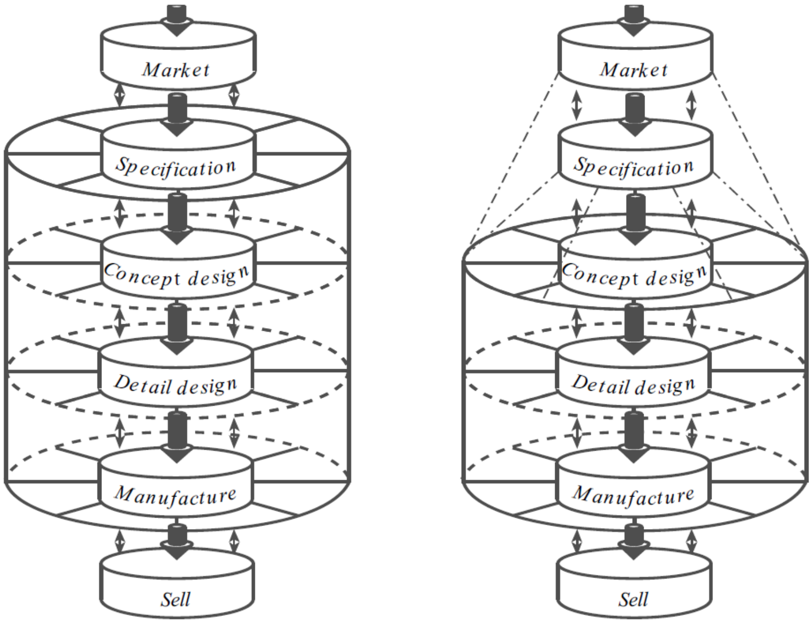


Figure 3.3 Total Design Process Model [45]

Pahl and Beitz [47] proposed five points that a design has impact on or aspects should be considered: affecting nearly all areas of human life, using the laws and insights of science, building upon special experience, providing the prerequisites for the physical realization of solution ideas, and requiring professional integrity and responsibility. Pahl and Beitz's five points argued the essence of the final design should have. In addition, Pahl and Beitz presented a systematic practical approach process to engineering design, as shown in Figure 3.4. The design process focuses on physical and mechanical products illustrating six design stages named as task clarification, requirement list, principle solution, preliminary layout, definitive layout, and product documentation. The stages listed here are considered to be the most useful strategic guidelines for designers. The model is a highly mechanical oriented model and the original aim of the model is to suit for designers and students in mechanical engineering domain. The model does not deal with the design of software or program during concept stage and cyber system that may require to be developed.

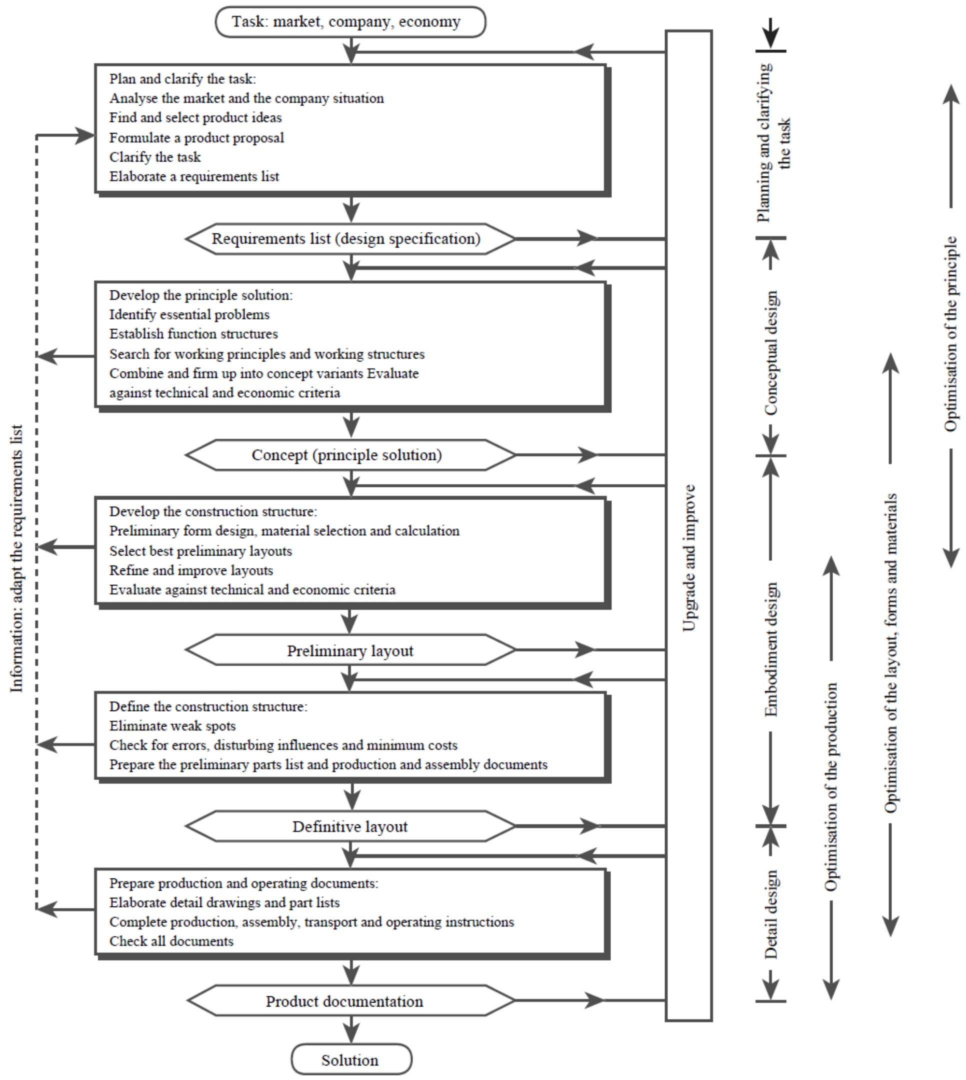


Figure 3.4 Pahl and Beitz Design Process Model [47]

French [48] also produced a model similar to Pahl's model which consists of four main stages as shown in Figure 3.5. The process begins with market needs and leads to an analysis of the needs which generate a list of requirements that system needs to fulfil. Several concepts are generated during conceptual stage and transformed into more concrete representations. The chosen concepts are further detailed in the embodiment and detailing stages where final drawings or workings for manufacture are produced. French's model provides a high level of abstraction of design process without providing much tools in each design stage.

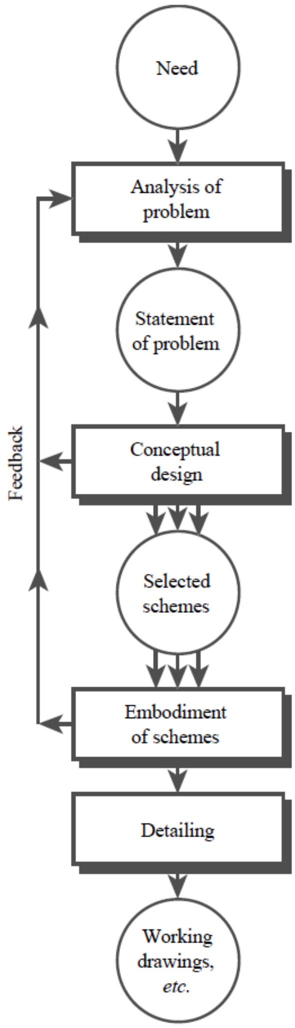


Figure 3.5 French Concept Model [48]

In Figure 3.6, Cross [49] mentioned a six stages process in a symmetrical form where the outline of the model consists problem solution model. The six stages cover the design process in clarifying objectives, establishing functions, setting requirements, generating alternatives, evaluating alternatives, and improving details. In each stage, different methods are used to achieve certain objectives in that stage. For example, objectives tree method is used to clarify objectives in the first stage and function analysis method is used to establish design boundaries for the second stage.

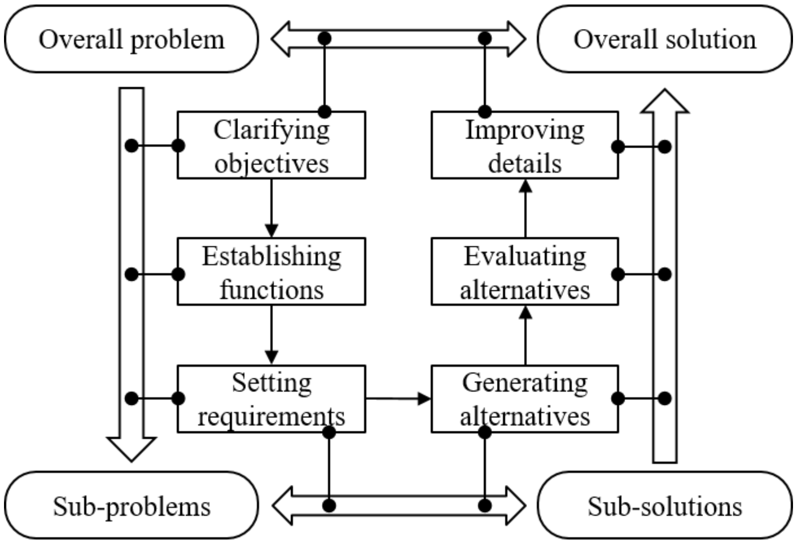


Figure 3.6 Design Model by Cross [49]

BS 7000-2-2015 [50] provides a very general guideline of design process from two different levels: organizational level and project level. The organizational level concerns about higher level of project management from organisational perspective including responsibility management, organisational design philosophy, investment management, infrastructure setup, market positioning, promoting and selling products, planning and communication, evaluation. The project level of 7000 series concerns about product design process including commissioning, operation, maintenance and end of life consideration, and human interface of the product concerning about aesthetics, operational expectations and ease of use of the product.

Except those design process models mentioned above, there are some other process models available in the literature addressing the mechanical engineering system. Asimow [51] proposes a three phase design approach early in the 1960s.

The feasibility study phase, the preliminary design phase, and the detailed design phase. Watts [52] describes a design model that in relation with the design environment and the model also has three processes: analysis, synthesis, and evaluation. Marples [53] proposes a structured design process with case studies to illustrate three principal phases in the design: synthesis, evaluation, and decision. Archer [54] describes design process into six stages as programming, data collection, analysis, synthesis, development, and communication. The six stages are grouped into three phases named as analytic, creative, and executive. Krick [55] illustrates design in five stages from problem formulation, problem analysis, search, to decision and specification. Harris [56] develops a design process model for education purpose with five stages: appreciation of the task, conception, appraisal of concepts, decision, checking and elaboration.

3.4.4 Software System Design Process Models

The waterfall model [57], in Figure 3.7, represents a sequential design process. Design starts from requirements and ends in operations. Each succeeding step can only be started after its previous step. It clears of what objectives need to be achieved in each stage. The big vision or big picture of the whole system is captured at the initiation stage of the whole design. The process of waterfall model goes through system requirements, software requirements, analysis, program design, coding, testing, and operations. One of most important aspects mentioned in waterfall model is that the model emphasis on the documentation of each design stages. Software designed by waterfall model seems to be difficult to change late in the project due to the inflexibility of the design process [58].

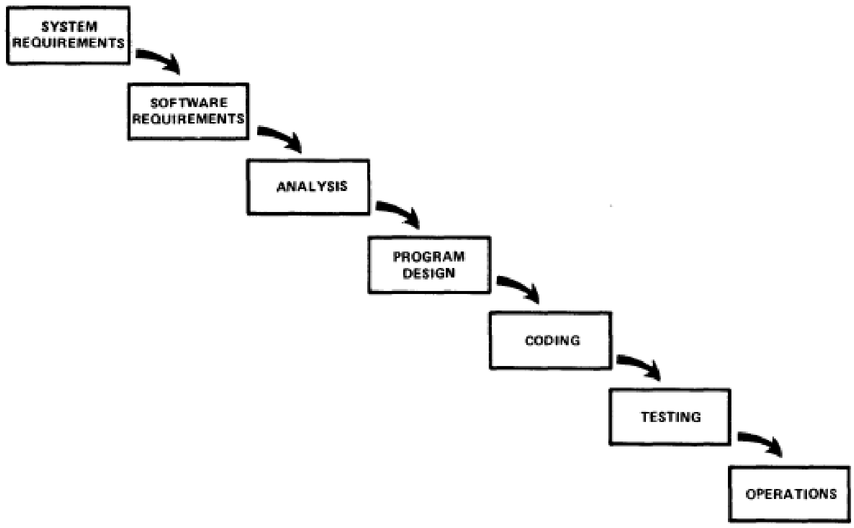


Figure 3.7 Waterfall Design Process [57]

Spiral model [58] is designed to provide more agility for the design process, as shown in Figure 3.8. It introduces a spiral shaped iterative process which puts design into four different phases. The first phase determines objectives of the design task, the second phase evaluates alternatives, identifies and resolves risks, and the third phase develops, verifies next level product, the final phase provides plan for next phase of iteration. Spiral process first introduces risk analysis into the design of software system.

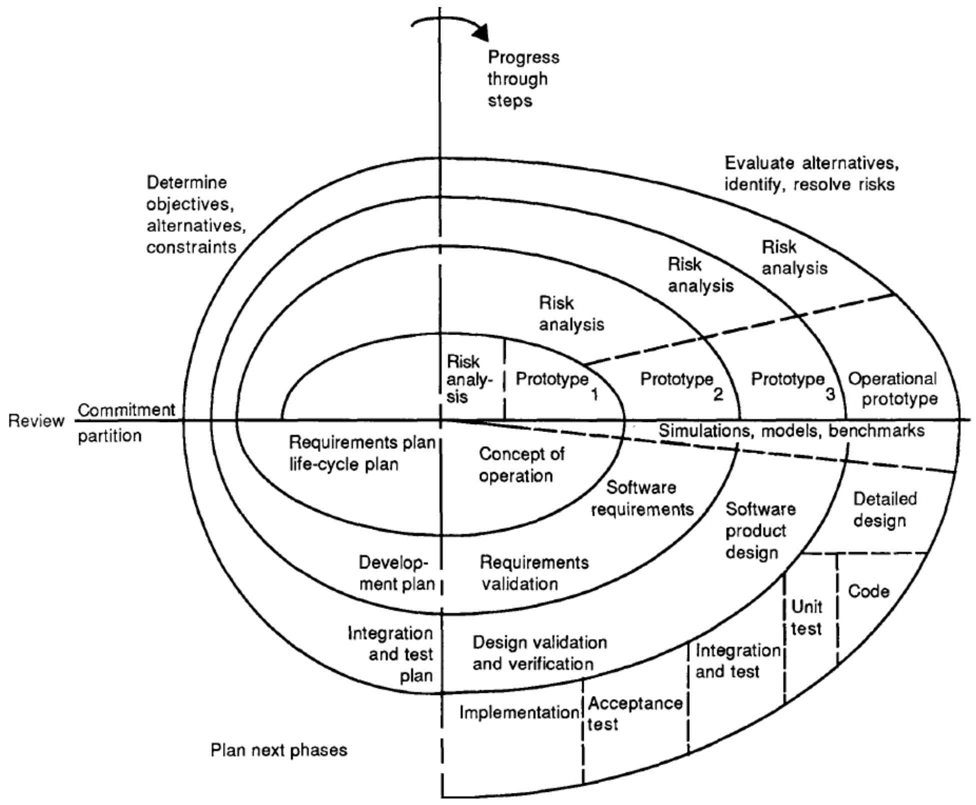


Figure 3.8 Spiral Design Process [58]

Watersluice [59], as shown in Figure 3.9, provides a view of combining iterative nature of the cyclical methodology (Spiral process) with the steady progression of the sequential methodology (Waterfall process). It is a combination of waterfall and spiral process providing a more accurate representation of current software engineering practices.

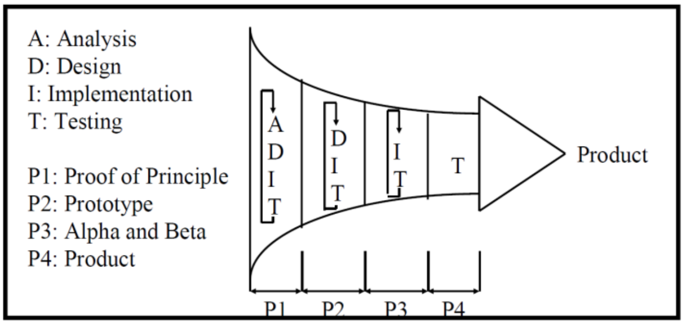


Figure 3.9 Watersluice Methodology [59]

The Rational Unified Process (RUP) [60] framework is proposed by Rational Corporation in 2003. The process describes a six best practices approach: developing software iteratively, manage requirements, using component based architectures, visualising software, verifying software quality, and controlling changes to software. The process splits design into four phases in the time axis named as inception phase, elaboration phase, construction phase, and transition phase. The process also has nine steps vertically along with the content of organization. The nine steps include business modelling, requirements, analysis and design, implementation, test, deployment, configuration and change management, project management, and environment. Figure 3.10 illustrates the view in rational unified process.

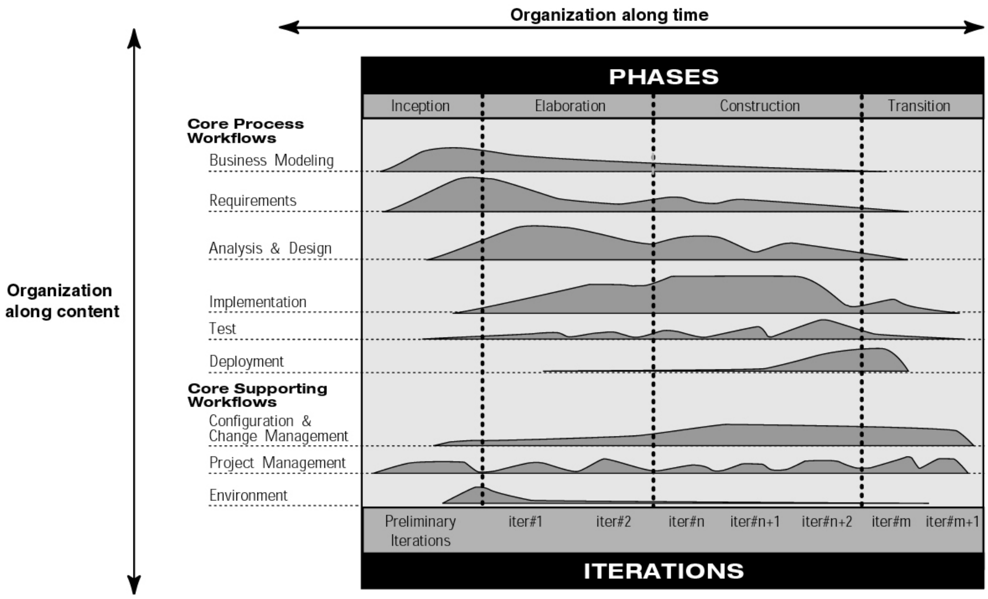


Figure 3.10 Rational Unified Process [60]

Software design process is a rapidly changing field of study, some of other methods are also need to be aware of. The extreme programing [61] aims to produce executable software in a very limited time to test the feasibility of the program. A model driven engineering [62] is also suitable for software system for usable systems and to alleviate the complexity of a design project.

3.4.5 Multidisciplinary Process Models

Prevailing ISO 15288 [63], provides a clear general framework of system engineering describing the life cycle of systems created by humans. It defines a set of processes from an engineering perspective that can be implemented in the hierarchy structure of a system. The standard also provides lifecycle process to define, control, and improve a project in an organisation. The standard defines four processes into four categories named as agreement processes, technical management processes, technical processes, and organisational project-enabling processes. Each category has been broken done into detailed processes for the reference of designers. The standard provides overview of the lifecycle process but does not provide detailed methods of completing certain design of systems.

Another model that is frequently used in the system and multidisciplinary design is the V-model [64]. V-model provides verification and validation process along the system development process and also illustrates the design lifecycle process for a product or a system. V-model is simple and easy to use and has various applications for many types of design. The application of V-model can be found in mechanical design and an adapted version of V-model is also used for designing software system.

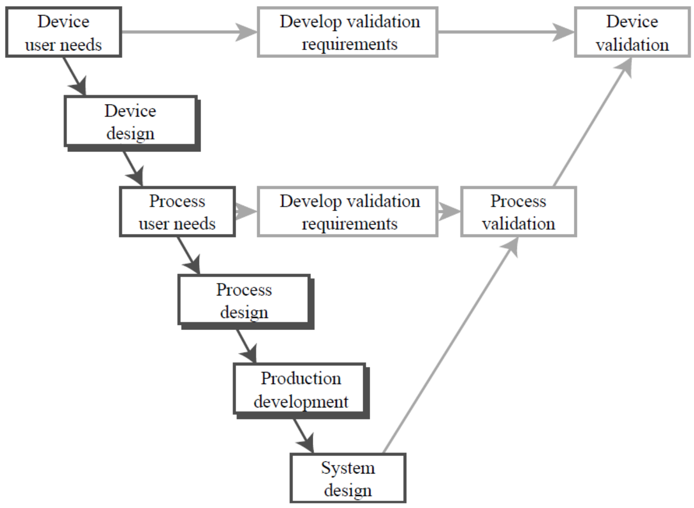


Figure 3.11 V-model for system design [64]

Jensen *et al* [65] proposed a model based design approach for the development of CPS. The model based design approach consists of ten steps: statement of the problem, model physical processes, characterise the problem, derive a control algorithm, select models of computation, specify hardware, simulate, construct, synthesise software, and verification of the system. A case study of a tunnel balling device is illustrated to demonstrate the use of the model based model.

