ABSTRACT

Manufacturing techniques are based on the principles of Flexible Manufacturing and Dedicated Manufacturing for mass production. Reconfigurable Manufacturing Systems (RMS) is a manufacturing system that can provide for Agile Manufacturing (AM). This has lead to research on the concept, design and equipment implementation for RMS.

RMS requires three key capabilities: rapid changeover between products, rapid introduction to new products and unattended operation. The relationship between these manufacturing techniques has been investigated. Research has been focused on the research and design of a Reconfigurable Modular Machine (RMM) for RMS. The research has addressed the design of subsystems for RMM by using the generic modular mechatronics control. This approach included modular machine controller hardware, software, mechanical design and generic "plug-and-play" capability. These designs of subsystems allowed for rapid reconfiguration of RMS that increased system efficiency and significantly minimized manufacturing change over downtime.

TABLE OF CONTENTS

CHAPTER 1. INTRODUCTION

CHAPTER 2. LITERATURE REVIEW

CHAPTER 3. MACHINING OPERATION CLASSIFICATION

CHAPTER 4. A CASE STUDY OF RMM DESIGN

CHAPTER 5. PC-BASED MODULAR MECHATRONIC CONTROL FOR RMM

CHAPTER 6. PERFORMANCE OF THE RMM

CHAPTER 7. MODULAR DIAGNOSTICS OF PC-BASED RMM MECHATRONIC CONTROL SYSTEMS

CHAPTER 8. CONCLUSION AND DIRECTIONS FOR FUURE STUDIES

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1. INTRODUCTION

1.1. Reconfigurable Manufacturing Systems (RMSs) and Reconfigurable Modular Machines (RMMs)

Manufacturing companies of today are faced with the challenge of unpredictable, high frequency market changes in both local and international markets. There is a need for greater, more effective responsiveness by manufacturers to change their manufacturing environment. To fulfill these purposes, the concept of Agile Manufacturing was introduced and it has three key capabilities: rapid changeover between products, rapid introduction of new products and unattended operation. Agile Manufacturing is a concept in the manufacturing environment that aims to achieve the flexibility and responsiveness required to meet the continually changing market needs. Critical to the success of an Agile Manufacturing environment is the ability to reconfigure the manufacturing system and to integrate the many disparate elements contained in the system. [1]

Reconfigurable Manufacturing System (RMS) technology enables the design of a "living", evolving factory that can be rapidly and cost-effectively reconfigured exactly when the market requires a change. The RMS is necessary for sustaining profits in the face of market fluctuations caused by global competition in the 21st century [2]. As one potential solution to achieve Agile Manufacturing, the relationship between RMS and Agile Manufacturing is illustrated below in Figure 1-1.

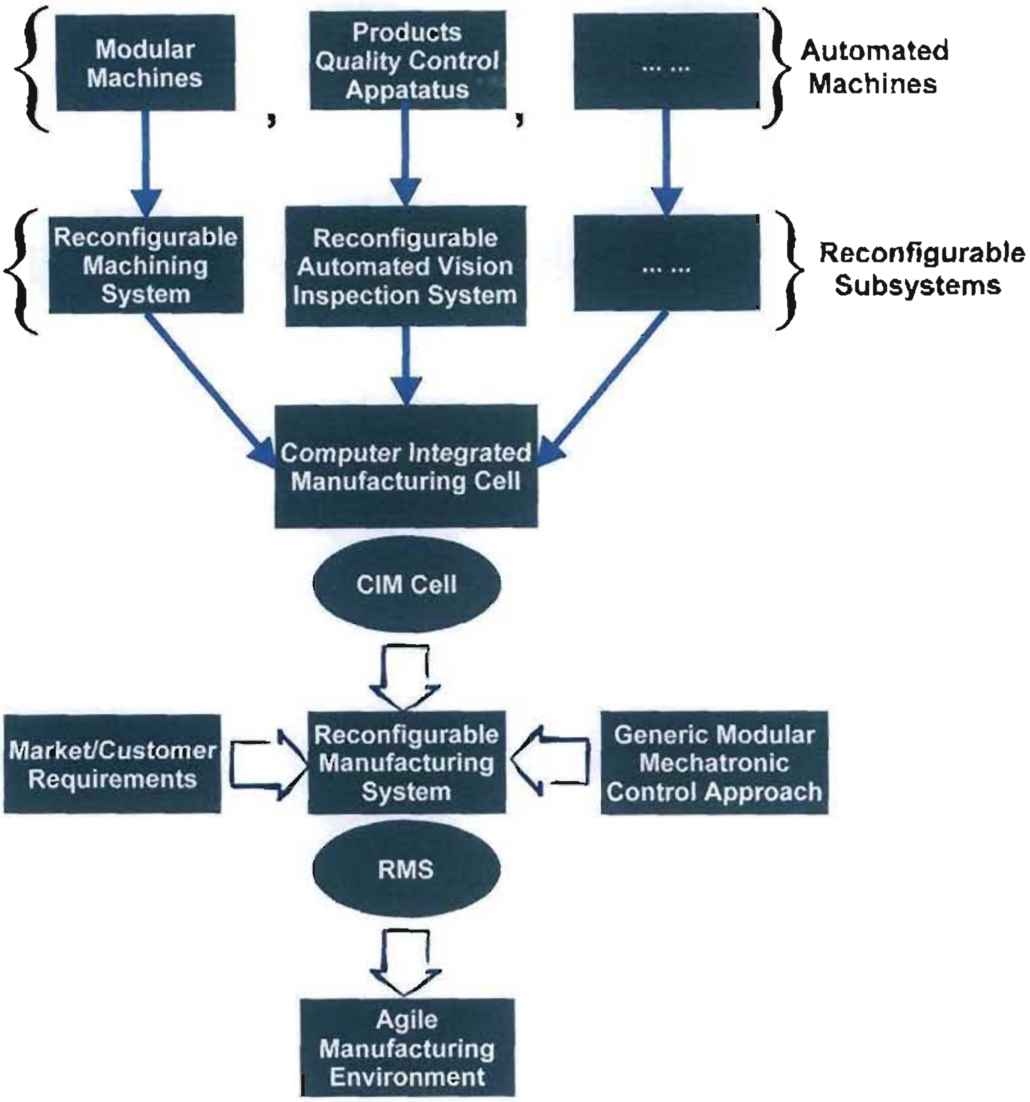


Figure 1-1 The architecture of RMS [8].

The concept of a RMS was introduced in response to rapid market changes. RMS is a new production system that is more flexible than the Dedicated Manufacturing System (DMS) and more productive and economically viable than the Flexible Manufacturing System (FMS) [3].

As one of the most important elements of RMS, the concept of Reconfigurable Machine Tool (RMT)/ Reconfigurable Modular Machine (RMM) was introduced to address these new demands. Current machine tools can be categorized into (1) transfer line machines and (2)

CNC machines. The transfer line machines are typically used in the high volume dedicated production systems and have very little flexibility to accommodate changes in the product designs and machining processes. In contrast, CNC machines possess flexibility to accommodate changes for a high product-mix but are very expensive, slow and are not suitable for high speed production. Due to intense global competition and the desire to meet the changing market demand, there is a growing trend toward mass customization to accommodate variations within a family of products. [2]

1.2. Outline of the Work

Many articles discussed the significance of RMS and predicted that it would become one of the most important production systems in the near future [4]. The methodology of RMM/RMS design is currently an important topic in RMS research. One of the basic steps to materialise the benefits of RMS is to optimize RMM design [5]. RMM design is different from traditional machine tool design in that reconfigurability, which determines the responsiveness of an RMM, is a critical factor for RMM design [6]. According to the most recent studies in RMM/RMT, reconfigurability (reflected by the number of configurations) and cost are two crucial criteria of RMM design. Therefore, by carefully choosing modules in an RMM and its module warehouse and balancing the requirements of configuration and cost, a RMM design, along with its module warehouse, can be optimized [7]. The objectives of the research project were to:

I. Research, design and analyze the modular mechanical structural that must allow the RMM design to have a large number of configurations;

II. Research and development of a low cost PC-based modular mechatronic control system, capable of smoothly coordinating the functioning of the RMM;

III. Research and development of a RMM design optimization methodology, by using multi-objective approach and establishing a module warehouse.

1.3. Thesis Layout

The layout of this thesis is based on the implementation of the generic modular mechatronic control approach to the RMM design. The chapters present the research into subtasks in sequential order, reflecting the design methodology of the generic modular mechatronic control approach.

Chapter 1 provides an introduction to the thesis and objectives of the research design.

Chapter 2 presents a review of studies that have been carried out so far by others in the RMS/RMM field. The limitations of previous work and the motivation of this study are also presented in this chapter.

Chapter 3 introduces a method to classify product family.

Chapter 4 presents a case study of RMM design. A modular design approach was utilized.

Chapter 5 discusses the development and implementation of the PC-based generic modular mechatronic control system required to ensure effective interaction of the modular components of the RMM.

Chapter 6 presents the experimental tests of the RMM.

Chapter 7 discusses the modular diagnostics of PC-Based RMM mechatonic system.

Chapter 8 is a conclusion to the thesis and offers suggestions for the further work.

CHAPTER 2. LITERATURE REVIEW

2.1. Manufacturing Systems

Manufacturing systems are defined as a collection of integrated equipment and human resources, whose function is to perform one or more processing and/or assembly operations on a raw material, part, or set of products. The integrated equipment includes production machines and tools, material handling and work positioning devices, and computer control systems. Human resources are required either periodically or even full time to keep the system running. The manufacturing systems are where the value-added work is accomplished on the part or product. In the given manufacturing systems, the following components are normally included [12]:

- production machines plus tools, fixtures, and other related hardware;
- material handling system;
- computer control systems to coordinate and/or control the above components;
- human resources.

2.2. The Development of Manufacturing Systems Theories

New technological development and market/customer requirements have major impacts on manufacturing systems design. As a result, several shifts in the focus of manufacturing systems can be observed, which can be conveniently divided into three major epochs: 1) pre computer numerical control, 2) computer numerical control (CNC), and 3) knowledge epochs. [13]

2.2.1. Dedicated Manufacturing Systems (DMSs)

In the pre-CNC epochs (before the 1970s), the emphasis was on increased production rate;

little demand existed for product variations and the market was characterized by local competition. Dedicated Manufacturing System (DMS) designed for production of a specific part was used during this epoch. The objective of DMS is to cost-effectively produce one specific part at high volumes and within a certain required quality. [13]

2.2.2. Flexible Manufacturing Systems (FMSs)

In the CNC epoch, (the 1970s and 1980s), the emphasis on cost-effective production was supplemented with a focus on improved product quality. Manufacturing systems were dramatically affected by the invention of CNC machines as they provide more accurate control and means for better quality. Furthermore, CNC machines provided necessary tools for easier integration/automation which, in turn, contributed to manufacturing of a product family on the same system. Flexible Manufacturing System (FMS) was introduced during this epoch to address changes in work orders, production schedules, part programs, and tooling for the production of several types of parts. The objective of FMS is to make possible the cost-effective manufacture of several types of parts that can change over time, on the same system at the required volume and quality. [13]

2.2.3. Reconfigurable Manufacturing Systems (RMSs)

In the knowledge epoch, (starting in the 1990s), the focus shifted to the responsiveness of the manufacturing systems characterized by intensified global competition, the fast pace of technological innovations, and enormous progress in computer and information technology [13]. These conditions require responsive manufacturing systems that can be rapidly designed, and are able to:

- convert quickly to the production of new product models;
- adjust capacity quickly, able to integrate progress technology;
- produce an increased variety of products in unpredictable quantities.

DMSs have the fixed tooling and automation. FMSs have the fixed hardware and fixed (but programmable) software. Both types of architectures do not allow for reconfiguration changes to be made. They have limited capabilities for upgrading, customization, and changes in production capacity due to the global competition. Reconfigurable Manufacturing System (RMS) was introduced at this time to respond to the new market-oriented agile manufacturing environment. A RMS, according to the definition provided by Koren *et al* [3], "is designed at the outset for rapid change in structure as well as in the hardware and software components." Koren *et al* [14] also proposed methodologies for systematic RMS design considering production capacity change, reconfiguration and ramp-up. His related work has been patented in 2002. The patent integrated the essential elements from all achievements in RMS research at the Engineering Research Centre (ERC) at the University of Michigan and shaped RMS into a valuable manufacturing design tool. RMS has a modular structure (software and hardware) that can be rearranged or replaced quickly and reliably. The reconfiguration of RMS will allow adding, removing, or modifying specific process capabilities, controls, software, or machine structure to adjust production capacity in response to changing market demands or technologies. RMS will provide customized flexibility for a particular part family, and will be open-ended, so that they can be improved, upgraded, and reconfigured, rather than replaced.

The objective of RMS is to provide the functionality and capacity that is needed, when it is needed. The given RMS configuration can be dedicated or flexible, and can change as required. RMS goes beyond DMS and FMS by permitting: 1) reduction of lead time for launching new systems and reconfiguration of existing systems, and 2) the rapid manufacturing modification and quick integration of new technology and/or new functions into existing systems. Figure 2-1 shows the main difference of three types of manufacturing systems. [13]

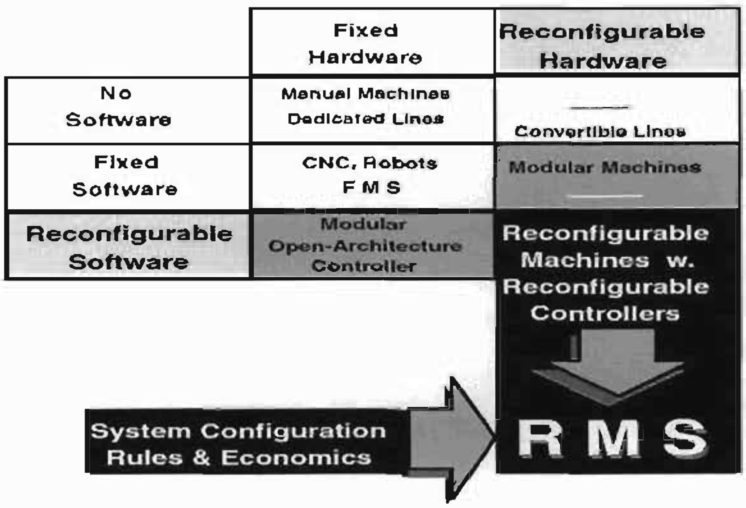


Figure 2-1 The comparison of three kinds of manufacturing systems [13].

2.3. Reconfigurable Modular Machine (RMM) Design

The design of RMM requires broad knowledge of machine design, machine tool design, kinematics and dynamics. There is no comprehensive theory or design methodology in the existing literature that is directly applicable to the RMM design. The concept of reconfiguration has been used in related fields including fixture design, assembly system design and reconfigurable robots. Researchers involved in reconfigurable robots have developed several design methodologies and some can be applied to the design of RMM. Researchers at the Carnegie Mellon University developed RMMS (Reconfigurable Modular Manipulator System). They identified the characteristics of reconfigurable machine design as a task based design and developed a design methodology for reconfigurable manipulators from the kinematics task requirements [15] [16]. I.-M. Chen applied the theory of graphs to the design of reconfigurable manipulators and proposed the concept of Assembly Incident Matrix. By manipulating the graphs, Chen generates all possible configurations which could fulfil the kinematics requirements [17].

The basic definition of the machine tool structure and its functionality is defined in ASME B5.54 [18]. S. B. Rao [19] summarized the development of the machine tools along various aspects and highlighted key research done in this field during the last decade. A. H. Slocumn [20] discussed the design process of precision machine tools with illustrative examples and pointed out the main considerations in designing new CNCs. Machining handbooks [21] [22] present calculations of machining forces and cutting speeds and provide an overview of the metal cutting practice and industrial trends.

There have been many research efforts which attempted to develop a design methodology for machine tools and most of them are based on expert systems [23] [24]. This is due to the nature of machine design itself which relies heavily on the human experience [25] [26].

In Europe, some RMT researchers suggest that RMT module designs should follow a worldwide standard so that these modules can be used as basic elements in different machines, similar to the hardware standards applied to the computer industry. However, given that the development of an internationally accepted standard is difficult and time consuming, this suggestion still remains at the theoretical stage. [27]

RMM design is often referred to as RMS hardware design, even though RMS hardware design consists of many other aspects such as machine tool positioning. As Koren and Ulsoy [14] point out, cost, quality and responsiveness are the three foundations on which every manufacturing company stands. As for an RMM, quality and responsiveness rely on processing accuracy and configurations, respectively. Hence, most studies on RMM hardware design focus mainly on these the aspects of configuration, cost and accuracy.

2.3.1. Reconfigurability of RMM

Given that RMM modular design is one of the crucial steps in achieving RMS reconfigurability, many recent studies on RMM design concentrate on seeking a method to systematically design modular and reconfigurable machines. One effort in RMM design has been to develop a method to modularize the basic components of machine tools. Walczyk *et al* [28] proposed a reconfigurable tool used in forming processes. Cecil [29] suggested adopting a computer-aided method to design a modular fixture. Perez et al [30] applied a concurrent design reference model to RMM development. Although those methods are capable of solving specific component modular design problems and the last method provided by Perez *et al* can even modularize all components of RMM, no solution has been offered to systematically design an RMM with all possible configurations.

The Engineering Research Centre (ERC) at the University of Michigan has made great contributions to RMM design. Moon and Kota [31] developed a generalized kinematics model of RMM. Koren *et al* [2] obtained a patent on an RMT, a machine tool having customized flexibility and reconfigurability, based on their research on a Virtual Arch Type Reconfigurable Machine Tool [32]. Landers et al [33] [34], Moon and Kota [31] and Moon et al [35] provided a methodology for systematic design of reconfigurable machine tools by using screw theory for kinematics and graph theory structural synthesis. Figure 2-2 includes an overview of the RMM kinematics design methodology.

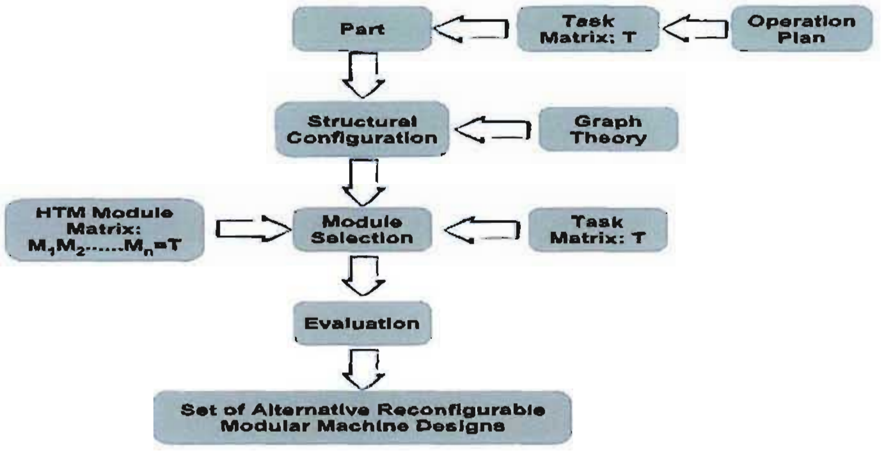


Figure 2-2 Overview of RMM kinematics design methodology [35].

In the RMM kinematics design methodology, a machining operation is transformed into a task matrix (e.g. a homogenous transformation matrix). Then, graph representations of candidate machine tools, which are generated in accordance with the functional requirements of the machining operation, are used to describe the structural configuration [33]. By verifying with the task matrix, modules of an RMM are then selected from a parameterized module library, which includes many modules with formal and unified representation schemes of mechanical functions [34]. The product of Homogeneous Transformation Matrixs (HTMs) of selected modules is compared to the task matrix. If these matrices are equal, the machine tool is kinematically viable. From this process, all possible configurations can be determined [31]. Finally, the viable configurations are examined by using other criteria such as degree of freedom, static and dynamic stiffness. As a result, by applying this methodology a set of alternative RMM designs can be automatically generated [35].

As shown in the above approach, new modules that have the same interface as that of one or more of existing modules can be easily added to a module library, thereby increasing design viability. Modular construction of an RMM ensures the reconfigurability of the machine tools. However, the inadequacies of this approach are:

I. Modules in the current module library are not optimal. Normally, modules in a module library are the basic components of an RMM, and most of them have only one function. In fact, most suppliers can provide relatively complex modules that have more than one function. These complex modules are less expensive than a combination of library modules which perform the same function as the complex modules do.

II. The above approach does not deal with module optimization, which is a critical step to reduce the difficulties associated with exchanging modules on an RMM and thereby shorts the reconfiguration time. For example, module merging, which is one part of module optimization, can reduce the numbers of modules in the reconfiguration process and thus reduce the reconfiguration time. Shortened reconfiguration time responsiveness is a critical characteristic of the RMS/RMM, the time for reconfiguring a product line and machine tools should be limited to less than 2 hours [36]. It should be pointed out that, in this study, "module warehouse" refers to a set of modules which are ready to be used as exchange modules in an RMM.

2.3.2. Cost of RMM

The costs of an RMT consist of the initial cost and operational costs [37]. Many studies have been focused on operational costs, especially processing cost [38]. Practical approaches to reduce the processing cost in machine tool design include, but are not limited to, carefully selecting the process sequence [39], machining parameters [40] [41], and tolerance design [42] [43] [39], or optimizing two or all of them [44].

Given that the approaches previously discussed are able to eventually reduce the initial investment in machine tools; few researchers have paid attention to this area. Initial investment in RMM includes both the investments in machine tools and in modules of the module warehouse, where the latter is determined by the configurations of an RMM. According to the latest information from industry, the initial cost of an RMM is estimated to be 40 percent more than that of current CNC machine tool. For this reason, reducing the initial investment cost of an RMM is still an essential part of RMM design, as it can considerably increase companies' acceptance and willingness to use RMS/RMM [39].

2.4. Summary

In this chapter, the background and priorities of RMS and RMM design problem have been discussed separately as follows:

- RMS has a modular structure, (software and hardware), that can be rearranged or replaced quickly and reliably. It will provide customized flexibility for a particular part family, and will be open-ended, so that it can be improved, upgraded, and reconfigured, rather than replaced. RMS goes beyond DMS and FMS by permitting:

1. reduction of lead time for launching new systems and reconfiguring existing systems;
2. rapid manufacturing modifications and quick integration of new technology and/or new functions into existing systems.

- RMM design is often referred to as RMS hardware design. In this research, two aspects will be focused on for RMM design: reconfigurability and cost.

CHAPTER 3. MACHINING OPERATION CLASSIFICATION

3.1. Introduction

Due to RMM is designed for a product family; there is one important element in identifying the RMM structure that is Machining Operation (MO) classification. Unlike in the FMS domain, where the assumed machine configurations strongly determine the product grouping criteria, in the RMS domain, the grouping rules are based on the product family machining requirements. Product quality, productivity and commonality across the product family are the key issues in the RMS domain. The product quality and productivity issues have a strong relationship with single product MO classification, while the product family commonality issue is based on the multiple products MO classification.

In this chapter the general MO classification was first discussed and then the development of a patterning algorithm was detailed which is a main ingredient in parallelism-based classification not only for a single product but for a product family. Because the limited research time, in this thesis only the hole type features which can be generated by drilling MOs are considered as the desired patterns.

3.2. Single Product Machining Operation Classification

Single product MO classification determines how to classify different MOs within a single product. There are two methods can be used for single product MO classification: tolerance-based method and parallelism-based method.

3.2.1. Tolerance-based Method

In order to satisfy the designer's quality requirement, tolerance-based classification is used to group MOs that have tight tolerance relationships.

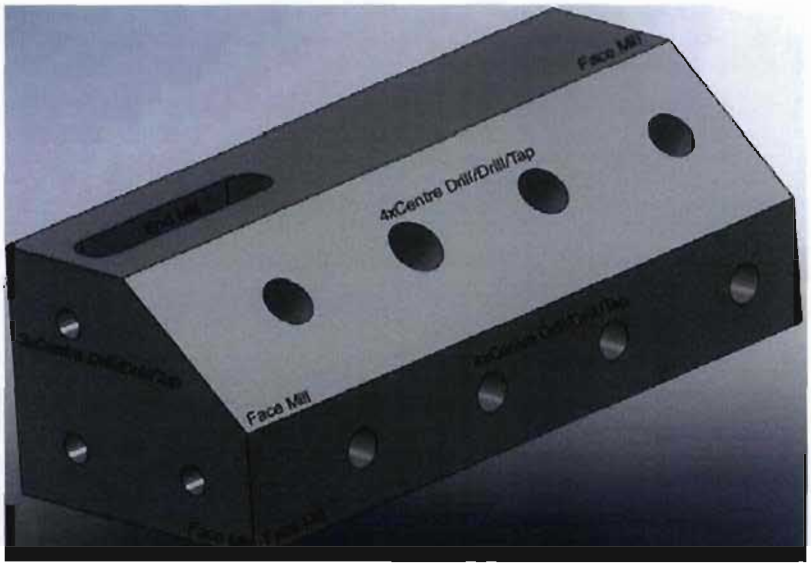


Figure 3-1 Required MOs on an example product.

The tolerance-based classifications are shown in Figure 3-2 according to the example product shown in Figure 3-1. By comparing the setup tolerance of a material handling system with the tolerance specifications of different hole machining processes required for different features, the classification rule can be shown in the rectangular block as illustrated in Figure 3-2. It can be seen that certain MOs have to be performed in the same setup according the classification rule and the material handling system's setup tolerance. $MO_{9,1}$ and $MO_{10,1}$ have to be grouped because the hole H_9 and H_{10}'s specified parallelism tolerance is tighter than the tolerance provided by the material handling system. Similarly, $MO_{8,1}$ and $MO_{10,1}$ also have to be grouped because the hole H_8 and H_{17}'s specified angularity tolerance is tighter than the tolerance provided by the material handling system. Tolerance-based method ensures that the same machine tool can always process the machining operations on features which share the tight tolerance relationship.

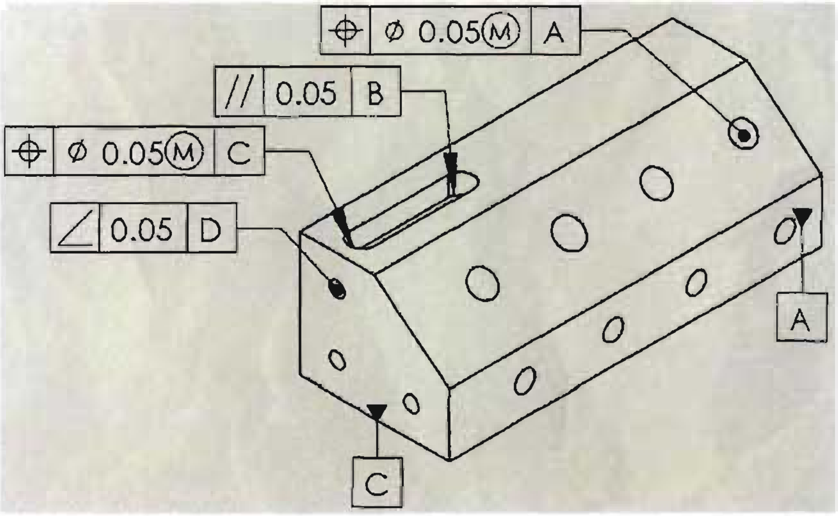


Figure 3-2 Tolerance-based classification

3.2.2. Parallelism-based Method

The group of MOs which are performed using the cutting tools mounted on a single machining unit and driven by a common set of axes can be identified by utilizing parallelism-based method. A gang spindle head can be used for this type of classification to eliminate the need for changing tools and the need for time consuming of moving single spindle tool tips. There are certain requirements that have to be satisfied for a group of MOs which can be processed in parallel. Normally, one necessary requirement is that they all have to affect the same surface and have the same approach direction. Apart from that, there are four requirements have to be checked as followings:

- *Similar type of MOs:* Different type of MOs can be grouped only if they require similar machining movements. For example, because of the different number of machine axes requirement, the drilling operation and the face milling operation cannot be grouped to be machined in parallel;
- *Operation limitation:* MOs that are grouped to be machined in parallel may cause dynamic interactions. The deflections in the cutting tool, the machine structure and the workpiece fixture may be caused by these interactions.

- *Mechanical limitation:* To evaluate a proposed parallelism-based operation, the mechanical constraints need to be considered. For example, due to the bearing size and housing limitations, there must be a lowest limitation on the distance among spindles in a gang spindle head. Therefore, for the features that will be processed in parallel, there should be enough space among them;
- *Geometrical limitation:* When the parallelism-based MOs are being processed at the same time, some kinds of interference may occur between the tool holders and the workpiece, the spindles and the fixtures. These geometrical limitations need to be considered and examined before the harmful interference happens.

When a MO falls in the scope of both a tolerance-based classification and a parallelism-based classification, the former one should be taken precedence over the latter one because the tolerance-based classification are created to meet the quality specifications' requirements. Although the parallelism-based classification has the potential to reduce a system's cycle time and the machine tools' complexity, this classification must follow certain limitations as discussed before. If some of these limitations are fail to satisfied, it becomes necessary to form new classifications by decomposing the groups of MOs.

3.3. Multiple Products Machining Operation Classification

The common and volatile machining operations across the product family are normally identified by multiple product operation classification. In order to classify common and volatile machining operations across the product family, the attributes with which the machining operations are compared needs to be defined. There are many such attributes: the type of machining process, the parameters of machining process, the machining tools, the directions of tool access and the spatial distributions of product features. In this thesis, the other attributes are assumed to be same, only the spatial distributions of product features will be considered. The concept of multiple products machining operation classification can be illustrated by the following two examples.

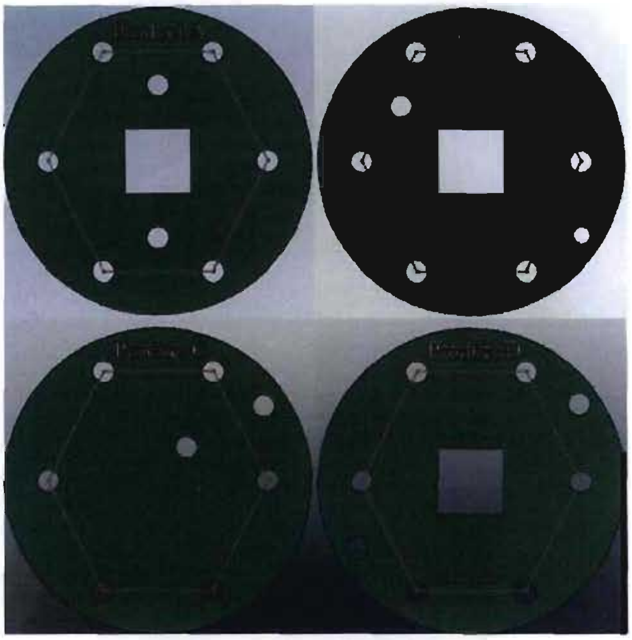


Figure 3-3 An example of product family with its common features.

Figure 3-3 shows an example of product family with four member products. All common features have been connected by the line contour. Further decision can be made toward the machine configuration and reconfiguration abilities only after recognizing these common features and identifying the common patterns. For instance, there would be two machining options for the example product family: the first one is for the common patterns, a less flexible machining machine equipped with gang spindle head can be used and for the other patterns, a more flexible machining machine equipped with a specific spindle head can be used; the second one is to built a new kind of modular machining machine equipped with a reconfigurable spindle to deal with the common patterns and other patterns at the same time.

By using rule-based techniques the tolerance-based classifications can be formed. The classification rule is comparing the setup tolerance of a material handling system with the tolerance specifications of the product features. However, for the parallelism-based classification, rule-based techniques would not work because the common patterns should be extracted to reduce the cost of redesigning machine spindle. In order to solve this problem, an analytical algorithm for patterning is needed. The patterning algorithm is a very useful tool for single product MO classification in generating the same spatial pattern subsets. Both the design effort and the manufacturing cost can be reduced by using the identical gang machining spindle heads or machines to produce these patterns. The algorithm can effectively identify the uncommon features, if they exist, and evaluate them so that common gang machining spindle head designer would know whether these uncommon features can be include in the design stage. For the multiple products MO classification, a machine designer is able to identify the common and uncommon features through the whole product family by using the patterning algorithm. After classifying the common features among different products in the product family, the same machines or machining spindle head can be designed to reduce the reconfiguration cost.

3.3.1. Mathematical Modelling of Patterning Algorithm Problem

In this section the mathematical modelling of the patterning algorithm problem will be discussed. The detailed discussions are as follows:

Given a set of H of points (the centre of a hole will be represented by each point of the set), find the minimum number of n subsets h_1, h_2, ... h_n, where

a. $\bigcap_{i=1}^{n} h_i = H$;

b. $\forall i \in [1,n]$, $\forall A,B \in h_i$, $d(A,B) \geq d_{\min}$ ($d_{\min}$ is a value of the bearing size limitation and *d(A,B)* is the distance between two points A and B);

c. $\forall i,j \in [1,n]$, $i \neq j$, a transformation $T_{ij}(h_i) = h_j$ can be found.