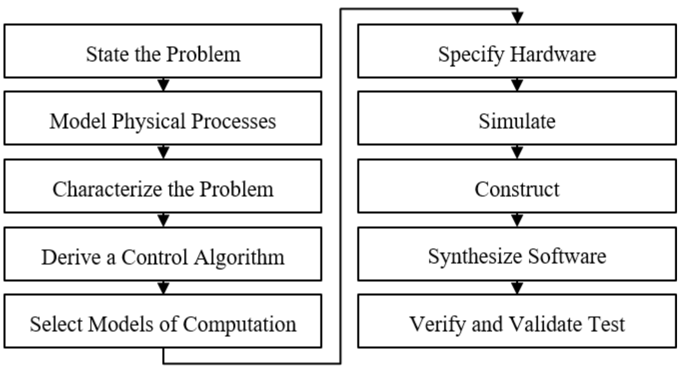


Figure 3.12 CPS Design Model by Jensen [65]

As shown in Figure 3.12, the model described by Jensen for CPS is a procedural process that encompasses ten different processes. As specified in the process model, Jensen's model is software oriented with emphasis on capturing control algorithm, acquiring model of computations. It does not illustrate the process of physical or mechanical side of a design. Comparing to the Axiomatic Design, it does not support analytical analysis of design at early stages.

Lean product development is a systematic approach widely adopted in industry to eliminate waste in manufacturing systems. The term lean was first coined by John Krafick [66] in 1988 and the management philosophy of lean manufacturing was largely derived from Toyota Production System [67]. The core philosophy of lean approach is to achieve totally waste free operations by continuously and simplifying all processes. It improves the delivery time of product, manufacturing productivity, and quality of final product. The objective of lean principles in product development is to identify value activities and eliminate non-value activities. The product is developed based upon customer requirements and must be based on proven knowledge and experience. Wastes need to be identified from all processes. For example, wastes in manufacturing have been identified in seven types: over production, waiting, transportation, inventory, motion, over processing, and rework [67]. Lean approach provides the philosophy for high quality and efficiency management and has been adopted by many companies in industry.

Another systematic approach that is also widely used in industry is concurrent engineering. Concurrent engineering is a relatively new design management system that matured in recent years to become a well-recognised approach in engineering design [68]. The basic idea of concurrent engineering is that all elements, such as functions, assemblies, and maintenances, in a product life cycle should be considered carefully early in the design phases [69]. Another idea in concurrent engineering, as the name suggested, is to conduct the design activities at the same time, concurrently. The concurrent process significantly increases productivity and product quality. The concurrent engineering approach is a favourable way in managing design teams in organisations.

3.5 Conclusions from Literature Review

This chapter reviews current literature in the following fields: problem solving and creative process, design theory and principle, physical/mechanical system design process models, software system design process models, system and multidisciplinary approaches.

A review of different process models are made in this chapter with a wide selection of process models in different engineering domains. The selection is focused on classical design process models. For example, in the field of mechanical engineering, Pugh, Pahl and Beitz, French, and Cross models are extensively reviewed. In the field of software engineering, Waterfall model, Spiral model, Watersluice model, and Rational Unified Process are reviewed. For the multidisciplinary field such as mechatronic systems, V model, BS 7000-2-2015, ISO 15288 are reviewed. In addition, the industrial utilised models such as lean approach and concurrent engineering process are also reviewed. To the best of the author's knowledge, there are limited literature on the process models particularly for CPS.

The current process models in the literature provide some in-depth views of a general design process. Process models have been developed by pioneers under different using scenarios and for different purposes. Furthermore, most of the design process models were developed and used for a few decades in the well-established disciplines, i.e. software engineering and mechanical engineering. It is essential to rethink and revisit those models for the new aim of CPS design and analysis. Since most of the process models are dealt with specific domains, i.e. only in mechanical domain or in software domain, however, for the scope of CPS it is important to propose a model that can handle both mechanical and software domain. The Axiomatic design is a powerful tool that intends to solve design problem through analytical analysis of design. It can be extended to the scope of CPS design.

The literature contains a large amount of papers and books talking about the design process. However, it seldom to find some comparisons among different models. Table 3.2 illustrates a comparison among the design processes reviewed in this chapter. Phases have been identified to use as the basic comparison datum. Those process models have been developed from different backgrounds but share some similar design activities from design phase perspective. To the aim of this research project, the comparison table shows the common points that a unified process should have to effectively combine different approaches that suit the need for CPS design and analysis.

Current process models are domain oriented, especially on software and mechanical engineering. Design and evaluation tools are designed for each domains of engineering. System engineering, on the other hand, provides some system approaches at the high level of design planning and management. But the approaches are very general and abstract. The approaches also need to be used in conjunction with other design processes. The process models in the literature do provide some system evaluation of design, but the evaluation is limited to certain domain. On the other hand, the Axioms approach provides high level decision making and design evaluation support in a generalised way. To combine general design process with Axioms approach would be a suitable approaches, and furthermore there is lacking a process in less abstractive manner that addresses CPS design and analysis. CPS is an emerging domain and it tends to be an independent discipline in the near future [5]. Therefore, it is essential to develop a design process that dedicated for CPS and could support general design analysis.

Table 3.2 Design Process Comparison

Models	**Establishing a need phase**	**Analysis of task phase**	**Conceptual design phase**	**Embodiment design phase**	**Detailed design phase**	**Implementation phase**
Pugh [46]	Market	Specification	Concept design		Detail design	Manufacture/Sell
Pahl and Beitz [47]	Task	Clarification of task	Conceptual design	Embodiment design	Detailed design	X
French [48]	Need	Analysis of problem	Conceptual design	Embodiment schemes	Detailing	X
Cross [49]	X	Exploration	Generation	Evaluation	Communication	X
BS7000 [50]	Concept/Feasibility		Implementation/Realisation			Termination
Waterfall [57]	System requirements	Software requirements	Analysis	Program design	Coding/Testing	Operations
Spiral [58]	Requirements plan	Risk analysis	Prototype	Operational prototype	Detailed design (return to first phase)	Testing/Implementation
Watersluice [59]	Proof of principle		Prototype	Alpha and Beta		Product
RUP [60]	Inception/Elaboration			Construction/Construction		
ISO 15288 [63]	Agreement processes	Organisational project-enabling processes	Technical management processes		Technical processes	
V-model [64]	Device user needs		Device design/Process user needs	Process design/production development	System design	Process validation/Device validation
Jensen [65]	Statement of problem	Model physical processes	Characterise the problem	Derive a control algorithm/Select models of computation/Specify hardware/Simulate/Construct/Synthesise		Verify, validate, and test

Chapter 4 Spine Design Process for CPS Design Analysis

4.1 Introduction

In the previous chapter, different design process models in various engineering fields were reviewed. It is found that the domain focused process models, for example Pugh's model [46] and Waterfall model [57], provide an in-depth understanding of certain domains but lack systematic view from system perspective. On the other hand, multidisciplinary and system engineering approaches are too general to provide guidance for a certain design project unless some domain process models are used. Axiomatic design, in contrast to design process models, provide abstract guidelines that can be used to the design of a system. This chapter proposes a process model using axiomatic theory for CPS design analysis that addresses objective five,

- *Develop process model to support early stage analysis and design of CPS. (O5)*

The process model presented in this chapter is named as Spine Design Process (SDP). The SDP incorporates general design process model and axiomatic design theory in order to provide a unified design approach for CPS. The process model adopts phases and stages process from general design process, like Pugh's model [46], and incorporate it with axiomatic design theory to support high level design decision making and early stage design evolution.

This chapter first provides a general discussion about human problem solving process in the design of CPS, and then introduces the overall structure of SDP followed by a detailed discussions about steps in each process stage.

4.2 Design Thinking for CPS

Design thinking of CPS consists of four phases according to the architecture of CPS: cyber, actuating, physical, and sensing. The four phases are linked by information exchange and integrated to a unified system. At the beginning of a design task, a designer should put him or her in a design neutral environment especially at early design stages [28]. Neutral environment is important that designer will not be affected by other unrelated solutions. The system shall be thought holistically as an overall design problem.

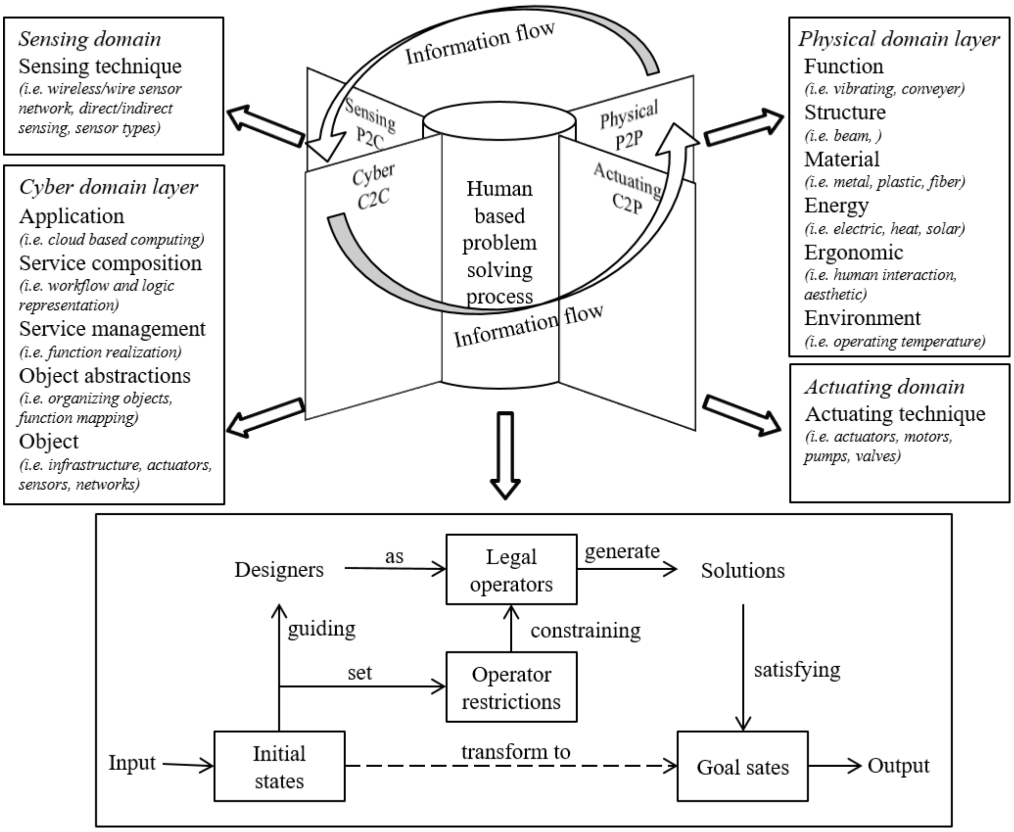


Figure 4.1 Design Thinking for CPS

Design as a human driven creative activity [39] is a systematic process which converts human needs into design solutions. The heterogeneity of CPS requires system practitioners developing more advanced reasoning skills to tackle with design problems and generate innovative ideas. By understanding thinking process, human, as the centre of design as shown in Figure 4.1, becomes the main actor in design task. A human centred view links the natural of design activity with problem solving process forms an integrated view for CPS design. The design processes, viewed from cognitive psychology perspective, are problem solving activities in which possible solutions can be found through the process supported by human reasoning and analytic. Problem solving process [21] mainly has four individual elements: *initial state*, *legal operators*, *operator restrictions*, and *goal state*.

Initial state is the starting point of the process and it should be treated as the first milestone of an activity or a problem. It outlines problem, question, or issue that need to be solved.

Legal operators provides theories, tools, techniques, and other supporting resources for design activity, and it is constrained by legal operators. Designer acts as an actor using operator to satisfy design requirements.

Operator restrictions, acted as constrains for a problem, sets restrictions on initial state. The restrictions outline the boundaries and limitations for a design task.

Goal state is final state of a design activity in which problems are solved by suitable solutions and it is the point of finishing a design activity.

The four elements in problem solving process illustrate a static process for a certain activity in a design process. However, a design is not always static. It is a dynamic activity that requires recursive and iterative analysis among initial state, legal operator, operator, and goal state [21, 70]. For example, design is a transformation process from ill-structured problem to well-define problem, constraints set by operator restrictions may not clear at early design activity. It is important to refine and revisit four elements after each design stage to ensure design activities are fully addressed the design requirements and ensure the design project is on right track. When conducting a design activity, it is useful to think from problem solving perspective in order to cover a comprehensive understanding of the situation and progression of a design activity.

As presented in Table 3.2, in general, a design process consists of different stages, as the design moves through each stage, certain objectives can be achieved. Similar stages in different models can be grouped into phase. A phase is a group of stages shared by different process models. Similarly, SDP adopts a six stages design approach to represent a systematic view to describe the process for analysing and designing CPS. The six stages are named as analysis, definition, interpretation, elaboration, construction, and operation. To avoid ambiguous understanding of each stage, stage names are chosen in a generalised form to represent a united and consolidated approach for CPS.

4.3 The Structure and Composition of SDP

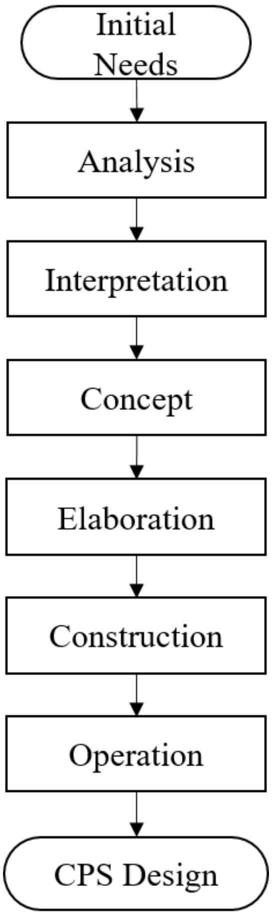


Figure 4.2 SDP Overall Process

SDP proposes a stage based, top down design approach for CPS, as illustrated in Figure 4.2, the overall design process is split into six stages named as analysis, interpretation, concept, elaboration, construction, and operation. The terms of each design stage are generalised from software and mechanical engineering design process.

- *Analysis* is the first stage of a design project converting initial needs into structured customer needs and conducting related market and product search.

- *Interpretation* is the second stage where top level functional requirements need to be defined based upon previously generated customer needs.
- *Concept* is the third stage in a design project where top level design parameters (top level solutions) are generated to fulfil top level functional requirements defined in the second stage. Axioms are used in this stage to evaluate top level solution.
- *Elaboration* is the detailing design stage where lower level functional requirements are defined and associated lower level design parameters are generated. Detailed design solutions are generated in this stage and Axioms are used in this stage to evaluate solutions at each level of system hierarchy.
- *Construction* is the manufacture, coding, and testing stage of each subsystems. Axioms are used in this stage to ensure that design parameters are fully illustrated in the coding and manufacture stage of a design project.
- *Operation* is the final stage in the design process where all subsystems are integrated and necessary testing are performed.

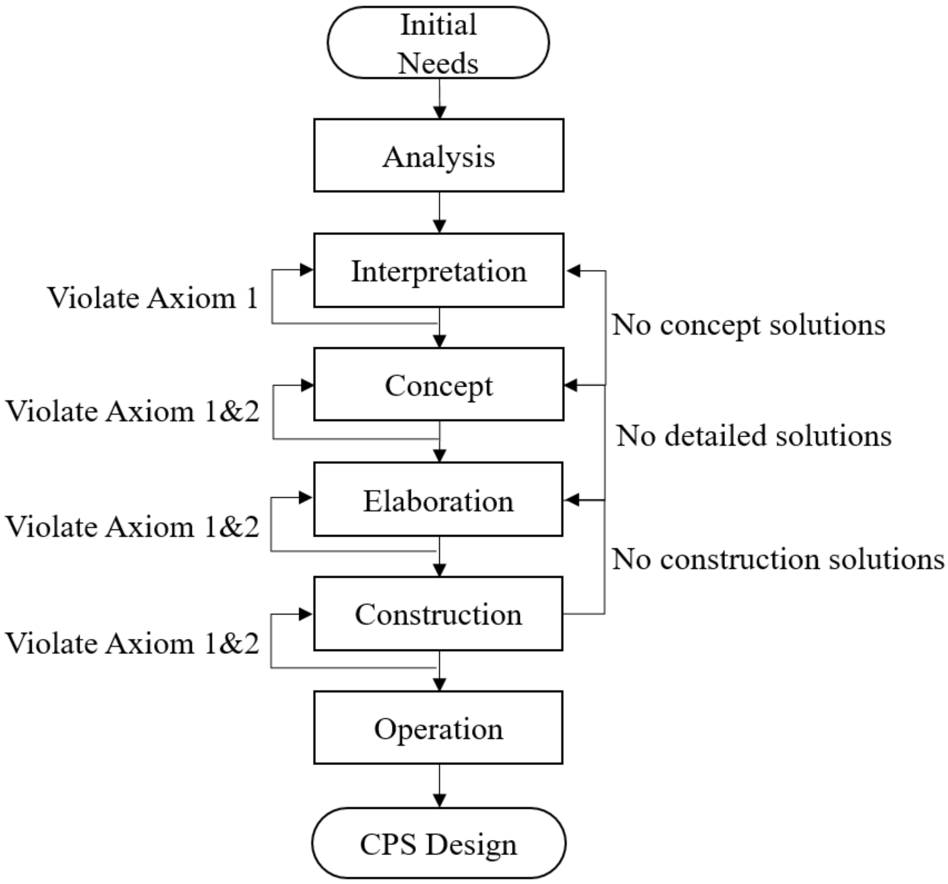


Figure 4.3 SDP Iterative Process

In addition, design is not a single loop process, it is a progressive and iterative process. SDP uses Axioms to evaluate each stage of the process, as shown in Figure 4.3. Axioms are used at each progression level of the design process to assess and select the most suitable solution to support decision making in a design project.

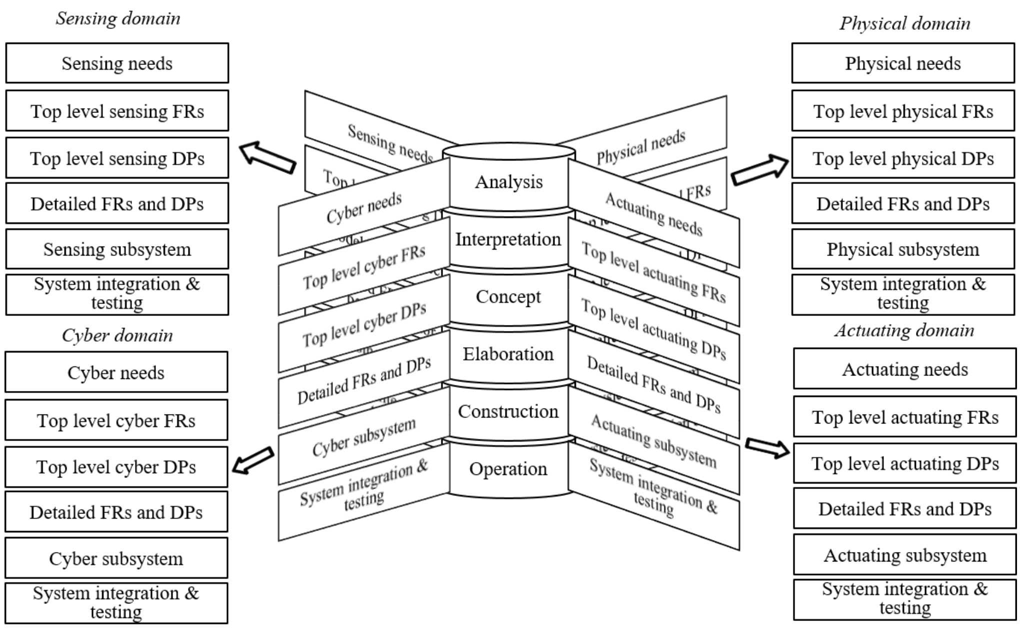


Figure 4.4 Design Stages Decomposition

Each design stage can be further decomposed into four domains, according to functionalities, as shown in Figure 4.4. The four domains are decomposed based upon the architecture of CPS. A typical CPS architecture can be divided into four domains: cyber, physical, actuating, and sensing. Domains are interconnected with each other through communication channels, sensors, and actuators.

- *Cyber domain* considers function, service, and management of computation and control in cyber domain.
- *Physical domain* deals with the interrelationships among functions, structures, materials, and energies in physical domain [47].
- *Actuating domain* seeks solutions that can link cyber and physical domain via actuators. It acts as a connector and an integrator from discrete cyber world to continuous physical world. Actuating domain is

comprised of actuators as a way of making changes to system self-conditions and its surrounding environments.

- *Sensing domain,* in contrast to actuating domain, represents an opposite direction in which cyber domain gathers required information from physical domain through sensing network. The collected data is used for control, computing and decision making in cyber domain.

To clarify the understanding of the terms used in the process and to build a rigour and solid approach, more definitions are given below,

- *Customer Need (CN)* is one or a set of problems, intentions, or expectations that stakeholders want to solve or achieve.
- *Functional Requirement (FR)* is a minimum set of independent requirements that completely address customer need of a product or a system [28].
- *Design Parameter (DP)* is a set of key variables that characterise the design satisfying the specified functional requirements [28].