Rule a make sure the completion of the subdivision, for example all points belonging to the original set have been assigned to a subset. Rule b ensures that the minimum spacing constraints can be satisfied by all points within a subset. Rule c guarantees all repeated patterns would be formed.

As shown in Table 3-1 and Figure 3-4, there are six categories for the general patterning issue to be classified. For a single product, the two-pattern, n-pattern, rotational-pattern, and partial pattern can be utilized. For a product family, the optional pattern and multiple products pattern can be utilized.

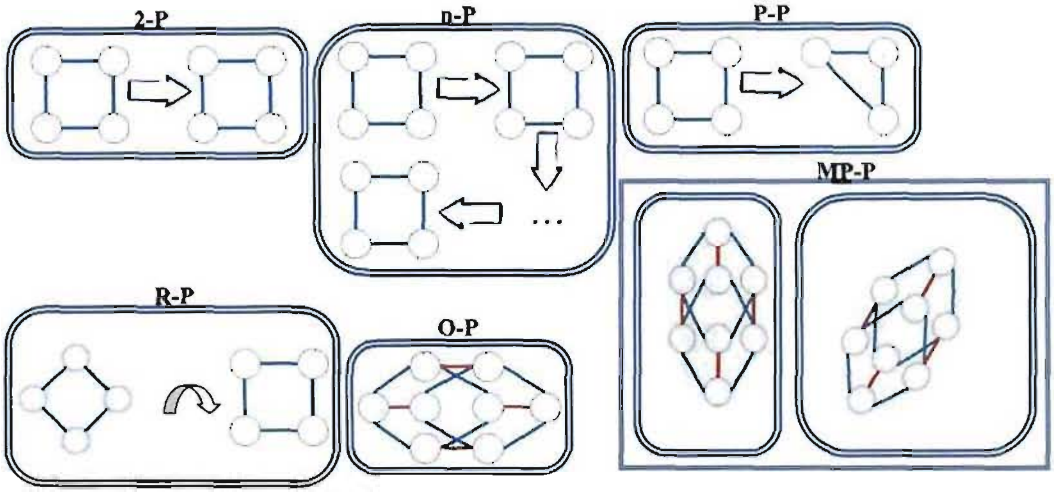
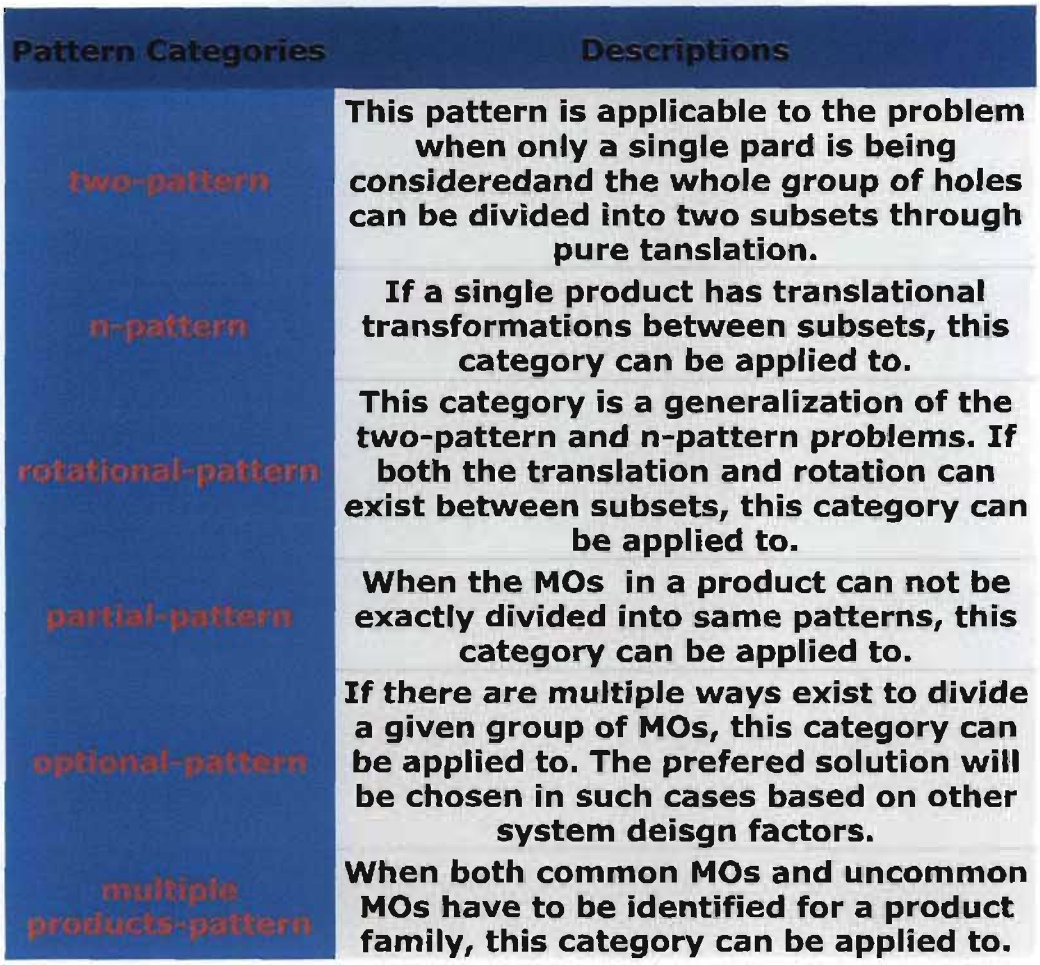


Figure 3-4 A categorization of different patterning problems.



Pattern Categories	Descriptions
two-pattern	This pattern is applicable to the problem when only a single pard is being consideredand the whole group of holes can be divided into two subsets through pure tanslation.
n-pattern	If a single product has translational transformations between subsets, this category can be applied to.
rotational-pattern	This category is a generalization of the two-pattern and n-pattern problems. If both the translation and rotation can exist between subsets, this category can be applied to.
partial-pattern	When the MOs in a product can not be exactly divided into same patterns, this category can be applied to.
optional-pattern	If there are multiple ways exist to divide a given group of MOs, this category can be applied to. The prefered solution will be chosen in such cases based on other system deisgn factors.
multiple products-pattern	When both common MOs and uncommon MOs have to be identified for a product family, this category can be applied to.

Table 3-1 General patterning categories.

There are many algorithms can be used to solve above mentioned patterning problems. First we will discuss the algorithm for 2-P problems. In this category, the MOs can be divided into two identical patterns which satisfy a translational relationship. As shown in Figure 3-5 (a), a 2-pattern problem with six holes, where two of them (h_2 and h_3) violate the minimum spacing limitation. Therefore, there is a need to divide this hole-pattern into sub-groups of MOs. A pattern separation algorithm can easily solve this problem even it is hard to recognize same patterns by pure observation. The algorithm is detailed in the following four steps:

- In the original point set, the slope and length of the line segments which connects any two points should be calculated. There are six holes in this example, so by connecting any two points, totally $k = c_2^6 = 15$ lines can be created. Figure 3-5 (b) shows some line connections together with the slope and length results;

- Among the line connections, the number of parallel lines should be counted. Figure 3-5 (c) shows some of the results for this step;
- The translational vectors should be identified;
- Pattern formation. By collecting all translational vectors' end-points, this step can be achieved. Figure 3-4 (d) shows the result of this step.

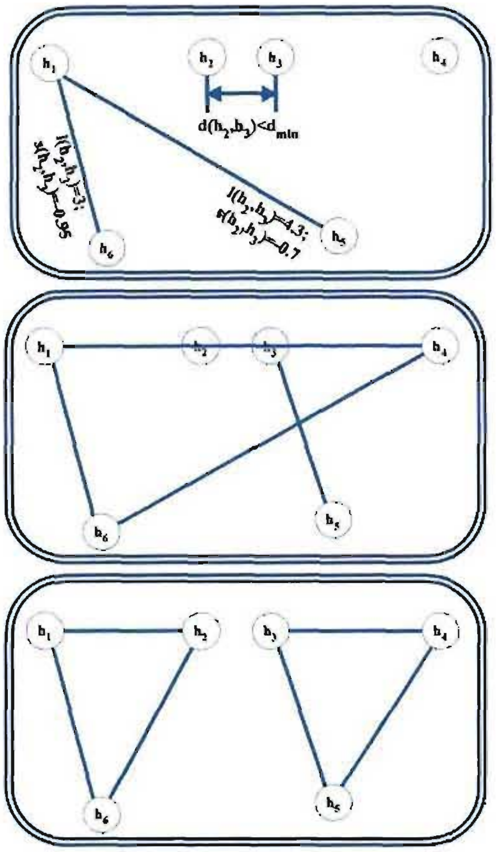


Figure 3-5 2-P algorithm example.

For n-P problems, given that there are n holes in an example case and they can be divided into a minimal of m patterns, then in each pattern the number of points will be $\frac{n}{m}$. If a set of translational vectors existed between two vectors in each group can be represented by a translational vector group, then in each group for $\frac{n}{m}$ translational vectors, there will be C_2^m translational vector groups in total and the total number of translational vectors will be $C_2^m g(\frac{n}{m})$. By adapting the 2-P algorithm, n-p algorithm can be generalized as follows:

- The first two steps will be the same as those in 2-P algorithm;
- To calculate the number of the same patterns. As shown in Figure 3-5 (a), the number of the same patterns will be 9/3=3;
- The translational vectors should be identified;
- Pattern formation. By collecting all translational vectors' end-points, this step can be achieved. Figure 3-6 (b) shows the result of this step.

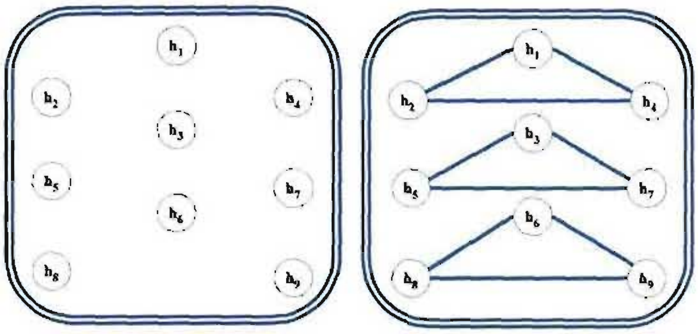


Figure 3-6 n-P algorithm example.

Figure 3-6 shows the application of above mentioned algorithm. The example has two kinds of features: casting features (two big holes) and machined features (small holes). Because there is a minimum spacing limitation exists, all the drilling MOs can not be processed at the same time by using one single gang machining spindle head. So there is a need to divide them into sub-MOs sets. Figure 3-6 (b) shows two sub-MOs sets can be classified: set 1 {h_1, h_2, h_3, h_4, h_9, h_{14}, h_8, h_{11}, h_7 and h_5} and set 2 {h_6, h_{12}, h_{13}, h_{10}, h_{19}, h_{20}, h_{18}, h_{16}, h_{17}, h_{15}}

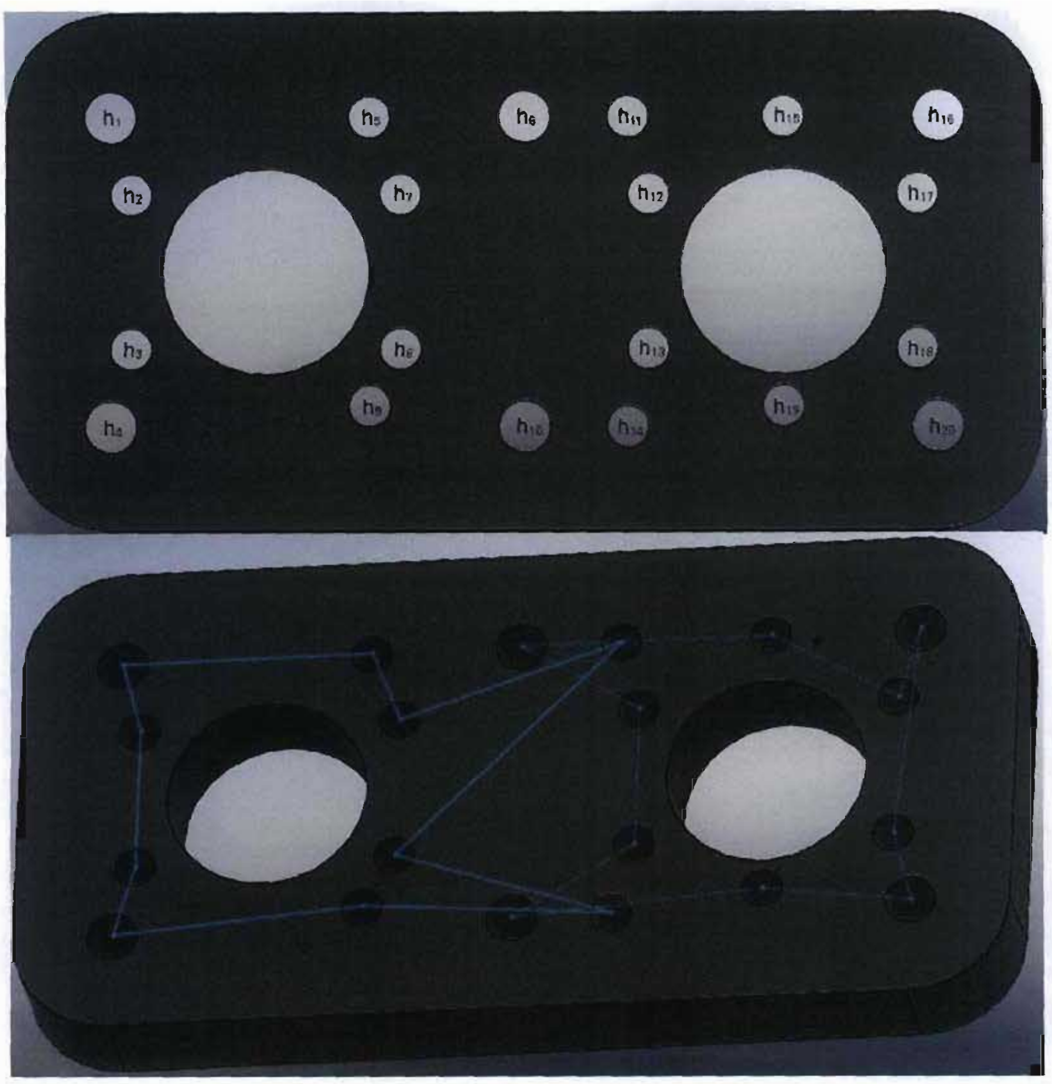


Figure 3-7 Patterning result of example product.

3.4. Summary

The translational patterning problems are the focus of this chapter. Algorithms have been designed to solve the 2-pattern and the n-pattern issues based on the assumption that the MOs can be divided into identical patterns and only translational transformations exist among patterns.

CHAPTER 4. A CASE STUDY OF RMM DESIGN

4.1. Mechanical Research Design of the RMM

The RMM was developed to be a robust, flexible and effective technology that can be implemented in agile manufacturing environments. This chapter details the mechanical design of the RMM, highlighting the conceptual development of the unique *modular mechanical structure* that is exhibited by this machine. Different design concepts are presented and discussed to illustrate the evolutionary development of the final design of the RMM. Thereafter the inception of the RMM is documented. The remaining sections of the chapter present a detailed analysis of the four independent motion systems and the foundation or *base frame* on which they are mounted.

RMM uses the modular technology that relies on the coordinated interaction of four principal motion systems: ***Automatic Part Transfer System (APTS), Automatic Part Clamping and Rotating System (APC/RS), Automatic Part Lifting System (APLS), and Automatic Tool Changing System (ATCS).*** Normally, the APTS will clamp the part and transfer it onto the APLS, and then APLS can adjust the workpiece position so that the APC/RS can clamp or rotate the workpiece properly. As soon as this is done, the ATCS will be moved to the machining start point on the workpiece. The automatic tool changing and workpiece rotation can be achieved by ATCS and APC/RS separately. Once all the machining jobs have been finished, the APTS will transfer it back onto the conveyor so that the next step: quality inspection can be performed. The integration of four motion systems enables the RMM to machine similar part from the product family without extensively changing the machine structure. The detailed CAD drawings of components for each sub-motion system can be found in Appendix A.

4.2. The Conceptual Development of APTS

4.2.1. Design Specifications

Two design methodologies were considered for the design of APTS; the modification of the existing transfer module that was worked on by previous designers in the MR^2G lab or a completely different concept would be created. Both methodologies were accordingly considered. The research that helped in the decision-making for choosing a relevant method was conducted. The research was done on the subject in order to assemble the possibilities. A ranking score sheet was developed for this purpose. What was found was that since the transfer of part involved motion from one point to another, there had to be actuation devices put in place to generate the motion. The common actuation devices that are used in this regard were motor drives, pneumatics and hydraulics. The existing transfer module used purely pneumatic actuators to generate the motions that were required. A device for picking and placing the part was also required. The dimension of the experimental workpiece is 150mm × 170mm × 100mm and the material is wood. It has to be transferred from conveyor to the APLS. In this thesis, only the functional specification of the APTS would be considered in the development of the concept design. The APTS was required to have two linear motions: one for clamping workpieces and the other for transferring workpieces to the lifting table.

4.2.2. Concept Design of APTS

Concept 1: Lead-Screw and Pneumatic Structure

In this concept some attention was paid to how the frame could be stabilized and also to have more flexibility along the vertical motion. Two frames would be designed and assembled to comprise the transfer system structure. Frame 1 would have a lead-screw that would provide vertical up and down motion which would move frame 2, and Frame 2 would be designed to carry the pneumatic cylinder which would provide the horizontal motion. The arrangement of this structure is shown in Figure 4-1.

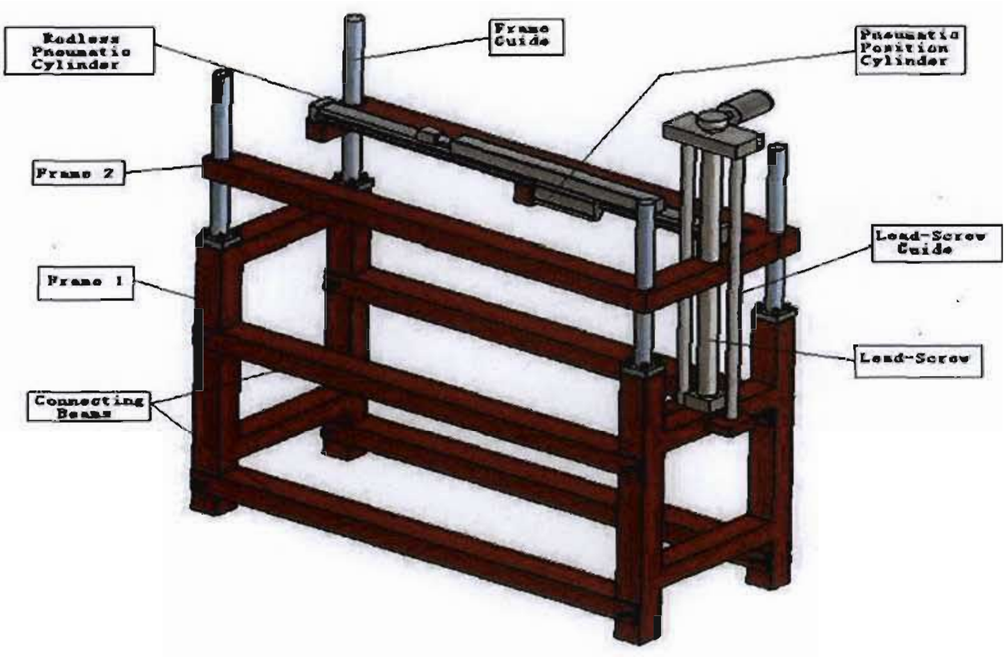


Figure 4-1 Lead-screw and pneumatic structure design for concept 1 [45].

In this design, using just one lead-screw on one side to provide vertical up and down motion might cause frame jamming because of the forces changing from one side to the other side. This would reduce the reliability of the APTS

Concept 2: Lead-Screw Structure

In this design, the horizontal and vertically motions were both provided by lead-screw. This design would be a modification of concept 1. There would be three lead-screws to provide vertical up and down motion. The rails that protruded beyond the frame would make the carriage to cover more distance. The arrangement of this structure is shown in Figure 4-2.

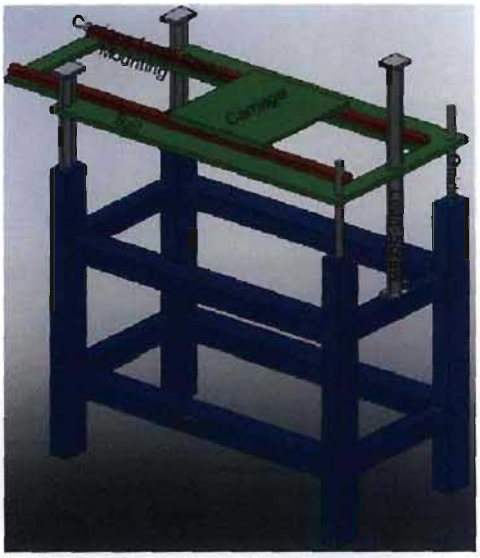


Figure 4-2 Lead-screw structure design for concept 2.

In this design, each lead-screw would require its own motor, synchronizing and hence controlling would be very difficult. Especially since there was no guarantee that the motors would operate in exactly the same way (e.g. Motor speed.). However the design is more reliable than concept 1.

Concept 3: The Final Design

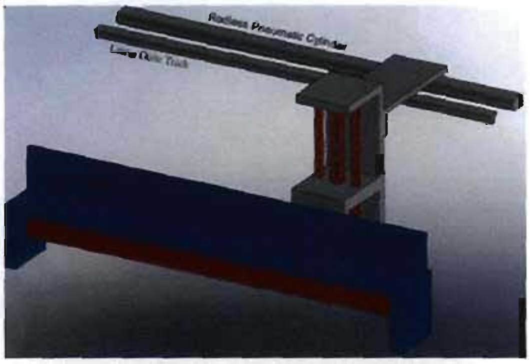


Figure 4-3 The final design concept for transfer structure.

The gripper is mounted on a lead-screw in the bracket. The lead-screw, with the gripper, can be moved horizontally by the linear pneumatic slide, and vertically by another lead-screw as shown in Figure 4-3.

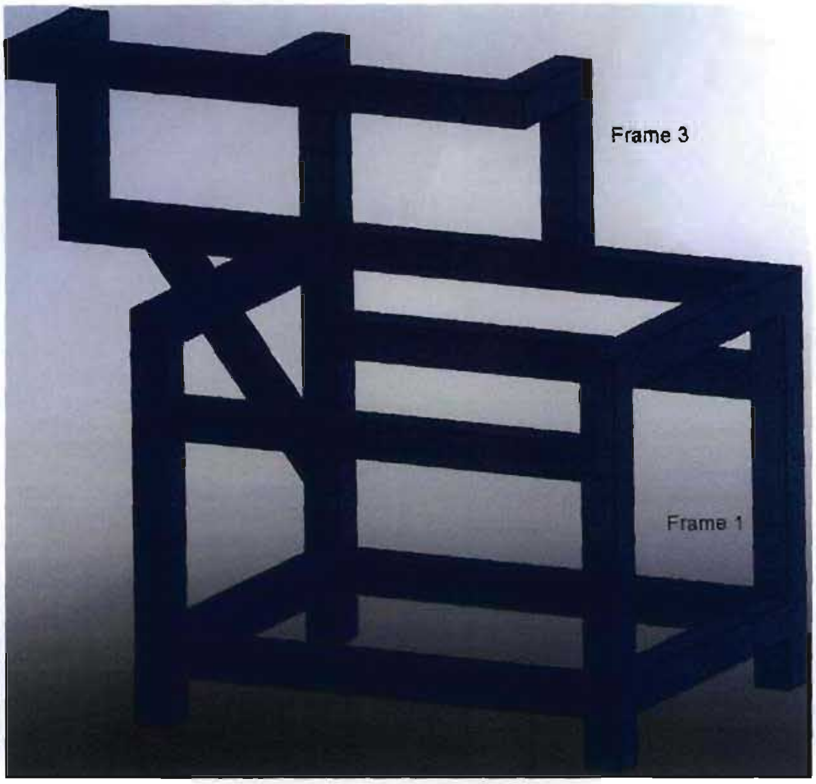
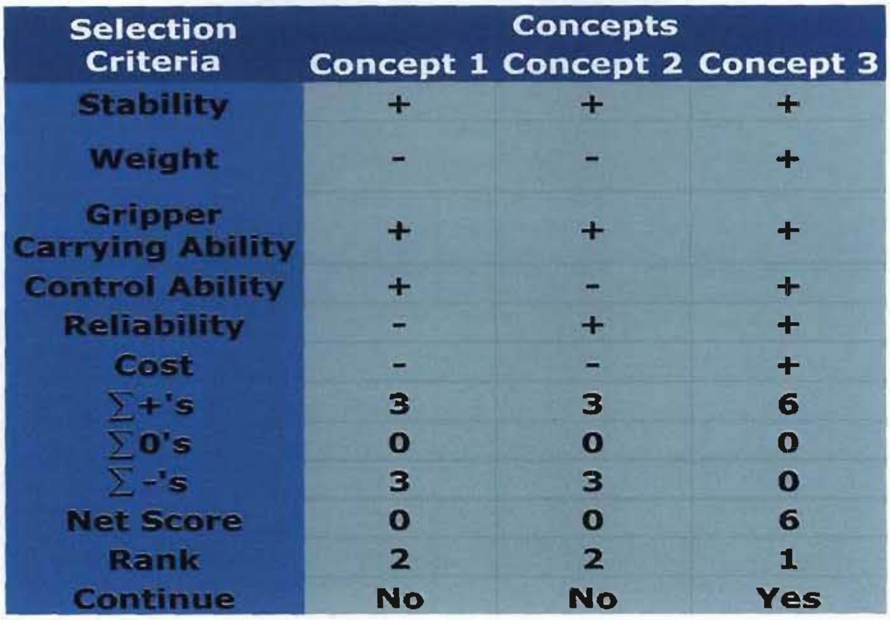


Figure 4-4 Frame 3 design for concept 4.

The whole mechanism would be mounted on frame 3 (see Figure 4-4), which in turn is mounted on top of frame 1. The top of fame 1 would have a conveyor which will deliver workpieces. The gripper and its bracket are mounted in such a manner that the gripper can pick work-pieces from the conveyor and place them on the lifting table.

4.2.3. The comparison of Four Conceptual Designs and Selection Criteria



Selection Criteria	Concepts		
	Concept 1	Concept 2	Concept 3
Stability	+	+	+
Weight	-	-	+
Gripper Carrying Ability	+	+	+
Control Ability	+	-	+
Reliability	-	+	+
Cost	-	-	+
$\sum$+'s	3	3	6
$\sum$0's	0	0	0
$\sum$-'s	3	3	0
Net Score	0	0	6
Rank	2	2	1
Continue	No	No	Yes

Table 4-1 *Comparison of three concept designs and the selection criteria.*

Table 4-1 shows the criteria that were used to evaluate the different concept designs. It uses criteria such as stability, weight, gripper carrying ability, etc. The concepts are then ranked from the highest to the lowest net score. As shown in Table 4-1, design of concept 3 was chosen as the final design for APTS.

4.3. The Conceptual Development of APLS

4.3.1. Design Description

The purpose of APLS was to receive the work piece from the APTS and moving it to the correct horizontal position for clamping. The APLS would then serve as a source of extra support during the machining process. The APLS had also to be removed during the rotation of the part by the clamping system. The main structure of the APLS was the lifting table and the table had to be at a sufficient height for the work piece to be removed (see Figure 4-5).

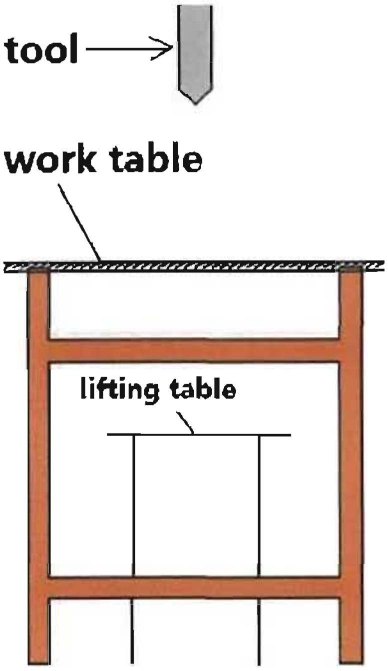


Figure 4-5 Special constraints for lifting table.

Lifting tables were used to rise and position work pieces for ergonomic access. They were typically used for material positioning, load positioning, or lifting. Lift tables are work platforms used to raise and lower material and work pieces for loading and work positioning. They were widely used in construction, automotive and garage service, electrical and power service, telecommunications, manufacturing, inventory management, wire and cable industries, painting and other applications where access to above ground locations was crucial.

Important specifications when considering lifting tables were lift capacity, vertical lift travel, and platform width and platform length. The lift capacity was the maximum force or load supported by the lift. Vertical lift travel described the difference between the fully lowered and the fully raised lift positions. The platform width was the narrow dimension of the lift platform. Platform length was the long dimension of the lift platform.

There were a number of mechanisms by which industrial lift tables might be raised or lowered. These included scissor lifts, screw lifts, rack and pinion lifts, telescoping lifts and articulated lifts. Scissor lifts used linked, folding support members to achieve lifts. The lifting action occurred when the members were drawn together, typically with a screw mechanism. Screw lifts employed the mechanical advantage of threads to lift work pieces vertically. Rack and pinion lifts functioned via a small pinion that drove a straight-toothed rack to lift the load. The telescoping lift mechanism has multiple sections that retracted and extended into and out of each other. Articulated lifts had multiple sections that unfold or articulated to lift a platform or bucket.

4.3.2. Concept Design of APLS

Concept 1: Quadruple vertical lead-screw design

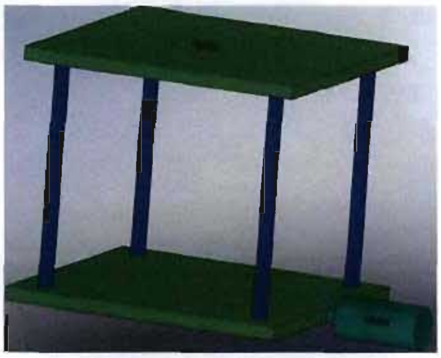


Figure 4-6 Quadruple vertical lead-screw design.

In this design four lead-screws were connected to the table. The motor drove all these lead-screws simultaneously allowing motion of the lead-screws and in-turn the table moved in the vertical direction. The possible problem of this design is that one motor might not be able to provide sufficient power.

Concept 2: Double vertical lead-screw design

Figure 4-7 Double vertical lead-screw design for concept 2.

In this concept, each of the lead-screws was driven by its own motor. The rectangular shape of the table combined with the correct positioning of the two lead-screws can offer sufficient stability of the table. Since two motors are being used, the load on the drive system is halved. The problem with having two motors is that their motions have to be perfectly synchronized in order for the table not to tilt and the lead-screws not to exert excessive force on their motors. Another disadvantage is that the design will not be economical as the motor contributes to majority of the cost.

Concept 3: Lead-screw driven scissors table

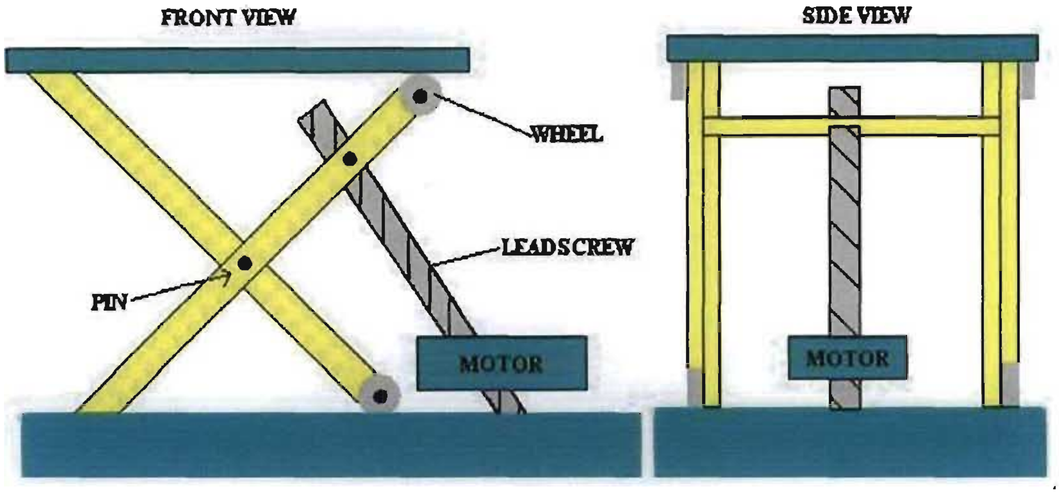


Figure 4-8 Lead-screw driven scissors table design for concept 3 [45].