The functional requirements and design parameters are all in hierarchy structures. The higher the level of functional requirements and design parameters in the hierarchy structures are more abstract and generalised than the lower level ones. The complexity of the hierarchy structure depends on the complexity of the design task. The levels in a hierarchy structure depends on the design solutions generated by the designers.

4.3.1 *Analysis*

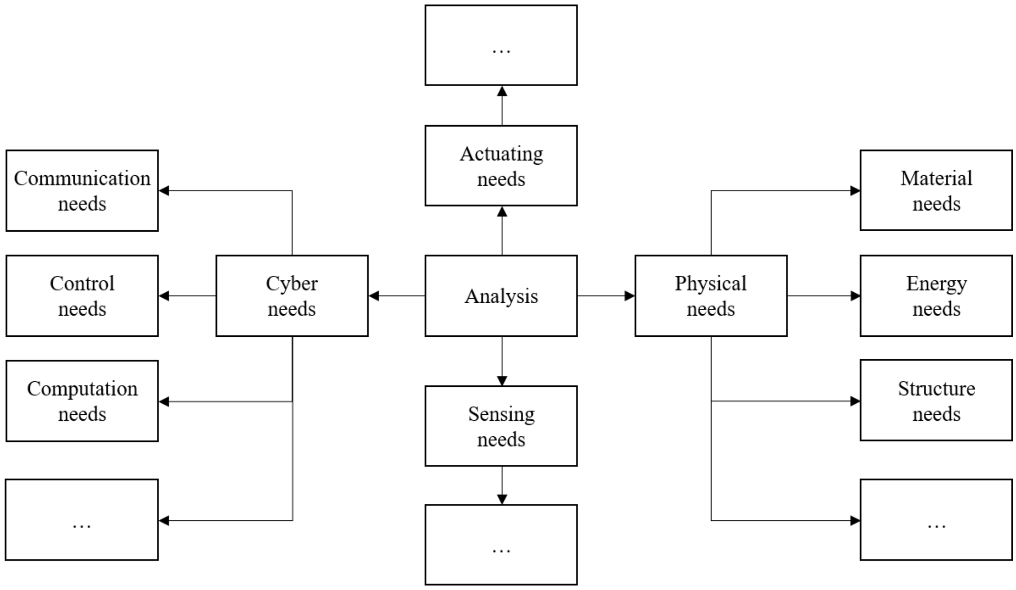


Figure 4.5 Examples of Searching Areas in Analysis Stage

As defined above, *analysis* stage focuses on the analysis of stakeholders' and customers' needs. The information input in this stage can be market needs, client's requests, and system upgrading needs, innovative ideas, or any other forms of requirements or problems that need to be addressed or solved. Various tools are designed to capture customer needs under this context.

In general, analysis stage consists of two main steps,

- Looking for stakeholders' needs
- Generating customer needs vector for customer needs summarization

The aim of these steps is to generate an overview of the current situations to assess the major needs from markets or stakeholders, and to systematically represent needs in a structured way for the use of next stage.

Looking for stakeholders' needs

One of the prevailing tools of identifying customer needs is the house of quality [71] designed by Hauser and Clausing. The house of quality approach is useful to compare different products based upon certain criteria and it can be used to map customer needs to desired product design. Customer needs can also be identified from in-depth market research through interviews, questionnaires, and statistical information, etc. Pugh's matrix method [46] is useful for quantitative analysis of markets and products by adding a set of criteria and using weighted matrix for the assessment of needs.

Identifying expectations and needs of all stakeholders are the beginning of a design project. Research on current market and current systems in the market, as well as a thorough understanding of the needs that might bring benefits to the development of the proposed system are the primary activities need to be done before the actual design activities. Figure 4.5 illustrates some example areas that a need may from and some areas that product or market research can be conducted.

Generating customer needs vector

After identifying customer expectations and stakeholder needs the next step in analysis stage is to generate a customer needs vector for summarizing and grouping of desired needs. A mathematical form can be used to illustrate the customer needs adapted from Axiomatic Design Theory [28].

$$\{\mathbf{CN}s\} = \begin{Bmatrix} CN_1 \\ CN_1 \\ \vdots \\ CN_i \end{Bmatrix} \tag{4.1}$$

where CNs represents all customer needs in the form of a vector, and each element of CN_i represents an identified customer need and the summary of all the needs comprise of the overall customer needs.

For example, customer needs for a consumer refrigerator door can be illustrated as,

- CN_1 = Doors can open and close
- CN_2 = Doors can reduce energy loss and saving running cost
- CN_3 = Items can be accessed easily in and out from the refrigerator

In this case, customer needs are identified into a three elements vector and design should address all the three needs later in the design project.

The macro and micro environmental analysis is better to be conducted

4.3.2 Interpretation

Interpretation is the second stage where top level functional requirements need to be defined based upon previously generated customer needs. The customer needs, as qualitative or quantitative inputs in this stage, are transformed into top level functional requirements. The outputs of interpretation stage are functional requirements vector, constraints, and customer needs to functional requirements matrix. It forms the design requirements and boundaries for a system from top level functional perspective.

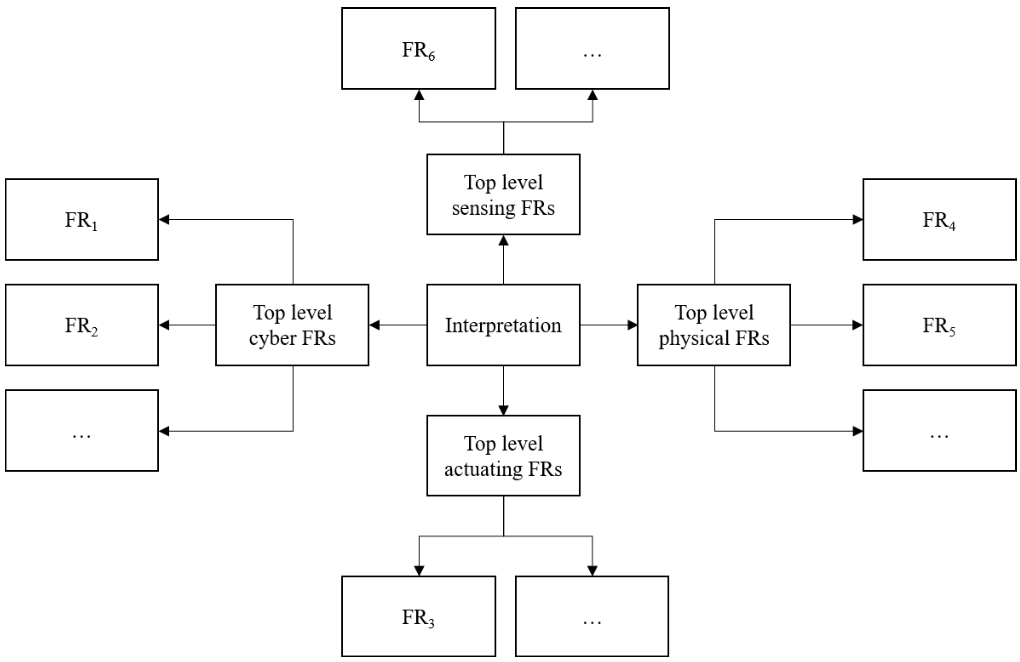


Figure 4.6 Top Level Functional Requirements Classification

Some steps are taken in this stage in order to generate required top level functional requirements,

- Generating top level functional requirements based on previously identified customer needs
- Generating constraints
- Generating customer needs to functional requirements matrix

Generating top level functional requirements

Top level functional requirements are the highest level of functionalities that a system is intended to achieve. A functional requirements vector is used to represent functional requirements that the proposed system need to achieve. Functional requirements intend to address customer needs generated in the analysis stage. Similar to customer needs vector, functional requirements vector is defined as [28],

$$\{\mathbf{FR}s\} = \begin{Bmatrix} FR_1 \\ FR_2 \\ \vdots \\ FR_j \end{Bmatrix} \tag{4.2}$$

where {FRs} represents functional requirements in the form of a vector, each FR_j represents an identified functional requirement, and the summary of all the functional requirements are comprised of the top level functionalities that the system needs to accomplish. The identified FR_j can be potentially grouped into four domains based upon CPS architecture as shown in Figure 4.6. The grouping of FR_j will be used in the concept stage for design parameter generation.

For example, the functional requirements for a consumer refrigerator door may be identified as [72],

- FR_1 = Give access to the items in the refrigerator
- FR_2 = Reduce energy loss

The functional requirements generated above can be grouped into CPS architecture based on Figure 4.6 as,

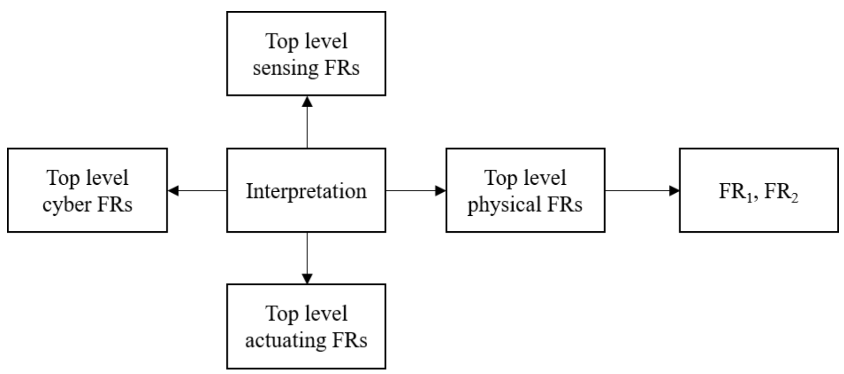


Figure 4.7 Top Level Functional Requirements for Refrigerator Door

Figure 4.7 groups the two functional requirements into physical domain under CPS context. Note that the main function for a refrigerator door is on physical side to provide access to the items in the refrigerator and to isolate heat from outside.

Generating constraints

Constraints in the context of CPS illustrate the boundary requirements for a system. For example, the size, weight, time of delivery, and cost of a system might be limited to some certain values in a design project. It is equally important to identify these constraints along with the identifying of functional requirements. Some customer needs might be considered as constraints not functional requirements, for example, the cost of a system.

For example, the constraints for the design of refrigerator door may be described as,

- C_1 = The outlines of the refrigerator door is equal to the refrigerator body
- C_2 = The manufacture cost of the refrigerator door is no more than £20
- C_3 = The maximum weight for the refrigerator is 4kg

Generate customer needs to functional requirements matrix

Needs to requirements matrix provides a way of showing the relationship between customer needs and functional requirements. It provides a mapping matrix to check if all the required customer needs are captured in functional requirements. A typical form can be described as,

$$\{\mathbf{CN}s\} = [X]\{\mathbf{FR}s\} \tag{4.3}$$

or in another form,

$$\begin{Bmatrix} CN_1 \\ CN_2 \\ \vdots \\ CN_i \end{Bmatrix} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1j} \\ X_{21} & X_{22} & \cdots & X_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ X_{i1} & X_{i2} & \cdots & X_{ij} \end{bmatrix} \begin{Bmatrix} FR_1 \\ FR_2 \\ \vdots \\ FR_j \end{Bmatrix} \tag{4.4}$$

[X] is a transformation matrix that shows the relationship between customer needs and top level functional requirements. X relates the relationship between {CNs} and {FRs}.

For example, customer needs to functional requirements for the refrigerator door design may be described as,

$$\begin{Bmatrix} CN_1 \\ CN_2 \\ CN_3 \end{Bmatrix} = \begin{bmatrix} X & 0 \\ 0 & X \\ X & 0 \end{bmatrix} \begin{Bmatrix} FR_1 \\ FR_2 \end{Bmatrix} \tag{4.5}$$

where X is nonzero element that indicates a relationship between CN_i and FR_j. Opening and closing doors can be illustrated as giving access to the items in the refrigerator, so that FR_1 transforms CN_1 into functional requirement. CN_3 can also be achieved using FR_1, due to that to access the items in the refrigerator doors must be open. By achieving FR_1 both CN_1 and CN_3 can be satisfied. Doors can reduce energy loss and saving cost can be described as reduce energy loss in functional requirements, so that CN_2 can be expressed by FR_2.

4.3.3 Concept

Concept stage generates top level design parameters for the design project. The top level design parameters define the highest level of solutions for a project. Top level design parameters are defined in concept stage, where functional requirements are transformed into design parameters using design matrix to evaluate top level design solutions. The outputs of concept stage are top level design parameter vector, top level design matrix, and top level functional requirements and design parameters hierarchy diagram.

Main tasks conducted in this stage,

- Generating top level design parameters based on top level functional requirements
- Generating top level design matrix

Generating top level design parameters

The top level design parameters describe architecture design solutions for a design project. All low level functional requirements and design parameters are constrained by top level design parameters. Generating suitable top level design parameters are extremely important for a design project, for it provides the directions for detailing design. Design parameters can also be described as a set of elements in a vector. In mathematical form, a design parameters can be written as [72],

$$\{\mathbf{DP}s\} = \begin{Bmatrix} DP_1 \\ DP_2 \\ \vdots \\ DP_k \end{Bmatrix} \tag{4.6}$$

where {DPs} represents design parameters in the form of a vector, and each DP_k represents an identified design parameter that provides solution to functional requirements and the sum of all design parameters are the solution for all functional requirements. Top level design parameters can also be grouped into a four domain CPS architecture model, as shown in Figure 4.8 similar to top level functional requirements classification.

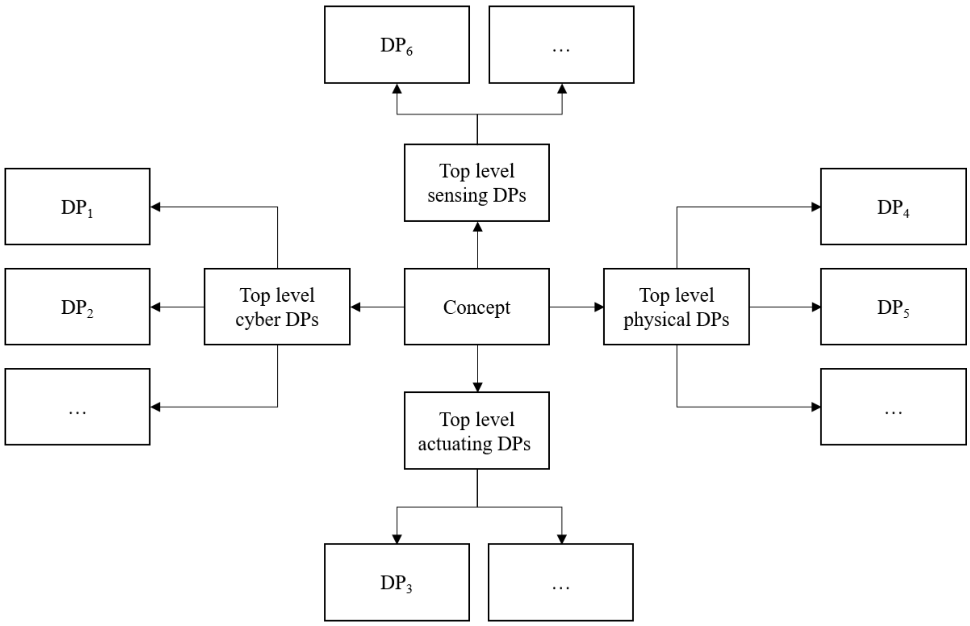


Figure 4.8 Top Level Design Parameters Classification

For example, the top level design parameters for the refrigerator door design might be described as [72],

- DP_1 = Horizontally hung door
- DP_2 = Thermal insulation materials in the door

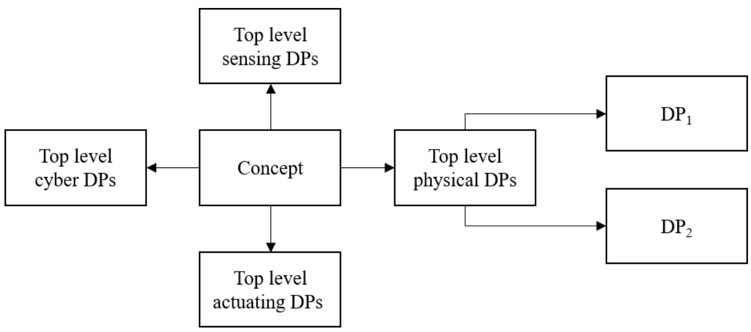


Figure 4.9 Top Level Design Parameters for Refrigerator Door

Generating top level design matrix

Design equation represents the mapping relationships between functional requirements and design parameters. Axiom 1 the independent axiom is used to evaluate the design matrix. Design equation can be defined as follow [28],

$$\{\mathbf{FR}s\} = [\mathrm{A}]\{\mathbf{DP}s\} \tag{4.7}$$

or in another form,

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ \vdots \\ FR_j \end{Bmatrix} = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1k} \\ A_{21} & A_{22} & \cdots & A_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ A_{j1} & A_{j2} & \cdots & A_{jk} \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ \vdots \\ DP_k \end{Bmatrix} \tag{4.8}$$

[A] is the design matrix that used to show the transformation relationships between functional requirements and design parameters. The form is adopted from axiomatic design theory [73] and used under CPS design context.

For example, the design equation for the refrigerator door design can be described as,

$$\begin{Bmatrix} FR_1 \\ FR_2 \end{Bmatrix} = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \end{Bmatrix} \tag{4.9}$$

The design equation above shows that, in this case, the design is an uncoupled design. The horizontally hung door can provide access to the items in the refrigerator which satisfies FR_1 and thermal insulation materials on the door can reduce energy loss. The cold air in the refrigerator is heavier than ambient air, so that horizontally hung door would not let cold air escape from the refrigerator. So that, the design is an uncoupled design that satisfies the Axiom 1 and could proceed to the next level of detailing design.

4.3.4 Elaboration

Continued from the concept stage, *elaboration* stage searches for low level functional requirements and design parameters which go into the details for a design. Decomposing of the overall design is an important task to capture all customer needs and functional requirements, and it is essential to put them into a structured and manageable framework. Elaboration is the detailing stage where detailed level of functional requirements are identified and detailed level of design parameters are generated. Depending on the design project, tools used in the elaboration stage may vary. Based on the hierarchy structure of functional requirements and design parameters, design matrix for each level in the hierarchy are generated and Axioms are used in each level of design hierarchy to support design evaluation and decision making. The outputs of elaboration are detailed functional requirements, detailed design parameters, detailed design matrix, and hierarchy diagram of functional requirements and design parameters.

To conduct this stage, some main tasks are identified as follow,

- Generating hierarchy diagram for functional requirements and design parameters
- Searching for low level functional requirements and design parameters and refine hierarchy diagram
- Generating design matrix for each level of hierarchy using Axioms for decision making and design evaluation

Generating hierarchy diagram

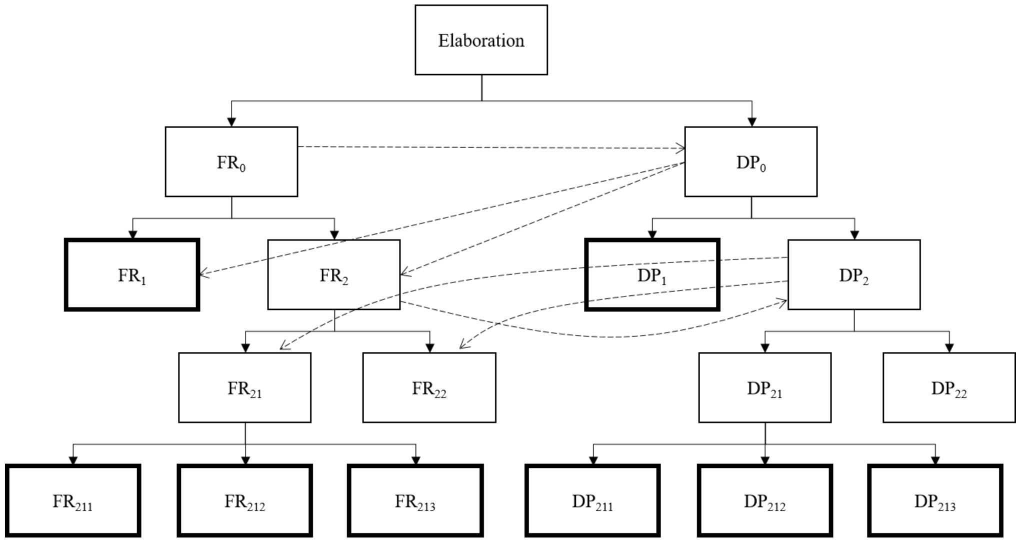


Figure 4.10 Elaboration Stage Hierarchy Diagram [72]

Figure 4.10 shows an example of hierarchy diagram, a hierarchy diagram is used to show the layered structure of a design in order to reduce the complexity of understanding a design project. Boxes with thick lines represent the leaves in a design that do not need further decomposition. The hierarchy diagram is used to support the generation of functional requirements and design parameters. After identifying and completing a certain level of functional requirements and design parameters, the hierarchy structure then moves to the next level of hierarchy. It follows a zigzagging process [72], the dash line shown in Figure 4.10, when generating the hierarchy diagram. Design parameters for a high level of hierarchy add constraints to the next level of functional requirements, and the functional requirements constrains design parameters at the same level of hierarchy. The zigzagging process will stop until all functional requirements and design parameters are decomposed to the lowest design level.

Searching for low level functional requirements and design parameters

As discussed in hierarchy diagram, the low level functional requirements and design parameters are identified through zigzagging process. Similar to the top level functional requirements and design parameters, the low levels share a similar representation. For example, if FR_2 can be further decomposed into two elements, the form of representing it is described as,

$$\{\mathbf{FR}_{2s}\} = \begin{Bmatrix} FR_{21} \\ FR_{22} \end{Bmatrix} \tag{4.10}$$

Similar to low level of functional requirements, low level of design parameters describe design in a more detailed manner. The mathematical form of DP_2 can be described as,

$$\{\mathbf{DP}_{2s}\} = \begin{Bmatrix} DP_{21} \\ DP_{22} \end{Bmatrix} \tag{4.11}$$

For example, the next level of FR_2 and DP_2 in the refrigerator door design may be described as,

- FR_{21} = The weight of the insulation materials is no more than 1 kg
- FR_{22} = Maintain or exceed the thermal insulation property
- DP_{21} = Microvoids in the insulation materials
- DP_{22} = Shape or characteristic of microvoids

The zigzagging process of searching functional requirements and design parameters can be described as in Figure 4.11,

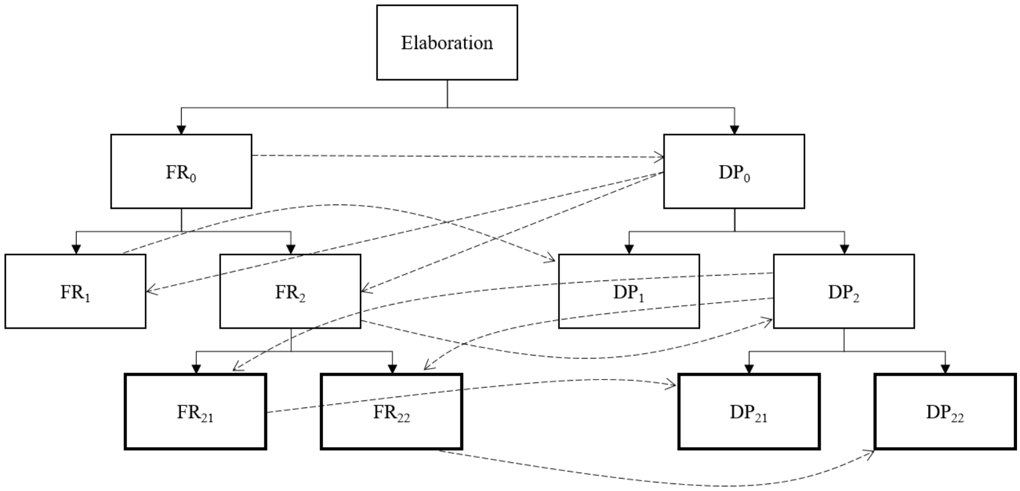


Figure 4.11 Zigzagging Process for Refrigerator Door

Generating detailed design matrix

The detailed design matrix is used to map low level functional requirements and design parameters, similar to top level design matrix but in a specific level of the decomposition structure. Axioms are used at this point to evaluate and select the design solution. The mathematical form for FR_{1m} detailed design matrix can be described as,

$$\begin{Bmatrix} FR_{11} \\ FR_{12} \\ \vdots \\ FR_{1m} \end{Bmatrix} = \begin{bmatrix} A_{111} & A_{112} & \cdots & A_{11n} \\ A_{121} & A_{122} & \cdots & A_{12n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{1m1} & A_{1m2} & \cdots & A_{1mn} \end{bmatrix} \begin{Bmatrix} DP_{11} \\ DP_{12} \\ \vdots \\ DP_{1n} \end{Bmatrix} \quad (4.12)$$

For example, the design matrix for FR_{2s} and DP_{2m} can be described as,

$$\begin{Bmatrix} FR_{21} \\ FR_{22} \end{Bmatrix} = \begin{bmatrix} A_{211} & 0 \\ 0 & A_{222} \end{bmatrix} \begin{Bmatrix} DP_{21} \\ DP_{22} \end{Bmatrix} \quad (4.13)$$

To reduce weight of insulation materials, one of the solutions is to add microviods in the materials to reduce the weight. So that, the weight of the insulation material is affected by the overall weight of the materials and the shape and characteristic of microvoid would not affect FR_{21}. The shape and characteristic of those microviods affect the thermal insulation properties of the structure, by choosing suitable shapes of microviods, FR_{22} can be satisfied. The design solution for FR_{2s} at this level of hierarchy is an uncoupled design. After determining the weight of the materials, suitable microviods and its characteristics can be chosen.

4.3.5 Construction and Operation

The overall system design process is proposed in the abovementioned four stages process to provide a holistic perspective of analysing and viewing a design project or conducting a design work. Due to the resource and time constrains for this MPhil research project, construction and operation stage are beyond the scope of this research project and will not be discussed in further details. Definitions of construction and operation are given in section 4.3.

4.4 Conclusions from Spine Design Process

This chapter presents a design process model for early stage design analysing and management for CPS. By adopting Axiomatic Design Theory, the SDP presents a synthesis process model that links several software and mechanical engineering design approaches together that suitable for an approach for a combined design approach for CPS. In contrast to previously established models, the synthesis model links previous separated models into a unified one by using Axioms that suitable for all types of systems. The model provides a perspective that uses Axioms in analysing CPS.

The SDP is comprised of six stages named as analysis, interpretation, concept, elaboration, construction and operation. In each stage, different tasks are conducted to achieve certain objectives in the stage. The overall objective of the process is to generate a comprehensive understanding of a design and produce solutions for a design problem. The process concerns four domains cyber, physical, actuating, and sensing. The four domains consists of four major areas of a system and should be considered carefully when conducting a design task.

Next chapter is going to implement the process model into case studies in order to demonstrate the usefulness of this process.

Chapter 5 Case Studies for Spine Design Process and Axiomatic Design Theory

5.1 Introduction

Following the previous chapter, this chapter demonstrates the use of SDP and Axioms in analysing and designing of system. Case studies are presented under different using scenarios to give an overall understanding of the implementation of the process model. This chapter addresses objective 6.

- *Evaluate process model in case studies. (O6)*

The evaluation has been done on two different case studies. The first case study demonstrates the use of SDP and Axioms for system architecture design, detailed analysis, and simulation of a cake manufacture system. The second case study uses the process to analysis and investigate the architecture design of virtual assembly coal shearer system.

5.2 Smart Cake Manufacture System

5.2.1 Case Study Briefing

The overall aim of the project is to build an automation system for pasting cake icing. The case study demonstrates the use of SDP and Axioms in analysing and designing of such systems.

5.2.2 Analysis of Smart Cake Manufacture System

The cake manufacture industry highly depends on human labour in producing cakes. The costs of traditional labour intensive cake manufacture approach have gradually increased each year. To improve the profit margins of cake production and maintain the competiveness of the company in the cake industry, a company is seeking for solutions to automate its production process, especially for high frequency complex tasks within the manufacture process.

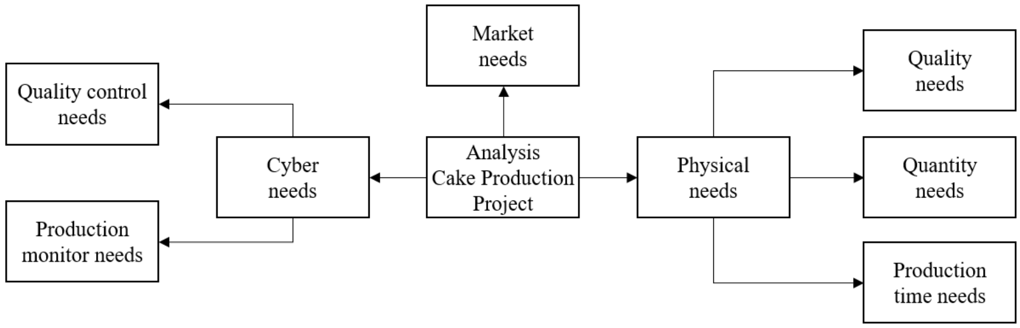


Figure 5.1 Investigation Areas during Analysis

The investigation areas of analysis stage are shown in Figure 5.1. Research shows that the market demands in biscuit and cake industry are constantly growing with an average speed of 4.4% each year estimating its worth at £3.78 billion and moreover, the market has risen by 19.3% in the past five years [74]. The high demands in industry need more high productivity manufacture lines to be developed to fulfil the needs in the market. Despite the increasing of productivity, it is essential to maintain the manufacture quality. To reduce the labour costs and to improve the quality of final products, it is necessary to build an automation system to compensate the industry needs.

Preliminary investigation in the production line shows a three staged manufacture process named as baking process, assembly process, and packaging process, a detailed manufacture process is given in Appendix B. Automations have been made in the process such as conveyor belt system, automate baking system, chocolate forming and cutting machines, and packaging machines. A missing part of automation is in the assembly process.

During the assembly process, cake sponges are placed on the conveyor belt where an icing dropping machine constantly squeeze sugar evenly on the conveyor belt. Cake sponges are passing through the icing dropping machine and covered by a blanket of icing. After that several employees, standing beside the conveyor belt, use hands to shape the icing around the cake. This process is tedious and requires a lot of people involving in the process. When the production volume increasing, more employees are needed in this process. Icing shaping process can be viewed as the bottleneck in the overall production process.

Analysis has been done in this stage to identify the market demands of automation in food manufacture system. The process of cake manufacture has been analysed to search for the critical process during production. To address the demands in the industry, the main customer needs for the project can be described as,

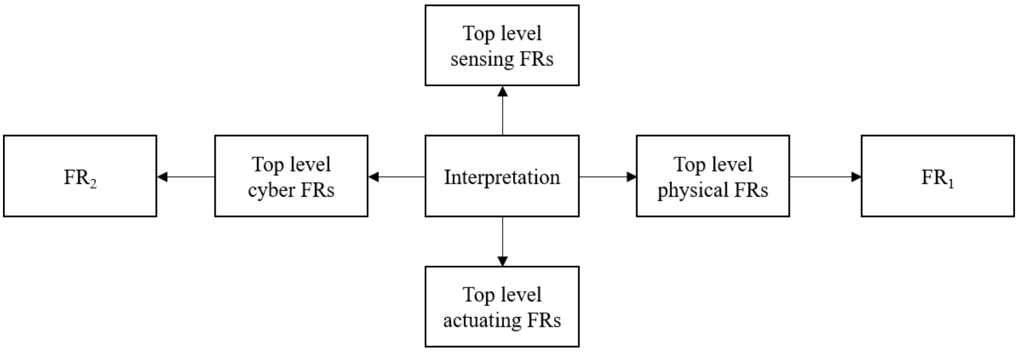
- CN_1 = Automate cake icing shaping process
- CN_2 = Quality check of formed cake icing
- CN_3 = Fit the system to current production line
- CN_4 = Improve production efficiency
- CN_5 = Reduce production costs and labours in the process

5.2.3 Interpretation of Smart Cake Manufacture System

After analysing basic needs in the project, the next stage of the project is to map customer needs to top level functional requirements. As described in Chapter 4, the top level functional requirements can be described as,

- FR_1 = Form cake icing on 9 inch round cake
- FR_2 = Monitor the geometry of formed cake icing

Round cakes are the majority cakes manufactured on the production line, so that FR_1 focuses on the icing shaping on round cakes. Human has the eyes to check the quality of cake icing when pressing the icing, it is equally important to monitor the geometry of cake icing to ensure the production quality, so that FR_2 is identified above.



Obviously, FR_1 requires actions in physical domain where cake icing is formed and shaped, so that FR_1 can be grouped in physical domain. In contrast to FR_1, FR_2 does not require physical actions but requires decision making of whether a cake icing is good or not, so that FR_2 is viewed as cyber domain.

The main functionalities of the system are defined by top level functional requirements. It is equally important to find constraints that govern the design solutions. To reduce the complexity of design project, CN_2, CN_3, and CN_4 can be transformed into constraints of the project. The major constraints of the cake manufacture system can be summarised as,

- C_1 = Install the system on the current production line
- C_2 = Return on investment

- C_3 = High repeatability

Based on the previously identified functional requirements and constraints. A needs to functional requirements matrix is generated below,

$$\begin{Bmatrix} CN_1 \\ CN_2 \end{Bmatrix} = \begin{bmatrix} X & 0 \\ 0 & X \end{bmatrix} \begin{Bmatrix} FR_1 \\ FR_2 \end{Bmatrix} \qquad (5.1)$$

From the matrix analysis, CN_1 and CN_2 have been addressed by FR_1 and FR_2 respectively and other CNs are defined as constraints in the design. The next stage of the design is to generate top level solutions to fulfil the identified top level functional requirements.

5.2.4 Concept of Smart Cake Manufacture System

This stage gives a top level decomposition of the system for design parameters. Continued from interpretation stage, concept stage looks for the solutions for the problem. This stage analyses the top level design parameters that used in the project. It provides a perspective to check that whether the current used design parameters could achieve the goal in functional requirements and gives improvements for the system.

Generating top level design parameters according to the top level functional requirements. A standard approach of pasting icing has two steps: dropping icing over the cake, pressing and shaping icing on the cake. The first step dropping icing over the cake has been achieved by an icing dropping machine, and the second step is where automation needs to be placed. Forming the cake icing requires using employees' hands to press and smooth the sugar surface, it is a simple but tedious process. One of the solutions is to replace human hands with robotic system. The industry robotic system is designed to conduct repeatable tasks with high accuracy and speed. To address FR_1, the DP_1 is chosen as,

- DP_1 = Robotic system for cake icing shaping

Robotic arm system is an industrialised method for manufacturing and process automation. It provides robust, accurate, and efficient automations in industry. As the applications for robotic arm system boost in recent years, the cost of implementing such system reduces quickly. The choice of using robotic system for cake icing is driven by the automation needs in the industry.

The FR_2 requires monitoring the geometry of the cake dimensions. The dimensions can be measured by special gauges. The gauges provide high accuracy of measurement, but may not be an effective method for cake dimension measurement. The contactless vision system is a better choice for measuring objects, as it provides cheap, fast, and reliable measuring of objects. Multiple objects can be measured at same time with enough accuracy for cake products. To fulfil FR_2, DP_2 is proposed as,

- DP_2 = Camera vision system for cake icing geometry detection

The top level design matrix is generated to map functional requirements and design parameters, as presented below.

$$\begin{Bmatrix} FR_1 \\ FR_2 \end{Bmatrix} = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \end{Bmatrix} \tag{5.2}$$

FR_1 and FR_2 are satisfied by DP_1 and DP_2 respectively. The independent axiom is satisfied, since DP_1 and DP_2 can be used individually to fulfil the functional requirements. After determining the top level design decision making, detail level of designing can be conducted.

5.2.5 Elaboration of Smart Cake Manufacture System

This stage provides a deep analysis of functional requirements and design parameters used in achieving identified customer needs. From previous identified design parameters, the FR_1 can be further decomposed as,

- FR_{11} = Locate geometry centre of each cake according to the robotic arm base coordinate
- FR_{12} = Use robotic arm to travel to the geometry centre of each cake respectively
- FR_{13} = Press down the end effector on the robotic arm to form cake icing

Some constraints are generated along with the chosen robotic system,

- C_{11} = Cakes are placed on the belt individually and not overlapping
- C_{12} = After the conveyor belt is fully loaded with cake, the belt will stop moving until all cake icings are formed

In order to address all the identified second level functional requirements for FR_1, the following DPs are proposed,

- DP_{11} = Vision camera for locating and positioning cakes on the conveyor belt within the range of robotic arm working space

- DP_{12} = Calculate joints positions on the robotic arms for each cake identified from DP_{11}
- DP_{13} = Robotic end effector for cake icing shaping and forming

As the choice of robotic arm and vision systems are made in DP_1 and DP_2, the related FRs are identified. Meanwhile the choices of DP_{11}, DP_{12}, and DP_{13} are based on the parameter required from the robotic arm and vision systems. Using vision system to locate the cake is a quickest way under this scenario without any mechanical interference since the cake body is fragile it is better to reduce any physical contact to the cake.

Design matrix is identified as,

$$\begin{Bmatrix} FR_{11} \\ FR_{12} \\ FR_{13} \end{Bmatrix} = \begin{bmatrix} A_{111} & 0 & 0 \\ 0 & A_{122} & 0 \\ 0 & 0 & A_{133} \end{bmatrix} \begin{Bmatrix} DP_{11} \\ DP_{12} \\ DP_{13} \end{Bmatrix} \tag{5.3}$$

A qualitative analysis is conducted here to evaluate whether the proposed second level design parameters fulfil the second level functional requirements. The location of cakes can be measured by a camera vision system, and the accuracy of the position depends on the resolution of the camera and the algorithms used for filtering image data. The geometry information for a cake is simple and the background environment of the conveyor belt is relatively stable, so that DP_1, the vision system, can be used to provide geometry information for the cake. The positions data of cakes are used in DP_2 for the robotic arms. The rotation angles for revolute joints and sliding offsets for the prismatic joints on the robotic arms are calculated for each identified cake. FR_2 can be satisfied by DP_2 if appropriate joints positions can be calculated. FR_3 is satisfied by robot arm end effector installed at the end of robot arm, the function of end effector is to press and push icing on the cake. DP_1, DP_2, and DP_3 are three independent design parameters that satisfy three functional requirement respectively. The tolerances of vision system and robotic arm system are in acceptance range, so that other elements in the design matrix are zero and satisfy Axiom 1.

First, considering FR_2, monitor the geometry of formed cake icing. As DP_2 is chosen to use camera vision system, the functionality of monitoring system is similar to FR_{11}. FR_2 can be decomposed as,

- FR_{21} = Measure the perimeter of formed cake icing (P_i)
- FR_{22} = Measure the surface area of formed cake icing (A_i)
- FR_{23} = Measure the roundness of formed cake icing (R_i)

where P_i, A_i, and R_i represent the perimeter, surface area, and roundness of the number *i* cake in a certain sensing frame.

The functionality of monitoring system can be achieved by using DP_2 the vision camera, to achieve the second level decomposition of FR_2, the following DPs are proposed. These DPs are determined based upon the parameters that required to acquire certain information from the cake.

- DP_{21} = Measure the number of perimeter pixels of objects in a certain frame to determine perimeter of cake icing ($b_i(x, y)$)
- DP_{22} = Measure the number of pixels within the perimeter pixels of objects in a certain frame to determine surface area ($p_i(x, y)$)

- DP_{23} = Calculate surface roundness for the cake icing (r_i)

where (x, y) represents the coordinates of a pixel in an image, $b_i(x, y)$, $p_i(x, y)$, and r_i represent the number *i* object's binary boundary pixels, pixels covered by object *i*, and object roundness.

The second level of design matrix of FR_2 and DP_2 can be represented as,

$$\begin{Bmatrix} FR_{21} \\ FR_{22} \\ FR_{23} \end{Bmatrix} = \begin{bmatrix} A_{211} & 0 & 0 \\ 0 & A_{222} & 0 \\ 0 & 0 & A_{223} \end{bmatrix} \begin{Bmatrix} DP_{21} \\ DP_{22} \\ DP_{23} \end{Bmatrix} \tag{5.4}$$

As the FRs and DPs have been quantified in this level of hierarchy, quantitative analysis will be used to evaluate the design in this stage. An image frame is captured by a camera and comprised of pixels distributed vertically and horizontally in an image. To use the information contains in an image, an image first need to be prepared, background subtraction and noise reduction techniques are used first. In this case, cakes are treated as objects in an image. To get objects binary image, background subtraction technique [75] is first used to extract objects from an image. The background image is subtracted from the foreground image, the differences of grayscale between the background and foreground image is caused by the objects in the image. It can be written as [75],

$$S_m(x,y) = \left| f_m(x,y) - b(x,y) \right| \tag{5.5}$$

where $S_m(x,y)$ is the subtraction image grayscale at pixel (x, y) at frame m, $f_m(x,y)$ is the foreground image grayscale at pixel (x, y) at frame m, $b(x,y)$ is the background image grayscale at pixel (x, y). By subtracting all pixels in frame m, a subtraction image can be obtained. Then, the subtraction image is transformed to a binary image by,

$$B_m(x,y) = \begin{cases} 0 & S_m(x,y) \le T \\ 1 & otherwise \end{cases} \tag{5.6}$$

where T is the grayscale threshold, zero represents the value for background pixel, one represents the value for foreground pixel. By using equation (5.5) and (5.6) a binary image can be obtained. A binary image may contain noises, a morphology operation [76] might be used to reduce the noise in the image. Figure 5.2 shows an example of a processed image after noise reduction.

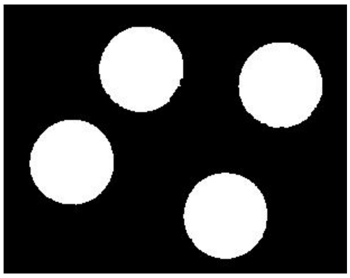


Figure 5.2 Example of Processed Binary Image

The geometry data can then be extracted from the processed binary image. Perimeter information can be obtained by adding the boundary pixels,

$$P_i = scale \times \sum_{BP_i} b_i(x,y) \tag{5.7}$$

BP_i represents the number of all the boundary pixels for object *i*. The *scale* is the unit transformation from pixels to millimetres. The summary of all the boundary pixels is the perimeter of the object, the unit is in pixel. The area covered by object *i* can be calculated by adding all the pixels within object *i*,

$$A_i = scale \times \sum_{W_i} p_i(x, y) \tag{5.8}$$

where W_i is the number of all the pixels for object *i*. The summary of all the pixels is the surface area for object *i*.