This design involves the use of a common lifting table structure as used in industry. As the motor turns the lead-screw, it turns the yellow bar connected to it causing it to move. This motion then will raise or lower the table in the vertical direction. The major advantage of this design is that most of the force is absorbed by the structure and not the motor. A disadvantage would be that the concept consists of many parts making assembly and disassembly slow.

Concept 4: Final design

Figure 4-9 Final design for concept 4.

Conceptual 4 was a modification of concept 1. The main reason for this choice was that the most important factor in the design was the cost of the table. Modifications were made on the concept to overcome the disadvantages of the concept design. It was decided that a thrust bearing be used to overcome the force on the motor and the guide bars be used to overcome the problem of stability. The load carried by the lead-screw is diminished significantly by manufacturing all lifted parts from aluminium.

Both the weight of the work piece (on the table) and the table are carried by the lead-screw. The lead-screw is coupled to the motor, causing all the downward force to be transmitted to the motor. The end of the lead-screw shaft is usually threaded into the motor to withstand small downward forces. These threads may shear if the force is too large. This means that the motor will fail if subjected to heavy loads. To avoid this, the threads were replaced with a thrust bearing on the table system. The guide bars were rods that simply ride in a sleeve in order to reduce the instability problem.

4.4. The Conceptual Development of the APC/RS

4.4.1. Motion Requirement

There were two fundamental motions had to be achieved by the APC/RS; both rotary and linear motions. The rotational motion was required to rotate the work-piece during the machining operation while the linear motion was required to move the system forwards and backwards in order to grip the work piece. Concepts that were generated and those which were developed are outlined below.

4.4.2. Rotary Motion Concept Design

Concept 1: Use of Bevel Gears

Gears can be found in many machines in workshops or factories and at homes as they are often an important part of mechanical devices. The concept of using bevel gears in the clamping and rotating system in order to produce rotational motion was explored. Bevel gears are made up of conical elements and are mounted on shafts having intersecting axes. It was found that a gear box will be required. Bevel gears are relatively expensive due to the complexity in their manufacturing. A driving mechanism will be required.

Concept 2: Use of Rack and Pinion Gears

For a rotational motion a pinion coupled to the shaft fixed on the clamper and a moving rack would be used. The movement of the rack forwards or backwards would rotate the pinion which will automatically rotate the clamper. This concept required a lot of space where the rack would be mounted. The hydraulic system and the spring would be used to drive the rack.

Concept 3: Use of Spur Gears

The concept of using spur gears instead of a moving rack was explored. A pinion would be coupled to the motor to produce a rotational motion. This would take less space in a sub table than a rack. The motor would be mounted on the linearly moving bracket with a shaft coupled to the pinion and the pinion would be coupled to the motor. This method would enable the clamp to rotate in any position along the track.

Concept 4: Direct Coupling

The concept of directly coupling motors on supporting brackets was also explored. The brackets would move forward and backward on the tracks which could be mounted on a rigid table. These motors would be directly coupled to the shaft that is fixed on the clamper. The motor would directly drive the clamping system. This would be efficient because it would enable rotation in any position along the track (see Figure 4-11).

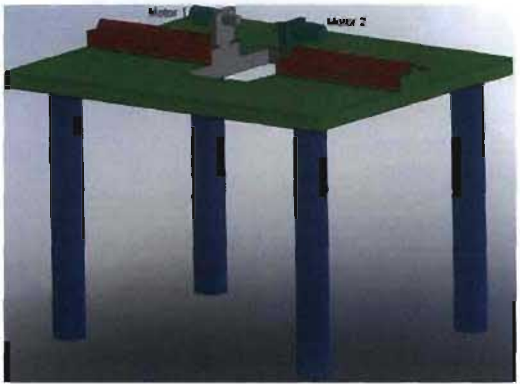
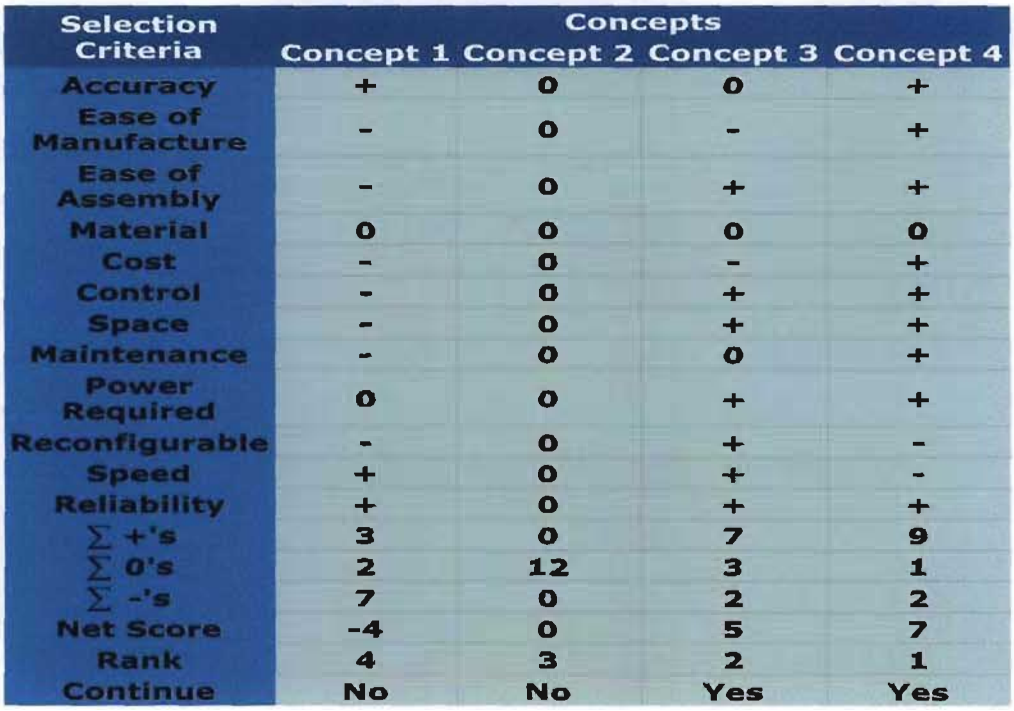


Figure 4-10 Table pinion guide design for concept 4.



Selection Criteria	Concepts Concept 1	 Concept 2	 Concept 3	 Concept 4
Accuracy	+	0	0	+
Ease of Manufacture	-	0	-	+
Ease of Assembly	-	0	+	+
Material	0	0	0	0
Cost	-	0	-	+
Control	-	0	+	+
Space	-	0	+	+
Maintenance	-	0	0	+
Power Required	0	0	+	+
Reconfigurable	-	0	+	-
Speed	+	0	+	-
Reliability	+	0	+	+
∑ +'s	3	0	7	9
∑ 0's	2	12	3	1
∑ -'s	7	0	2	2
Net Score	-4	0	5	7
Rank	4	3	2	1
Continue	No	No	Yes	Yes

Table 4-2 Comparison of four concept designs and the selection criteria.

Table 4-2 shows the criteria that were used to evaluate the different concept designs of rotary motion. It uses criteria such as accuracy, material, cost, etc. The concepts are then ranked from the highest to the lowest net score. As shown in Table 4-2, design 4 was chosen as the best design.

4.4.3. Linear Motion Concept Design

In an APR/CS, linear motion was required. Different types of linear guided motion systems were considered. Different concepts based on available guide systems were considered are presented below.

Concept 1: Friction Guides

This concept implemented friction guides that had gibs and counter gibs. The gibs would allow free play adjustment and counter-gibs would prevent the slide from lifting off the bed. A stick slip effect at low feed rates could be caused by a limiting friction. The guide ways needed to be kept free from dust required clean smooth surface to prevent wear and surface damage. Friction guide systems should be sufficiently lubricated at all times (see Figure 4-12).

Figure 4-11 Friction guides design for concept A.

Concept 2: Roller Guides

This concept implemented track roller guides that provide uniform low friction and high accuracy. The accuracy would dependend on the guiding surface and roller body. Roller guides would be preloaded. Compared to friction guides, less effort would be required to drive the roller guides. The system would not require a clean smooth surface and no special covering.

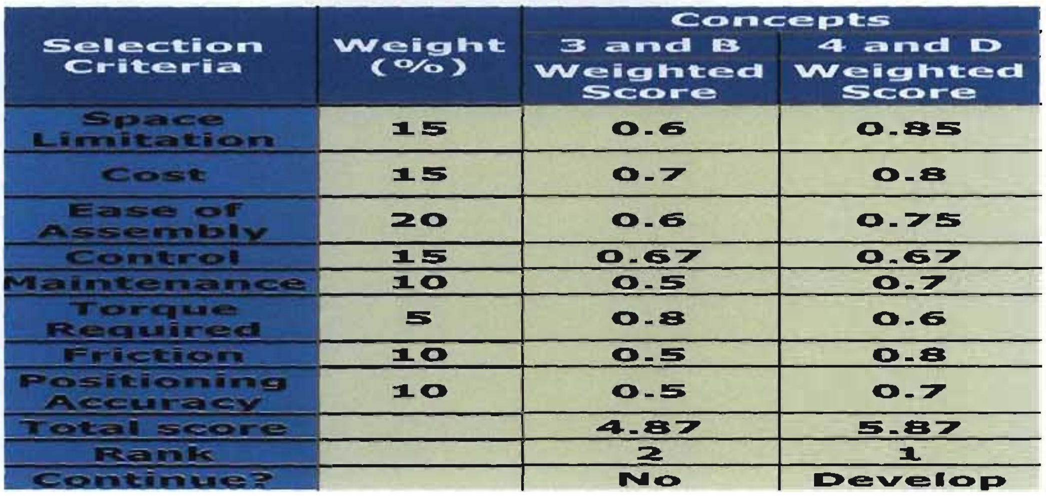
Concept 3: Hydrostatic Guides

This concept implemented hydrostatic guides that used oil under pressure to separate the two surfaces. The slide would virtually float on an oil film that is maintained by a special oil supply unit. This would result in no direct contact, and excellent friction and wear behaviour would be achieved. Oil would provide good damping characteristics that would absorb any imbalances and oscillations. The designs and maintenance of these guides would be complex. Pressure control would be important and would need to be maintained, and the system would need to be properly sealed.

Concept 4: Linear Bushing

The concept of using linear bushing which would have a high precision linear motion rolling guide that traveled along a shaft to achieve linear motion was explored. It was easy to use linear bushings instead of conventional plain bushings, because both types were used with a round shaft, and no major redesign was necessary. For each dimensional series, standard, adjustable clearance and open types were available.

4.4.4. APC/RS Final Design



Selection Criteria	Weight (%)	Concepts	
		3 and B	4 and D
		Weighted Score	Weighted Score
Space Limitation	15	0.6	0.85
Cost	15	0.7	0.8
Ease of Assembly	20	0.6	0.75
Control	15	0.67	0.67
Maintenance	10	0.5	0.7
Torque Required	5	0.8	0.6
Friction	10	0.5	0.8
Positioning Accuracy	10	0.5	0.7
Total score		4.87	5.87
Rank		2	1
Continue?		No	Develop

Table 4-3 Comparison of concept designs for rotary motion and linear motion.

Table 4-3 shows the criteria that were used to evaluate the different concept designs of rotary motion and linear motion. It uses criteria such as space limitation, cost, torque required, etc. The concepts are then ranked from the highest to the lowest net score. As shown in Table 4-3, design 4 was chosen as a better design.

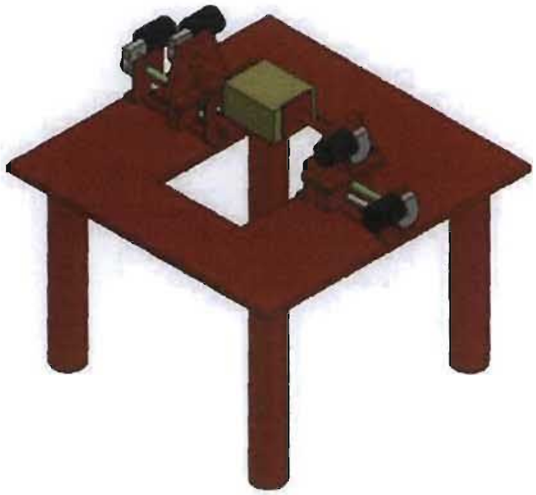


Figure 4-12 Full assembly of APC/RS [45].

4.5. THE CONCEPTUAL DEVELOPMENT OF ATCS

4.5.1. Motion Requirements

There were three fundamental motions that had to be achieved by ATCS: X, Y and Z axes linear motion. These motions were required to correctly position the tool relative to the work-piece during the machining operation.

4.5.2. Concept Design of ATCS

Concept 1: Circle plate automatic tool changing unit

In this design one circle plate would be fixed on the main shaft and it would have eight rooms for different tool modules. The whole structure was designed as a completely modular structure (see Figure 4-14). All of those modular components could be easily connected to each other, and be driven by motor and belt drive system.

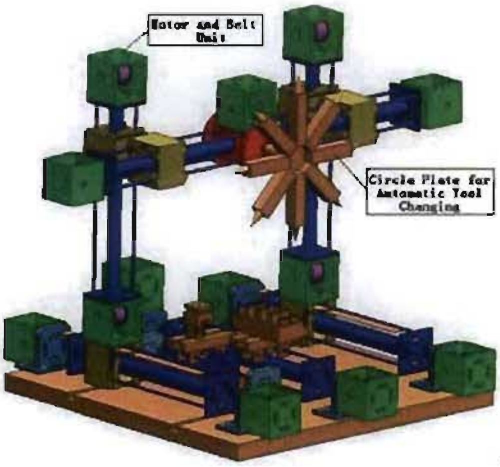


Figure 4-13 Circle plate automatic tool changing unit design for concept 1 [11].

A motor and belt drive system was widely used in this design. The system would be easy to build and control but the main disadvantage would be that they can not afford reliable X-Y-Z motion for the whole structure. The other problem would be that this structure would be difficult to implement into the MR^{2}G CIM cell.

Concept 2: Final design based on the Co-ordinate Measuring Machine (CMM)

In this design the CMM machine in MR^{2}G laboratory was chosen as the base frame for RMM (see Figure 4-15).

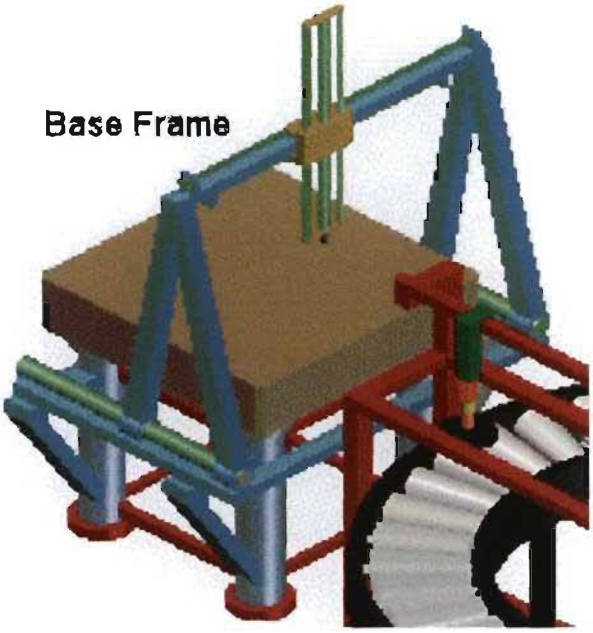


Figure 4-14 Base frame for RMM.

The CMM was a rigid structure allowing 3D movement of the tool on the work piece. A new designed tool box would be fixed onto the Z-axis (ref Figure 4-15). The double guide rail X-axis system was the primary load support. The columns had a pair of linear bearings on each side, sitting on the X-axis shaft. The shaft was driven by one 12V DC motor and it drove a lead-screw. On the Y-axis a lead-screw was directly mounted on the upper section of the columns. It was also driven by one 12V DC motor. On the Z-axis a lead-screw coupled with an automatic tool changing box was fixed on the columns.

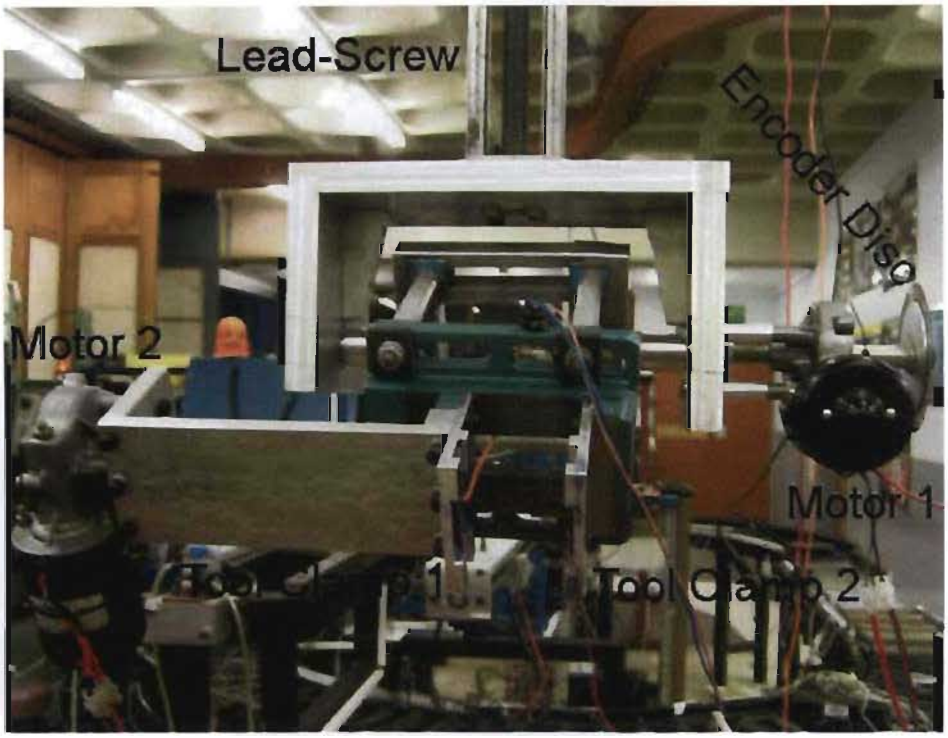


Figure 4-15 Automatic tool changing box of RMM.

In this new designed tool box, lead-Screw will first lift the whole tool box up and down to the required position and then motor 2 will drive tool clamp 1 towards tool clamp 2 to clamp a needed tool. The encoder disc controlled motor 1 will be use to rotate the selected tool from -180 degree to +180 degree. Different machining position on the workpiece can be achieved as a result.

4.6. Summary

In this chapter, the conceptual designs of RMM have been discussed and analyzed. The final design of RMM has been selected as well. In order to achieve the purpose of modular structure, four sub-systems have been designed and analyzed separately so that each sub-system can be easily modified based on the requirements.

CHAPTER 5. PC-BASED RECONFIGURABLE MECHATRONIC CONTROL FOR RMM

5.1. State of the Problem

Historically, based on the development of different manufacturing systems, the machine tool controllers used by manufacturing industries are also quite differently. With the development of new technologies and customer requirement changes, new machine control system has to be designed to keep pace. In this chapter, we will propose and development a new reconfigurable mechatronic control (RMC) method to address the problems existing in today's machine control system.

Before we start to detail our new approach, some existing machine control technologies have to be reviewed first. The first generation of machine tool controller is called dedicated machine tool controller which was widely used before the numerically control (NC) technology was invented. At that time, the controllers were purely mechanical or electromechanical systems. Because each machine controller was tailored for a specific product, so as a result, such machine controller can not be changed or upgraded without a big effort. This is its biggest disadvantage. With the development of NC technology, the second generation of machine controller which is called computer numerical control (CNC) had significantly changed modern manufacturing. In spite of many advantages of CNC systems, there are two distinct disadvantages existing in current CNC controllers: 1) the programming code used by CNC is quite old. The process of creating those codes and feeding them into machine tool controllers is tedious, inefficient, and error-prone; 2) closed machine tool controller architecture. Under this old paradigm, the controllers used by one machine are incompatible with other machines'. So once these controllers were programmed, built and delivered to end users, it was extremely difficult to be upgraded with customized functionalities. In these conditions, the machine tool controllers would be worked as a black box because for end users, they will have limited or no access to internal control algorithms or hardware. In order to solve above mentioned problems, during the past decade, many university research institutions as well as manufacturing industries had been focusing on developing a new open architecture control methodology. In this thesis, we developed a reconfigurable mechatronic control (RMC) method which falls in the open architecture control scope. It would allow the RMM control system to be modular.

5.2. Reconfigurable Mechatronic Control (RMC)

As shown in Figure 5-1, the RMC architecture is designed to control a machine tool working like part printing device. This reconfigurable control architecture can be dynamically reconfigured for direct control based on the hierarchical and modular software structure, for even though each software module is designed and built separately. By simply implementing a different control module for a mechanism, the mechanism can be reconfigured to perform differently.

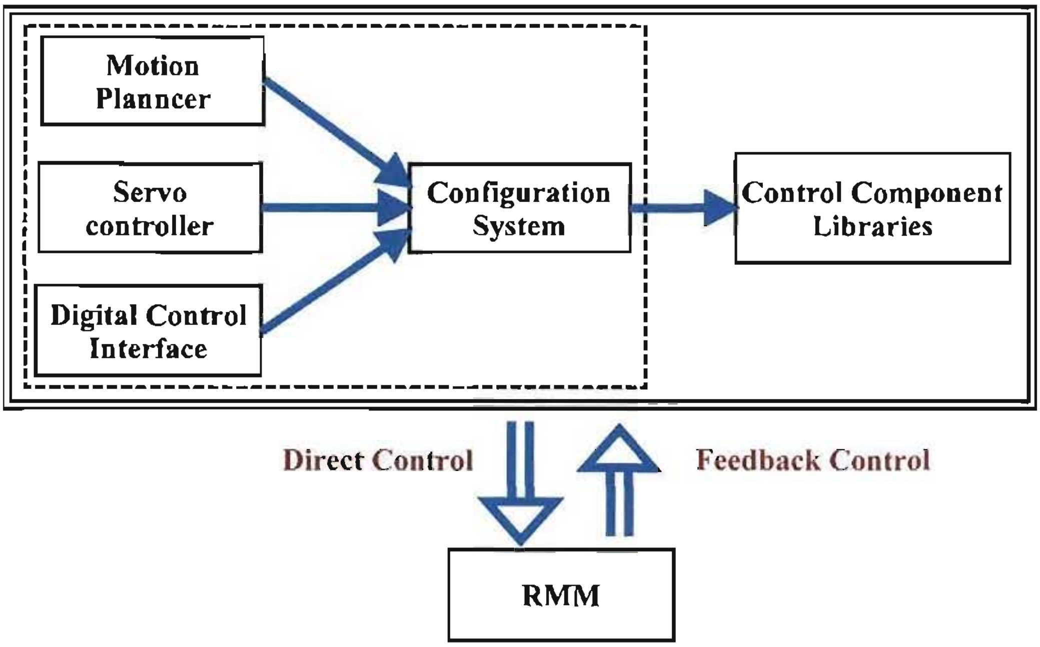


Figure 5-1 RMC architecture.

5.2.1. Interfacing Using the Standard Printer Port

The Eagle Data Acquisition technique enabled the development of the motion control system using the standard printer port available on all computers. This research has established the mechatronic principles required using the standard printer port as an interfacing tool and details of which have been appended to this thesis in Appendix D.

5.2.2. Interfacing Using the Eagle DAQ Box

The PC-based control system for the RMM was implemented using an Eagle MicroDAQ Data Acquisition Box USB-120A . This section summarizes the information about the operation of Eagle MicroDAQ Data Acquisition Box USB-120A and its corresponding EDR Software Developers Kit User Manual, which was used to develop custom operation of the digital I/O functions of the Eagle MicroDAQ Data Acquisition Box USB-120A. Figure 5-2 shows the connections of Eagle MicroDAQ Data Acquisition Box USB-120A.

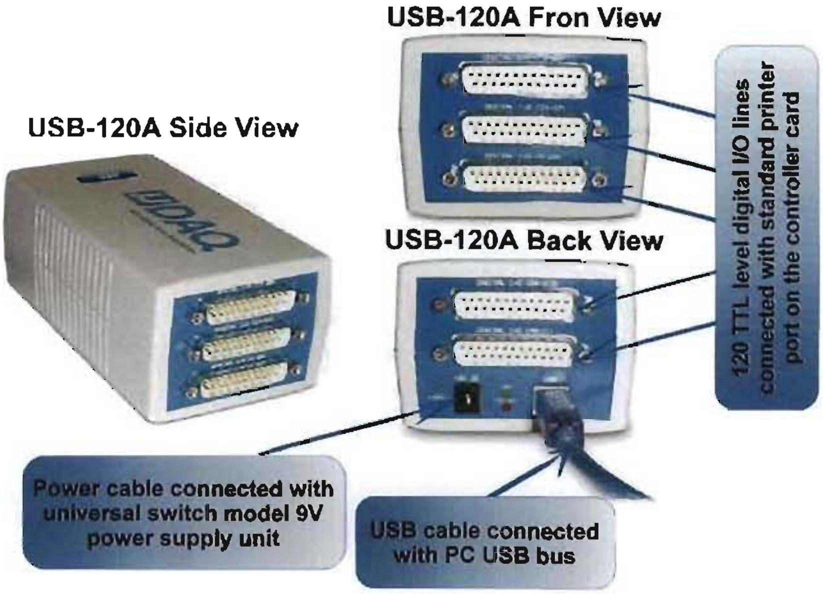


Figure 5-2 DAQ USB-120A and its connections [8].

The MicroDAQ USB-120A is general purpose digital I/O products for the USB bus. Based on the industry standard 82C55 PPI device, it communicates with the PC via the USB bus featuring 120 TTL level digital I/O lines. The I/O can be programmed in banks of 8 as inputs or outputs.

5.2.3. Interfacing Using the Pulse Width Modulation (PWM) H-Bridge Controller Card

The Pulse Width Modulation (PWM) principle utilizes chopper type circuits. The principle operates by applying the full supply voltage to the motor for short pulses of variable duration. This is done by timing the opening and closing of high frequency switch. In practice a power MOSFET (Metal-Oxide Field-Effect Transistor) is used to do this switching. A signal similar to the waveform desired across the motor is sent to the gate of the MOSFET, which is either open or closed with the signal to its gate being high (~11V) or low (~0V).

The value of the average voltage applied to the motor is varied by adjusting the ratio of the time that the 'switch' is closed to the period of the switching, i.e. duty cycle. Duty cycle is the proportion of time during which a component, device, or system is operated (ref Figure 5-3.). Advantages of the chopper circuit are that losses in a MOSEFET, just as in any semi-conductor, are less when operated in the saturated region as compared to operation in the linear region. This results in lower losses such as dissipation of energy as heat from the driver circuit and leads to improved on-board battery life.

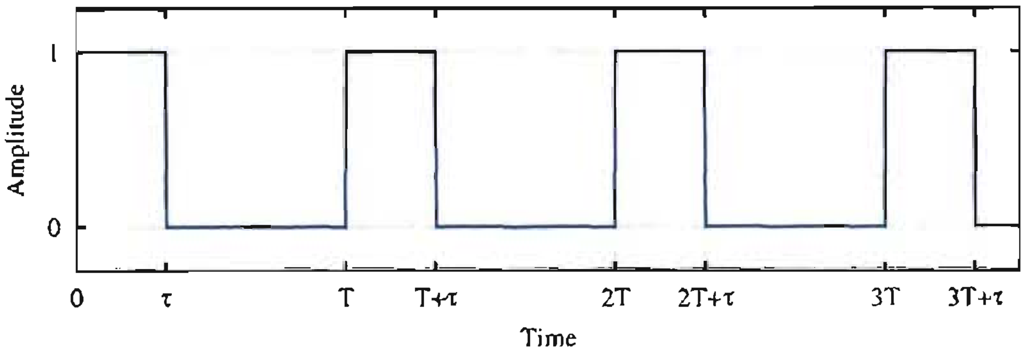


Figure 5-3 Output from a Pulse Width Modulation (PWM) circuit [46].

$$D = \frac{\tau}{T}$$

Where,

D is duty cycle;

τ is the duration that the function is non-zero;

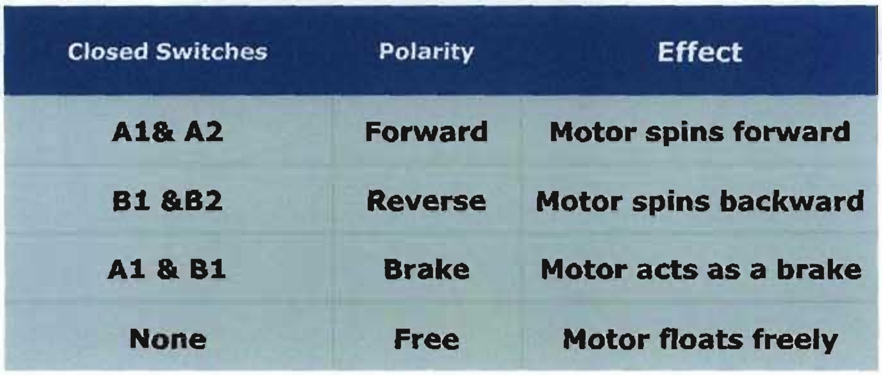
T is the period of function.

The timing circuit which controls the switching of the transistors has small currents and negligible power passing through it and so does not suffer from heating effects such as drifting. Drifting occurs because parameters like current amplification are highly dependant on temperature and also differ from transistor to transistor.

In practice, a frequency of over 15KHz must be used for the switching circuits as at lower frequencies motor operation will not be smooth and audible noise will be apparent.

The disadvantage of the PWM circuit is that it does not provide for direction reversal of the motor rotation. This would have to be done by a separate circuit using a Double-Pole Double-Throw (DPDT) relay configured specifically for polarity changing of the voltage fed to the motor, or by combining PWM with an H-bridge circuit. A basic H-bridge has 4 switches or transistors that form a circuit to drive a motor.

Since each of the four switches can be either open or closed, there are $2^4 = 16$ combinations of switch settings. Many are not useful and in fact, several should be avoided since they short out the supply current (e.g., A1 and B2 both closed at the same time). There are four combinations that are useful (see Table 5-1).



Closed Switches	Polarity	Effect
A1& A2	Forward	Motor spins forward
B1 &B2	Reverse	Motor spins backward
A1 & B1	Brake	Motor acts as a brake
None	Free	Motor floats freely

Table 5-1 Switch position for forward and reverse combination.

The H-Bridge driver circuit (see Figure 5-5) overcomes this by utilizing a push-pull approach. There are two logic level inputs to the H-Bridge circuit, A and B. If input A is made high while input B is held low, outputs A goes high which then drives the motor in one direction. If input B is made high while holding A low, output B goes high and drives the motor in the opposite direction. If both inputs are kept low, the motor is not driven and thus delivers no torque. To perform speed control, pulse width modulated signals must be provided to the inputs of the circuit. This results in the output to the motor following the waveform of the PWM input, resulting in the required speed control. Thus, the H-Bridge driver circuit with PWM inputs obtains speed as well as direction control using only solid-state electronics.

Since the circuit uses Darlington power transistors and current must flow through two transistors, forward losses are typically 1 to 2 volts. This is a significant volt drop when the maximum supply voltage is only 12V.

Figure 5-4 H-Bridge PWM speed controller card (made by MR^{2}G workshop).

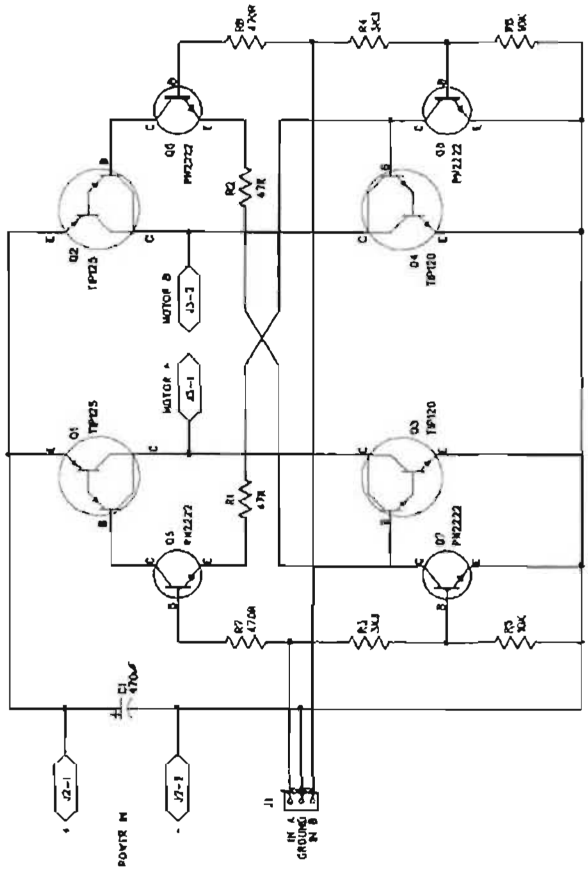


Figure 5-5 H-Bridge Driver Circuit [46].

5.3. DESIGN SPECIFICATIONS OF THE RMM CONTROL SYSTEM

The following is a list of the design specification for the RMM controller:

- *The control system had to ensure the smooth and co-ordinated functioning of every component of the RMM.* The control system had to be able to control and co-ordinat the functioning of these systems.
- *The control system needed to be modular and computer based.* The RMM had been designed to primarily function within a RMS. The RMS agile environment typically implemented CIM technologies. The design of a computer based control system was required to facilitate the interaction of the RMM with the separate CIM technologies.
- *The control system had to be designed by using modular mechatronic design principles.* Mechatronic can be defined as the systematic integration of mechanical engineering, electrical endgineering and IT to product a high quality working system. Modular mechatronic design of the RMM control system was achieved by breaking down each and every control module of the sub-system into its elementary modules. When a number of elementary modules are combined to from a working module with a particular purpose, then the combined modules also form another module. This process is continued until the working, high-level module can be identified.

5.4. MODULAR MECHATRONICS CONTROL STRATEGY FOR RMM

In keeping with the *modular mechatronics control* design protocol the development of the control system for the RMM was decomposed into four control modules as follows:

- Automatic Part Transfer System control module;

- Automatic Part Clamping/Rotating System control module;
- Automatic Part Lifting System control module;
- Automatic Tool Changing System control module;

The development of four separate control modules for the four independent motion sub-systems of the RMM. These control modules were then used as the building blocks to development the final control system for the RMM.

This section details the development and implementation of the control modules for the four motion sub-systems of the RMM (see Figure 5-6). Each of the four motion sub-systems are reviewed in turn. The review for each of the motion sub-systems discusses the development and implementation of the control algorithms specific to that system.

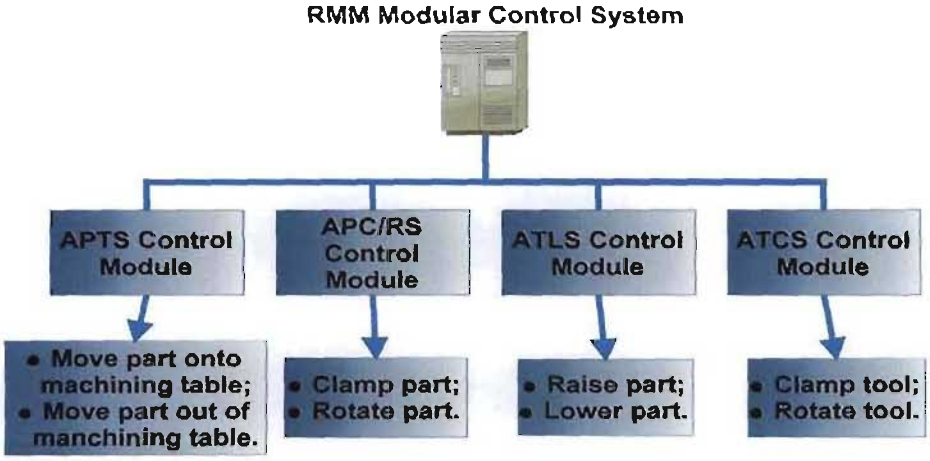
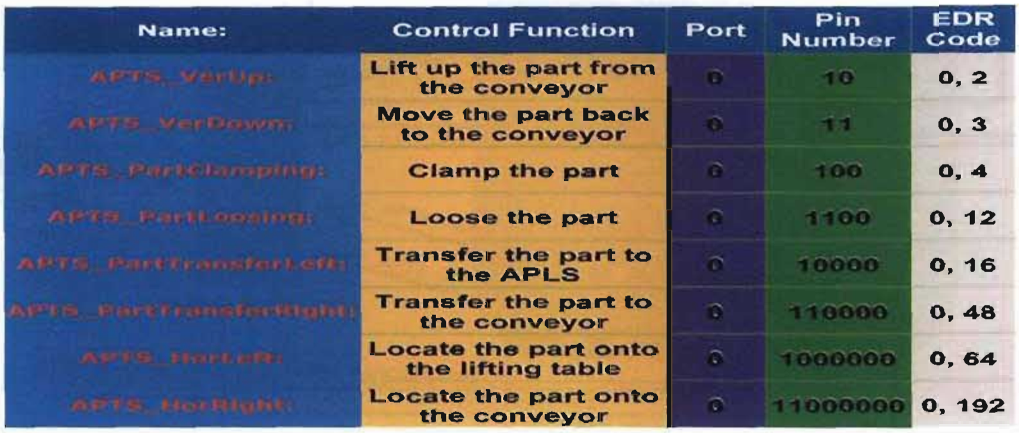


Figure 5-6 RMM four motion sub-systems' control modules.

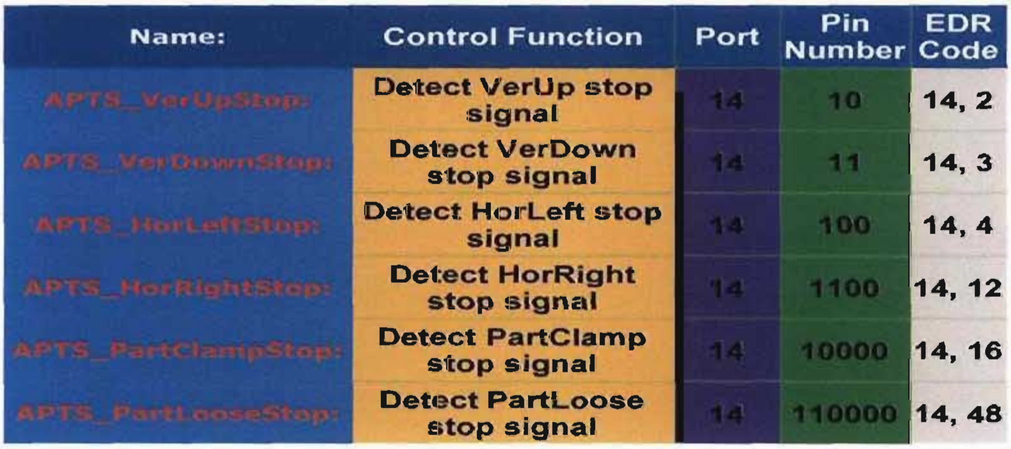
5.4.1. APTS Control Module

The control module for the Automatic Part Transfer System (APTS) required the generation of 4 activation signals to control the functioning of the drive motor in order to implement control decisions (see Figure 5-7). The control decisions were made based on the feedback information collected from the APTS using 6 feedback signals (see Figure 5-7). The activation and feedback signals for the APTS are listed in Table 5-2 and Table 5-3 respectively.



Name:	Control Function	Port	Pin Number	EDR Code
APTS_VerUp:	Lift up the part from the conveyor	0	10	0, 2
APTS_VerDown:	Move the part back to the conveyor	0	11	0, 3
APTS_PartClamping:	Clamp the part	0	100	0, 4
APTS_PartLoosing:	Loose the part	0	1100	0, 12
APTS_PartTransferLeft:	Transfer the part to the APLS	0	10000	0, 16
APTS_PartTransferRight:	Transfer the part to the conveyor	0	110000	0, 48
APTS_HorLeft:	Locate the part onto the lifting table	0	1000000	0, 64
APTS_HorRight:	Locate the part onto the conveyor	0	11000000	0, 192

Table 5-2 A list of the activation signals used in the APTS control module. The port and binary pin number of each signal is shown.



Name:	Control Function	Port	Pin Number	EDR Code
APTS_VerUpStop:	Detect VerUp stop signal	14	10	14, 2
APTS_VerDownStop:	Detect VerDown stop signal	14	11	14, 3
APTS_HorLeftStop:	Detect HorLeft stop signal	14	100	14, 4
APTS_HorRightStop:	Detect HorRight stop signal	14	1100	14, 12
APTS_PartClampStop:	Detect PartClamp stop signal	14	10000	14, 16
APTS_PartLooseStop:	Detect PartLoose stop signal	14	110000	14, 48

Table 5-3 A list of the feedback signals used in the APTS control module. The port and binary pin number of each signal is shown.

The feedback information for the APTS was collected using the position feedback sensor technique. The decision logic of the APTS control algorithm is presented as a flowchart (see Figure 5-8). A VB 6.0 project, *APTS control module.vbp*, was developed to implement the APTS control algorithm. A full scripting of the documented source code for *APTS Control Module.vbp* is given in Appendix C.

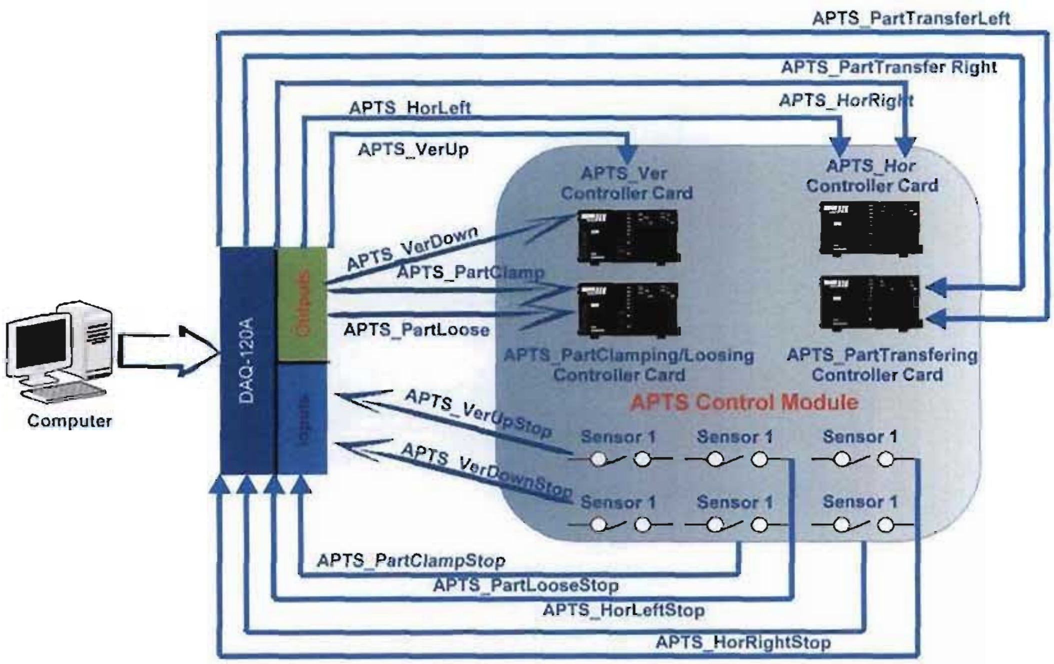


Figure 5-7 The block diagram for the APTS.

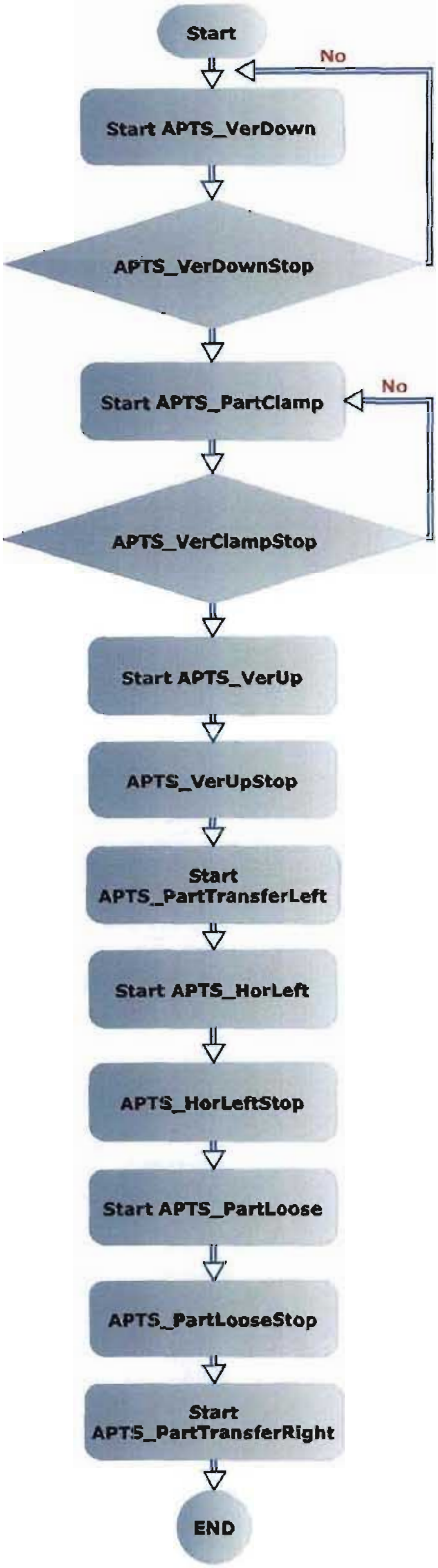
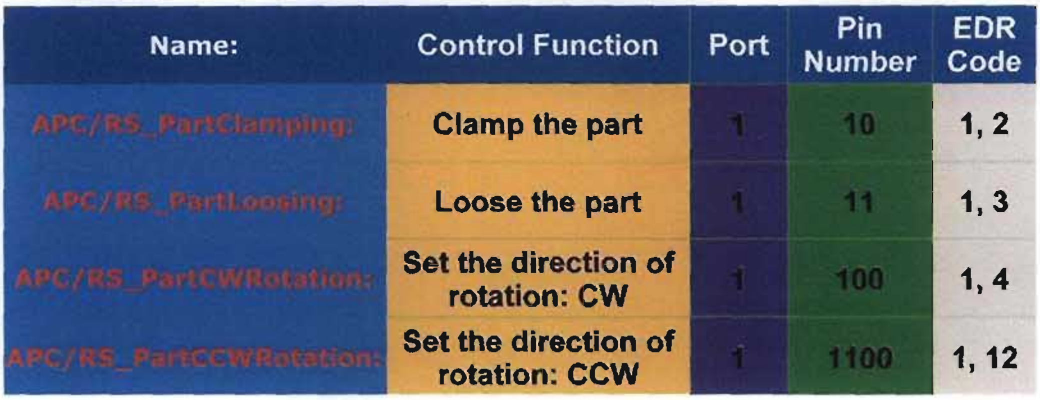


Figure 5-8 A Flow chart detailing the decision logic of the APTS control algorithm.

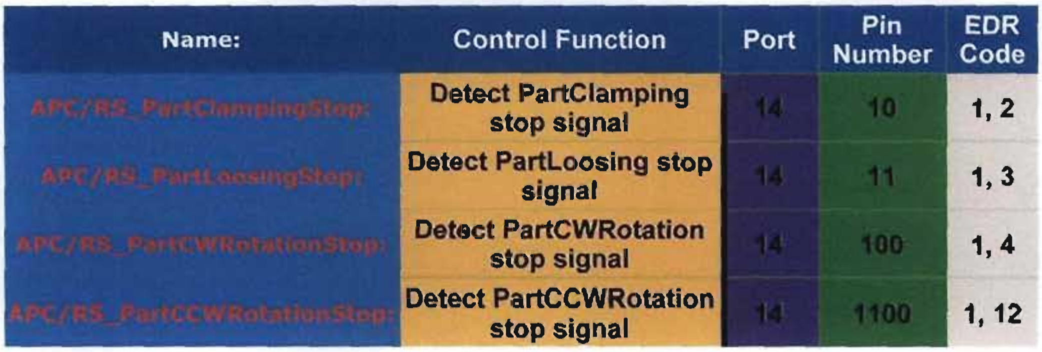
5.4.2. APC/RS Control Module

The control module for the Automatic Part Clamping/Rotating System (APC/RS) required the generation of 4 activation signals to control the function of the drive motor in order to implement control decisions (see Figure 5-9). The control decisions were made based on the feedback information collected from the APC/RS using two feedback signals (see Figure 5-9). The activation and feedback signals for the APC/RS are listed in the Table 5-4 and Table 5-5 respectively.



Name:	Control Function	Port	Pin Number	EDR Code
APC/RS_PartClamping:	Clamp the part	1	10	1, 2
APC/RS_PartLoosing:	Loose the part	1	11	1, 3
APC/RS_PartCWRotation:	Set the direction of rotation: CW	1	100	1, 4
APC/RS_PartCCWRotation:	Set the direction of rotation: CCW	1	1100	1, 12

Table 5-4 A list of the activation signals used in the APC/RS control module. The port and binary pin number of each signal is shown.



Name:	Control Function	Port	Pin Number	EDR Code
APC/RS_PartClampingStop:	Detect PartClamping stop signal	14	10	1, 2
APC/RS_PartLoosingStop:	Detect PartLoosing stop signal	14	11	1, 3
APC/RS_PartCWRotationStop:	Detect PartCWRotation stop signal	14	100	1, 4
APC/RS_PartCCWRotationStop:	Detect PartCCWRotation stop signal	14	1100	1, 12

Table 5-5 A list of the feedback signals used in the APC/RS control module. The port and binary pin number of each signal is shown.

A VB 6.0 project, *APC/RS control module.vbp*, was developed to implement the APC/RS control algorithm. A full scripting of the documented source code for *APC/RS control module.vbp* is given in Appendix C.

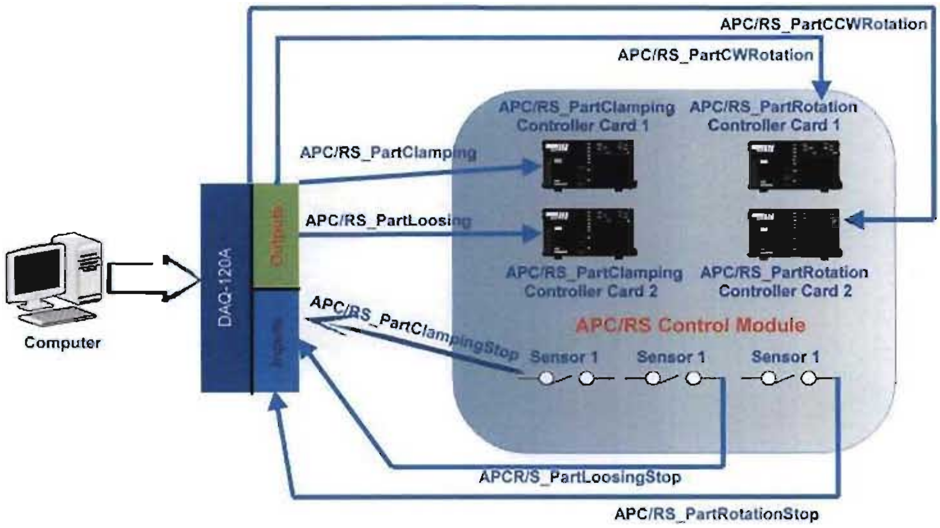


Figure 5-9 The block diagram for the APC/RS.

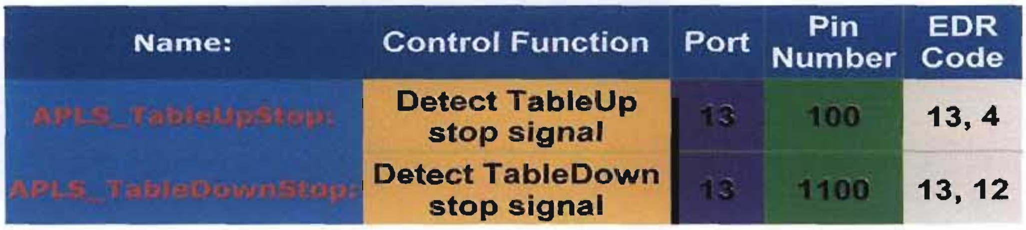
VB 6.0 includes a standard timing function, the *Timer* control, which was used to implement the pulse generation for the excitation signal used to drive the DC motor. The timer event included a counter variable which was used to stop the generation of timer event after the appropriate number of steps had been generated. The timer event was disabled by setting the Enabled property to false, e.g. Timer1.Enable = False. *APC/RS Control Module.vbp* utilized a timer control event to control the DC motor.

5.4.3. APLS Control Module

The control module for the Automatic Part Lifting System (APLS) required the generation of two activation signals to control the functioning of the drive motor in order to implement control decisions (see Figure 5-10.). The activation and feedback signals for the APC/RS are listed in the Table 5-6 and Table 5-7 respectively.

Name:	Control Function	Port	Pin Number	EDR Code
APLS_TableUp:	Raise the lifting table	1	10000	1, 16
APLS_TableDown:	Lower the lifting table	1	110000	1, 48

Table 5-6 A list of the activation signals used in the APLS control module. The port and binary pin number of each signal is shown.



Name:	Control Function	Port	Pin Number	EDR Code
APLS_TableUpStop:	Detect TableUp stop signal	13	100	13, 4
APLS_TableDownStop:	Detect TableDown stop signal	13	1100	13, 12

Table 5-7 A list of the feedback signals used in the APLS control module. The port and binary pin number of each signal is shown.

A VB 6.0 project, *APLS control module.vbp*, was developed to implement the APLS control algorithm. A full scripting of the documented source code for *APLS control module.vbp* is given in Appendix C.

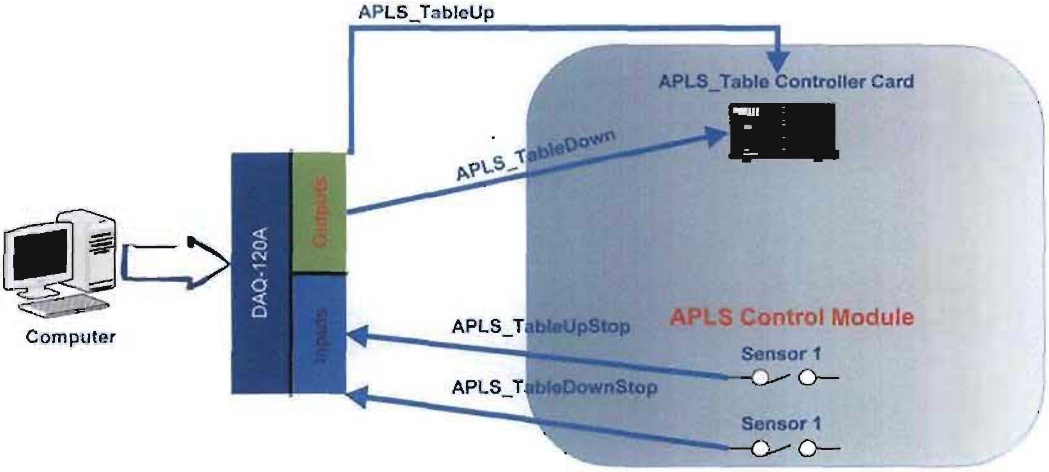
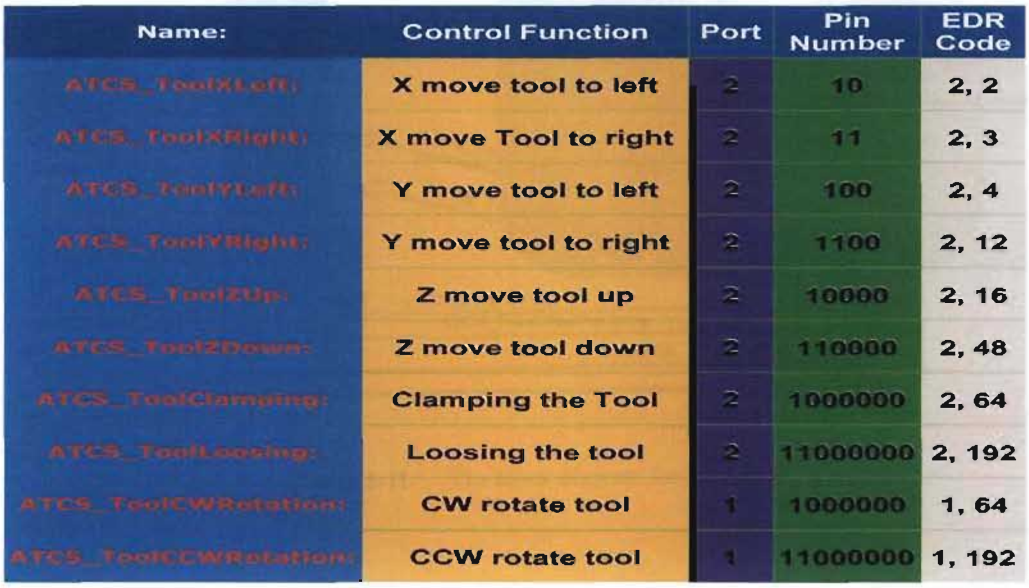


Figure 5-10 The block diagram for the APLS.

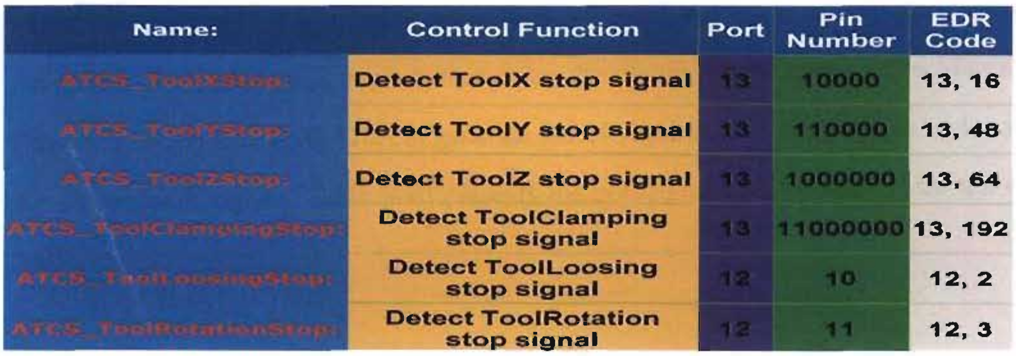
5.4.4. ATCS Control Module

The control module for the Automatic Tool Changing System (ATCS) required the generation of 2 activation signals to control the functioning of the drive motor in order to implement control decisions (see Figure 5-11.). The activation and feedback signals for the APC/RS are listed in the Table 5-8 and Table 5-9 respectively.



Name:	Control Function	Port	Pin Number	EDR Code
ATCS_ToolXLeft:	X move tool to left	2	10	2, 2
ATCS_ToolXRight:	X move Tool to right	2	11	2, 3
ATCS_ToolYLeft:	Y move tool to left	2	100	2, 4
ATCS_ToolYRight:	Y move tool to right	2	1100	2, 12
ATCS_ToolZUp:	Z move tool up	2	10000	2, 16
ATCS_ToolZDown:	Z move tool down	2	110000	2, 48
ATCS_ToolClamping:	Clamping the Tool	2	1000000	2, 64
ATCS_ToolLoosing:	Loosing the tool	2	11000000	2, 192
ATCS_ToolCWRotation:	CW rotate tool	1	1000000	1, 64
ATCS_ToolCCWRotation:	CCW rotate tool	1	11000000	1, 192

Table 5-8 A list of the activation signals used in the ATCS control module. The port and binary pin number of each signal is shown.



Name:	Control Function	Port	Pin Number	EDR Code
ATCS_ToolXStop:	Detect ToolX stop signal	13	10000	13, 16
ATCS_ToolYStop:	Detect ToolY stop signal	13	110000	13, 48
ATCS_ToolZStop:	Detect ToolZ stop signal	13	1000000	13, 64
ATCS_ToolClampingStop:	Detect ToolClamping stop signal	13	11000000	13, 192
ATCS_ToolLoosingStop:	Detect ToolLoosing stop signal	12	10	12, 2
ATCS_ToolRotationStop:	Detect ToolRotation stop signal	12	11	12, 3

Table 5-9 A list of the feedback signals used in the ATCS control module. The port and binary pin number of each signal is shown.

A VB 6.0 project, *ATCS control module.vbp*, was developed to implement the ATCS control algorithm. A full scripting of the documented source code for *ATCS control module.vbp* is given in Appendix C.

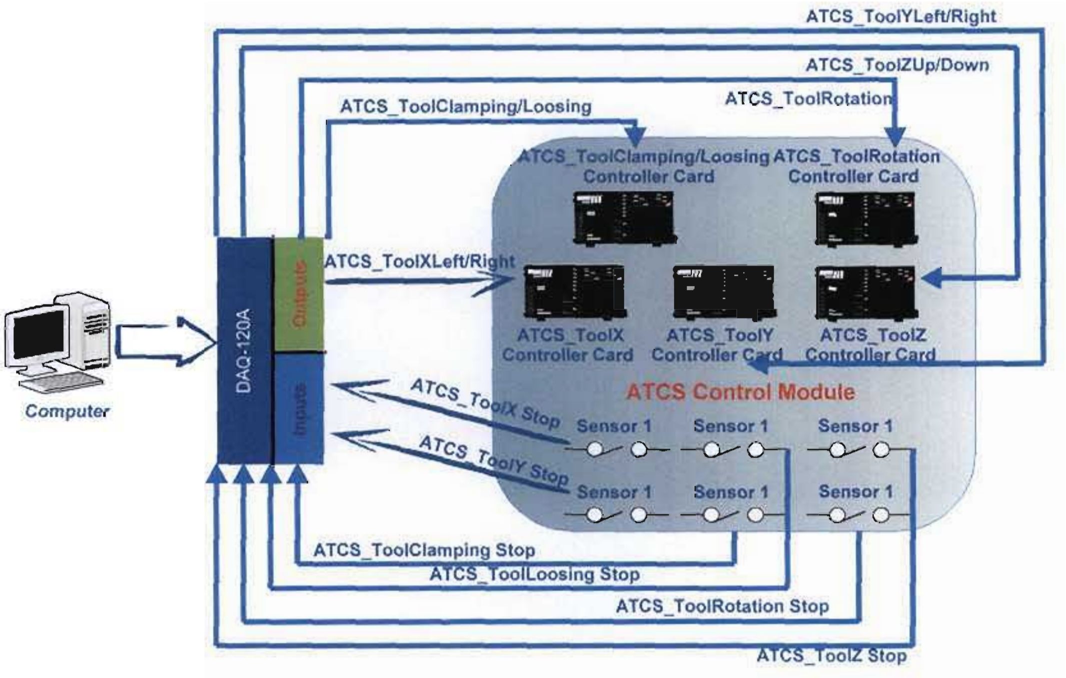
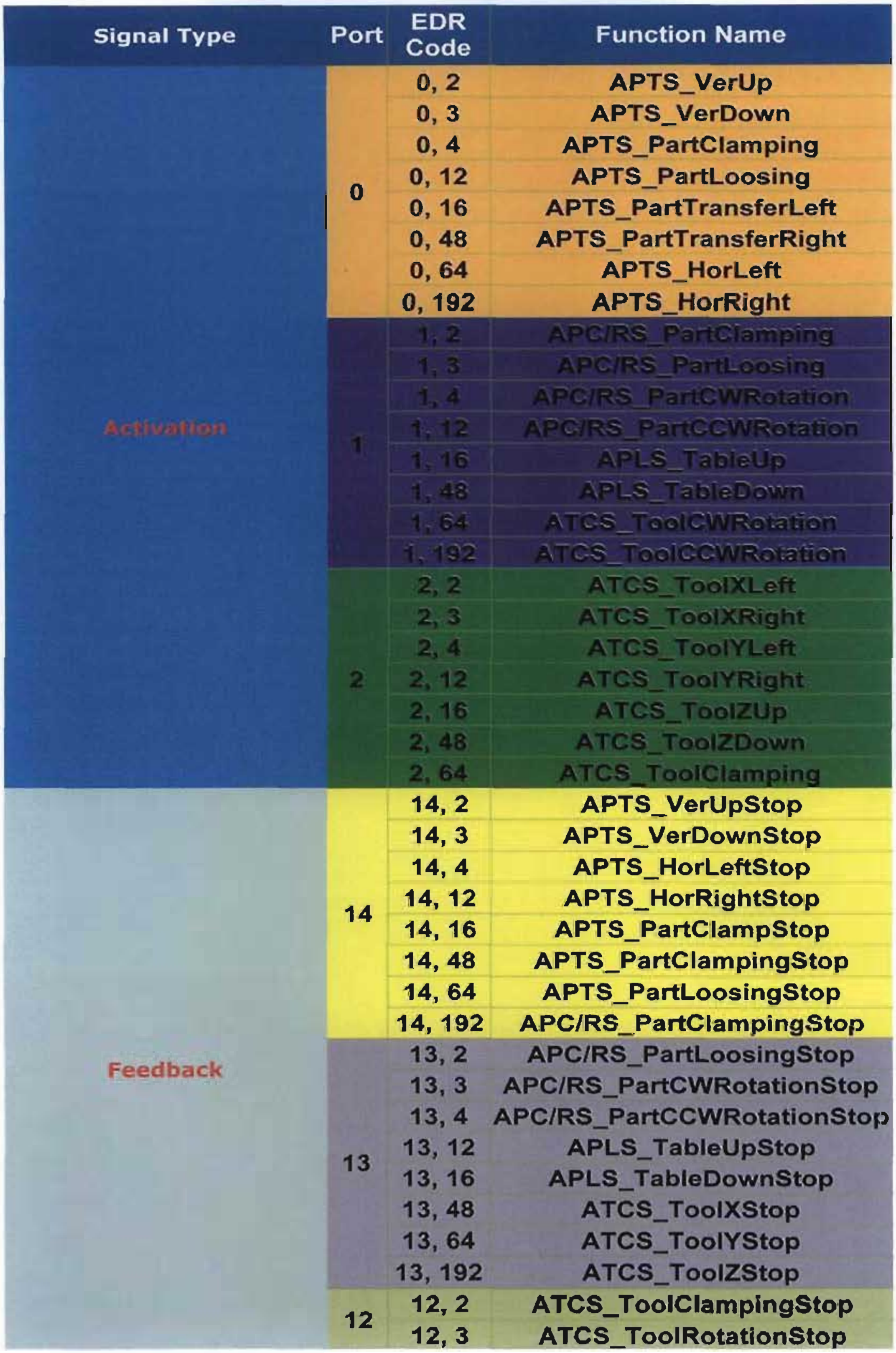


Figure 5-11 The block diagram for the ATCS.