The roundness of object *i* can be calculated by,

$$R_i = r_i = \frac{4 \times \pi \times A_i}{P_i^2} \tag{5.9}$$

where R_i is equal to one only if the area is a perfect circle, it is less than one for any other shape. An appropriate threshold can be set to *r* to determine whether the finalised shape is acceptable.

Reviewing equation (5.4), using equation (3.4) the elements in design matrix can then be determined as,

$$A_{211} = \frac{\partial FR_{21}}{\partial DP_{21}} = scale \times BP_i \tag{5.10}$$

$$A_{222} = \frac{\partial FR_{22}}{\partial DP_{22}} = scale \times W_i \tag{5.11}$$

$$A_{233} = \frac{\partial FR_{23}}{\partial DP_{23}} = 1 \tag{5.12}$$

where for a certain object *i* in a frame *m*, BP_i and W_i are constant. The design matrix has been evaluated here, so that, the second level of design for FR_2 is an uncoupled design which satisfies Axiom 1. The design matrix for FR_2 can be rewritten as,

$$\begin{Bmatrix} P_i \\ A_i \\ R_i \end{Bmatrix} = \begin{bmatrix} scale \times BP_i & 0 & 0 \\ 0 & scale \times W_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} bp_i \\ p_i \\ r_i \end{Bmatrix} \tag{5.13}$$

Furthermore, the leaf elements for FR_2 have been found for detailing and elaboration. The hierarchy diagram and the zigzagging process can be illustrated in Figure 5.3 the hierarchy diagram.

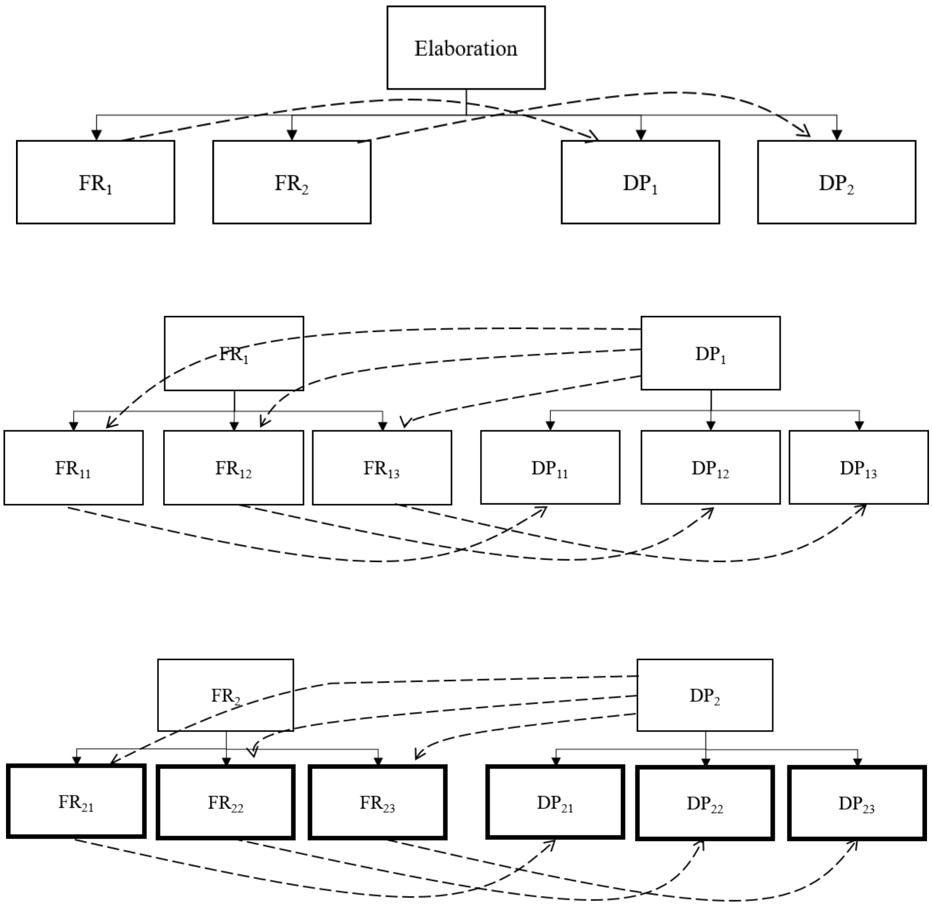


Figure 5.3 Hierarchy Diagram for Cake Icing System

Continuing from FR_{11}, FR_{12} and FR_{13}, further decomposition is needed to search for the leaf elements in each functional requirements and design parameters. The third level of hierarchy is generated for detailing the elements in second level of hierarchy. Considering FR_{11}, FR_{11} is to locate the positons of each cake on the conveyor belt and DP_{11} is to use vision camera for locating and positing.

Comparing FR_{11} with FR_{21}, FR_{22}, and FR_{23}, similar system can be deployed to achieve FR_{11}. The same vision system can be used for different purposes to determine coordinates for each cake.

Thus, FR_{11} can be further decomposed as,

- FR_{111} = The x_i coordinates for cake *i* relative to robotic arm (geometric centre of cake *i*)
- FR_{112} = The y_i coordinates for cake *i* relative to robotic arm (geometric centre of cake *i*)

The height of each cake is constant in this design, so that z_i is constant in this case. The corresponding DPs are identified as,

- DP_{111} = $p_i(x)$ coordinates set for pixels in object *i* (*x* is the coordinate for one pixel in object *i* relative to robotic arm base coordinate)
- DP_{112} = $p_i(y)$ coordinates set for pixels in object *i* (*y* is the coordinate for one pixel in object *i* relative to robotic arm base coordinate)

The design matrix for FR_{11} and DP_{11} can be illustrated as,

$$\begin{Bmatrix} FR_{111} \\ FR_{112} \end{Bmatrix} = \begin{bmatrix} A_{1111} & 0 \\ 0 & A_{1122} \end{bmatrix} \begin{Bmatrix} DP_{111} \\ DP_{112} \end{Bmatrix} \tag{5.14}$$

The x_i coordinate for object *i* can be calculated by adding all the *x* coordinates for the pixels in object *i* and divided by the total number of pixels. Similarly, the y_i coordinate for object *i* can be calculated by adding all the *y* coordinates for the pixels in object *i* and divided by the total number of pixels. The mathematical form can be described as,

$$x_i = \frac{\sum_{W_i} p_i(x)}{W_i} \tag{5.15}$$

$$y_i = \frac{\sum_{W_i} p_i(y)}{W_i} \tag{5.16}$$

The unit of x_i and y_i is in pixel. So that, the geometry centre for object *i* can be calculated by equation (5.15) and (5.16). A_{1111} and A_{1122} can then be determined by differentiating $p_i(x)$ and $p_i(y)$ from (5.15) and (5.16).

$$A_{1111} = \frac{\partial FR_{111}}{\partial DP_{111}} = 1 \tag{5.17}$$

$$A_{1122} = \frac{\partial FR_{112}}{\partial DP_{112}} = 1 \tag{5.18}$$

A_{1111} and A_{1122} are evaluated through (5.17) and (5.18). The design matrix does not violate Axiom 1.

The FR_{12} can be further decomposed as,

- $FR_{121} = x$, required position on *x* axis for robotic arm end effector
- $FR_{122} = y$, required position on *y* axis for robotic arm end effector
- $FR_{123} = z$, required position on *z* axis for robotic arm end effector
- $FR_{124} = \psi$, the angle between robotic arm end effector and *x* axis

As the position requirements for the robotic end effector have four elements, to form an ideal design, the corresponding design parameters should have four elements either. So that, a four degrees of freedom robotic arm is an ideal solution in this case. The design chooses a Staubli TS80L robotic arm with four degrees of freedom. The related design parameters for FR_{12} can be described as,

- DP_{121} = θ_1, the revolution degree of the first rotational axis on the robotic arm
- DP_{122} = θ_2, the revolution degree of the second rotational axis on the robotic arm
- DP_{123} = d_3, the sliding length of the third prismatic axis on the robotic arm
- DP_{124} = θ_4, the revolution degree of the fourth rotational axis on the robotic arm

Figure 5.4 shows the positions of robotic arm and conveyor belt system. In the figure below, the relationship between robotic arm and conveyor belt has been illustrated. As the example shown in the figure, the required positon and orientation of the end effector is illustrated as (x, y, z, ψ) and the positions of each robotic arm joints have been illustrated as (θ_1, θ_2, d_3, θ_4). The base coordinate of robotic arm has been illustrated as x, y, z axis respectively.

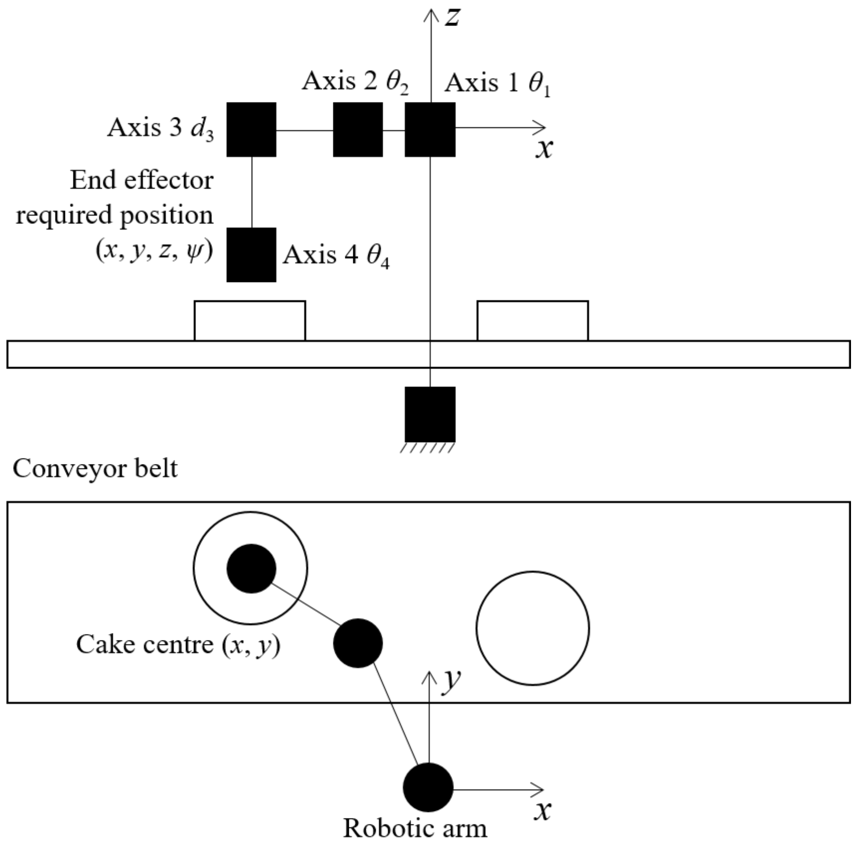


Figure 5.4 Relationship of cake position and robotic arm

In accordance with the defined relationship in Figure 5.4. A design matrix for FR_{12} and DP_{12} can be formed as,

$$\begin{Bmatrix} FR_{121} \\ FR_{122} \\ FR_{123} \\ FR_{124} \end{Bmatrix} = \begin{bmatrix} A_{1211} & A_{1212} & 0 & 0 \\ A_{1221} & A_{1222} & 0 & 0 \\ 0 & 0 & A_{1233} & 0 \\ A_{1241} & A_{1242} & 0 & A_{1244} \end{bmatrix} \begin{Bmatrix} DP_{121} \\ DP_{122} \\ DP_{123} \\ DP_{124} \end{Bmatrix} \tag{5.19}$$

In order to identify and evaluate each element in the design matrix, robotic arm kinematic analysis needs to be conducted. The kinematic analysis of manipulator is the fundamental step towards the analysis of motions of robotic manipulator. There are two different paths for solving a kinematic problem: direct kinematic analysis and inverse kinematic analysis [77]. Figure 5.5 shows the analysis process for robotic manipulators. In order to evaluate the elements in (5.19), direct kinematic will be conducted first followed by inverse kinematic analysis.

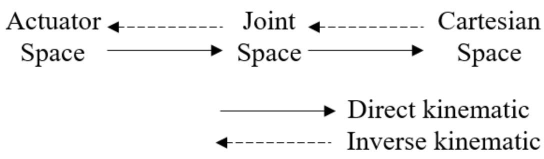


Figure 5.5 Robotic Manipulator Kinematic Analysis Process [77]

Direct kinematic analysis of four degrees of freedom robotic arm

To obtain the motion relationship between robotic end effector and the base coordinate of robotic arm, Denavit-Hartenberg parameters [77] (DH parameters) need to be defined first for a robotic system. DH parameters use four parameters associated with a particular convention to represent references frames to the base coordinates. However, to obtain DH parameters, the first thing is to define reference frames on the robotic arm.

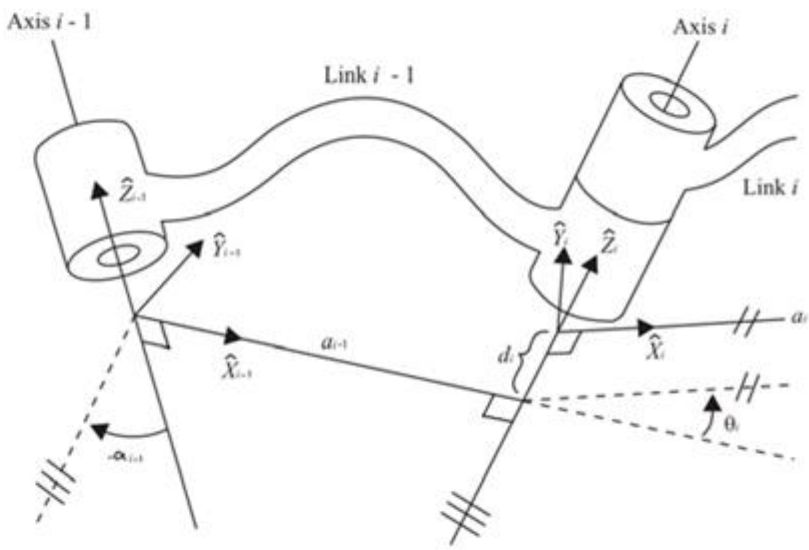


Figure 5.6 Frame Relationships [77]

A convention used to define frames are [77],

- a_i = the distance from $\hat{Z}_i$ to $\hat{Z}_{i+1}$ measured along $\hat{X}_i$;
- α_i = the angle from $\hat{Z}_i$ to $\hat{Z}_{i+1}$ measured about $\hat{X}_i$;
- d_i = the distance from $\hat{X}_{i-1}$ to $\hat{X}_i$ measured along $\hat{Z}_i$; and
- θ_i = the angle from $\hat{X}_{i-1}$ to $\hat{X}_i$ measured about $\hat{Z}_i$.

The detailed process of defining frames have been presented in [77]. By using the conventions above, a robotic DH parameters could be defined. There exists multiple ways of defining frames in a same robotic arm system. For the convenience of the robotic system used in this case study the frames are defined in Figure 5.7. Where $(\hat{X}_0, \hat{Y}_0, \hat{Z}_0)$ is the base coordinate (frame) for the robotic arm which is the same as the (*x*, *y*, *z*) coordinate in Figure 5.4. $(\hat{X}_1, \hat{Y}_1, \hat{Z}_1)$, $(\hat{X}_2, \hat{Y}_2, \hat{Z}_2)$, $(\hat{X}_3, \hat{Y}_3, \hat{Z}_3)$, and $(\hat{X}_4, \hat{Y}_4, \hat{Z}_4)$ are the frames for the four joints on the robotic arm respectively. L_1 and L_2 are the linkage length of robotic arm, and θ_1, θ_2, and θ_4 are the rotation angles for the three rotational joints. d_3 is the offset for the prismatic joint.

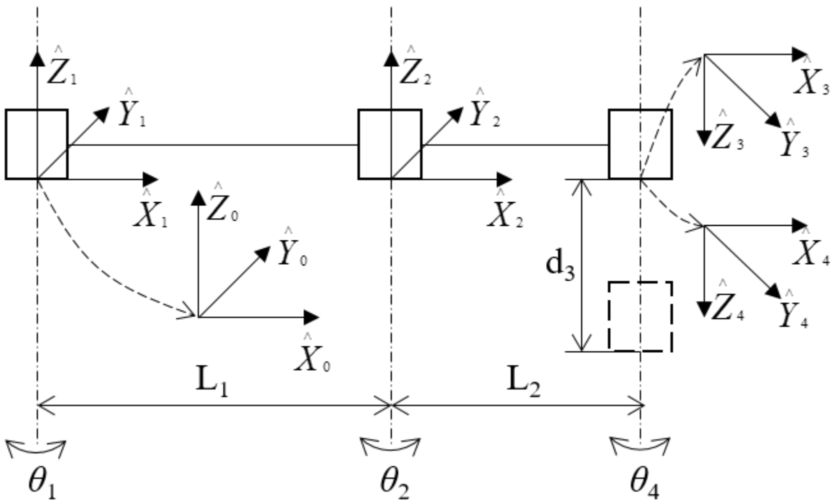


Figure 5.7 Frame Relationships for 4 Axis Robotic Arm

Based upon the frame defined in Figure 5.7, a DH parameter table is generated below, the table shows the key parameters and variables for the robotic arm.

Table 5.1 Link Parameter for 4 Axis Robotic Arm

i	α_{i-1}	a_{i-1}	d_i	θ_i	Variables	Range
1	0	0	0	θ_1	θ_1	-140°-140°
2	0	L_1 (430mm)	0	θ_2	θ_2	-155°-155°
3	π	L_2 (370mm)	d_3	0	d_3	0-400mm
4	0	0	0	θ_4	θ_4	-500°-500°

To calculate the kinematic of the end effector relative to the base frame, a homogeneous transformation matrix is used to define the relationship of frame *i* relative to frame *i*-1, a transformation equation is given as [77],

$$ {}^{i-1}_{i}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i \cos\alpha_{i-1} & \cos\theta_i \cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1} d_i \\ \sin\theta_i \sin\alpha_{i-1} & \cos\theta_i \sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.20) $$

Using (5.20), the frames for each neighbour frame defined in Figure 5.7 can be calculated as,

$$
{}_{1}^{0}T = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.21}
$$

$$
{}_{2}^{1}T = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & L_1 \\ \sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.22}
$$

$$
{}_{3}^{2}T = \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.23}
$$

$$
{}_{4}^{3}T = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ \sin\theta_4 & \cos\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.24}
$$

Thus, ${}_{0}^{4}T$ can be calculated by,

$$
{}_{0}^{4}T = {}_{1}^{0}T\,{}_{2}^{1}T\,{}_{3}^{2}T\,{}_{4}^{3}T \tag{5.25}
$$

$$
{}_{4}^{0}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.26}
$$

where the rotational transformations are illustrated by,

$r_{11} = \cos(\theta_1 + \theta_2 - \theta_4)$, $r_{12} = \sin(\theta_1 + \theta_2 - \theta_4)$, $r_{21} = \sin(\theta_1 + \theta_2 - \theta_4)$,

$r_{22} = -\cos(\theta_1 + \theta_2 - \theta_4)$, $r_{13} = r_{23} = r_{31} = r_{32} = 0$, $r_{33} = -1$

The position of end effector relative to robotic arm base frame can be described as,

$$p_x = L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) \tag{5.27}$$

$$p_y = L_1 \sin\theta_1 + L_2 \sin(\theta_1 + \theta_2) \tag{5.28}$$

$$p_z = -d_3 \tag{5.29}$$

The end effector's position (p_x, p_y, p_z, ψ) can be calculated through equation (5.26). When certain θ_i and d_i are given, the position can be calculated. As the position requirements in this project, (p_x, p_y, p_z, ψ) are functional requirements and related positions of each joint need to be calculated. So that, an inverse kinematic analysis need to be conducted here.