5.5. RMM Control System

The final control system for the RMM was successfully developed as a PC-based system that implemented digital I/O interfacing techniques to control and coordinate the functioning of the RMM. The system was implemented using the digital portion of an Eagle MicroDAQ Data Acquisition Box USB-120A to generate the 24 required activation signals and monitor the 18 feedback signals. Table 5-10 presents the port/signal correspondence for the entire RMM control system.



Signal Type	Port	EDR Code	Function Name
Activation	0	0, 2	APTS_VerUp
		0, 3	APTS_VerDown
		0, 4	APTS_PartClamping
		0, 12	APTS_PartLoosing
		0, 16	APTS_PartTransferLeft
		0, 48	APTS_PartTransferRight
		0, 64	APTS_HorLeft
		0, 192	APTS_HorRight
	1	1, 2	APC/RS_PartClamping
		1, 3	APC/RS_PartLoosing
		1, 4	APC/RS_PartCWRotation
		1, 12	APC/RS_PartCCWRotation
		1, 16	APLS_TableUp
		1, 48	APLS_TableDown
		1, 64	ATCS_ToolCWRotation
		1, 192	ATCS_ToolCCWRotation
	2	2, 2	ATCS_ToolXLeft
		2, 3	ATCS_ToolXRight
		2, 4	ATCS_ToolYLeft
		2, 12	ATCS_ToolYRight
		2, 16	ATCS_ToolZUp
		2, 48	ATCS_ToolZDown
		2, 64	ATCS_ToolClamping
Feedback	14	14, 2	APTS_VerUpStop
		14, 3	APTS_VerDownStop
		14, 4	APTS_HorLeftStop
		14, 12	APTS_HorRightStop
		14, 16	APTS_PartClampStop
		14, 48	APTS_PartClampingStop
		14, 64	APTS_PartLoosingStop
		14, 192	APC/RS_PartClampingStop
	13	13, 2	APC/RS_PartLoosingStop
		13, 3	APC/RS_PartCWRotationStop
		13, 4	APC/RS_PartCCWRotationStop
		13, 12	APLS_TableUpStop
		13, 16	APLS_TableDownStop
		13, 48	ATCS_ToolXStop
		13, 64	ATCS_ToolYStop
		13, 192	ATCS_ToolZStop
	12	12, 2	ATCS_ToolClampingStop
		12, 3	ATCS_ToolRotationStop

Table 5-10 The port/signal correspondence of the final RMM control system.

In keeping with the integrated mechatronics design technique, the final control system of RMM was developed by integrating the previously established control modules for each motion systems. The RMM control system was implemented in the VB 6.0 project, *RMM Main Control Software.vbp* which utilized the software techniques developed in the control modules for the individual systems. RMM is an acronym for Reconfigurable Modular Machine. A full scripting of the documented source code of *RMM Main Control Software.vbp* is presented in Appendix C.

5.6. Summary

The complex nature of the motion interactions required for RMM to function correctly and efficiently, required the design of a motion control and function coordinating system. This chapter details the development of a control system for RMM that employed generic modular mechatronic control approach.

The primary task of RMM is to perform automated part machining functions in RMS environment. RMS is characterized by the large scale implementation of computer based technologies. RMM has been designed by using computer based modular mechatronic control technology. This technology facilitated the integration of RMM into RMS.

CHAPTER 6. PERFORMANCE TEST OF THE RMM

6.1. Motion Accuracy of the RMM

The efficient functioning of the RMM is dependent on the ability of the four motion systems to repeatedly position either the tool or the work piece at precise locations whilst remaining within the bounds of accurate tolerances. The positioning tolerance of the separate motion systems is determined from the overall measurement accuracy requirement of the RMM. The performance testing and analysis of the four separate motion systems are reviewed individually as below:

- the accuracy of the APTS to move the parts from the conveyor and to position parts on the APLS.
- the accuracy of the APC/RS to clamp the parts from the APLS and to rotate the parts as required angle.
- the accuracy of the APLS to move the parts to the required height.
- the accuracy of the ATCS to move the tool box from X-Y-Z axis and to position the tool at the right place on the parts.

The design specifications of the four motion systems are presented in the following table:

Specification		Tolerance
APTS	Function 1	X-axis: ± 1mm Y-axis: ± 1mm
APC/RS	Function 1	± 1°
	Function 2	± 1°
APLS	Function 1	Z-axis: ±1mm
ATCS	Function 1	± 1°

Table 6-1 Design specifications of four motion systems of RMM.

6.2. Performance of the APTS

The APTS was developed to transfer parts on the conveyor and to position parts on the APLS. The APTS utilized two lead-screw drive systems, one linear rack drive system both are powered by 3 DC motor to finish the function of picking up work piece. The system incorporated an electro-pneumatic system used to transfer the part between the conveyor and lifting table (see Figure 6-1).

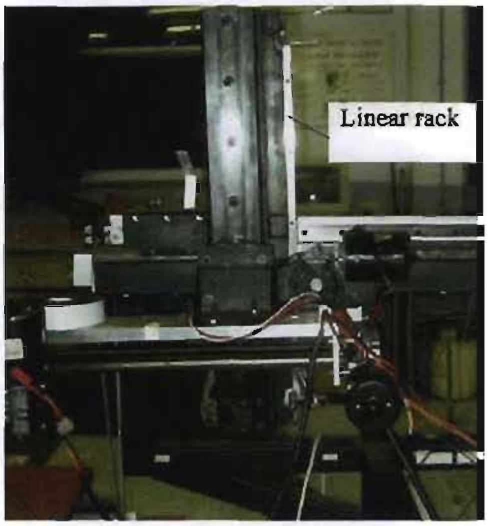


Figure 6-1 Modified design of APTS.

This section reviews the tests and analyses performed on the APTS in order to determine its performance characteristics with respect to clamping precision and operational time. Firstly the motion accuracy of the system is reviewed and then operation times are discussed.

6.2.1. Motion Accuracy of the APTS

The APTS was designed as a close loop system using three 12V DC motor to drive the entire system. During the manufacturing and installation of the system, two permanent stoppers were made on both sides of linear rack to avoid the damage of the slider.

Procedure: Motion accuracy test of the APTS

- Ensure that the APTS has been correctly connected and set the reference to the start position.
- Using software (Visual Basic 6.0) test the APTS function 1: moving the part from conveyor to lifting table.
- Repeat step 2 ten times.

6.2.2. APTS Motion Test

The APTS-function 1 motion accuracy analysis was focused on whether APTS can locate parts at the correct place on the lifting table. The test part dimension was 170 mm x 170 mm x 100 mm. The lifting table dimension was 800 mm x 600 mm. The top view of the test part and the lifting table is illustrated in Figure 6-2.

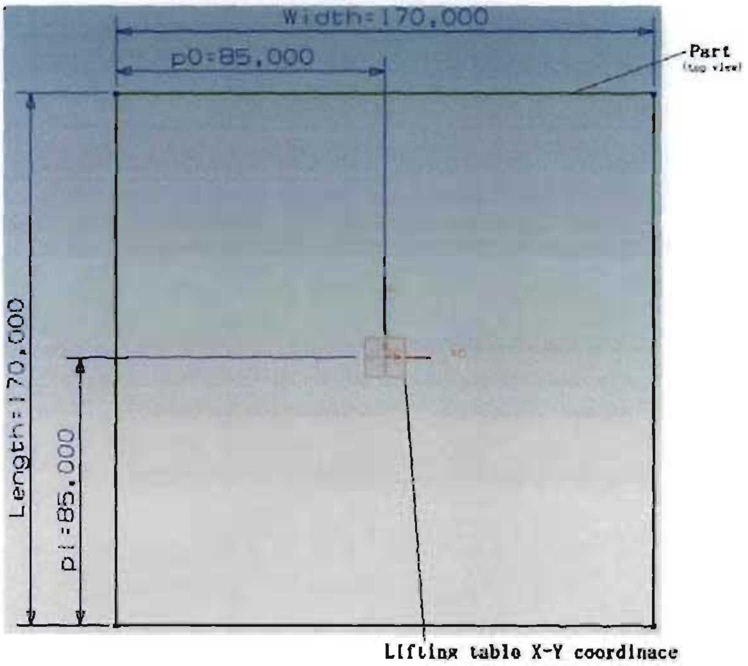
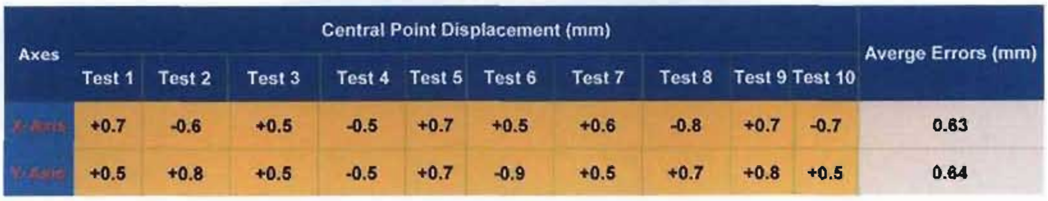


Figure 6-2 Top view of part and lifting table X-Y coordinace.



Axes	Central Point Displacement (mm)										Averge Errors (mm)
	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Test 8	Test 9	Test 10	
X-Axis	+0.7	-0.6	+0.5	-0.5	+0.7	+0.5	+0.6	-0.8	+0.7	-0.7	0.63
Y-Axis	+0.5	+0.8	+0.5	-0.5	+0.7	-0.9	+0.5	+0.7	+0.8	+0.5	0.64

Table 6-2 The central point displacement test of the function 1 of the APTS.

According to the design specifications, the average errors of the APTS-function 1 was within the acceptable range (tolerance see Table 6-1).

6.3. Performance of the APC/RS

The APC/RS was developed to perform the clamping and rotation required during the part machining procedure. The system utilized two lead-screw systems that were operated by four 12V DC motors. Figure 6-3 shows 2 motors on the right-hand side of APC/RS.

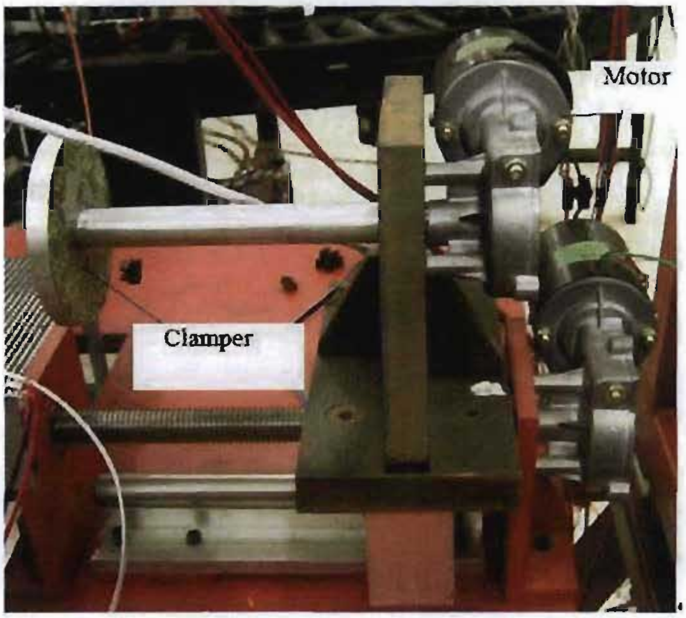


Figure 6-3 Two motors of the APC/RS

This section reviews the performance of the APC/RS with respect to the rotating accuracy and the recorded operational times for standard working motions of the system. Firstly the motion accuracy of the rotating function is discussed and thereafter the operational time measured for the standard motions are documented.

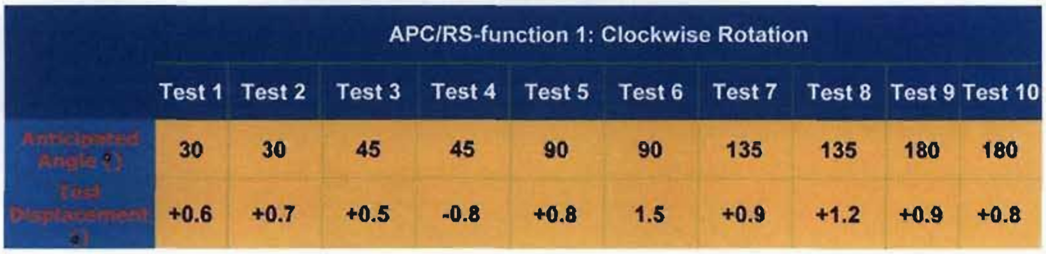
6.3.1. Motion Accuracy of the APC/RS

Procedure: Motion accuracy test of the APC/RS

- Ensure the APC/RS has been right connected. Position a part on the lifting table.
- Using the software (Visual Basic 6.0) test APC/RS function 1: clamping the part and clockwise rotating it.
- Using the software (Visual Basic 6.0) test APC/RS function 2: count-clockwise rotating the part and losing it.
- Repeat steps 2 and 3 for 10 times.

6.3.2. APC/RS Motion Test

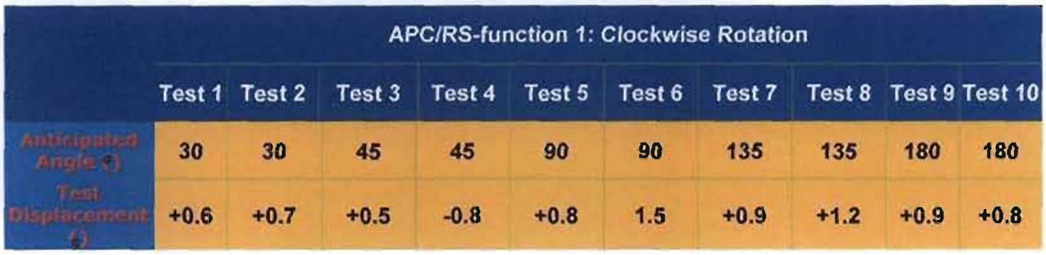
The APC/RS-function 1 motion accuracy analysis was focused on whether APC/RS can clockwise rotate the parts to the desired degree.



	APC/RS-function 1: Clockwise Rotation									
	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Test 8	Test 9	Test 10
Anticipated Angle (°)	30	30	45	45	90	90	135	135	180	180
Test Displacement (°)	+0.6	+0.7	+0.5	-0.8	+0.8	1.5	+0.9	+1.2	+0.9	+0.8

Table 6-3 Angle displacement of APC/RS-function 1.

The APC/RS-function 2 motion accuracy analysis was focused on whether APC/RS can counter-clockwise rotate the parts to the desired degree.



	APC/RS-function 1: Clockwise Rotation									
	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Test 8	Test 9	Test 10
Anticipated Angle (°)	30	30	45	45	90	90	135	135	180	180
Test Displacement (°)	+0.6	+0.7	+0.5	-0.8	+0.8	1.5	+0.9	+1.2	+0.9	+0.8

Table 6-4 Angle displacement of APC/RS-function 2.

The average errors of the degree displacement of APC/RS two functions were 0.87 and 0.85. They were both within the range of design specifications (tolerance see Table 6-1).

6.4. Performance of the APLS

The APLS was developed to provide a working platform required during the part machining procedure. The system utilized one lead-screw that was operated by one 12V DC motor (see Figure 6-4).

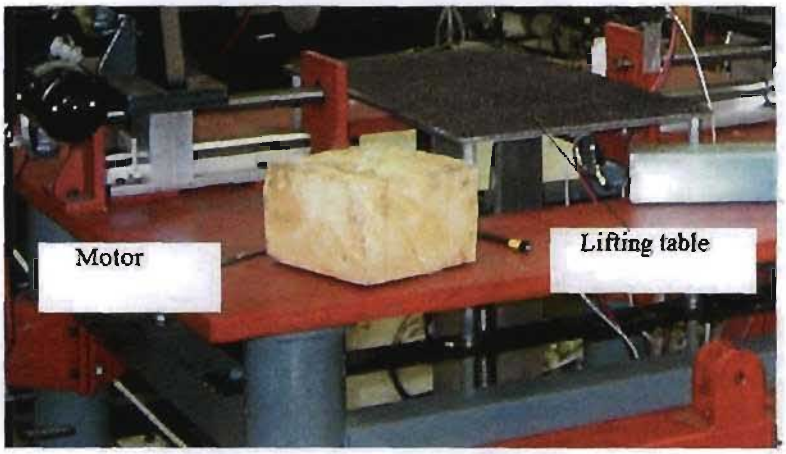


Figure 6-4 APLS

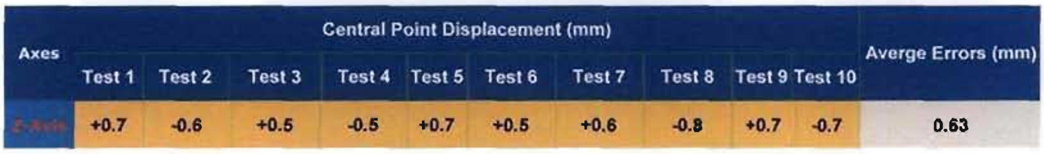
6.4.1. Motion Accuracy of the APLS

Procedure: Motion accuracy test of the APLS

- Ensure the APLS has been correctly connected. Position a part on the lifting table.
- Using the software (Visual Basic 6.0) test the APLS function 1: lifting the part to the required height.
- Repeat step 2 ten times.

6.4.2. APLS Motion Test

The APLS-function 1 motion accuracy analysis was focused on whether APLS can lift parts to the right height. The test part dimension is 170 mm x 170 mm x 100 mm. The lifting table dimension is 800 mm x 600 mm.



Axes	Central Point Displacement (mm)										Averge Errors (mm)
	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Test 8	Test 9	Test 10	
Z-Axis	+0.7	-0.6	+0.5	-0.5	+0.7	+0.5	+0.6	-0.8	+0.7	-0.7	0.63

Table 6-5 The central point displacement test of the function 2 of the APLS..

The Z-axis displacement of APLS function 1 was within the range of design specifications (tolerance see Table 6-1).

6.5. Performance of the ATCS

The ATCS was developed to provide automated tool changing. This was required during part machining procedure. The system utilized four lead-screws that were operated by five 12V DC motors to perform the 3-axis movement and tool clamping motions (see Figure6-5).

Figure 6-5 ATCS.

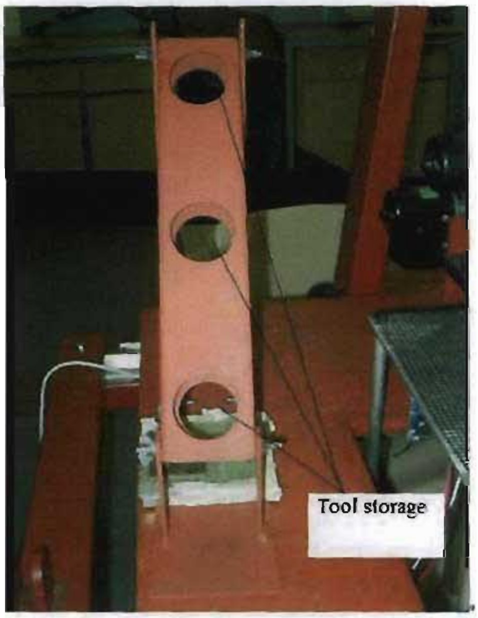


Figure 6-6 Tool Storage.

6.5.1. Motion Accuracy of the ATCS

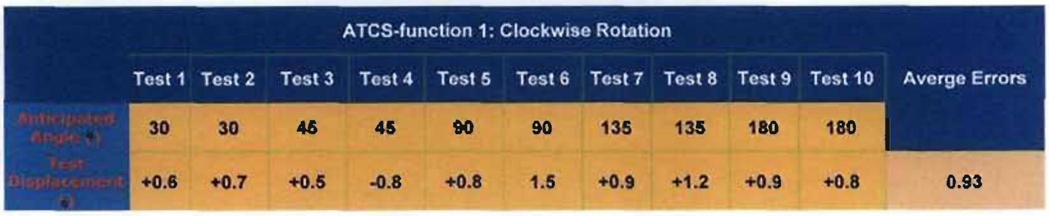
__Procedure: Motion accuracy test of the ATCS__

- Ensure the ATCS has been correctly connected. Position a tool on the tool storage.

- Using the software (Visual Basic 6.0) test the ATCS function 1: rotating the tool to the required degree during the machining procedure.
- Repeat step 2 ten times.

6.5.2. ATCS Motion Test

The ATCS-function 1 motion accuracy analysis was focused on whether ATCS can rotate the tool to the desired degree.



ATCS-function 1: Clockwise Rotation											
	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Test 8	Test 9	Test 10	Averge Errors
Anticipated Angle (°)	30	30	45	45	90	90	135	135	180	180	
Test Displacement (°)	+0.6	+0.7	+0.5	-0.8	+0.8	1.5	+0.9	+1.2	+0.9	+0.8	0.93

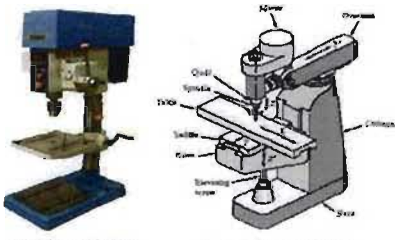
Table 6-6 The degree displacement test of ATCS-function 1.

The average error of the degree displacement of ATCS function 1 was 0.93. It was within the range of design specifications (tolerance see Table 6-1).

6.6. Cycle Time Comparison

Machine cycle time is very important factor in determining machining system throughput. A shorter machine cycle time equates to a higher machine throughput. For a given throughput target, shorter machine cycle times usually translate to fewer machines, which means less capital investment. Two cycle time comparisons have been made. A cycle time comparison is made among the RMM, traditional single-spindle drilling machine and traditional vertical milling machine, under the same tool path and drilling/milling operation of on a sample block. A summary of the cycle time comparison is given in the following figures.

The traditional single-spindle drilling machine and vertical milling machine are illustrated in Figure 6-7 which are used in the MR^{2}G lab.



Single-spindle drilling machine

Vertical milling machine

Figure 6-7 Traditional drilling and milling machined.

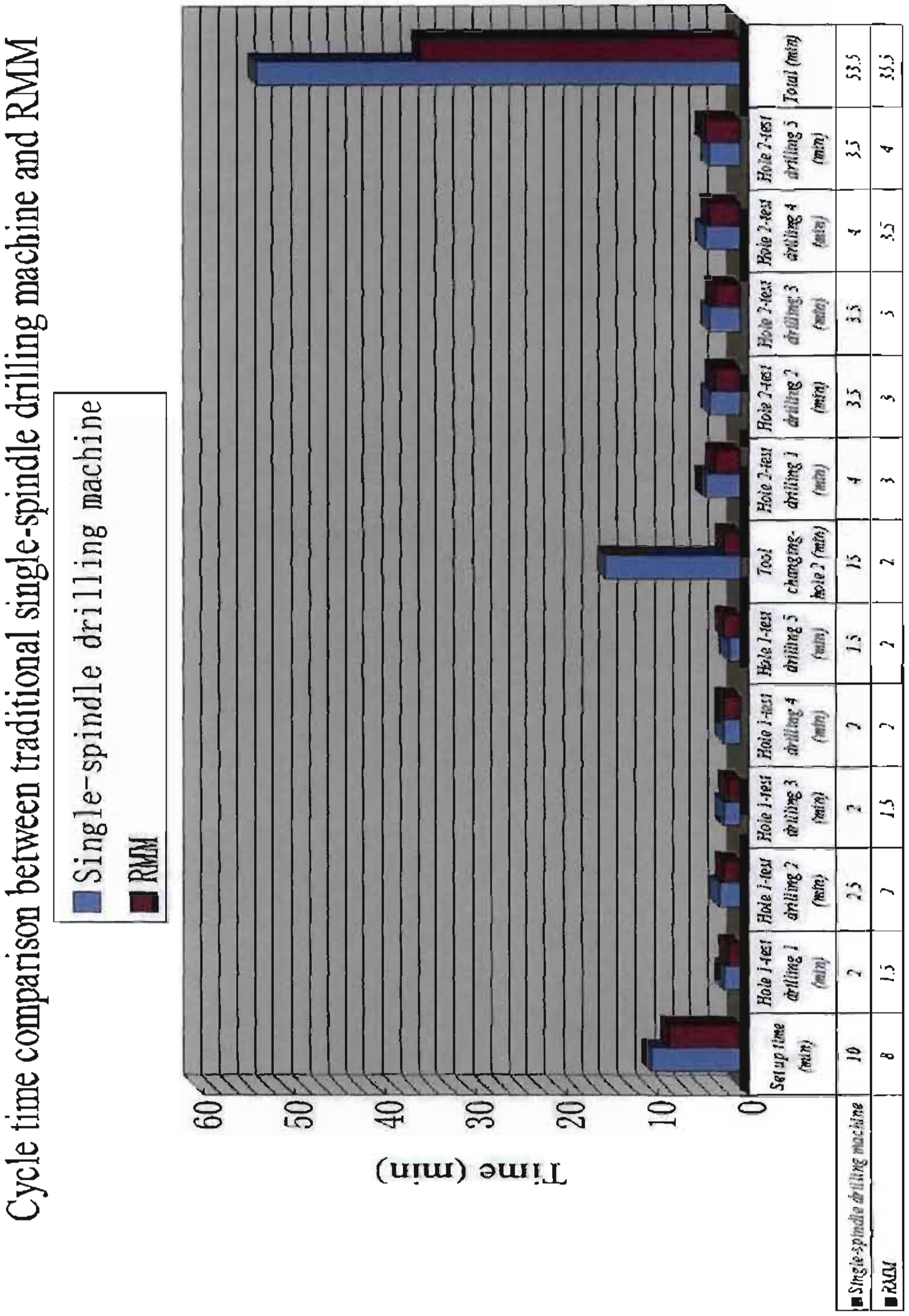


Figure 6-8 Cycle time comparison between traditional single-spindle drilling machine and RMM.

A 33.6% of cycle time reduction was obtained by using RMM compared with traditional single-spindle drilling machine.

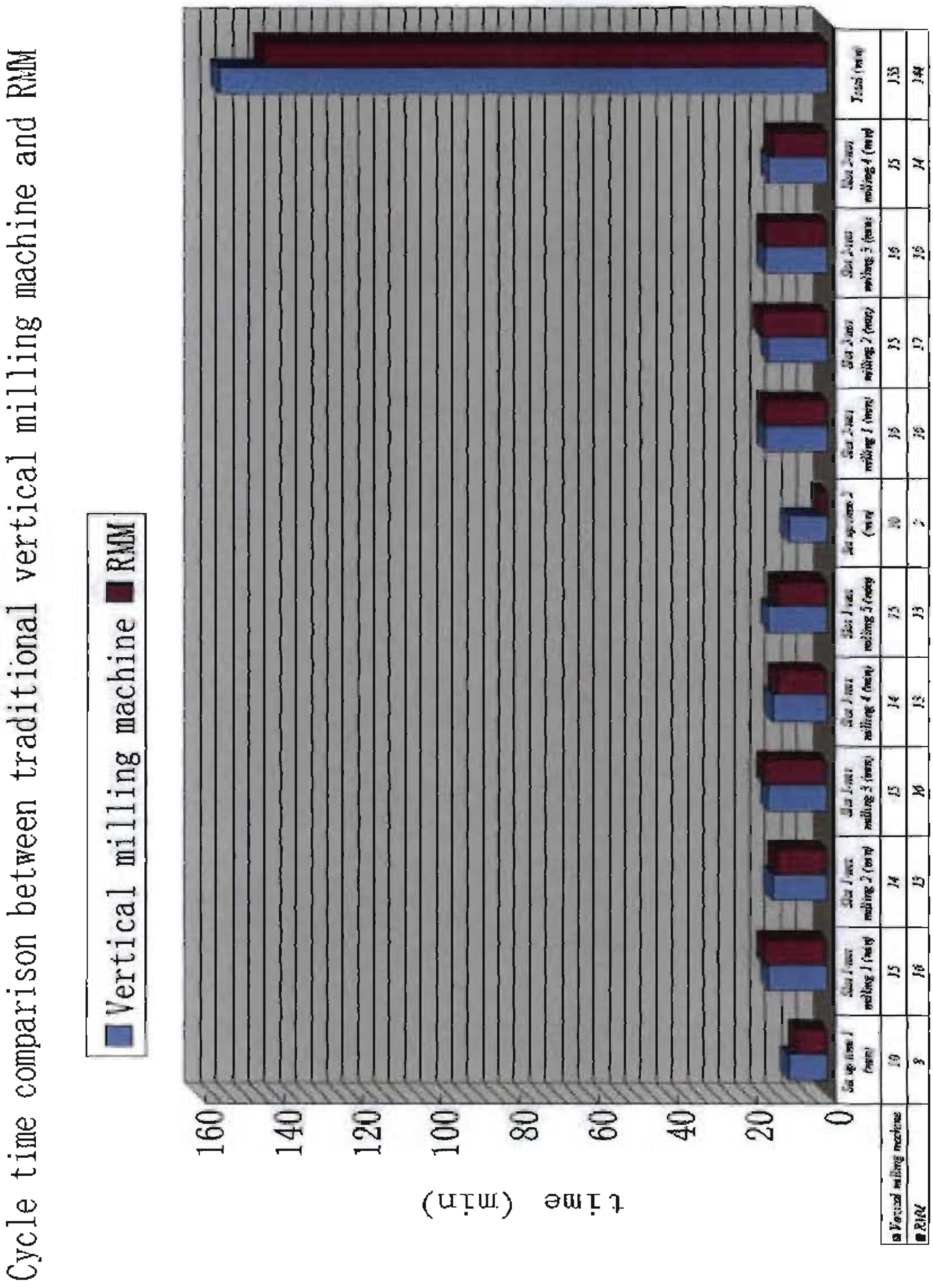


Figure 6-9 Cycle time comparison between traditional vertical milling machine and RMM.

A 7% of cycle time reduction was obtained by using RMM compared with traditional vertical milling machine.

6.7. Drilling/milling Accuracy Comparison

Drilling and milling are two machining functions performed by RMM. The machining accuracy comparisons between traditional single-spindle drilling machine/traditional vertical milling machine and RMM were illustrated in the Figure 6-11, 6-12, 6-13 and 6-14.

The dimension of the experimental block was 170mm x 170mm x 100mm. The material was wood.