Inverse kinematic analysis of four degrees of freedom robotic arm

Inverse kinematic can be obtained by solving the transformation matrix in terms of θ_i and d_i. By square both (5.27) and (5.28) and add them, get,

$$p_x^2 + p_y^2 = L_1^2 + L_2^2 + 2L_1L_2 \cos\theta_2 \tag{5.30}$$

Solving (5.30), $\cos\theta_2$ can be obtained as,

$$\cos\theta_2 = \frac{p_x^2 + p_y^2 - L_1^2 - L_2^2}{2L_1L_2} \tag{5.31}$$

The right hand side of (5.31) must have a value between -1 and 1, and this equation can be used to check if there exist solutions or in a range that the robotic arm can reach. To ensure all solutions can be found, both sine and cosine of the desired joint angle are calculated. Assuming there exist solutions in the workspace, $\sin\theta_2$ can be written as,

$$\sin\theta_2 = \pm\sqrt{1-\cos^2\theta_2} \tag{5.32}$$

So that two solutions may exist for θ_2, using two-argument arctangent, obtain,

$$\theta_2 = A\tan 2(\sin\theta_2, \cos\theta_2) \tag{5.33}$$

Rewritten (5.27) and (5.28) in the form,

$$p_x = k_1\cos\theta_1 - k_2\sin\theta_1 \tag{5.34}$$

$$p_y = k_1\sin\theta_1 + k_2\cos\theta_1 \tag{5.35}$$

where,

$$\begin{aligned} k_1 &= L_1 + L_2\cos\theta_2 \\ k_2 &= L_2\sin\theta_2 \end{aligned} \tag{5.36}$$

if,

$$\begin{aligned} r &= \sqrt{k_1^2 + k_2^2} \\ \gamma &= A\tan 2(k_2, k_1) \end{aligned} \tag{5.37}$$

then,

$$\begin{aligned} k_1 &= r\cos\gamma \\ k_2 &= r\sin\gamma \end{aligned} \tag{5.38}$$

Equations (5.34) and (5.35) can then be written as,

$$\frac{p_x}{r} = \cos\gamma\cos\theta_1 - \sin\gamma\sin\theta_1 \tag{5.39}$$

$$\frac{p_y}{r} = \cos\gamma\sin\theta_1 + \sin\gamma\cos\theta_1 \tag{5.40}$$

so,

$$\cos(\gamma + \theta_1) = \frac{p_x}{r} \tag{5.41}$$

$$\sin(\gamma + \theta_1) = \frac{p_y}{r} \tag{5.42}$$

So that θ_1 can be calculated using two-argument arctangent as,

$$\theta_1 = A\tan 2(p_y, p_x) - A\tan 2(k_2, k_1) \tag{5.43}$$

Using equation (5.25) to solve θ_4 by multiplying inverse matrix of ${}^0_1T$, ${}^1_2T$, and ${}^2_3T$ respectively. Then obtain,

$${}^3_4T(\theta_4) = [{}^2_3T(d_3)]^{-1}[{}^1_2T(\theta_2)]^{-1}[{}^0_1T(\theta_1)]^{-1}{}^0_4T \tag{5.44}$$

Based on the property of transformation matrix, the inverse matrix can be calculated by,

$$\begin{aligned} {}^B_AT &= {}^A_BT^{-1} \\ {}^B_AT &= {}^A_BT^{T} \end{aligned} \tag{5.45}$$

So that equation (5.44) can be written as,

$$\begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ \sin\theta_4 & \cos\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta_1+\theta_2) & \sin(\theta_1+\theta_2) & 0 & 0 \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) & 0 & 0 \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.46}$$

By using equation (5.46) $\cos\theta_4$ and $\sin\theta_4$ can be calculated as,

$$\cos\theta_4 = r_{11}\cos(\theta_1+\theta_2) + r_{21}\sin(\theta_1+\theta_2) \tag{5.47}$$

$$\sin\theta_4 = -r_{12}\cos(\theta_1+\theta_2) - r_{22}\sin(\theta_1+\theta_2) \tag{5.48}$$

So that θ_4 can be calculated using two-argument arctangent as,

$$\theta_4 = A\tan 2(\sin\theta_4, \cos\theta_4) \tag{5.49}$$

When a desired position is given, a target position in a frame can be written as,

$${}^0_4T = \begin{bmatrix} \cos\psi & -\sin\psi & 0 & p_x \\ \sin\psi & \cos\psi & 0 & p_y \\ 0 & 0 & -1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & 0 & p_x \\ r_{21} & r_{22} & 0 & p_y \\ 0 & 0 & -1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.50}$$

So that, based on the previous calculation, θ_1, θ_2, d_3, and θ_4 can be calculated by (5.43), (5.33), (5.29), and (5.49) respectively. Thus, the inverse kinematic analysis is finished. The elements in the detailed design matrix can then be evaluated using the above calculation.

A_{1211} and A_{1212} can be evaluated by differentiating θ_1 and θ_2 from (5.27) respectively,

$$A_{1211} = \frac{\partial FR_{121}}{\partial DP_{121}} = \frac{\partial p_x}{\partial \theta_1} = -L_1 \sin\theta_1 - L_2 \sin(\theta_1 + \theta_2) \tag{5.51}$$

$$A_{1212} = \frac{\partial FR_{121}}{\partial DP_{122}} = \frac{\partial p_x}{\partial \theta_2} = -L_2 \sin(\theta_1 + \theta_2) \tag{5.52}$$

A_{1211} and A_{1212} are nonlinear design elements affected by the value of θ_1 and θ_2, and do not affected by other design parameters. Similarly, A_{1221} and A_{1222} can be calculated by (5.28) as,

$$A_{1221} = \frac{\partial FR_{122}}{\partial DP_{121}} = \frac{\partial p_y}{\partial \theta_1} = L_1 \cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) \tag{5.53}$$

$$A_{1222} = \frac{\partial FR_{122}}{\partial DP_{122}} = \frac{\partial p_y}{\partial \theta_2} = L_2 \cos(\theta_1 + \theta_2) \tag{5.54}$$

A_{1233} can be calculated by differentiating d_3 from (5.29),

$$A_{1233} = \frac{\partial FR_{123}}{\partial DP_{123}} = \frac{\partial p_z}{\partial d_3} = -1 \tag{5.55}$$

From (5.50), r_{11}, r_{12}, r_{21}, and r_{22} can be replaced by,

$$r_{11} = \cos\psi, r_{12} = -\sin\psi, r_{21} = \sin\psi, r_{22} = \cos\psi \tag{5.56}$$

Replacing r_{11} and r_{21} in (5.47), get,

$$\cos\theta_4 = \cos\psi \cos(\theta_1 + \theta_2) + \sin\psi \sin(\theta_1 + \theta_2) \tag{5.57}$$

So that, ψ can be calculated as,

$$\psi = \theta_1 + \theta_2 + \theta_4 \tag{5.58}$$

Then A_{1241}, A_{1242}, and A_{1244} can be calculated by differentiating θ_1, θ_2, and θ_3 from (5.58) respectively,

$$A_{1241} = A_{1242} = A_{1244} = 1 \tag{5.59}$$

Up to this point, all the design elements in FR_{12} are evaluated. All the four functional requirements can be achieved by the four proposed design parameters. FR_{121} and FR_{122} are coupled, since the rotation of θ in x-y plane affects both x and y coordinates and these cannot be decoupled. FR_{123} can be achieved by adjusting d_3 the height of the prismatic joint parallel to z axis. The rotation requirement ψ is a decoupled design, since the rotation of θ_1 and θ_2 affect the orientation of ψ. θ_4 as the final rotational joint can compensate the changes made by θ_1 and θ_2.

Continuing to FR_{13}, FR_{13} requires an end effector mounted at the end of robotic arm to complete the shaping process for the icing. Based on the size of the cake, FR_{13} can be further decomposed as,

- FR_{131} = End effector for a cake with 9 inches in diameter
- FR_{132} = End effector for a cake with 5 inches in height

A simplest method of forming the icing is to use a housing that is slightly larger than the cake in order to get the final result. The complemented design parameters may be described as,

- DP_{131} = Pressing die with 10 inches inner diameter
- DP_{132} = Pressing die with 7 inches inner height

The design equation for the end effector obviously can be written as,

$$\begin{Bmatrix} FR_{131} \\ FR_{132} \end{Bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} DP_{131} \\ DP_{132} \end{Bmatrix} \tag{5.60}$$

Where the diameter of the cake only affect the diameter of the end effector, and the height of the cake only affect the height of the end effector. The relationship between the height and diameter are linear, so that the design matrix is written as (5.60). So that FR_{13} is an uncoupled design. Figure 5.8 shows the pressing process for the end effector. The pressing process is achieved by d_3 defined DP_{123} after the robotic end effector moves to the geometry centre of the cake.

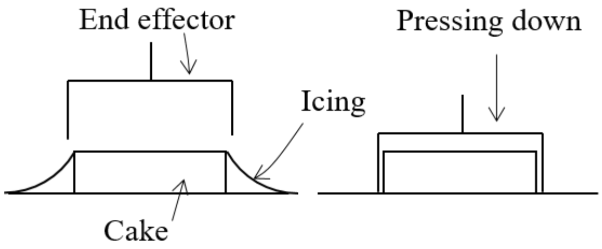
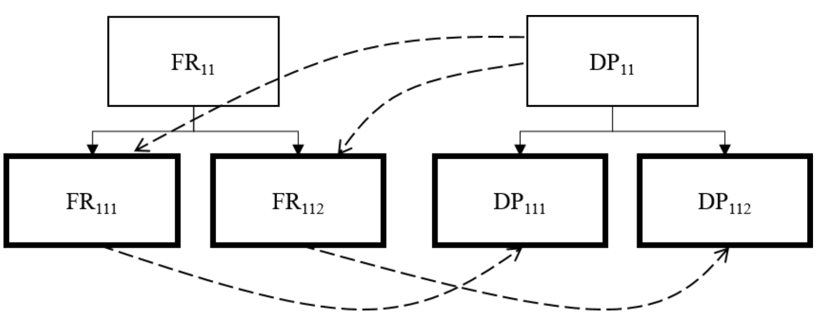


Figure 5.8 Robotic Arm End Effector

After the completing of FR_{13} decomposition, all the leaf elements in functional requirements and design parameters are found. Hierarchy tree for FR_1 is depicted in Figure 5.9.



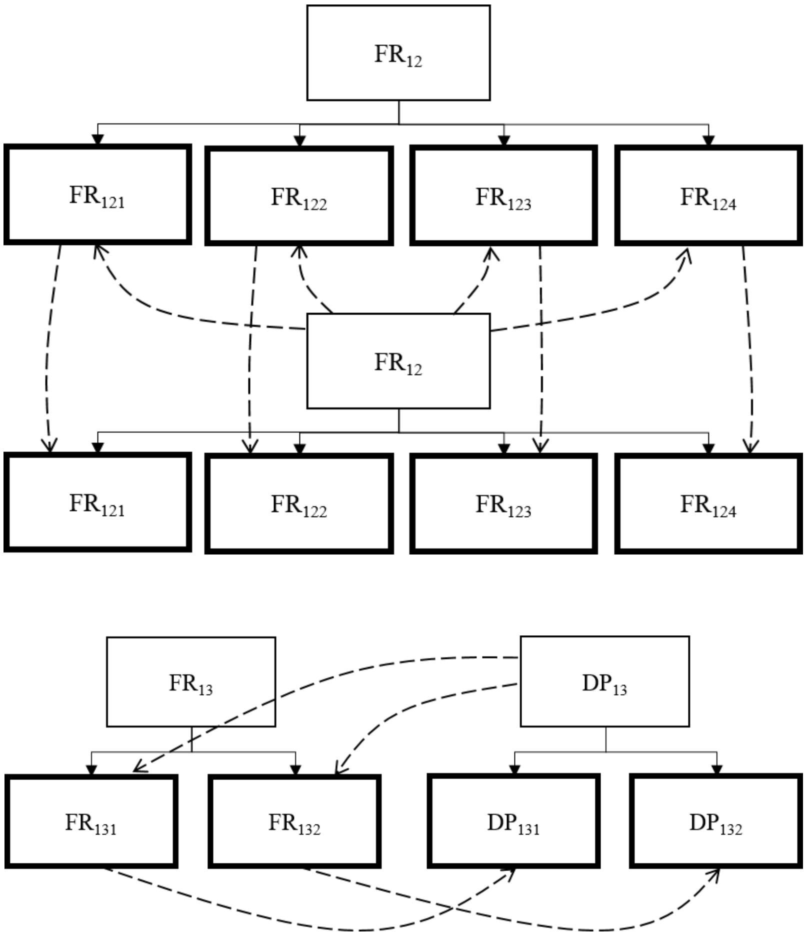


Figure 5.9 Hierarchy Diagram for Cake Icing System 2

To validate the calculations presented above, Matlab simulation has been made to calculate the workspace for the robotic arm and model kinematic relationships of each joints. Workspace for a robotic arm can be calculated using an angle iteration method. Using the parameters identified in Table 5.1. The workspace of this four degrees of freedom robot arm is presented in Figure 5.10. All the Matlab codes can be found in Appendix C.

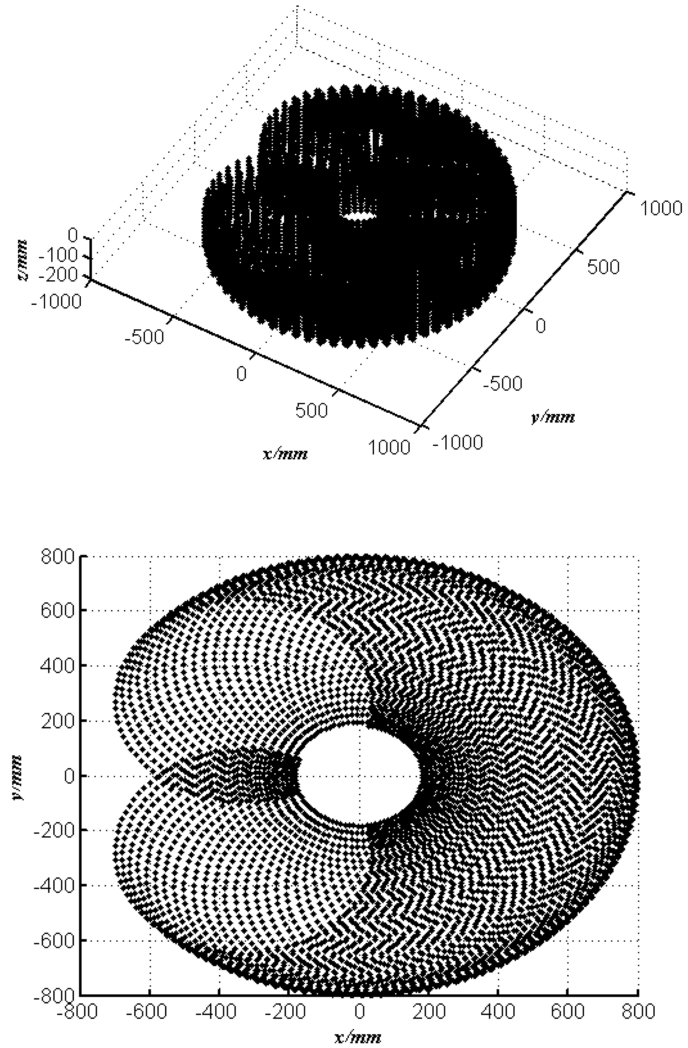


Figure 5.10 Workspace for Four Degrees of Freedom Robotic Arm

Using Matlab Robotic Tool box, the four degrees of freedom robot can also be modelled, as shown in Figure 5.11. The position of joints can be changed based upon the desired positions for the robotic arm. By using (5.29), (5.33), (5.43) and (5.49), the position of the end effector can be calculated. The actual workspace for the robotic arm is the common spaces covered by the conveyor belt and robotic arm.

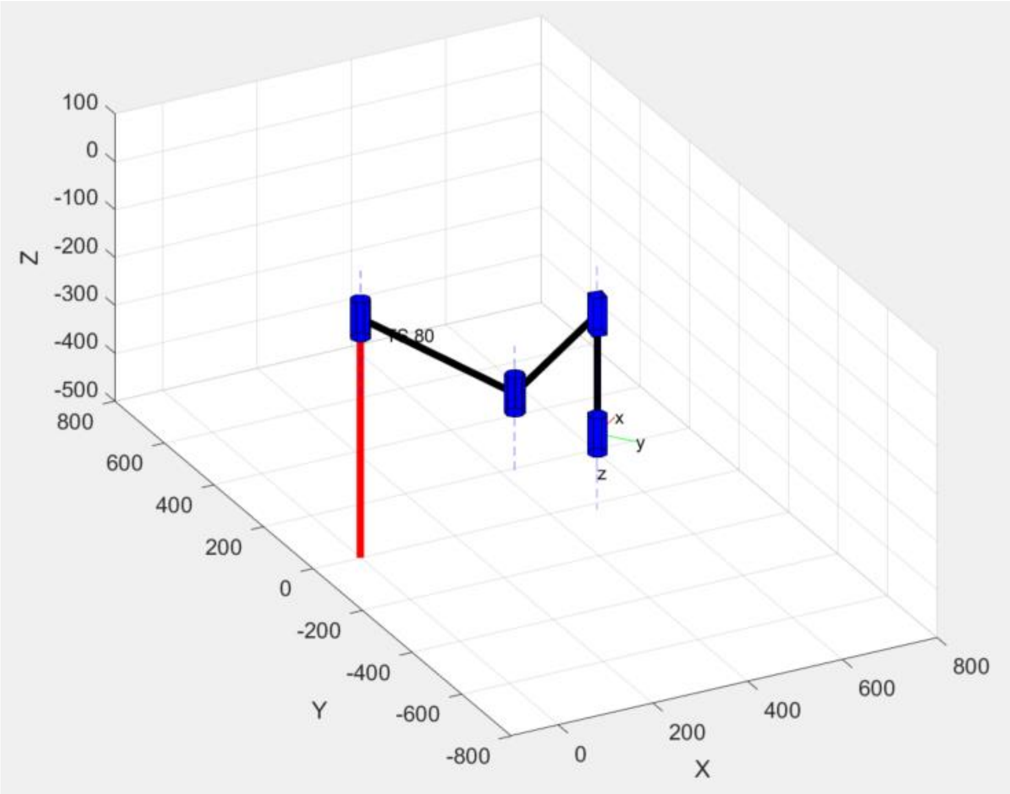


Figure 5.11 Matlab Modeling of Robot Arm

For example, if there are three cakes within the range of the robotic arm with coordinates of geometry centre identified as (x_1=450, y_1=-600), (x_2=550, y_2=-300), and (x_3=500, y_3=-100) (z_i is chosen as -200 to give a space between end effector and top surface of a cake, ψ_i is chosen to be zero to ensure that the end effector would not rotate relative to the conveyor belt surface.

The positions of each joint can then be calculated using inverse kinematic equations calculated above. Only one solution is selected for the joint position, and the calculation result is presented in Table 5.2.

Table 5.2 Positions of Joints for Each Cake

Number	θ_1 (radius)	θ_2 (radius)	d_3 (mm)	θ_4 (radius)
1	-1.2558	0.7129	200	-0.5429
2	-1.1130	1.3467	200	0.2338
3	-0.9895	1.7663	200	0.7768

The results can then be tested using Robotic Tool box, using functions below,

```
q1=[-1.2558, 0.7129,200, -0.5429];   %position of first cake
q2=[-1.1130, 1.3467,200, 0.2338];    %positon of second cake
q3=[-0.9895, 1.7663,200, 0.7768];    %position of third cake
t1=[0:0.050:10];                     %define the time from q1 to q2
t2=[0:0.050:12];                     %define the time from q2 to q3
[qa,qav,qaa]= jtraj(q1,q2,t1);       %generate trajectory from q1 to q2
[qb,qbv,qba]=jtraj(q2,q3,t2);        %generate trajectory from q2 to q3
plot(TS80,qa),hold on                %draw trajectory graph
plot(TS80,qb),hold on
```

5.2.6 Discussions on Separation and Integration of Domains

The previous four sections discussed the detailed design of cake icing automation system from market research to formalising customer needs, from the interpretation of higher level functional requirements to the elaboration of low level design parameters. This section is going to discuss the rationales made for the domain separation early in the design and the integration of different domains later in the design.

5.2.6.1 The Separation of Domains

As indicated in Axiomatic design, functional requirements and design parameters can be represented in a tree structure as shown in Figure 4.10. The tree structure separates a design in two ways. First, it clearly outlines the structure of design in each level of hierarchy and it maps to the correlated design parameters at the same level in the tree structure. In this case, Axioms and related theorems can be used to make analytical reasoning and support decision making for a design at each stage of the design process. For example, at the analysis stage of cake icing automation system, it is used to validate the top level FRs for the cake system, as shown in Equation (5.1). To capture the main requirements from the customers, CN_1 and CN_2 are identified as the requirements that the system need to achieve and the other CNs are viewed as constraints for the system. It is not to say that the other CNs are less important than the others, but it is a mechanism that emphasises the most important design requirements in a system. Forming cake icing and monitoring the geometry of cake icing are the two most important design requirements identified in this project.

Second, the tree structure also provides a way of separating the system into different domains. This can reduce the complexity of the overall system and separate the system into subsystems. The Axiomatic theory emphasises on the independence of functional requirement and minimises the interactions among different domains. According to Axiom 2, the best design always contains the minimum of information. In other words, a functional uncoupled design provides less information than a decoupled or coupled design. If domains can be separated early in the project with minimum interactions among different domains and without diminishing any design requirements, this can greatly reduce the information entropy in a design. For example, as in Equation (5.2), to fulfil FR_1 and FR_2, robotic arm system and vision system are proposed to satisfy FRs since the forming cake icing and monitoring cake geometry can be treated separately. This results the top level separation of domains, the robotic arm is in the physical side of the system and the vision system is in the cyber side of the system.

As the design goes into elaboration stage, each side of the system is investigated into detailed level. The Axiomatic theory is still a powerful tool in guiding the design in each domains of the system. The second level of physical side of the system uses vision system to locate the position of the cake, see FR_{11} and DP_{11}. The vision system used for locating the cake is the same system that is used to fulfil FR_2, in this case, it reduces the number of components or parts that may use to achieve the same results. FR_1 is then transformed into a subset of CPS system that uses both vision and robotic arm to achieve its goals. FR_1 and FR_2 are sharing the same hardware system, it is not to say that FR_1 and FR_2 are coupled. They are still independent functional requirements but using the same hardware system to achieve different goals. This leads to the minimising of information content in a design. When the design goes further into the third and fourth level of the design tree, design parameters are quantified. The finest level of design parameters are validated using Equation (3.4) to ensure that all the elements in design matrix are matched with relative functional requirement and design parameter. When the design reaches the finest level, the domain separation is finished. By adding independent functional requirements in each domain, the main design objective is achieved. The next section is going to discuss domain integration in details.

5.2.6.2 The Integration of Domains

After the design reaches its lowest level, it is the time to look back at the overall design structure, especially focusing on the tree structure. The Axiomatic design allows a decent way of separating design into small independent chunks and by adding the small chunks, the overall objectives can be achieved. Under the design scenarios for CPS, the small chunks might represent different domains, for example, the vision system to acquire cake information is in the software domain, handling cake icing using robotic arm is in mechanical domain. Both domains using quantitative approaches to achieve certain objectives and the quantitative approaches can be represented in design matrix as different design elements. Despite of the differences among domains, i.e. different properties for variables, the structure of quantitative calculations remain the same. So that, all the functional requirements and design parameters in different domains can be validated using the same Equation (3.4). By using the Axiomatic design structure in CPS design, the following points are observed:

- Design matrix provides a way of separating domains in CPS.
- The cyber and physical domains share same metadata information.
- The cyber and physical domains are intertwined in a unified way and form a coordinated system.

Design matrix provides a way of separating domains in CPS. Design matrix described in Axiomatic design provides a method of representing design in according to related functional requirements. In addition, design matrix is a useful tool to separate design into different domains and reduces the complexity of a design. For example, in Equation (5.3) the robotic arm system is further decomposed to FR_{11}, FR_{12} and FR_{13} based upon the choices made in DP_1 and DP_2. This decomposition separates the locating of geometry centre and manipulation of robotic arm. Separation of domains makes the design simple and clear, it divides a large chunk of design into smaller pieces that can be handled individually.

The cyber and physical domains share same metadata information. Although the domains are separated in the functional tree, there exists some similarities between the cyber and physical domains that need to be addressed. The design of a system is driven by initial needs and requirements. For a creative design work, multiple methods might be found to achieve the original goals. No matter what types of approaches are used to solve the design problem, the object (the problem itself) remains the same. So that the information conveyed by the object is the metadata information shared by the cyber and physical domains. For example, in the cake icing automation system, the main object in this system is the cake on the conveyor belt and both cyber and physical parts of the system are all concentrated on and dealt with the properties and information derived from the cake. In this example, the cyber and physical domains share the physical properties from the cake, such as the geometry information and the position of the cake. It is the metadata from the cake, in this case, that links the cyber part and physical part in an integrated manner.

The cyber and physical domains are intertwined in a unified way and form a coordinated system. Despite that the cyber and physical domains have their own functional requirements, the two parts of the system are cooperated with and relied on each other to accomplish the final goals. For example, in the cake icing automation system, the physical system represented as robotic arm requires the geometry information from the vision system to determine the positon of the end effector. Without this information the system would not work properly. On the other hand, the vision system requires the physical system to actually form the cake icing, as the physical system is the actuator to conduct such work. Without the actions by the robotic arm, the geometry inspection by the vision system is useless. The separation of domains are based on the functional requirements, but both cyber and physical part of the system are closely linked and worked together. So that, it is still an integrated system that accomplishes the design goals.

5.2.7 Case Study Summary

This case study uses SDP and axiomatic design to analyse an automation cake icing system. Each stage mentioned in SDP has been shown in the case study with relative detailed information. The process showed above demonstrates the usefulness of a unified design approach bringing design theory with design process.

- Successfully demonstrate the use in design analysis and validation.
- Clearly outline the system architecture early in the project to avoid unnecessary changes later in the project.
- The case study shows a way of implementing process model and design principles in a combined analysing manner.
- Axiomatic design provides a method to analytical analysis the relationship between DPs and FRs.
- Domains are dealt individually after the separation of independent functional requirements. As indicated by Axiom 1, the more thorough and specific classification in the detailed design can reduce the information contained in a design project.

5.3 Virtual Assembly Simulation System for Coal Shearer

5.3.1 Case Study Briefing

The second case study presented here uses SDP and axiomatic design in analysing the architecture of a virtual assembly system for coal shearer machine. The goal of the project is to develop a virtual assembly system to represent coal shearer machinery assembly and simulate the working conditions. It always takes a relatively longer design and testing cycle for mining machinery to develop and implement these heavy machineries. The virtual assembly system could reduce the design difficulties and presents an overall design work at early stage of a design project. Therefore, the development of such systems have recently obtained more attentions in industry. The virtual assembly system is considered to be an application of CPS for the following reasons. It is a highly integrated system using various software packages to build, calculate, and render the virtual models. Physical systems such as 3D monitors, servers, and computers need to be integrated. Sensing system would be preferably installed to provide monitoring capabilities for coal shearer, actuating devices are essential in virtual controlling and monitoring the system.

5.3.2 Analysis of Virtual Assembly Simulation System

The customer expectation of this project is to develop a virtual assembly system that supports coal shearer design and analysis, to minimise design and design validation time. The customer needs of this project can be summarised below,

- CN_1 = System can simulate assembly and movement of coal shearer machine
- CN_2 = System can be extended to install force sensors, position trackers, and digital gloves to interact with the shearer machine
- CN_3 = System can be installed in classroom for education purposes

5.3.3 Interpretation of Virtual Assembly Simulation System

Definition stage defines the top level functional requirements that need to be achieved in this project, and the customer needs can be summarised in terms of functional requirements as,

- FR_1 = Coal shearer machine assembly and disassembly
- FR_2 = Extension for digital gloves for position tracking
- FR_3 = Suitable hardware system for classroom installation

A need to requirement matrix can be obtained as,

$$\begin{Bmatrix} CN_1 \\ CN_2 \\ CN_3 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ 0 & X & 0 \\ 0 & 0 & X \end{bmatrix} \begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{Bmatrix} \tag{5.61}$$

5.3.4 Concept of Virtual Assembly Simulation System

After analysing top level requirements, then top level design parameters are being identified as,

- DP_1 = Open Source Graphs (OSG) simulation environment for coal shearer machine
- DP_2 = 5DT gloves for controlling and manipulating
- DP_3 = Supporting hardware system

The basic software used in this case is 3DsMax, OSG is an open source graphs codes that can be implemented into 3DsMax. In OSG, there are coded functions can be directly used to represent 3D models in the system. The chosen DPs can be easily implemented using existing software.

The design matrix can be illustrated as,

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ 0 & X & 0 \\ X & X & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \end{Bmatrix} \tag{5.62}$$

As it can be seen in (5.62), the design is a decoupled design, since the performance of the hardware would significantly affect the DP_1 and DP_2. The selection of DP_1 and DP_1 would have influence on the selection of DP_3. This design cannot be uncoupled due to the relationship between the hardware and the software, but it is still an acceptable design that all the functional requirements can be achieved by selecting suitable design parameters.

5.3.5 Elaboration of Virtual Assembly Simulation System

Continued from concept stage, the case study then jumps to the detailed analysis of the system architecture for the design. Based upon previous decomposition, FR_1 can then be further decomposed as,

- FR_{11} = Manipulation of models
- FR_{12} = Path recording and replaying

To achieve the manipulation of models and path recording functions, the following design parameters are used,

- DP_{11} = osg::TrackballManipulator and osg::MatrixTransform classes for models manipulation and positioning
- DP_{12} = osg::AnimationPath and osg::AnimationPathCallBack class for path recording and replying

The corresponding design matrix can be described as,

$$\begin{Bmatrix} FR_{11} \\ FR_{12} \end{Bmatrix} = \begin{bmatrix} X & 0 \\ 0 & X \end{bmatrix} \begin{Bmatrix} DP_{11} \\ DP_{12} \end{Bmatrix} \quad (5.63)$$

For FR_{11}, the model can be selected using osg::TrackballManipulator class provided in the OSG library and can move, rotate, and zoom by using osg::MatrixTransform class. For FR_{12}, the path can be obtained by using osg::AnimationPath class when the user moves or rotates parts in the assembly, the path will be saved. osg::AnimationPathCallBack class is used to replay the path of moved parts. So that, FR_{11} and FR_{12} can be achieved by DP_{11} and DP_{12} respectively, and it is an uncoupled design for FR_1.

DP_2 is a plugin device for the system, it is an input device for a system similar to the functions provided by mouse. The manipulation of parts using digital gloves is similar to the use of mouse, the digital gloves need to be calibrated first before using. So that, the leaf element of FR_2 is DP_2 and does not need to be decomposed further.

A system architecture diagram is shown in Figure 5.12. DP_1, DP_2, and DP_3 are being identified from the architecture diagram as shown below.

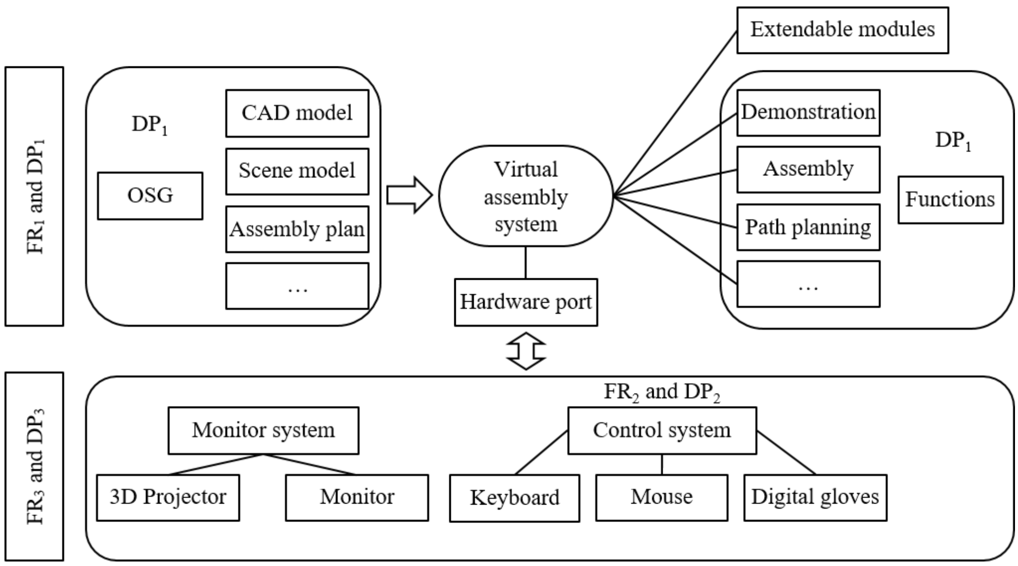


Figure 5.12 Coal Shearer Simulation System Architecture

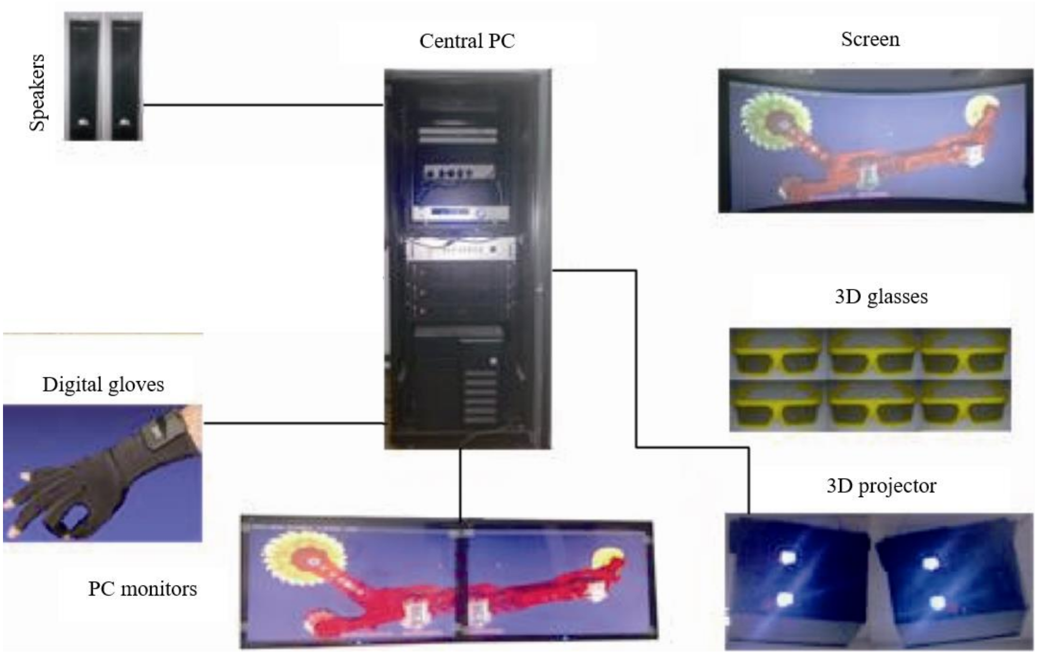


Figure 5.13 System Hardware Structure [78]

The hardware systems chosen to achieve DP_3 are standard components, 3D projector, 17 inches monitors, graphic cards etc, as shown in Figure 5.13. Figure 5.14 shows an example of an assembled coal shearer, detailed information of coal shearer can be found on the author's publication [78].

Figure 5.14 Example of 3D Assembly Model for Coal Shearer [78]

5.3.6 Case Study Summary

The process model presented in this case study can be used to analysis a system from a design perspective to extract system structure and functions. The analysis takes a breakdown approach by looking at top level design requirements at beginning without looking at detailed design of the project. This could improve the understanding of the system without losing in the complex structure of the design.

The proposed process successfully captured the main functions provided in the case study in a structured approach. This case study demonstrate some benefits of using the process model in understanding a system,

- The process model can be used for analysis and decomposition of a system.
- The process model put analyser in a design position to view and think the project.
- SDP and Axiomatic theory are used to analysis the architectural design of virtual coal shearer.

5.4 Conclusions from Case Studies

This chapter presents two case studies in different application scenarios to demonstrate the use of SDP and Axiomatic design for the design and analysis of CPS. The first case study uses a cake automation manufacture system to illustrate how the process model can be used to analyse, evaluate and validate system design using a top down approach. The second case study uses virtual assembly simulation system for coal shearer to demonstrate the use of the process model in analysing the architecture of a design.

Through the above two case studies, the process proposed in the model is used to conduct design and analysis of systems. By following the stages in the process model, a design project can be carried out in a systematic manner. Since the design is a top down approach and uses Axioms to validate design at each level, so that the design requirements can be ensured. Due to the use of Axioms, design can be evaluated at each stage of the process, the architecture design and detailed level design can be checked to ensure that the solution is satisfied functional requirements so as to customer needs. Using Axioms in the design is helpful to extract key design parameters and isolate unnecessary information at certain stage of a design. The process model is useful in analysing different types of system, since it incorporates cyber, physical, actuating, and sensing domain by using commonly accepted Axioms. By finishing the above case studies, the process model shows benefits in support of system architecture analysis, hierarchy approach in understanding and designing system, integrating design principles into process model, system checking and validating.

Chapter 6 Conclusion and Recommendations

6.1 Introduction

The final chapter draws conclusions on the research to satisfy the final objective:

- *Review and proposals for further work. (O7)*

6.2 Discussion

This thesis contributes to knowledge in the field of design process model to support CPS design and analysis by proposing a design process model to help bridging the missing part between cyber world and physical world from design process perspective. Furthermore, by using Axiomatic Design Theory in the process, the process can support design evaluation at all stages in the process.

CPS was briefly reviewed in Chapter 2 to address the first objective *"Brief introduction of Cyber Physical System"*. History background was first reviewed followed by the characteristics of CPS and the major application areas for the system.

In Chapter 3 three objectives were addressed. The second objective *"Review of problem solving and human creative thinking process that support system design"* The problem solving and creative process has been well established in management and psychology, however, there is lacking a mechanism that links problem solving and creative process with engineering design process. Prevailing problem solving and creative models were reviewed to be adapted to the design thinking for the CPS. Problem solving process can be adopted to design process models to provide further support for designers to search for the best solutions. The third objective, *"Review of Axiomatic Design Theory design principles for design and decision making"* was addressed by investigating design principles that are used for general systems. Axiomatic Design Theory provides fundamental approach to many types of systems not only for the product design and system design but for the design of software and organisation. The broad coverage of Axiomatic Design Theory can be implemented to any human made systems and it is very useful for a systematic design approach. So that, Axiomatic Design Theory can also be adapted for the needs of CPS design. The fourth objective, *"Review of system design process models in mechanical engineering process, software engineering process, and multidisciplinary process."* was carried out to identify prevailing design process that are being used in education and industry. It was found that for the development of CPS it is essential to build a process that similar to software and mechanical system but in a more unified manner. System engineering and multidisciplinary approaches, on the other hand, are general processes that are used to support design of systems. However, the multidisciplinary approaches does not have analytical methods for evaluating and validating design. It lacks the property provided by Axiomatic Design Theory to judge a design solution. So that, a combination of Axiomatic Design Theory and general design process would be important to provide a holistic process approach and an analytical tool to support design.

Chapter 4 addressed Objective 5 *"Develop process model to support early stage analysis and design of CPS".* A design process model named as Spine Design Process (SDP) was proposed in Chapter 4. The process contains six stages named as *Analysis*, *Interpretation*, *Concept*, *Elaboration*, *Construction,* and

Operation. In each stage, a domain based structure was identified dividing the process to cyber, physical, actuating, and sensing domains. Axiomatic Design Theory is embedded into the overall process to provide decision making and design evaluation support. The hierarchy structure of system functional requirements and design parameters need to be identified for a given design, and it is essential to use Axioms at every level of hierarchy to evaluate and check the design. The top down decomposition is useful when the design task is complex and the use of Axioms can ensure that better design solutions can be identified and selected.

In Chapter 5, the penultimate objective *"Evaluate design thinking model and process model in case studies"* was satisfied. Two case studies were chosen to evaluate the process model in different implementation scenarios. The first case study uses a cake icing automation system to demonstrate the use of the process and Axioms in analysing and designing of the system. The case study starts with reviewing of the markets, followed by identifying customer expectations. Then the case study jumps to the design of the system, top level design decisions are made and detailed designs are conducted. Detailed technical approaches of the case studies are conducted to show the use of Axioms in analysing of the design. The second case study uses a coal shearer virtual assembly system to demonstrate the use of the proposed process in analysing the hierarchy structure of a design, software and physical part of the system are identified. The process model along with the Axioms could support the design and analysing of such systems in a logical and unified manner.

In conclusion, the overall aim of this research project was achieved following the investigation of the above objectives.

6.3 Limitations and Recommendations for Further Work

Although the overall aim has been achieved in this research, some limitations are also need to be addressed for further work.

Within the scope of this MPhil project, time and resources are the main constrains that may have adverse influence on this project. Only the first four design stages in the process are investigated in greater details. Since the time and resources restrictions, it is unlikely to cover the whole process in a short period of time. Then, further investigations would go for the remaining two stages of the process to address the manufacturing and assembling part of a system. Furthermore, limited number of case studies have been conducted to evaluate the process. More implementation scenarios should be considered in the future work. To improve the usability of the model, it is important to incorporate more practitioners and designers to use the model. Using the model in software dominated design need to be further investigated in the future based upon the architectural analysis in the virtual coal shearer case.

In addition, as a model to support the design process, it is essential to provide more supporting tools in order to get more application coverage. A tool set and a tool set instruction would give designers more ideas of how the design need to be conducted. This would help to consolidate the process model and adapt the model to a more generalised form. By adding more support tools and resources, the process model would be acted as the central pillar of a design project.

References

1. Randolph, J.J., *A guide to writing the dissertation literature review.* Practical Assessment, Research & Evaluation, 2009. **14**(13): p. 1-13.

2. Gall, M.D., W.R. Borg, and J.P. Gall, *Educational research: An introduction*. 1996: Longman Publishing.

3. Yin, R.K., *Case study research: Design and methods*. 2013: Sage publications.

4. Lee, E.A., *Cyber-Physical Systems - Are Computing Foundations Adequate.* Proceedings of the NSF Workshop on Cyber-Physical Systems: Research Motivation, Techniques and Roadmap, 2006.

5. Lee, E.A. *Cyber Physical Systems: Design Challenges*. in *Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on*. 2008.

6. Kyoung-Dae, K. and P.R. Kumar, *Cyber-Physical Systems: A Perspective at the Centennial.* Proceedings of the IEEE, 2012. **100**(Special Centennial Issue): p. 1287-1308.

7. Khaitan, S.K. and J.D. McCalley, *Design Techniques and Applications of Cyberphysical Systems: A Survey.* Systems Journal, IEEE, 2014. **PP**(99): p. 1-16.

8. Jianhua, S., et al. *A survey of Cyber-Physical Systems*. in *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*. 2011.

9. Park, K.-J., R. Zheng, and X. Liu, *Cyber-physical systems: Milestones and research challenges.* Computer Communications, 2012. **36**(1): p. 1-7.

10. Lui, S., et al. *Cyber-Physical Systems: A New Frontier*. in *Sensor Networks, Ubiquitous and Trustworthy Computing, 2008. SUTC '08. IEEE International Conference on*. 2008.

11. Rajkumar, R., et al., *Cyber-physical systems: the next computing revolution*, in *Proceedings of the 47th Design Automation Conference*. 2010, ACM: Anaheim, California. p. 731-736.