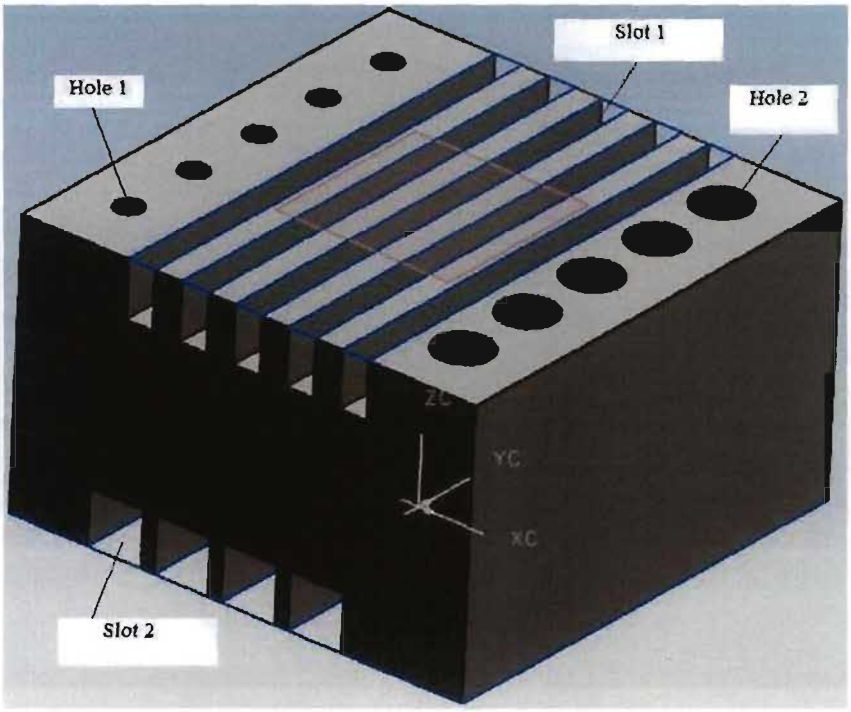


Figure 6-10 Experimental block 170mm x 170mm x 100mm.

The slot 1, (width 10mm x depth 20mm), milling test results are presented in Figure 6-11.

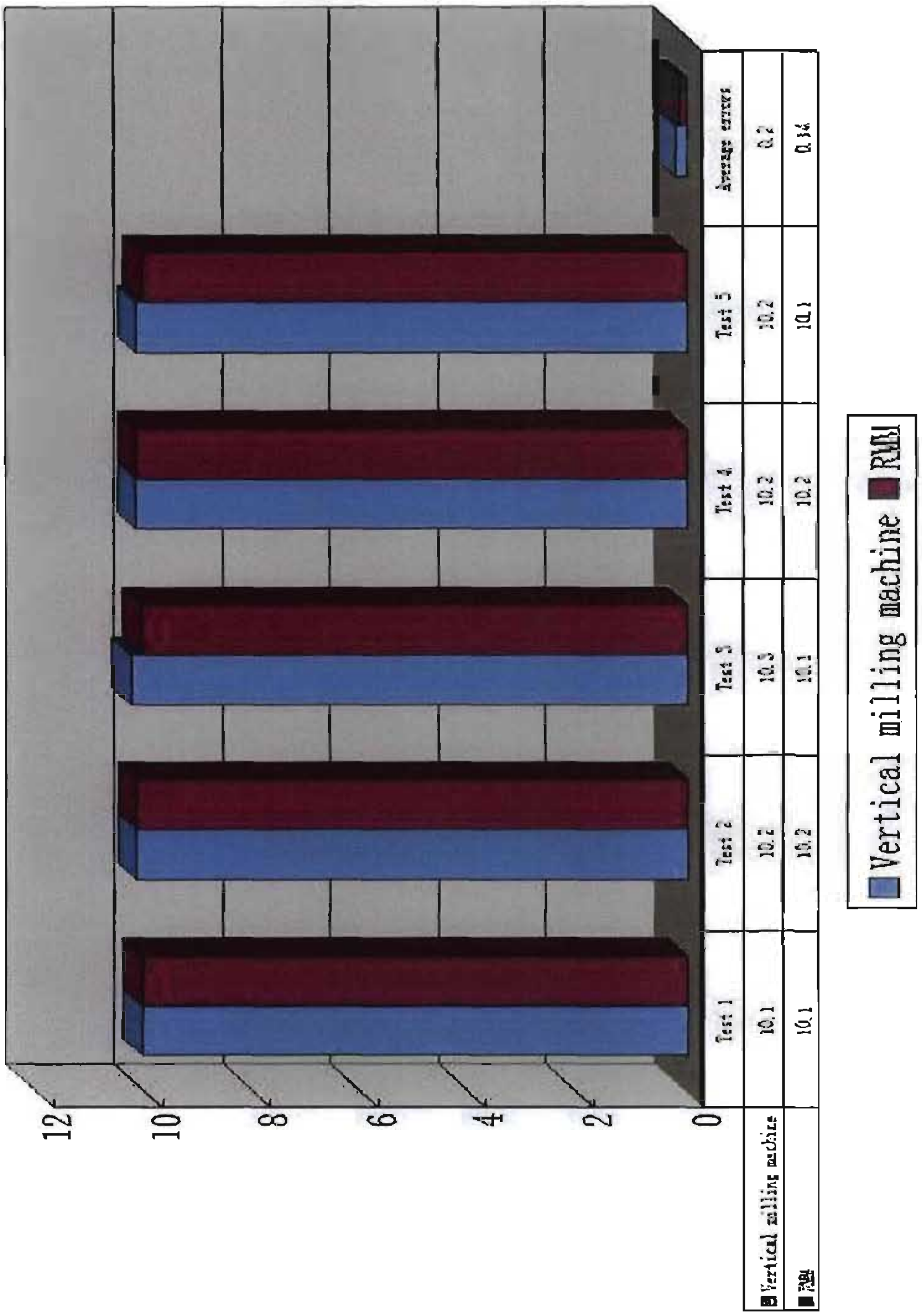


Figure 6-11 Milling accuracy comparison 1.

A 30% reduction of milling errors, (width 10mm x depth 20mm), was obtained by using RMM compared with traditional vertical milling machine.

The slot 2, (width 20mm x depth 20mm), milling test results are presented in Figure 6-12.

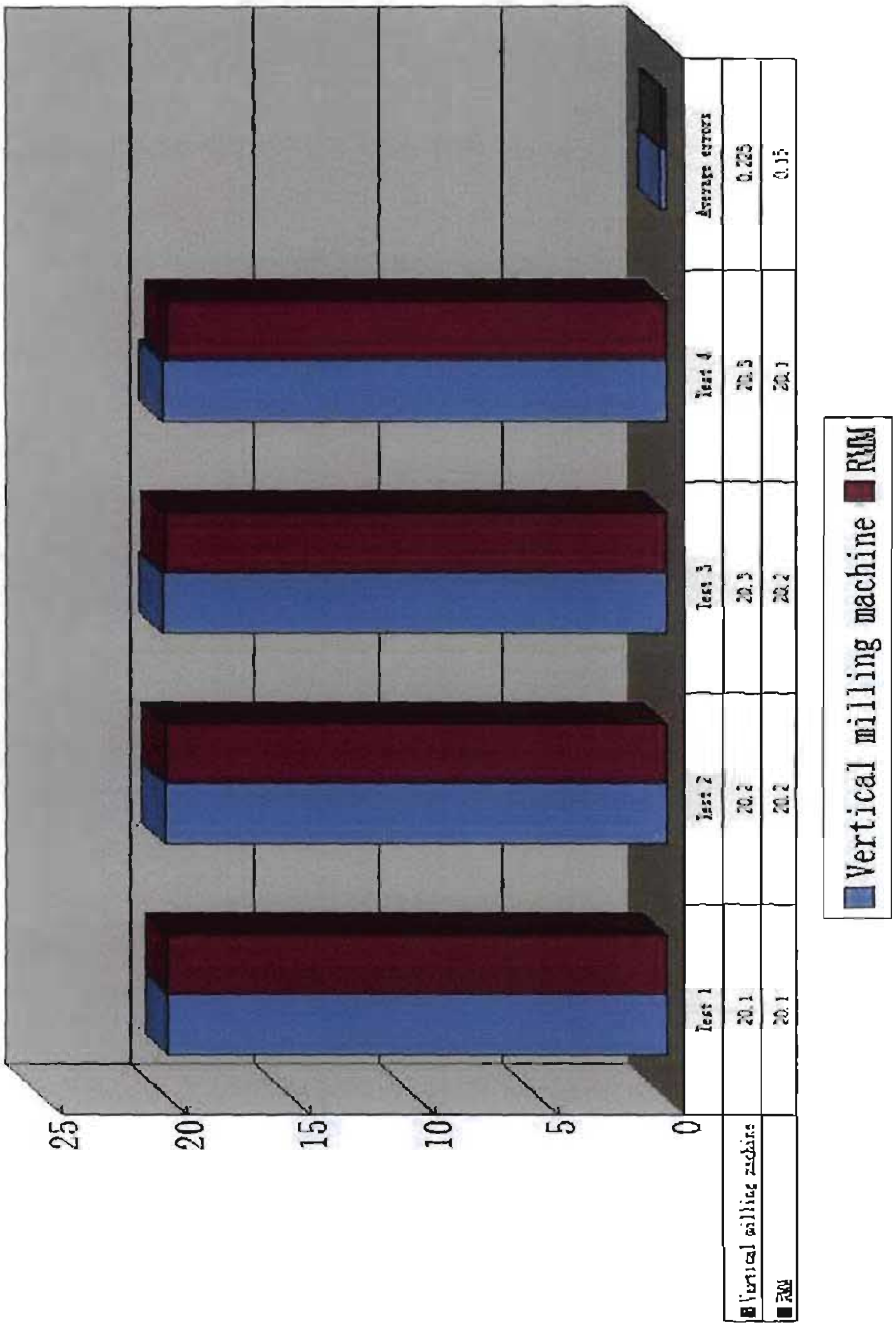
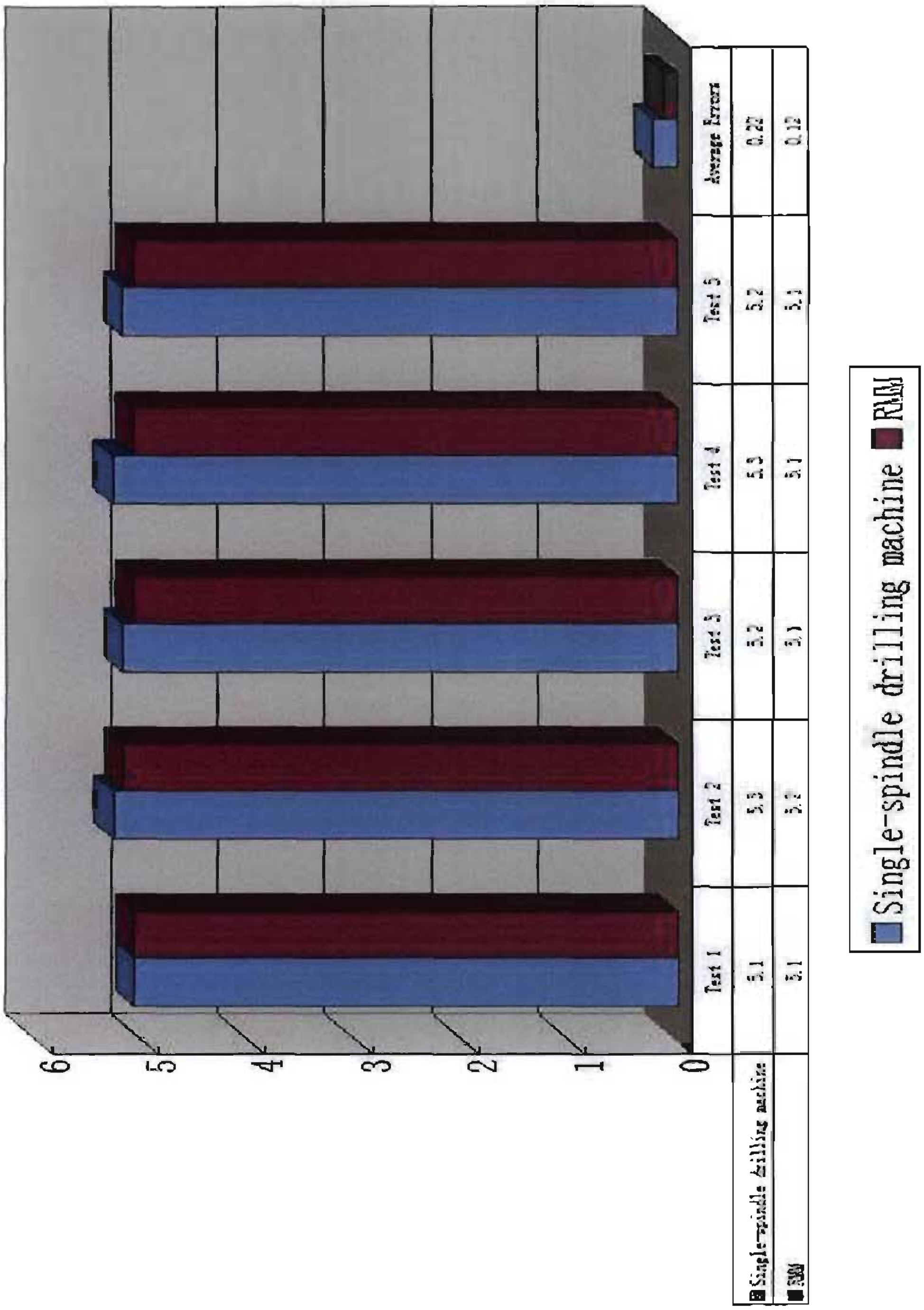


Figure 6-12 Milling accuracy comparison 2.

A 33.3% reduction of milling errors, (width 20mm x depth 20mm), was obtained by using RMM compared with traditional vertical milling machine.

The through hole 1, (Φ 5mm), drilling test results are presented in Figure 6-13.



	Test 1	Test 2	Test 3	Test 4	Test 5	Average Errors
▨ Single-spindle drilling machine	5.1	5.3	5.2	5.3	5.2	0.22
■ RMM	5.1	5.2	5.1	5.1	5.1	0.12

Figure 6-13. Drilling accuracy comparison 1.

A 45.5% reduction of drilling errors, (Φ 5mm), was obtained by using RMM compared with traditional single-spindle drilling machine.

The through hole 2, (Φ 20mm), drilling test results are presented in Figure 6-14.

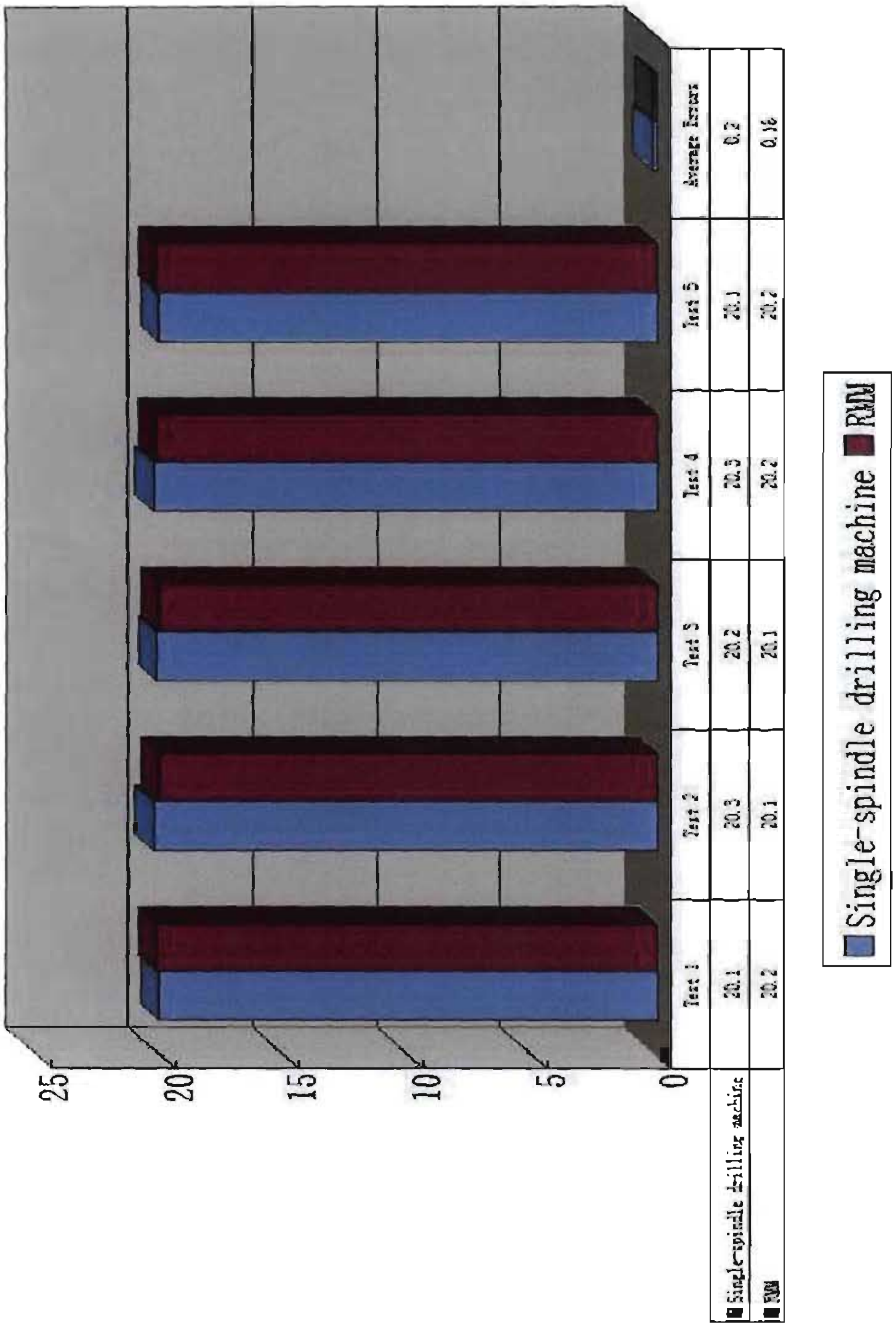


Figure 6-14. Drilling accuracy comparison 2.

A 20% reduction of drilling errors, (Φ 20mm), was obtained by using RMM compared with traditional single-spindle drilling machine.

6.8. Summary

The performance testing and analysis of the four separate motion systems was detailed in this chapter. The drilling and milling functions performed by RMM were also tested. The results compared with traditional single-spindle drilling machine and traditional vertical milling machine were also presented.

The motion accuracy tests performed on the RMM were focused on the four sub-systems motion accuracy. The results showed that four sub-systems can perform desired jobs individually and meet the requirements of the design specifications. From the motion accuracy point of view, RMM was designed successful.

The machining tests performed on the RMM were focused on two aspects: cycle time comparison and drilling/milling accuracy comparison. The comparison results of the cycle time showed that RMM can achieve a better performance than traditional drilling and milling machines. The drilling/milling accuracy tests were carried out by drill different diameter of holes and mill different size of slot. The results showed for these two fundamental machining functions, RMM had better performance than traditional drilling and milling machines.

CHAPTER 7. Modular Diagnostics of PC-Based RMM Mechatronic Control System

7.1. Introduction

With the development of modular and complicated machine tool systems, the failure probability of such complex machining system is higher than ever before. In spite of this, only little efforts have been invested to enhance the complex systems' diagnostics. Although in some newly designed machine modules, there are some kinds of built-in self-diagnostics features exist, these diagnostics modules can not be well used for system-level diagnostics. However, for some advanced diagnostic approaches, the system function and its failure models are required to be modelled before the model-based diagnostic techniques can be utilized. These problems can be detailed as the following three aspects in the diagnostics of modular systems:

- The accumulation of a failure from the component level and module level through the whole system;
- The optimization of the sensors' placement;
- The development of model-based diagnostic approaches for a modular structured system.

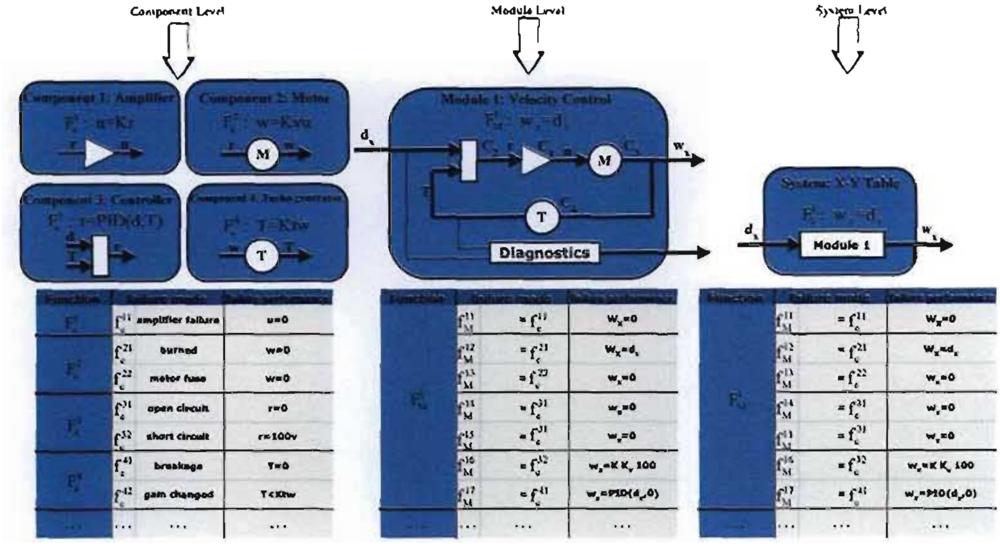


Figure 7-1 Failure diagnostics illustration from components to a system.

As shown in Figure 7-1, there are some issues can be concluded corresponding to the failure diagnostics:

- All the components have some kinds of possible failures which can lead to the system failure. However, for the component level, module level and system level, the failure behaviours of each are different. For instance, at the component level, the zero output volts is the behaviour of an amplifier fuse failure. However, at the module level, the zero output speed is the behaviour for the same amplifier fuse failure;
- Module level failure diagnostics are normally design before the system configuration is determined. But in some cases, the module can be reused. For example, a module designer who designs the failure diagnostics of an x-axis slide module might not know whether it would be reused for an x-y slide table;
- There are some undetectable failures existed at the lower level which can only be detected at the higher level. For instance, due the axis controller's feedback effect, some of the feedback gain error can only be detected at the system level by using the specific measurement device instead of realizing it at the axis control module level;
- In a modular structure, sub-system sometimes can be treated as a module depends on the size of the whole system. For instance, in this research the sub-systems APTS, APLS,

ATCS and APC/RS can be treated as different module separately.

All of above mentioned problems need a method to deal with the failure diagnostics of the modular dynamic structures. Due to less efforts have been invested on these issues, in this research, we will develop a method to determine a modular system's failure diagnostics based on the failure accumulation. The primary objective of this method would be: 1) failures' modelling and diagnostic approaches for failures in a modular mechatronic controlled system; 2) system diagnostics' design and modification by using the existed failure detection and isolation approaches; 3) the diagnosability evaluation of a system's diagnostics.

7.2. Setup A Diagnostic Model

There are some methods exist in dealing with functional modelling, but none of them deals with the diagnosability issue. Therefore, the new method we proposed will mainly cover the following two aspects which are not well investigated in the previous research work: 1) failure modes and performances of a component; 2) reconfigurability of the model structure for a multi-level represented system. If these two aspects can be successfully applied in a model, then we would have a systematic approach to determine whether a diagnostics could or could not be completed.

7.2.1. Failure Modes and Performances

Diagnostic approaches have used two basic modelling methods to identify the performances of the diagnosed system: *proper performance* and *failure performance*. However, by only using the proper performance model, sometimes it is still difficult to identify a root failure reason of a system. So in this research, we introduced a functional modelling method which is composed of failure modes and performances. In mechatronic systems, the proper performance model is very complex because of the system dynamics. So this method is significantly beneficial for the diagnostics of these systems.

7.2.2. Reconfigurability of the Model Structure

Reconfigurability is the most important aspect of the modelling approach if a system needs to be modelled in different resolutions or accurate levels. A system can be modelled as the sum of its modules if it can be divided into modules instead of modelling it as a whole. Similarly, each module can also be divided into components. In this research, a system where modules and components can be modelled by the same approach and the same diagnostics approach as well will be called reconfigurable. For instance, Figure 7-2 shows the model structure of an experimental RMM that has been built. As shown in the figure, the RMM can be modelled as one machining system if only the system model if of interest in RMS research. However it can also be modelled by a set of modules like APTS, ATCS, APLS and APR/CS. Because of the reconfigurability of modules, each module can be further divided into a set of components if a higher resolution is needed. The system analysis will be much easier if the components, the modules and the systems these three different levels can be modelled in the same approach. For a reconfigurable system where the component or the modules needed to be reused, this approach can benefit a lot.

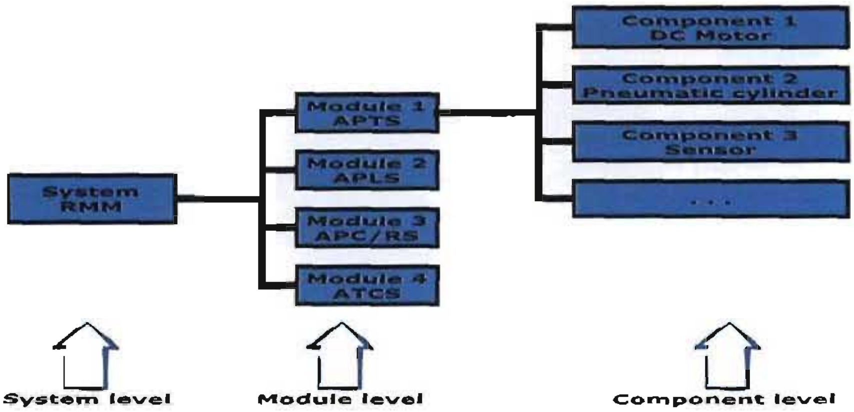


Figure 7-2 Reconfigurability of multi-level model.

7.3. Structure of Component Diagnostic Model

In order to develop a diagnostic modelling framework which can be used to evaluate the diagnosability of a modular system, the above mentioned two aspects of system model have to be considered in each level of the model. The first step is the development of component functional models which can be used to facilitate diagnostics. A structure of the functional model of a component was illustrated in Figure 7-3.

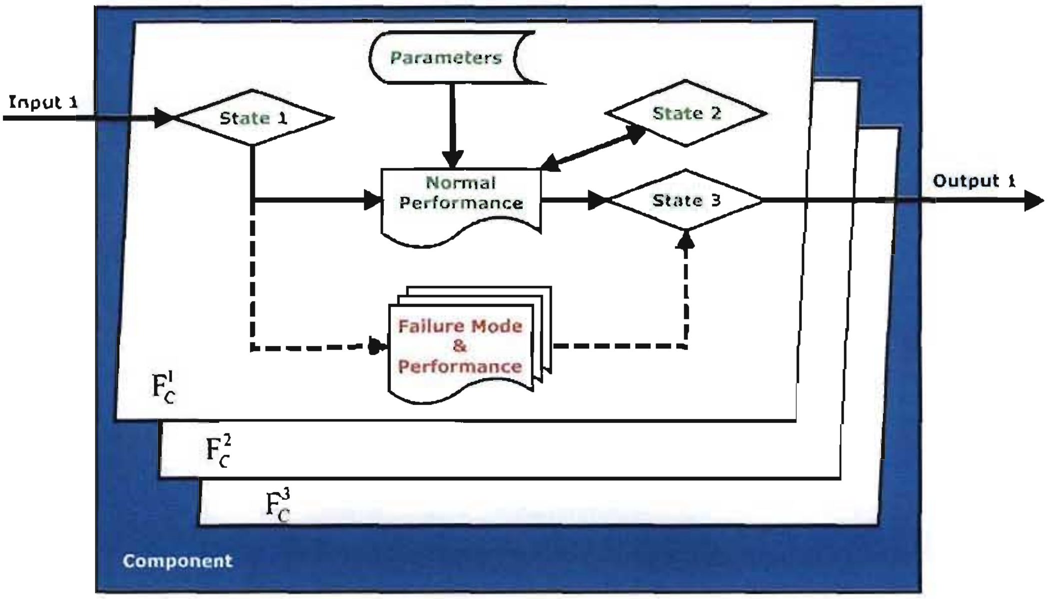
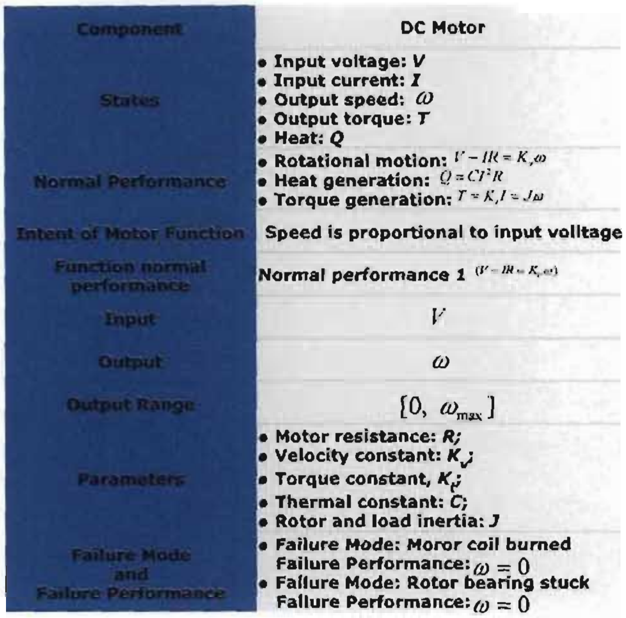


Figure 7-3 Functional model of a component.

7.3.1 Elements of A Model

A component consists of its own states and performances. The properties of a component such as its dynamic or static state variables that represent the current status of the component are called the component's states. In a component, one state's action or reaction corresponding to other states is called the component's performance. The function is the specific performance of a component which is designed to achieve the component's functional goal. An example of a function and other model elements of a DC motor was shown in Table 7-1.



Component	DC Motor
States	• Input voltage: V • Input current: I • Output speed: ω • Output torque: T • Heat: Q
Normal Performance	• Rotational motion: $V - IR = K_v\omega$ • Heat generation: $Q = CI^2R$ • Torque generation: $T = K_tI = J\omega$
Intent of Motor Function	Speed is proportional to input volltage
Function normal performance	Normal performance 1 $(V - IR = K_v\omega)$
Input	V
Output	ω
Output Range	$[0, \omega_{max}]$
Parameters	• Motor resistance: R; • Velocity constant: K_v; • Torque constant, K_t; • Thermal constant: C; • Rotor and load inertia: J
Failure Mode and Failure Performance	• Failure Mode: Moror coil burned Failure Performance: $\omega = 0$ • Failure Mode: Rotor bearing stuck Failure Performance: $\omega = 0$

Table 7-1 An example of a functional model.

7.3.2. Reconfigurabilty of the Model Structure

The higher level system can be modelled by integrating the modules and components which was shown in Figure 7-4.

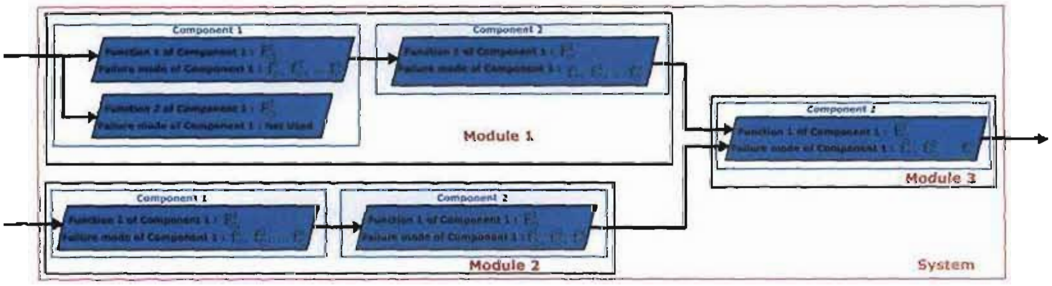


Figure 7-4 System model.

7.4. Module Integration and Diagnostics

In this section, we will use the component functional models to construct a model of a module. First, the principles which combine the components inside the module were developed; second, the module-level diagnostics was detailed; third, we developed a modular system diagnosability evaluation method based on the module level functional model.

7.4.1. Modelling a Module

As discussed in the previous section, a set of components are included in each module and each system contains a set of modules. The selection of components is based on the module functional goals which are needed to be achieved. The module's performance and failure mode can be derived from the components' properties. In this section, the same notations which were used for the component level model will be used again to derive the module level model.

Given the models of a module consists of two components, so they can be described by using the basic configurations. If component C_1 consists of only one function F_C^1 and failure modes f_C^1 and f_C^2, and component C_2 consists of only one function F_C^2 and failure mode f_C^3, then the component models can be derived as follows:

$$\begin{cases} \text{Function}(C_1)=\{\, F_{C_1}^1 \,\}; \\ y_{Output}^1(t) = C_1 g F_{C_1}^1 (\mathbf{x}(t),\ \mathbf{u}_1(t)); \\ \text{failure mode}(C_1 g F_{C_1}^1)=\{f_{C_1}^1,\ f_{C_1}^2\}; \end{cases} \tag{7-1}$$

$$\begin{cases} \text{Function}(C_2)=\{\, F_{C_2}^1 \,\}; \\ y_{Output}^2(t) = C_2 g F_{C_2}^1 (\mathbf{x}(t),\ \mathbf{u}_2(t)); \\ \text{failure mode}(C_2 g F_{C_2}^1)=\{\, f_{C_2}^3 \,\} \end{cases} \tag{7-2}$$

For instance, if we suppose that an amplifier is component 1 and a motor is component 2. Amplifying the voltage with a gain K_a is the only one function that the amplifier has. Equation $v=K_a u$ is the normal performance of this function. The amplifier also has two failure modes of the function: $f_{C_1}^1$, zero volts of output, and $f_{C_1}^2$, 150 volts of output. Rotation is the only one function that the component 2 has. Equation $\omega=K_v v$ is the normal performance of the function. The motor also has a failure mode of the function: $f_{C_1}^3$, motor fail to rotate.

- **Serial Connection**

  When a function F_M^1 of a module is built by two component functions F_C^1 and F_C^2 which are serial connected as shown in Figure 7-5, the functional model of module 1 can be expressed by using the component functions. The expressions are as follows:

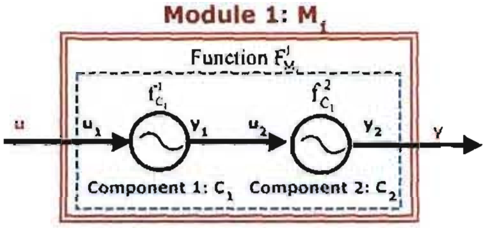


Figure 7-5 Serial connection of component functions.

- **Parallel Connection**

  When a module consist of two parallel functions as shown in Figure 7-6, the module functions of M_1 are as follows:

$$\text{Function}(M_1)=\{ F_{M_1}^1, F_{M_1}^2 \} \qquad (7\text{-}3)$$

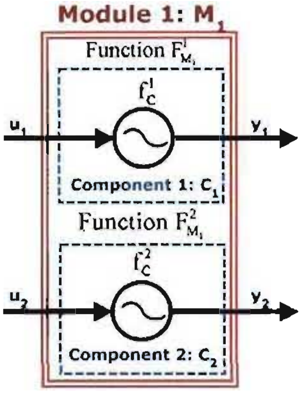


Figure 7-6 Parallel composition of component functions.

- **Feedback Connection**

As shown in Figure 7-7, based on the serial connection modelling, the feedback in the module can also be modelled. The only difference is the function output will be fed back to the input.

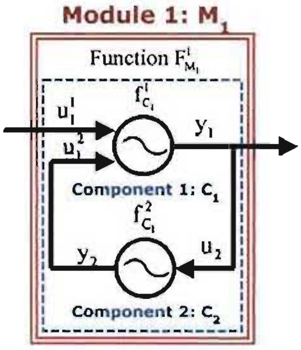


Figure 7-7 Feedback connection of component functions.

In this section, we describe the derivation of a higher-level function based on three basic manipulations: serial, parallel and feedback. A model of a more complex module can also be build by using these three manipulations. The procedures of establishing the model can follow the following steps:

- Module function determination;
- Achieving the module-level function by choosing the component functions;
- Using the performance of the component functions to derive the performance model of the module function.

7.4.2. Module-level Single-process Diagnostics

In this section, in order to evaluate whether a failure mode in a module is diagnosable, an approach is developed based on the diagnostic functions and the adjacency matrices. There are two classes of methods can be used to diagnose or to monitor a module: 1) to check whether a module is working properly as shown in Figure 7-8 (a); 2) to check whether a specific failure mode is happening in a module as shown in Figure 7-8 (b). In this thesis, the first one is called function diagnostics, and the second one is called failure mode diagnostics. For instance, if the diagnostics which compares ideal tool tip position and real tool position is used to check whether the RMM is working properly, it is called function diagnostics. Similarly, if the diagnostics which checks the cutting force to monitor tool tip breakage, it is called failure mode diagnostics. As shown in the figure, the input and output of the function are often used by the function diagnostics for comparison. We can use any state variable of the module to represent the failure mode diagnostics inputs. In figure 7-8 (b), the dash and dot line represents that the input signal of the function can be used for the failure mode diagnostics. The output of the module deviates from the designed performance will result in a module-level failure based on the failure in a module function. If the inputs and output of a module function can be directly achieved for the diagnostics, this module-level failure can be detected by using the function diagnostics. However, it is sometimes difficult to use the function inputs and output directly for the diagnostics. For instance, we normally use a motor shaft encoder to measure the speed of the machine tool table. So due to the sensor position of the diagnostics, the ball screw failure are not able to be detected by using the motor shaft encoder.

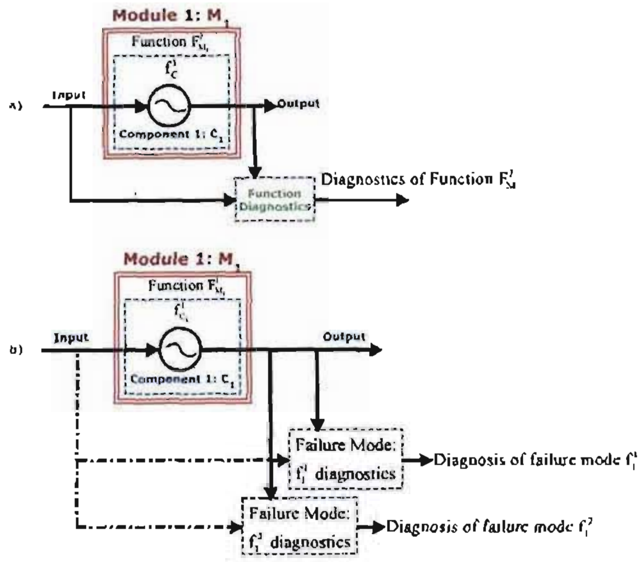


Figure 7-8 Classes of diagnostics.

There are undiagnosable failures existing in the module if it's not properly diagnosable by a module-level failure diagnostics. The states consists of the undiagnosable failure of a module will propagate to the other modules. Based on the system-level diagnostic function, the propagated failure can be detected. It will cause the system failure if the propagated failure can not be detected. So at a module-level, it is very important to evaluate the undiagnosable modes. In this section we will discuss some general procedures for the undiagnosable failure mode evaluation. As shown in Figure 7-9, it is an ideal case of the function diagnostics where the input and output of module function F^1_M can be accessed directly by using diagnostics function F^1_D. We assumed, in the figure, that there is only one failure mode f^1_C exists for the component function F^1_C.

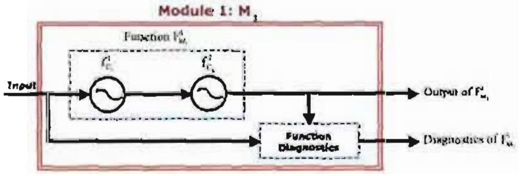


Figure 7-9 Function diagnostics.

The function diagnostics F_D^1 can be used to detect the input, output and all possible failures. So in this case, there is no undiagnosable mode exist. The detectable failure modes of the diagnostic function can be derived by using the following equations:

$$\text{Diagnostics}(\mathbf{M}_1)=\text{Diagnostics}(\mathbf{M}_1 g F_D^1)=\{\{f_C^1, f_C^2\}\} \qquad (7\text{-}4)$$

In Figure 7-10, a module that has the similarly structure as Figure 7-9, but the diagnostics cannot be used to directly monitor the input module function F_M^1.

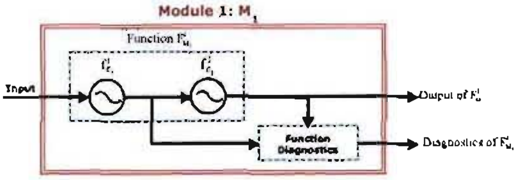


Figure 7-10 Function diagnostics if input cannot be accessed.

In this figure, the diagnostic function becomes the following equations because of the undiagnosability of the failure modes.

$$\text{Diagnostics}(\mathbf{M}_1)=\text{Diagnostics}(\mathbf{M}_1 g F_D^1)=\{f_C^1\} \qquad (7\text{-}5)$$

In Figure 7-11(a), the diagnosability is affected by the feedback loop if there is a feedback loop in the module-level function and a sensor used for the feedback loop is also used for the diagnostics. Figure 7-11(b) shows the failure is detectable if the failure mode of the component function is function loss.

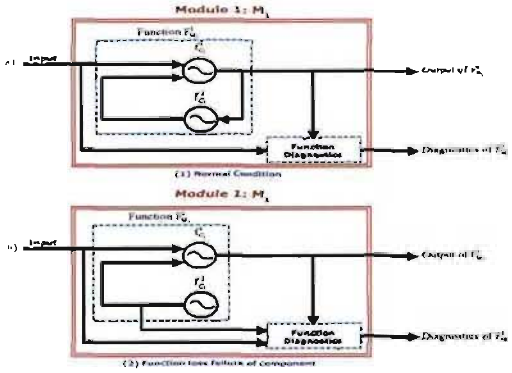


Figure 7-11 Function diagnostics using feedback function.

7.5. Summary

In this chapter, we discussed the implementation of module-level diagnostics and diagnosability of a module. In Figure, the relationships of failure modes in a module were illustrated.

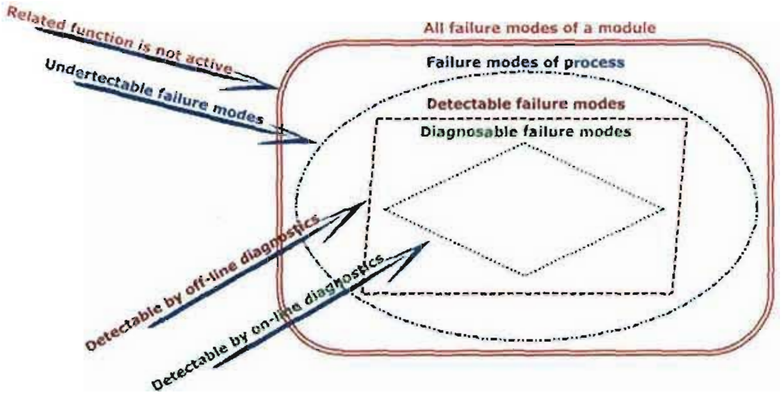


Figure 7-12 Relationships of failure modes in a module.

As can be seen from the figure, the currently used functions in the system can be utilized to determine the possible failure modes. The failure will never occur if the function which consists of a certain specific failure mode is not active.

CHAPTER 8. Conclusion and Directions for Future Studies

8.1. Conclusions

8.1.1. Advanced Manufacturing Technologies

The work in this thesis detailed the development of a unique, original RMM that could be used to implement automated part machining tasks into RMS. The RMM was successfully integrated into a research oriented RMS at the School of Mechanical Engineering of the University of KwaZulu-Natal, Durban, South Africa (ref Figure 8-1).

Figure 8-1 MR^{2}G research Agile Manufacturing environment

The RMM was developed as a reconfigurable, PC-based machine that implemented a modular mechanical structure and a modular mechatronics control system. The RMM was developed using an alternative modular design approach, including modular mechanical structure design and modular mechatronics control system design, which resulted in the separate development of four motion sub-systems used to implement the advanced reconfigurable manufacturing. The coordinated interaction of the APTS, the APLS, the APC/RS and the ATCS enabled the RMM to position the work piece at the right place and then accurately position the tool on the work piece and machine it from multiple sides. The design and analysis of the individual motion sub-systems was presented in Chapter 4.

The RMM was developed as an autonomous technology that implements a PC-based digital I/O control system to control, sequence and coordinate the interaction of the four motion systems to perform the standard routines were developed and coded in Microsoft Visual Basic 6 (VB 6). The graphical user interface with which the RMM is operated was also developed in VB 6. The structure of the digital I/O system and the development of the control algorithms implemented for the RMM were discussed in Chapter 6.

The RMM was developed as an in-line technology that could be able to perform in-process verification tasks in the reconfigurable agile manufacturing environment. The RMM was therefore developed to exhibit a high level of reconfigurability in its capability to integrate with RMS. The two key design factors that guaranteed this level of reconfigurability were the modular mechanical structure and the modular mechatronics control system (see Chapter 5). The combination of these factors resulted in the integration of the RMM in the AMS.

8.1.2. The Application and Integration of Modular Mechatronic Control

The RMM was controlled using a digital control system that was based on the TTL digital standard. Digital interfaces were low cost to implement. The work in this thesis demonstrated the application of mechatronics principles required to implement four independent motion sub-systems using a digital interface.

The development of the RMM discussed in this thesis provided an example of the implementation of an alternative design methodology, termed *modular mechatronics*. Modular mechatronics is a powerful rapid system control development methodology that allows for the rapid development of reconfigurable, agile manufacturing technologies and therefore represents a new and original contribution to the fields of advanced manufacturing.

Modular mechatronics control is a modular technique that enables the overall development task to be subdivided into smaller sub-tasks, or mechatronic function modules (MFMs), that are grouped according to their mechatronic functionality. These sub-tasks involve the development of motion system and in the case of RMM are represented by the APTS, APC/RS, APLS and ATCS.

Modular mechatronics control requires that an overall concept of the final system/project is developed through an evolutionary process of conceptual development. Once this has been finalized the physical criteria for the motion sub-systems can be determined using typical engineering analysis techniques. If other functionality tasks are required, the appropriate analysis and research must be conducted to determine the design criteria for the system. The conceptual development of the RMM and the required engineering analyses of the motion systems were presented in Chapter 4.

8.2. Suggestions for Future Work

To further improve RMM design, the above study could be extended in the following directions:

- **Multi-Spindle RMM Design:** In this thesis, the author has developed the RMM design procedure by assuming there is only one spindle for each machine tool, although a tool storage device has been designed in the experiment case study. In order to reduce cycle time in the real manufacturing environment, the synthesis methodology of the RMM design should be extended to multiple spindles' domain.

- **Fully integrated with CIM centre:** In a CIM centre, there are many factors affect the final position and orientation of the workpiece on the machining table such as conveyor transfer system, material handling system, etc. Meanwhile, the RMM design is significantly affected by those configurations of workpiece. As a result, the RMM design procedure should also be extended by considering other factors in a CIM centre.
- **The design of a RMS:** RMM is one of the basic components of RMS. Due to the limited research budget, in this research work, there is only one RMM has been build and to some extent this RMM can only be tested in a experimental CIM centre which is available in the research laboratory. In order to get more accurate test results, more RMMs have to be designed and constructed so as the whole RMS can be formulated
- **Machining accuracy estimation:** Machining accuracy is one of the most important criteria for machine tool quality evaluation. In RMM design, because of the modular structure, each module may have its own manufacturing errors and these errors may be accumulated and amplified after different modules have been connected. As a result, the machining accuracy of RMM may be significantly affected by these manufacturing errors. So the establishment of a methodology which can predict the machine tool accuracy is a crucial topic that should be covered in the future research work.
- **Design software development:** In this research, all of the 3D features of different modules of RMM were created by using UniGraphics. There are also some other commercial CAD softwares available but none of them can deal with RMM design specifically. So there is a need to develop a computer program which can help both RMM designers and users. On the designer side, this software is supposed to provide assistance for RMM designer in the progress of designing new modules or new possible structure solutions for RMM. On the user side, it supposed to assist user in reconfiguring an existing RMM to cope with the unexpected changes. In order to follow the modular idea throughout the RMM design, RMM-CAD software itself should also

be programmed in a modular fashion which allows designers and users to enter, modify specification, and collect the information of various machine modules.

References

[1]. Goebel, P., 2004, ***Reconfigurable manufacturing systems***, Proceedings of the International Conference on Competitive Manufacturing, pp.69-79, COMA'04, Stellenbosch South Africa.

[2]. Koren, Y. and Kota, S., 1999, ***Reconfigurable machine tools***, U.S. Patent, No.5,943,750

[3]. Koren, Y. and Ulsoy, A.G., 1997, ***Reconfigurable manufacturing systems***, Engineering Research Center for Reconfigurable Manufacturing Systems (ERC/RMS) Report1, ERC/RMC, University of Michigan, Ann Arbor, MI.

[4]. Mehrabi, M.G., Ulsoy A.G. and Koren, Y., 2000, ***Reconfigurable manufacturing systems: key to future manufacturing***, Journal of Intelligent Manufacturing, 11(4), pp.403-419.

[5]. Mehrabi, M.G., Ulsoy, A.G., Koren, Y. and Heytler, P., 2002, ***Trends and perspectives in flexible and reconfigurable manufacturing systems***, Journal of Intelligent Manufacturing, 13(2), pp.135-146.

[6]. Mellor, C., 2002, ***Quick-change artist: why techs must get ready for reconfigurable manufacturing***, The Ontario Technologist, Jan/Feb, pp.12-15

[7]. Rogers, G. G. and Bottaci, L., 1997, ***Modular production systems: a new manufacturing paradigm***, Journal of Intelligent Manufacturing, 8(2), pp.147-156.

[8]. Glen Bright, Bo Xing, Stephen Craig, 2005, ***Modular Machine Design for Reconfigurable Manufacturing***. CIRP-3rd International Conference on Reconfigurable Manufacturing, 10-12 May 2005, Ann Arbor, Michigan, USA.

[9]. Glen Bright, Bo Xing, 2005, ***Reconfigurable Machines for Mass Production***, 3rd Interdisciplinary World Congress on Mass Customization and Personalization, Hong Kong, China.

[10]. Bo Xing, Jesse Eganza, Glen Bright, 2006, ***Reconfigurable Manufacturing System for Agile Manufacturing***, 12th IFAC Symposium on Information Control Problems in Manufacturing, Saint-Etiennne, France.

[11]. Bo Xing, Glen Bright, Nkgatho S. Tlale, 2006, ***Reconfigurable Manufacturing System for Agile Mass Customization Manufacturing***, 22nd International Conference on CAD/CAM Robotics and Factories of the Future CARs & FOF 2006, Vellore Institute of Technology, Deemed University, India.

[12]. Mikell P. Groover, 2001, ***Automation, production systems and computer-integrated manufacturing***, 2 Edition, Prentice-Hall, pp.375-377.

[13]. Osita D.I.Nwokah and Yildirim Hurmuzlu, 2002, ***The mechanical systems design handbook: modeling, measurement, and control***, Southern Methodist University, Dallas, Texas, CRC Press, pp.18-27.

[14]. Koren, Y. and Ulsoy, A.G., 2002, ***Vision, principles and impact of reconfigurable manufacturing systems***, Powertrain International Magazine, pp. 14-21

[15]. Paredis, C.J.J., Brown, H.B., and Khosla, P.K., 1996, ***A rapidly deployable manipulator system***, In Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, pp.1434-1439.

[16]. Paredis, C.J.J., 1996, ***An agent-based approach to the design of rapidly deployable fault tolerant manipulators***, PhD Dissertation, Department of Electronical and Computer Engineering, Carnegie Mellon University, Pittsburgh, Pennsylvania.

[17]. Chen, I.-M., 1993, ***Theory and application of modular reconfigurable robotic systems***, PhD Dissertation, California Institute of Technology, CA.

[18]. ASME, 1992, ***Method for performance evaluation of computer numerically controlled machining centers***, B5.54.

[19]. Rao, S.B., 1997, ***Metal cutting machine tool design – a review***, Journal of Manufacturing Science and Engineering, Vol. 119, pp.713-716.

[20]. Slocum, A.H., 1992, ***Precision machine design***, Prentice-Hall. Inc., USA.

[21]. Walsh, R.A., 1994, ***McGraw-Hill machining and metalworking handbook***, McGraw-Hill, USA.

[22]. Weck, M. and Bibbring, H., 1980, ***Handbook of machine tools***, Awiley Heyden Publication, Vol 1-4.

[23]. Garro, O., Martin, P., and Marth, H., 1992, ***Algebraic description for machine tool design***, International Journal of Production Research, Vol.30, No.11, pp.2603-2641.

[24]. Cariapa, V., 1991, ***Multimode machine tools—a concept that improves operation of flexible manufacturing systems***, International Journal of Production Research, Vol.92, No.5, pp.1069-1079.

[25]. Nemeth, I., Fisette, P. and Brussel, H.V., 1999, ***Conceptual design of 3-axis machine tools***, In proceedings of the 32nd CIRP International Seminar on Manufacturing Systems, Leuven, Belgium, pp.239-248.

[26]. Hatamura, Y., Nagao, T., Mitsuishi, M. and Nagao, T., 1995, ***Actual conceptual design process for an intelligent machining center***, Annals of the CIRP, Vol.44, pp.123-128.

[27]. Park, M.-W., Cha, J.-H. and Park, J.-H., 1999, ***An intelligent design system for machine tools***, Proceedings of the 32nd CIRP, International Seminar on Manufacturing Systems, Leuven, Belgium, pp.249-254.

[28]. Walczyk, D.F., Lakshmikanthan, J. and Kirk, D.R., 1998, ***Development of a reconfigurable tool for forming aircraft body panels***, Journal of Manufacturing Systems, 17(4), pp.287-296.

[29]. Ceil, J., 2001, ***Computer-aided fixture design—a review and future trends***, International Journal of Advance Manufacturing Technology, 18(11), pp.790-793.

[30]. Perez R., Aca, S. J., Valverde T. A., Ahuett, G.H., Molina, G.A. and Riba,

R. C., 2004, ***A modularity framework for concurrent design of reconfigurable machine tools***, Y. Luo(Ed): CDVE 2004, LNCS3190, Verlag Berlin Heideberg: Springer, pp.87-95.

[31]. Moon, Y. M. and Kota, S., 2002, ***Design of reconfigurable machine tools***, Transactions of the ASME, 124(2), pp.480-483.

[32]. Katz, R. and Moon Y.-M., 2000, ***Virtual arch type reconfigurable machine tool design: principles and methodology***, NSF ERC Virtual RMT report 41, ERC/RMS, University of Michigan, Ann Arbor, MI.

[33]. Landers, R. G., Min, B.-K. and Koren, Y., 2001, ***Reconfigurable machine tools***, Annals of the CIRP, 50 (1), pp.269-274.

[34]. Landers, R. G., 2000, ***A new paradigm in machine tools: reconfigurable machine tools***, Proceedings of the 2000 Japan-USA Flexible Automation Conference, Ann Arbor, Michigan, pp.361-364.

[35]. Moon, S-K., Moon, Y. M. and Kota, S., 2001, ***Screw theory based metrology for design and error compensation of machine tools***, Proceedings of ASME 2001 Design Engineering Technical Conferences, Pittsburgh, PA, pp.697-707.

[36]. Heisel, U. and Meitzner, M., 2003, ***Progress in reconfigurable manufacturing systems***, 2nd International Conference on Reconfigurable Manufacturing, Ann Arbor, Michigan, CD-ROM, B11.

[37]. Zhang, C. and Wang, H.P., 1993, ***Optimal process sequence selection and***

manufacturing tolerance allocation, Journal of Design and Manufacturing, 3, pp.135-146.

[38]. Ming, X.G. and MAK, K.L., 2001, ***Intelligent approaches to tolerance allocation and manufacturing operations selection in process planning***, Journal of Material Processing Technology, 117(1), pp.75-83.

[39]. Yeo, S. H. and Hgoi, B.K.A., 1996, ***A cost-tolerance model for process sequence optimization***, International Journal of Advanced Manufacturing Technology, 12(6), pp.423-431.

[40]. Juan, H., Yu, S.F. and Lee, B.Y, 2003, ***The optimal cutting parameter selection of production cost in HSM for SDK61 tool steels***, International Journal of Machine Tools & Manufacture, 43(7), pp.679-686.

[41]. Hui, Y. V., Leung, L. C. and Linn, R., 2001, ***Optimal machining conditions with costs of quality and tool maintenance for turning***, International Journal of Production Research, 39(4), pp.647-665.

[42]. Wu, Z., ElMaraghy, W.H. and ElMaraghy, H.A., 1988, ***Evaluation of cost-tolerance algorithms for design tolerance analysis and synthesis***, ASME Manufacturing Review, 1(3), pp.168-179.

[43]. Diplaris, S. C. and Sfantsikopoulos, M. M., 2000, ***Cost-tolerance function: a new approach for cost optimum machining accuracy***, International Journal of Advanced Manufacturing Technology, 16(1), pp.32-38.

[44]. Wang, P., 2003, ***Simultaneously solving process selection, machining parameter***

optimization and tolerance design problems: a bi-criterion approach, M.A.Sc. Thesis, Department of Mechanical Engineering, University of Ottawa, Canada.

[45]. MR^{2}G Group, 2005, ***Modular Mechatronics Control for Reconfigurable Manufacturing System***, Research Report, University of KwaZulu-Natal, Durban, South Africa.

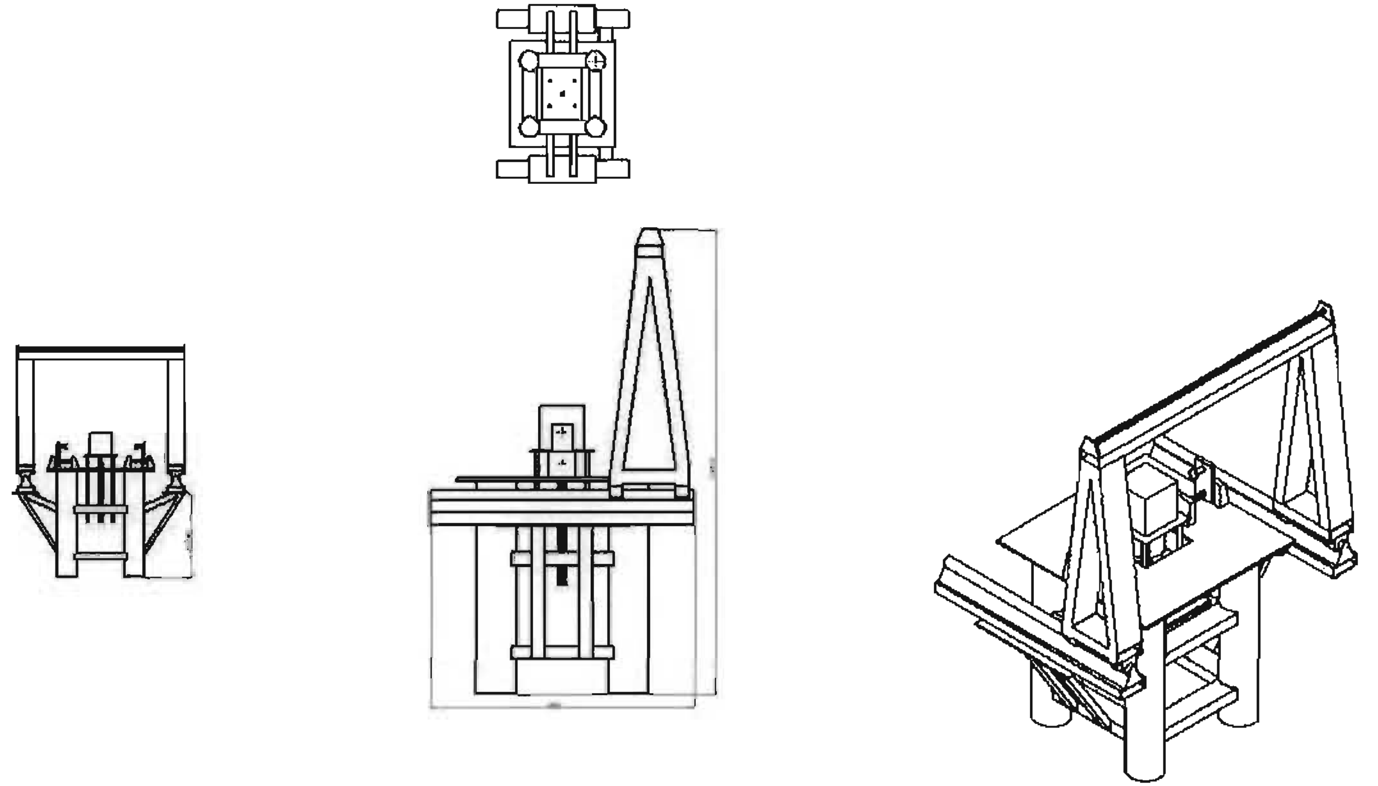
[46]. Mayor R., 2000, ***Automated Vision Inspection Apparatus for Flexible Manufacturing System***, University of Natal, Durban South Africa.

[47]. Rexroth, 2005, ***Linear Bushings and Shafts***, Bosch Group The Drive & Control Company.

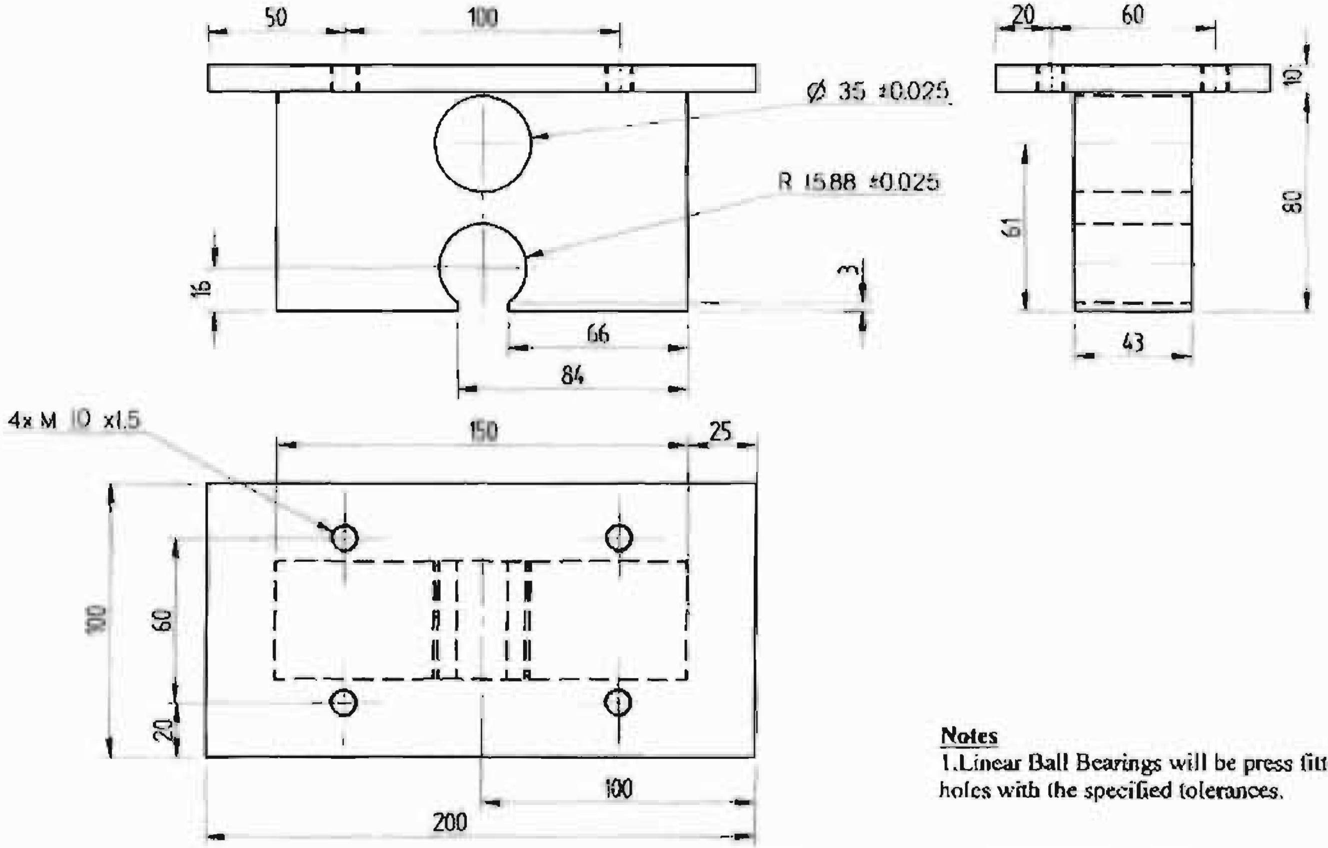
Appendix A

RMM CAD Manufacturing Drawings

The RMM was designed in the Unigraphical 3.0 and SolidEdge V17 computer aided design environment. The working drawings of the mechanical design of the RMM have been included in this appendix.