12. Derler, P., E.A. Lee, and A.S. Vincentelli, *Modeling Cyber-Physical Systems.* Proceedings of the IEEE, 2012. **100**(1): p. 13-28.

13. Horvath, I. and B.H. Gerritsen. *Cyber-physical systems: Concepts, technologies and implementation principles*. in *Proceedings of TMCE*. 2012.

14. Gerritsen, B.H.M. and I. Horvath, *Current Drivers and Obstacles of Synergy in Cyber-Physical Systems Design.* Proceedings of the Asme International Design Engineering Technical Conferences and Computers and Information in Engineering Conference 2012, Vol 2, Pts a and B, 2012: p. 1277-1286.

15. Insup, L., et al., *Challenges and Research Directions in Medical Cyber-Physical Systems.* Proceedings of the IEEE, 2012. **100**(1): p. 75-90.

16. Lim, S., et al. *An interactive cyber-physical system (CPS) for people with disability and frail elderly people*. in *Proceedings of the 5th international conference on ubiquitous information management and communication*. 2011. ACM.

17. Lee, J., B. Bagheri, and H.-A. Kao, *A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems.* Manufacturing Letters, 2015. **3**(0): p. 18-23.

18. Ilic, M.D., et al. *Modeling future cyber-physical energy systems*. in *Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE*. 2008. IEEE.

19. Baddeley, A., *Working memory.* Science, 1992. **255**(5044): p. 556-559.

20. Lee, E.A. and S.A. Seshia, *Introduction to embedded systems: A cyber-physical systems approach*. 2011: Lee & Seshia.

21. Kahney, H., *Problem solving : a cognitive approach*. 1986: Milton Keynes ; Philadelphia : Open University Press.

22. Inc., M.-W., *Merriam-Webster's collegiate dictionary*. 2004: Merriam-Webster.

23. Wallas, G., *The art of thought*. 1931: London Cape.

24. Dewey, J., *How we think*. 1910: Heath.

25. Osborn, A.F., *Applied imagination : principles and procedures of creative thinking*. 1958: Scribner.

26. Basadur, M., G.B. Graen, and S.G. Green, *Training in creative problem solving: Effects on ideation and problem finding and solving in an*

industrial research organization. Organizational Behavior and Human Performance, 1982. **30**(1): p. 41-70.

27. Thompson, G. and M. Lordan, *A review of creativity principles applied to engineering design.* Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering, 1999. **213**(1): p. 17-31.

28. Suh, N.P., *The principles of design*. 1990: New York : Oxford University Press.

29. Suh, N.P., *Axiomatic design of mechanical systems.* Journal of Mechanical Design, 1995. **117**(B): p. 2-10.

30. Suh, N.P., *Complexity : theory and applications*. 2005: Oxford : Oxford University Press.

31. Thompson, M.K., *Improving the requirements process in Axiomatic Design Theory.* CIRP Annals - Manufacturing Technology, 2013. **62**(1): p. 115-118.

32. Kandjani, H., et al., *Using extended Axiomatic Design theory to reduce complexities in Global Software Development projects.* Computers in Industry, 2015. **67**: p. 86-96.

33. Tarenskeen, D., R. Bakker, and S. Joosten, *Applying the V Model and Axiomatic Design in the Domain of IT Architecture Practice.* Procedia CIRP, 2015. **34**: p. 263-268.

34. Vardarlıer, P., Y. Vural, and S. Birgün, *Modelling of the Strategic Recruitment Process by Axiomatic Design Principles.* Procedia - Social and Behavioral Sciences, 2014. **150**: p. 374-383.

35. Shannon, C.E., *A mathematical theory of communication.* ACM SIGMOBILE Mobile Computing and Communications Review, 2001. **5**(1): p. 3-55.

36. Wilson, D.R., *An exploratory study of complexity in axiomatic design*. 1980, Massachusetts Institute of Technology.

37. Dym, C.L., et al., *Engineering design: a project-based introduction*. 2004: Wiley New York.

38. Finkelstein, L. and A. Finkelstein, *Review of design methodology.* IEE Proceedings A (Physical Science, Measurement and Instrumentation, Management and Education, Reviews), 1983. **130**(4): p. 213-222.

39. Evbuomwan, N., S. Sivaloganathan, and A. Jebb, *A survey of design philosophies, models, methods and systems.* Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 1996. **210**(4): p. 301-320.

40. John, C. and E. Claudia, *Design process improvement [internet resource] : a review of current practice*, ed. C. John, E. Claudia, and Springerlink. 2005: London U.K. : Springer.

41. Bañares-Alcántara, R., *Representing the engineering design process: two hypotheses.* Computer-Aided Design, 1991. **23**(9): p. 595-603.

42. Gero, J.S., *A System for Computer-Aided Design in Architecture.* Computer-Aided Design. J. Vlietstra and R. Wielinga (eds). North-Holland, Amsterdam, 1973: p. 307.

43. Gero, J.S., *Design prototypes: a knowledge representation schema for design.* AI magazine, 1990. **11**(4): p. 26.

44. Sim, S.K. and A.H. Duffy, *Towards an ontology of generic engineering design activities.* Research in Engineering Design, 2003. **14**(4): p. 200-223.

45. Pugh, S., *Total design : integrated methods for successful product engineering*. 1991: Wokingham, England : Addison-Wesley.

46. Pugh, S., *Total design : towards a theory of total design*, ed. E.M. Ian and D. University of Strathclyde. Design. 1988: Glasgow, Scotland : Design Division, University of Strathclyde.

47. Pahl, G., *Engineering design*, ed. W. Wallace Ken Beitz and C. Design. 1988: Design Council.

48. French, M.J. and D. Council, *Conceptual design for engineers*. 1985: Springer.

49. Cross, N., *Engineering design methods*. 1989: Chichester ; New York : Wiley.

50. Institution, B.S., *Design management systems*, in *Part 2: Guide to managing the design of manufactured products*. 2015, BSI Standards Limited.

51. Asimow, M., *Introduction to design*. Vol. 394. 1962: Prentice-Hall Englewood Cliffs, NJ.

52. Watts, R.D., *The elements of design*, in *The design method*. 1966, Springer. p. 85-95.

53. Marples, D.L., *The decisions of engineering design.* Engineering Management, IRE Transactions on, 1961(2): p. 55-71.

54. Archer, L.B., *Systematic method for designers*. 1964: Council of Industrial Design.

55. Krick, E.V., *An introduction to engineering and engineering design.* 1965.

56. Harris, A., *Can design be taught.* Proceedings of Institution of Civil Engineers part, 1980. **1**: p. 68.

57. Royce, W.W. *Managing the development of large software systems*. in *proceedings of IEEE WESCON*. 1970. Los Angeles.

58. Boehm, B.W., *A spiral model of software development and enhancement.* Computer, 1988. **21**(5): p. 61-72.

59. Burback, R., *Software engineering methodology: the Watersluice*. 1998, Citeseer.

60. Kruchten, P., *The rational unified process: an introduction*. 2004: Addison-Wesley Professional.

61. Beck, K., *Embracing change with extreme programming.* Computer, 1999. **32**(10): p. 70-77.

62. Douglas, C.S., *Guest Editor's Introduction: Model-Driven Engineering*. 2006. p. 25-31.

63. British Standards, I., *Systems and software engineering - System life cycle processes*, in *BS ISO/IEC/IEEE 15288:2015*. 2015, BSI Standards Publication.

64. Forsberg, K. and H. Mooz. *The relationship of system engineering to the project cycle*. in *INCOSE International Symposium*. 1991. Wiley Online Library.

65. Jensen, J.C., D.H. Chang, and E.A. Lee. *A model-based design methodology for cyber-physical systems*. in *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International*. 2011.

66. Krafcik, J.F., *Triumph of the lean production system.* MIT Sloan Management Review, 1988. **30**(1): p. 41.

67. Liker, J.K., *The toyota way*. 2004: Esensi.

68. Ma, Y.-S., G. Chen, and G. Thimm, *Paradigm shift: unified and associative feature-based concurrent and collaborative engineering.* Journal of Intelligent Manufacturing, 2008. **19**(6): p. 625-641.

69. Kusiak, A., *Concurrent engineering: automation, tools, and techniques*. 1993: John Wiley & Sons.

70. Chandrasekaran, B., *Design problem solving: A task analysis.* AI magazine, 1990. **11**(4): p. 59.

71. Hauser, J.R. and D. Clausing, *The house of quality*. Harvard business review, 1988. **66**(3).

72. Suh, N.P., *Axiomatic Design: Advances and Applications (The Oxford Series on Advanced Manufacturing).* 2001.

73. Suh, N.P., *Axiomatic design theory for systems.* Research in engineering design, 1998. **10**(4): p. 189-209.

74. Keynote. *Biscuits & Cakes, Market Report*. 2013 [cited 2014 13 January]; Available from: http://www.keynote.co.uk/biscuits-%26-cakes?medium=download.

75. Sen-Ching, S.C. and C. Kamath. *Robust techniques for background subtraction in urban traffic video*. in *Electronic Imaging 2004*. 2004. International Society for Optics and Photonics.

76. Peters, R.A., *A new algorithm for image noise reduction using mathematical morphology*. Image Processing, IEEE Transactions on, 1995. **4**(5): p. 554-568.

77. Craig, J.J., *Introduction to robotics: mechanics and control*. Vol. 3. 2005: Pearson Prentice Hall Upper Saddle River.

78. Wang, X., et al., *Study on Virtual Assembly Simulation of Coal Shearer (in Chinese).* Journal of Graphics, 2015. **36**(2): p. 268-273.

79. http://www.jfe-21st-cf.or.jp/chapter_3/3e_1.html.

80. Tan, Y., et al., *A prototype architecture for cyber-physical systems.* SIGBED Rev., 2008. **5**(1): p. 1-2.

Appendix A Axiomatic Design Theorems and Corollaries

The AD theory is a useful tool to analyse and manage design. By defining philosophy and principle of design, AD provides a guideline to all types of design in engineering domain. This appendix collected theorems and corollaries that used in the thesis to help design and validate of design process, design projects, and systems. The key points of AD theory are summarized below.

There are two basic Axioms that form the basis of all theorems and corollaries, the independence theory and the information theory [28].

Axiom 1 the Independence Theory

- Maintain the independence of functional requirements;
- Or, an optimal design always maintains the independences of functional requirements;
- Or, in an acceptable design, the design parameters and the functional requirements are related in such a way that specific design parameter can be adjusted to satisfy its corresponding functional requirement without affecting other functional requirements.

Axiom 2 the Information Theory

- Minimize the information content of the design;
- Or, the best design is a functional uncoupled design that has the minimum information content.

According to AD theory, there are three types of design: uncoupled design, coupled design, and decoupled design. The design type depends on the design matrix that identified from a design.

Typically, an uncoupled design matrix has nonzero diagonal elements and all the other elements in the design matrix are zero, an example matrix is shown below [28],

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ 0 & X & 0 \\ 0 & 0 & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \end{Bmatrix}$$

A coupled design has nonzero elements on the upper and lower triangle of design matrix. It can be written as,

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{Bmatrix} = \begin{bmatrix} X & X & X \\ 0 & X & 0 \\ X & 0 & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \end{Bmatrix}$$

A decoupled design only has nonzero elements at lower triangle of design matrix. For example, a decoupled matrix has the form like,

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ 0 & X & 0 \\ X & 0 & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \end{Bmatrix}$$

If a design has nonzero elements at upper triangle of design matrix, the design can be transformed to decoupled design. For example,

$$\begin{Bmatrix} FR_1 \\ FR_2 \\ FR_3 \end{Bmatrix} = \begin{bmatrix} X & 0 & X \\ 0 & X & 0 \\ 0 & 0 & X \end{bmatrix} \begin{Bmatrix} DP_1 \\ DP_2 \\ DP_3 \end{Bmatrix}$$

can be transformed into decoupled design by change line one and line three,

$$\begin{Bmatrix} FR_3 \\ FR_2 \\ FR_1 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ 0 & X & 0 \\ X & 0 & X \end{bmatrix} \begin{Bmatrix} DP_3 \\ DP_2 \\ DP_1 \end{Bmatrix}$$

The use of design axioms and theorems are affected by the types of the design. So that it is essential to define the design type early in the design.

Appendix B Cake Manufacture Process

This appendix presents essential information supporting the analysis made in Chapter 5.2. Figure B.1 presents a typical cake manufacture process that is being used for cake production. The overall process is split into three main sub-processes, the baking process, the assembly process, and the packaging process. The cake automation project presented in Chapter 5.2 focused on assembly process especially on icing pasting and decorations. Three different lines are used to illustrate sub-processes in processing sequences.

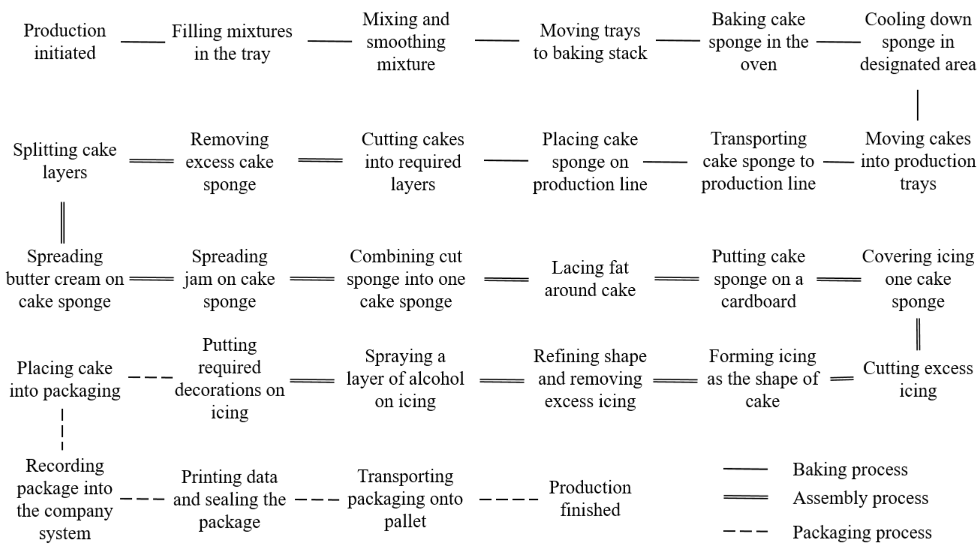


Figure B.1 Cake Manufacture Process

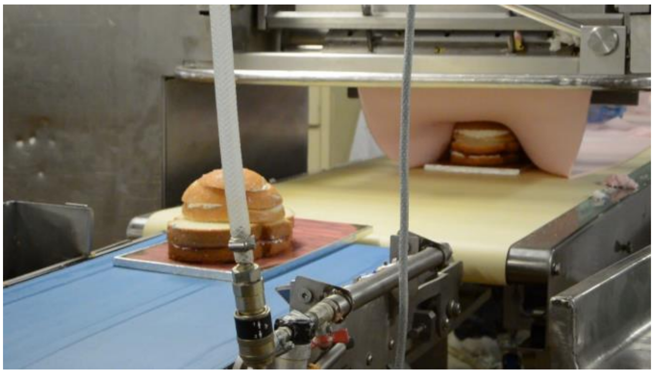


Figure B.2 Icing Dropping Machine

Figure B.3 Shaping Icing Manually

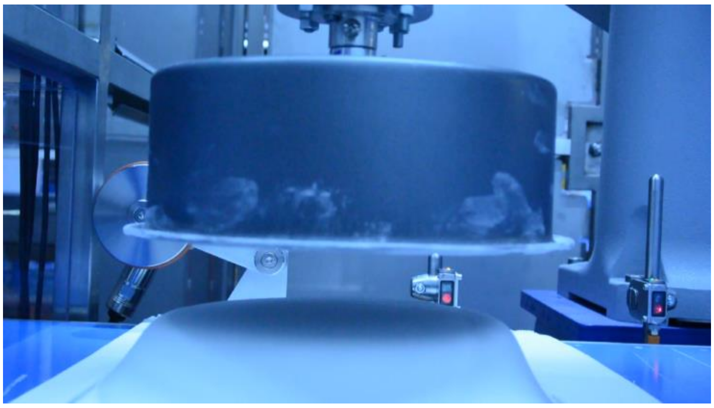


Figure B.4 End Effector Mounted at the End of Robitc Arm

Figure B.5 TS80 Robotic Arm Used in the Manufacture Process

Appendix C Matlab Codes for Simulation

Robot Arm Definition

```
clear L
L[79] = link([0  430 0   0   0   0]);
L{2} = link([pi 370 0   0   0   0]);
L[80] = link([0  0   0   0   1]);
L{4} = link([0  0   0   0   0   0]);

TS80 = robot(L, 'TS 80', 'Strathclyde');

clear L
TS80.name = 'TS 80';
TS80.manuf = 'Strathclyde';

%Position animation

q1=[-1.3673,1.5607,200,0.1934];
q2=[-0.9028,1.6000,200,0.6972];
q3=[-0.0550,2.1318,200,2.0769];

t1=[0:0.01:1];
t2=[1:0.005:4];

[qa,qav,qaa]= jtraj(q1,q2,t1);
[qb,qbv,qba]=jtraj(q2,q3,t2);

plot(TS80,qa),hold on
plot(TS80,qb),hold on
```

Robot Arm Workspace Calculation

```
clear all
clc
close all

%a=theta1 -140-140
%b=theta2 -155-155
%c=d3 0-200/0-400
%d=theta4 -500-500

tic
l1=430;
l2=370;
for a=-140:4:140
   for b=-155:4:155
```

```
        for c=0:20:200
                x=l1*cosd(a)+l2*cosd(a+b);
                y=l1*sind(a)+l2*sind(a+b);
                z=-c;
                plot3(x,y,z,'.k');
                hold on;
                grid on;
        end
    end
end

toc
```

Direct Kinematic Matrix Calculation

```
%calculate kinematic transformation matrix

clear all
clc
syms theta1 theta2 theta4 l1 l2 d3
T01=[cos(theta1) -sin(theta1) 0 0;
    sin(theta1) cos(theta1) 0 0;
    0 0 1 0;
    0 0 0 1];
T12=[cos(theta2) -sin(theta2) 0 l1;
    sin(theta2) cos(theta2) 0 0;
    0 0 1 0;
    0 0 0 1];
T23=[1 0 0 l2;
    0 -1 0 0;
    0 0 -1 -d3;
    0 0 0 1];
T34=[cos(theta4) -sin(theta4) 0 0;
    sin(theta4) cos(theta4) 0 0;
    0 0 1 0;
    0 0 0 1];

T=T01*T12*T23*T34
T41=T12*T23*T34

T32=transpose(T23)
T21=transpose(T12)
T10=transpose(T01)

T30=T32*T21*T10

syms r11 r12 r13 r21 r22 r23 r31 r32 r33 px py pz
```

```
T=[r11 r12 r13 px;
   r21 r22 r23 py;
   r31 r32 r33 pz;
   0 0 0 1];
```

Inverse Kinematic Calculation

```
clear all

syms theta1 theta2 theta3 d3 theta4

r11=1; % -1-1
r12=0; % -1-1
x=100;
y=100;
z=100;

l1=430;
l2=370;

%Calculate first answer in radius
disp('Calculate first answer in radius')
theta2=atan2(sqrt(1-((x^2+y^2-l1^2-l2^2)/(2*l1*l2))^2),(x^2+y^2-l1^2-
l2^2)/(2*l1*l2))
theta1=atan2(y,x)-atan2(l2*sin(theta2),l1+l2*cos(theta2))
d3=-z
theta4=atan2((-
r12*cos(theta1+theta2)+r11*sin(theta1+theta2)),(r11*cos(theta1+theta2)+r12*
sin(theta1+theta2)))

%Calculate first answer in degree
disp('Calculate first answer in degree')
theta2=atan2d(sqrt(1-((x^2+y^2-l1^2-l2^2)/(2*l1*l2))^2),(x^2+y^2-l1^2-
l2^2)/(2*l1*l2))
theta1=atan2d(y,x)-atan2d(l2*sind(theta2),l1+l2*cosd(theta2))
d3=-z
theta4=atan2d((-
r12*cosd(theta1+theta2)+r11*sind(theta1+theta2)),(r11*cosd(theta1+theta2)+r
12*sind(theta1+theta2)))

%Calculate second answer in radius
disp('Calculate second answer in radius')
theta2=atan2(sqrt(1-((x^2+y^2-l1^2-l2^2)/(2*l1*l2))^2),(x^2+y^2-l1^2-
l2^2)/(2*l1*l2))
theta1=atan2(y,x)-atan2(l2*sin(theta2),l1+l2*cos(theta2))
d3=-z
theta4=atan2((-
r12*cos(theta1+theta2)+r11*sin(theta1+theta2)),(r11*cos(theta1+theta2)+r12*
sin(theta1+theta2)))
```

```
%Calculate second answer in degree
disp('Calculate second answer in degree')
theta2=atan2d(sqrt(1-((x^2+y^2-l1^2-l2^2)/(2*l1*l2))^2),(x^2+y^2-l1^2-
l2^2)/(2*l1*l2))
theta1=atan2d(y,x)-atan2d(l2*sind(theta2),l1+l2*cosd(theta2))
d3=-z
theta4=atan2d((-
r12*cosd(theta1+theta2)+r11*sind(theta1+theta2)),(r11*cosd(theta1+theta2)+r
12*sind(theta1+theta2)))
```