UNIVERSITY OF KWA-ZULU NATAL School of Mechanical Engineering		DATE	CHECKED	SCALE 1:5	UNITS mm	TITLE: RMM Assembly	No. 4
	W'Shop Technician			MATERIAL Mild Steel	NOTES None		
	Draftsperson			STUDENT NAME. Bo Xing		TEL No 2601227 E-MAIL [illegible]	
	Project Supervisor			Project : RMS			

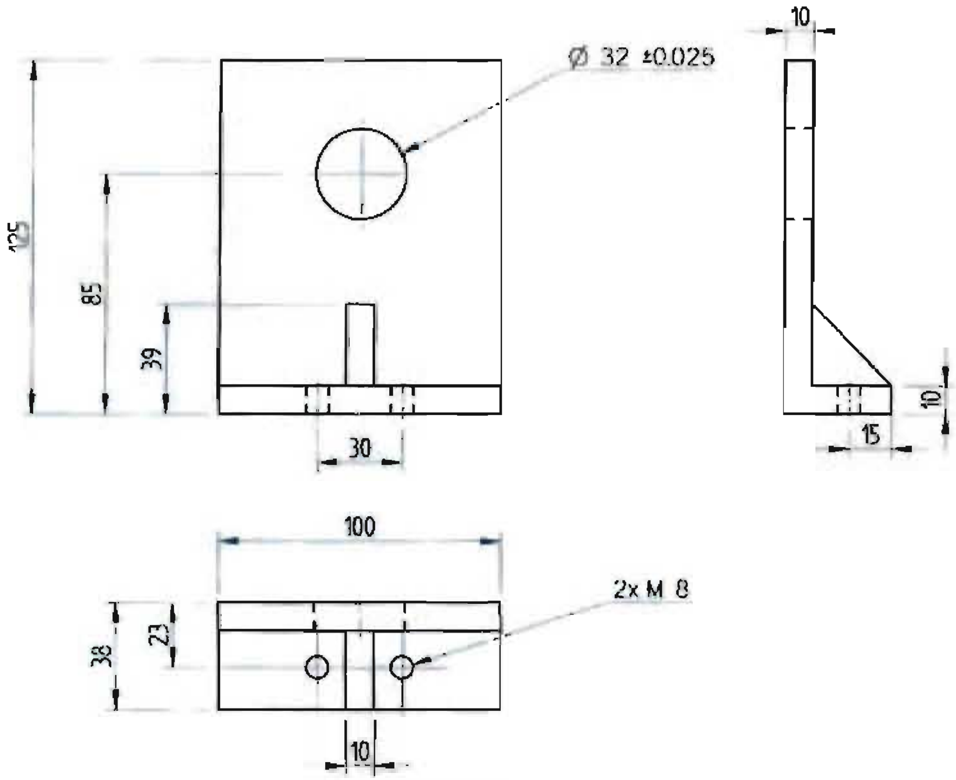


Notes
1.Linear Ball Bearings will be press fitted to the holes with the specified tolerances.

UNIVERSITY OF
KWA-ZULU NATAL
School of Mechanical Engineering

	DATE	CHECKED
W'Shop Technician		
Draftsperson		
Project Supervisor		

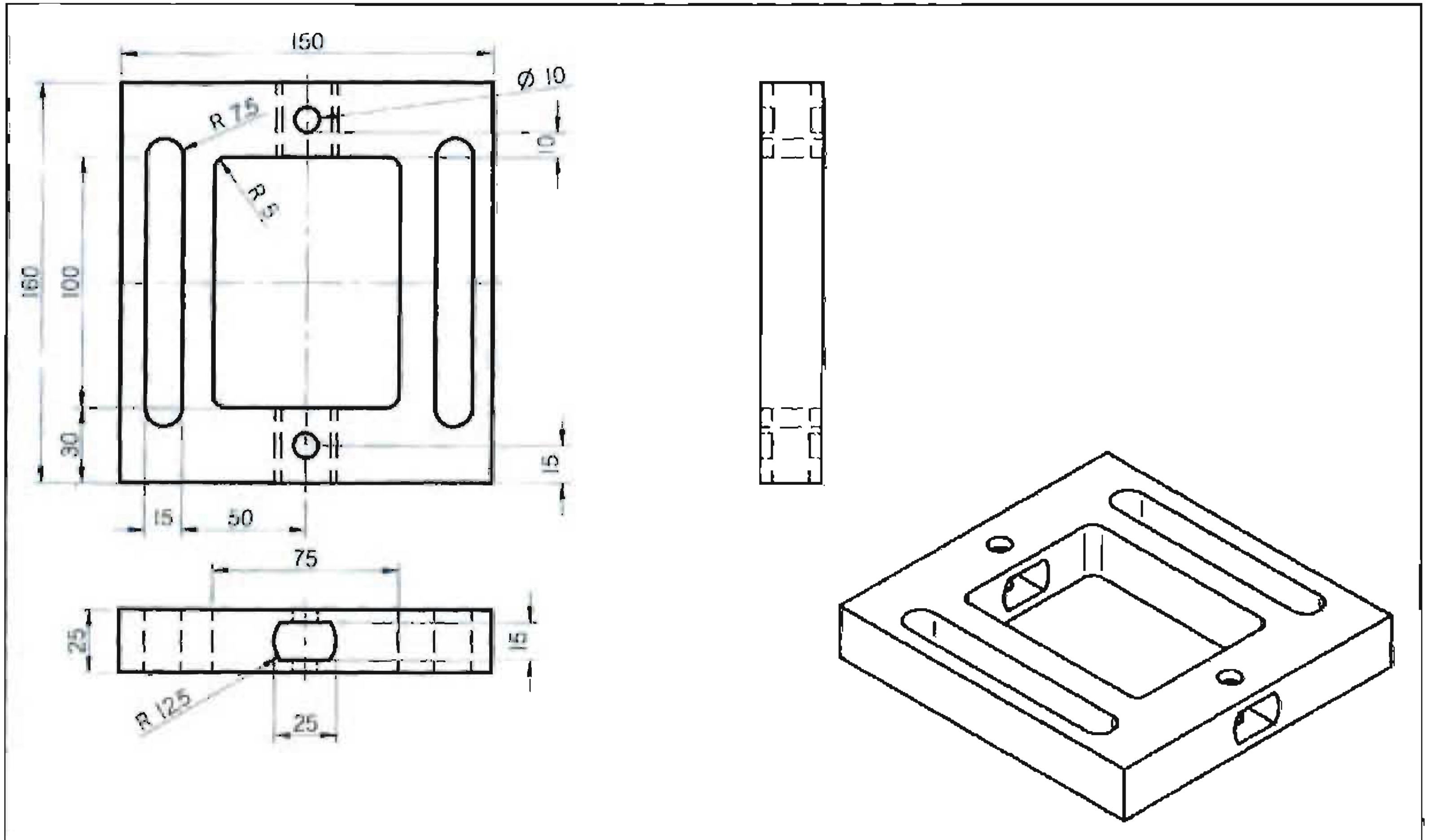
SCALE 1:2
UNITS : mm
MATERIAL Mild Steel(EN2)
NOTES
STUDENT NAME : Bo Xing
Project : BMS
TITLE: Bearing Housing
ST. No 2061227
E-MAIL [illegible]
No. 2



Notes

1. A radial roller bearing with OD 32 will be press fitted to the hole with a specified tolerance

UNIVERSITY OF KWA-ZULU NATAL School of Mechanical Engineering		DATE	CHECKED	SCALE 1:2	UNITS : mm	TITLE: L Bracket	No. 2
	W'Shop Technician			MATERIAL: Mild Steel (EN2)	NOTES:	TEL No. 073 147 5934	
	Draftsperson			STUDENT NAME : Bongumusa Mashazi		E-MAIL: 201504189@ukzn.ac.za	
	Project Supervisor			Project : Mechatronics Control of Adv. Manufacturing			



UNIVERSITY OF
KWA-ZULU NATAL
School of Mechanical Engineering

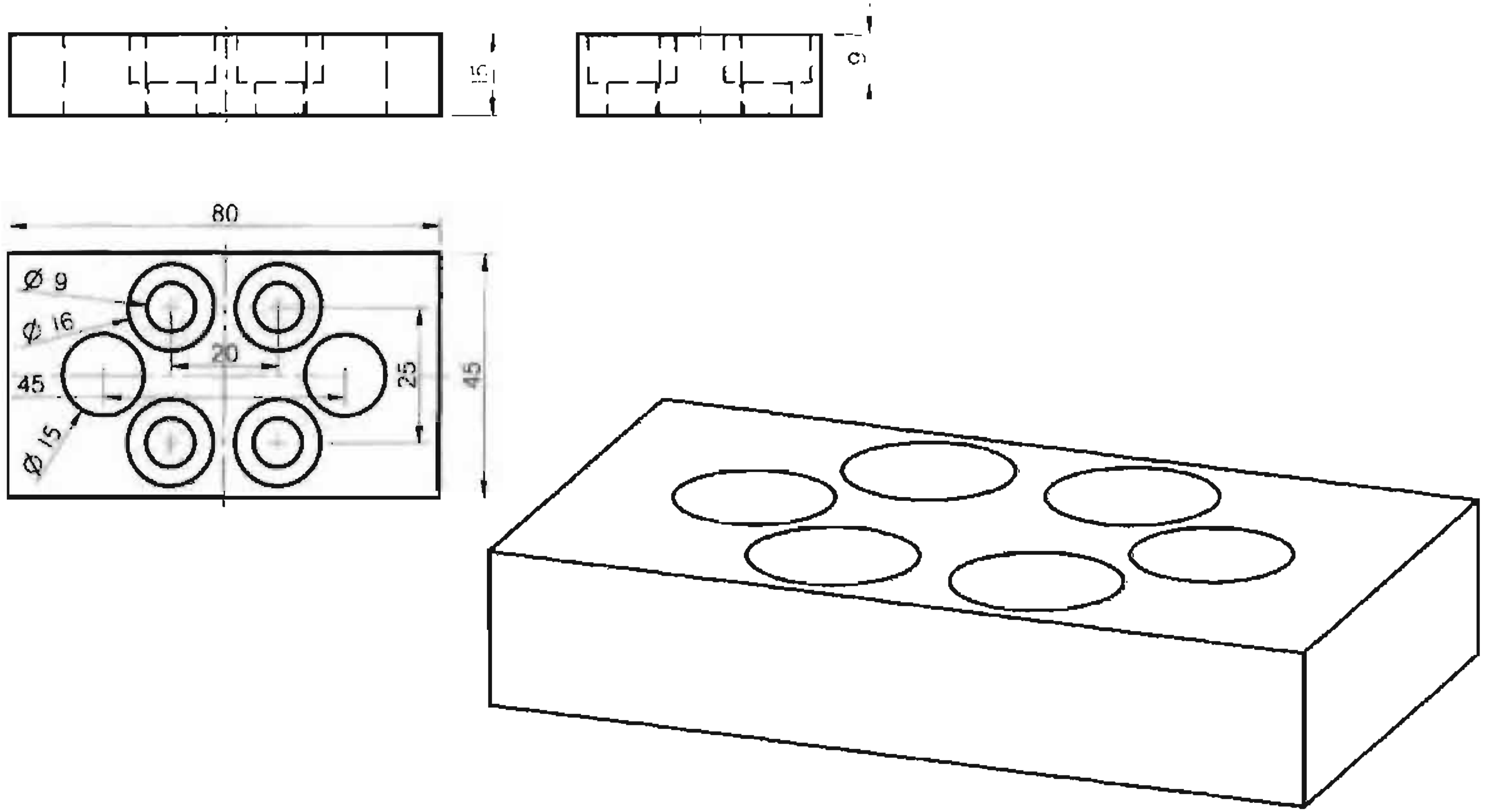
	DATE	CHECKED
W'Shop Technician		
Draftsperson		
Project Supervisor		

SCALE 1:2
UNITS : mm
MATERIAL
NOTES
STUDENT NAME :
Project :

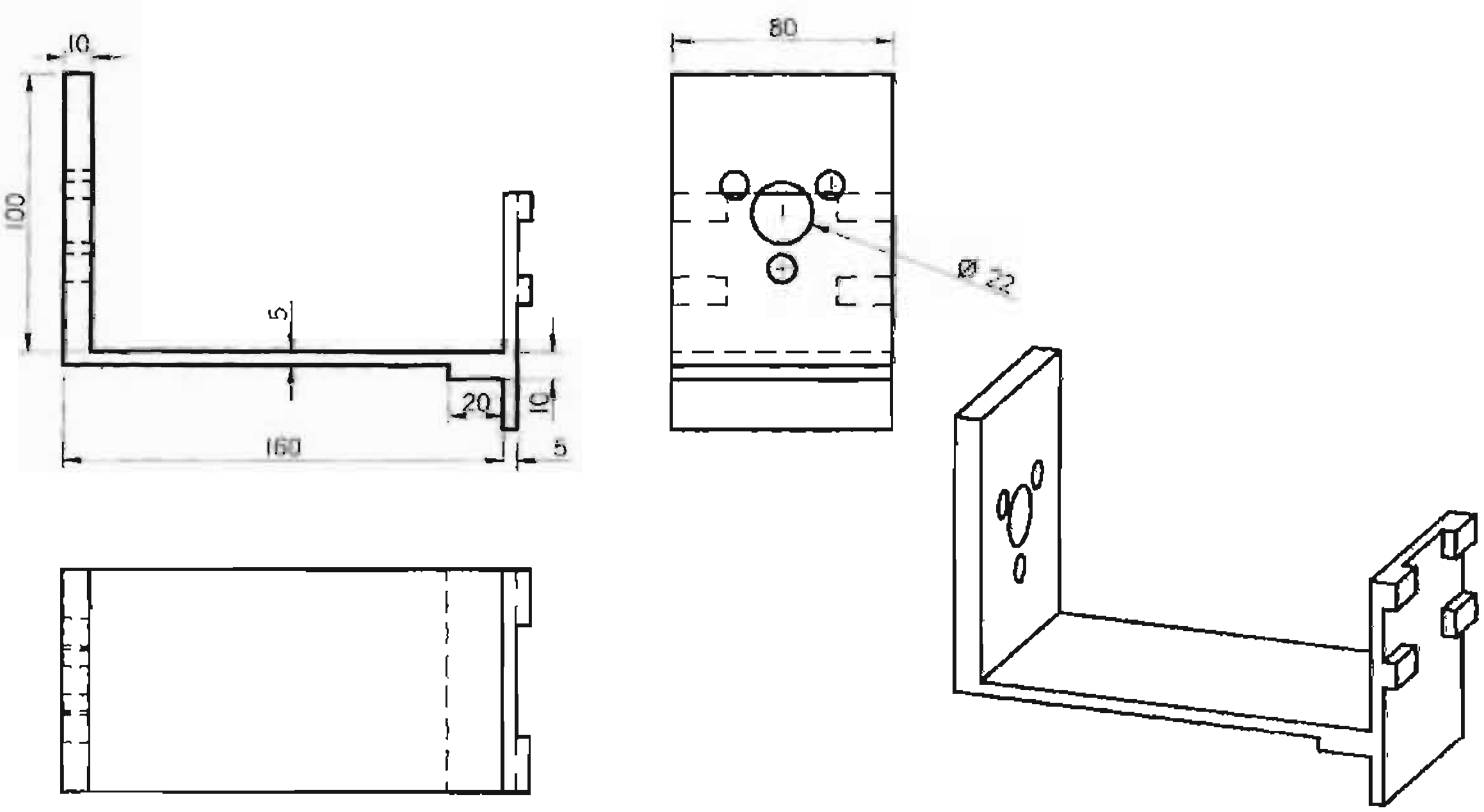
TITLE: vice holder

No. 2

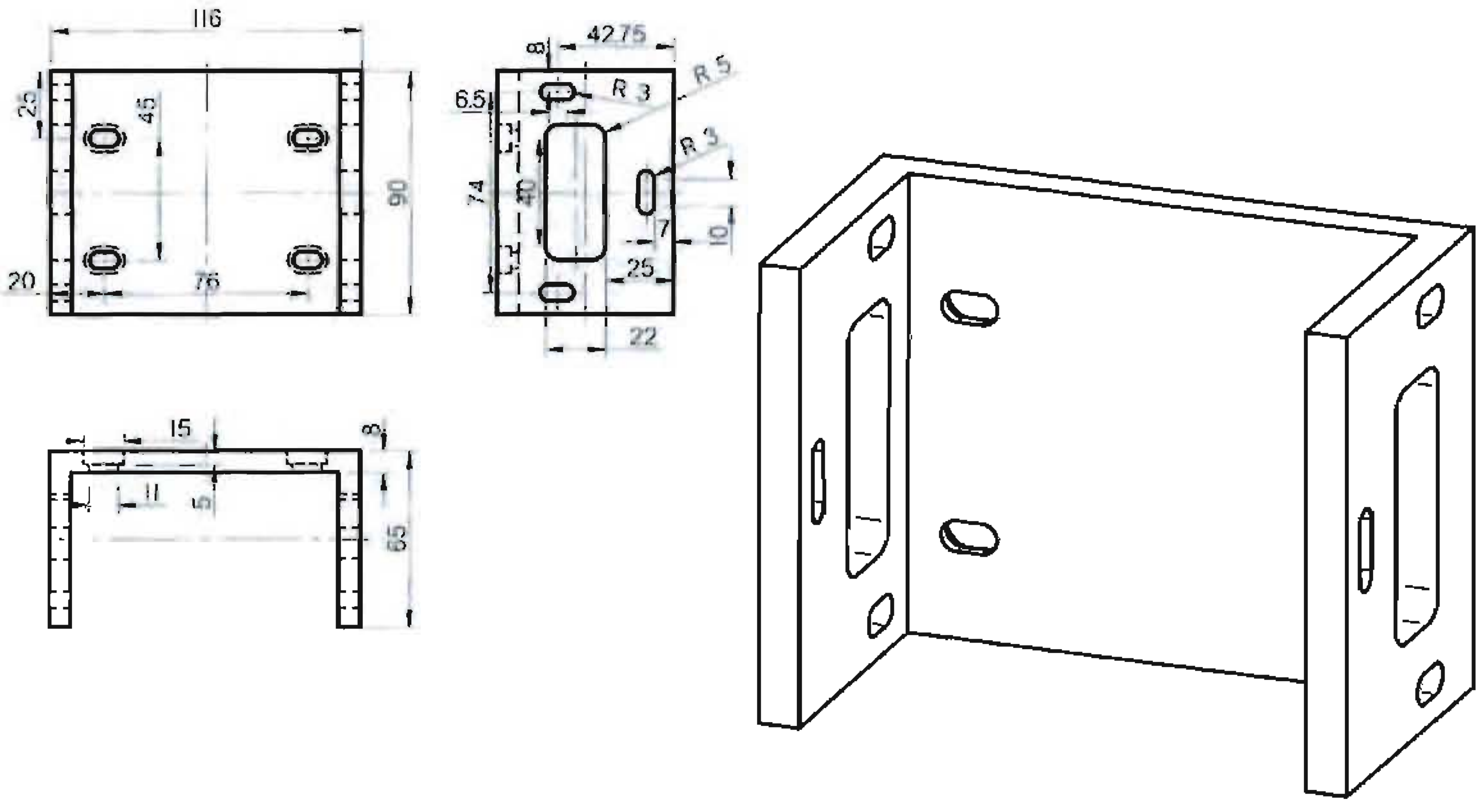
TEL No [illegible]
E-MAIL [illegible]



UNIVERSITY OF KWA-ZULU NATAL School of Mechanical Engineering		DATE	CHECKED	SCALE 1:	UNITS : mm	TITLE: Leadscrew	No. 2
	W'Shop Technician			MATERIAL	NOTES		
	Draftsperson			STUDENT NAME :		TEL No	
	Project Supervisor			Project :		E-MAIL	



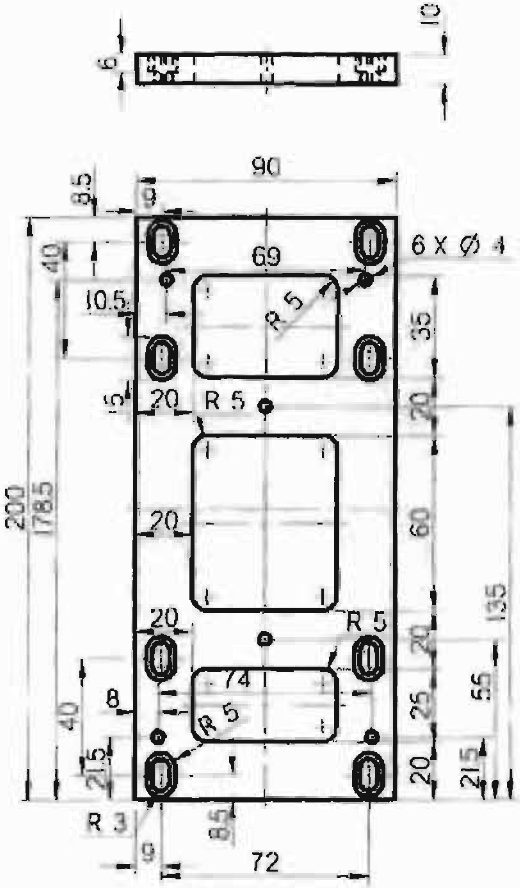
UNIVERSITY OF KWA-ZULU NATAL School of Mechanical Engineering		DATE	CHECKED	SCALE 1:1 MATERIAL	UNITS : mm NOTES	TITLE: Shaft	No. 2
	W'Shop Technician			STUDENT NAME :		TEL No. [illegible]	
	Draftsperson					E-MAIL [illegible]	
	Project Supervisor			Project : RMS			

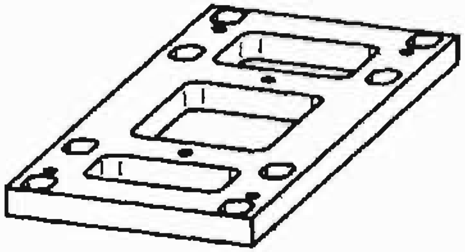


UNIVERSITY OF
KWA-ZULU NATAL
School of Mechanical Engineering

	DATE	CHECKED
W'Shop Technician		
Draftsperson		
Project Supervisor		

SCALE 1:
UNITS : mm
MATERIAL
NOTES
STUDENT NAME :
Project :
TITLE Leadscrew
TEL No
E-MAIL
No. 2





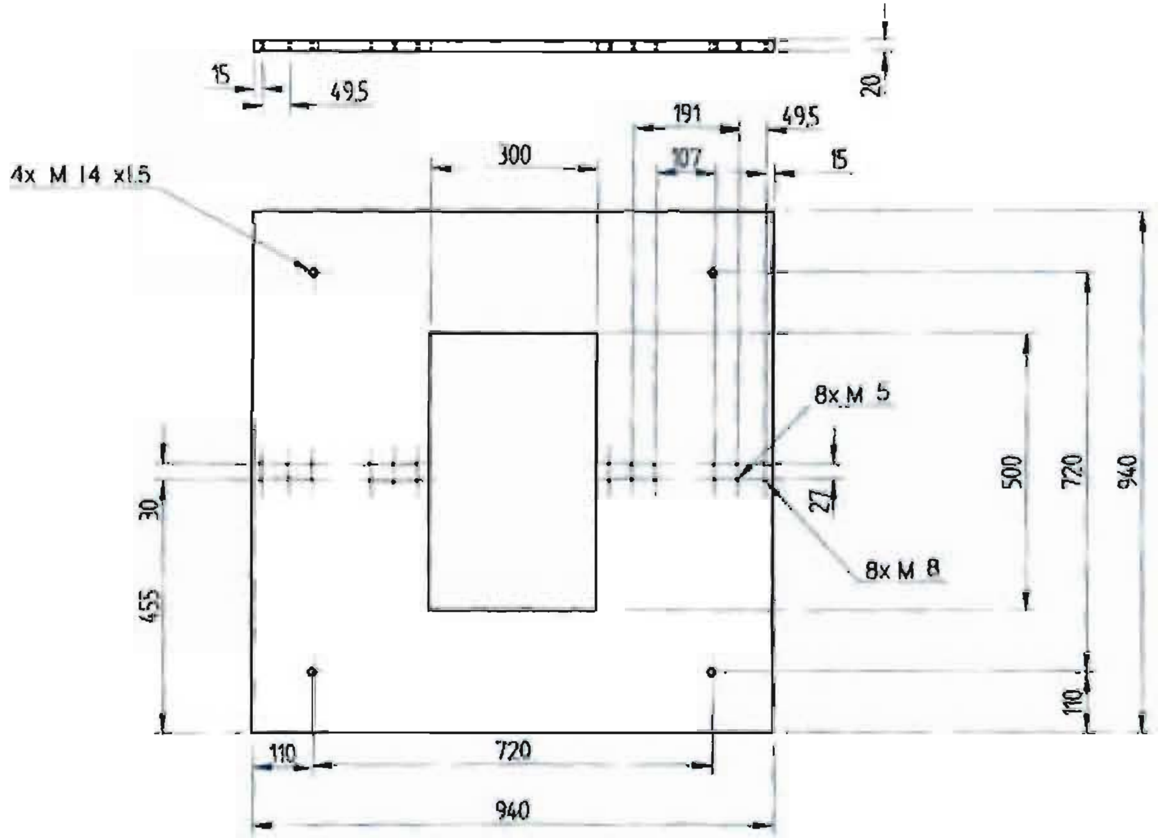
UNIVERSITY OF KWA-ZULU NATAL School of Mechanical Engineering		DATE	CHECKED	SCALE 1:1	UNITS : mm
	W'Shop Technician			MATERIAL [illegible]	NOTES
	Draftsperson			STUDENT NAME : Bo Xing	
	Project Supervisor			Project : Reconfigurable Machine	

TITLE: [illegible]

No. 1

TEL No [illegible]

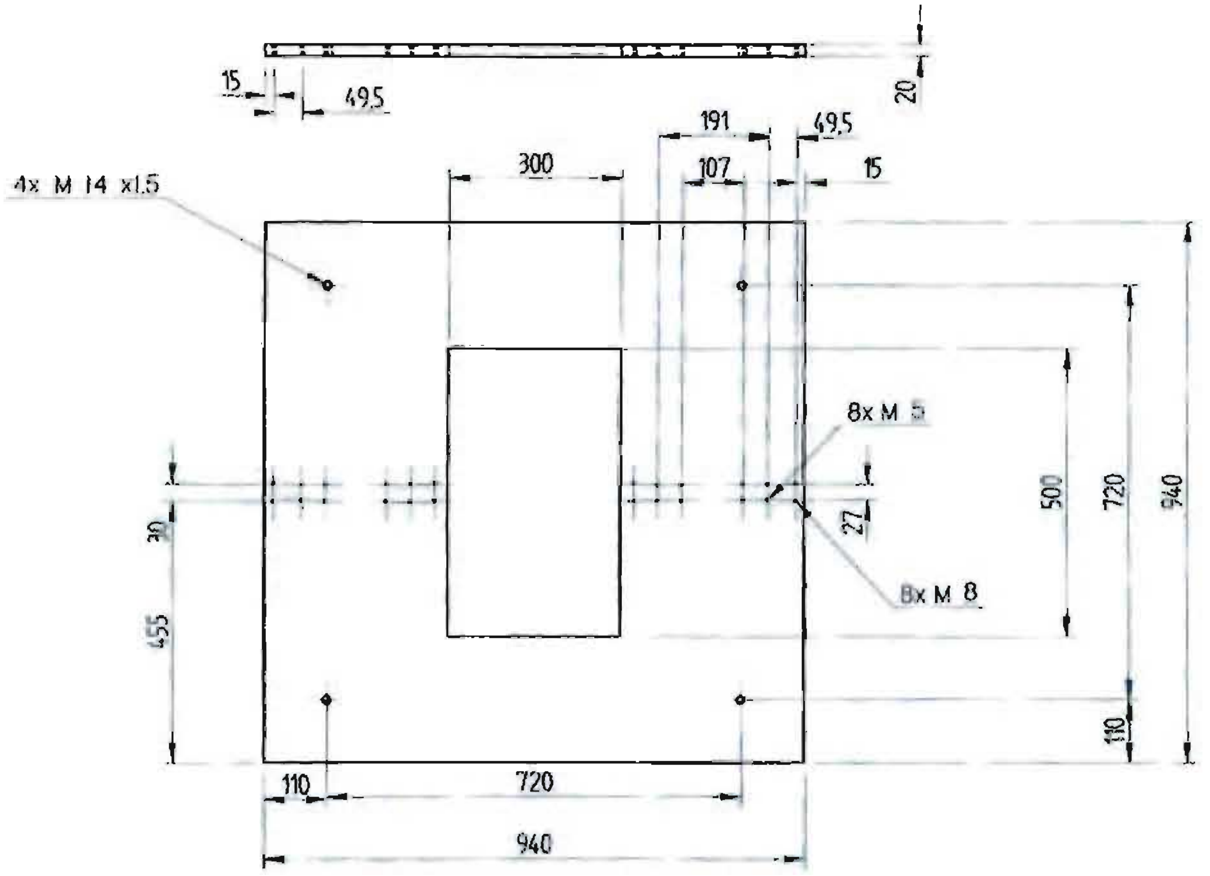
E-MAIL [illegible]



UNIVERSITY OF
KWA-ZULU NATAL
School of Mechanical Engineering

	DATE	CHECKED
W'Shop Technician		
Draftsperson		
Project Supervisor		

SCALE 1:
UNITS : mm
MATERIAL Mild Steel(EN2)
NOTES
STUDENT NAME : Bo Xing
Project : RMS
TITLE: Table
TEL No 2601227
E-MAIL [illegible]
No. 1



		DATE	CHECKED				
UNIVERSITY OF KWA-ZULU NATAL School of Mechanical Engineering				SCALE 1:	UNITS : mm	TITLE: Table	No. 1
	W'Shop Technician			MATERIAL Mild Steel(EN2)	NOTES		
	Draftsperson			STUDENT NAME : Bo Xing		TEL No 2601227 E-MAIL [illegible]@ukzn.ac.za	
	Project Supervisor			Project : RMS			

Appendix B

DC Motor Specification

The RMM used a fractional horsepower DC motor supplied from Smiths Manufacturing, Pinetown, South Africa.

The motor produces a stall torque of 20 Nm at 20 A.

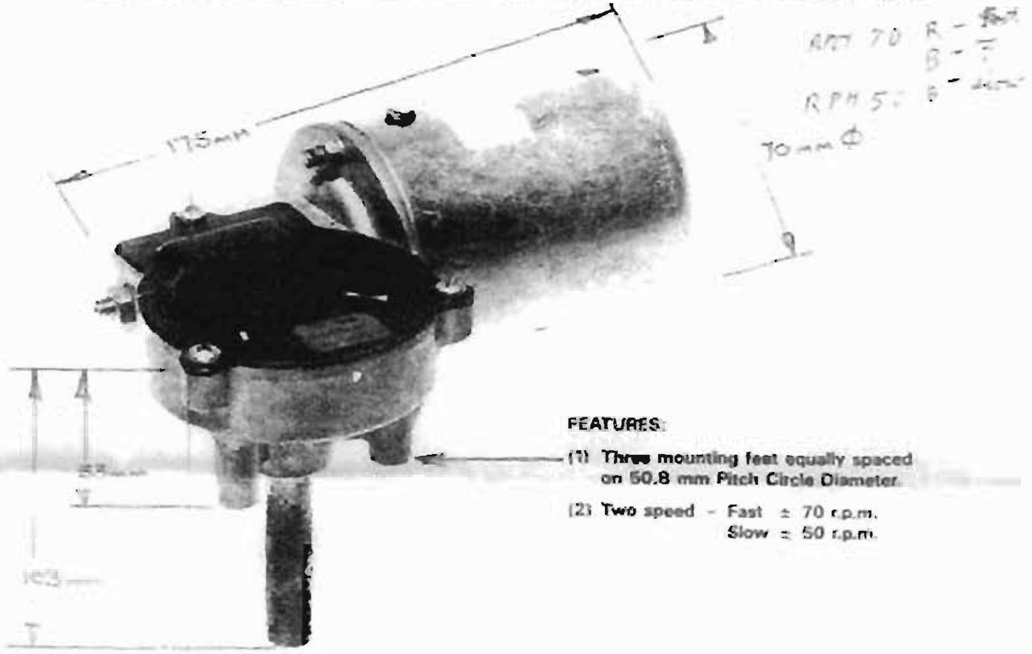
The speed/torque and current/torque relationships are presented in the following to graphs.

METPART

AUTOMOTIVE COMPONENT

ATTENTION.
Basil.

AUTOMATIC MOTOR AND GEARBOXES



FEATURES:

(1) **Three mounting feet equally spaced on 50.8 mm Pitch Circle Diameter.**

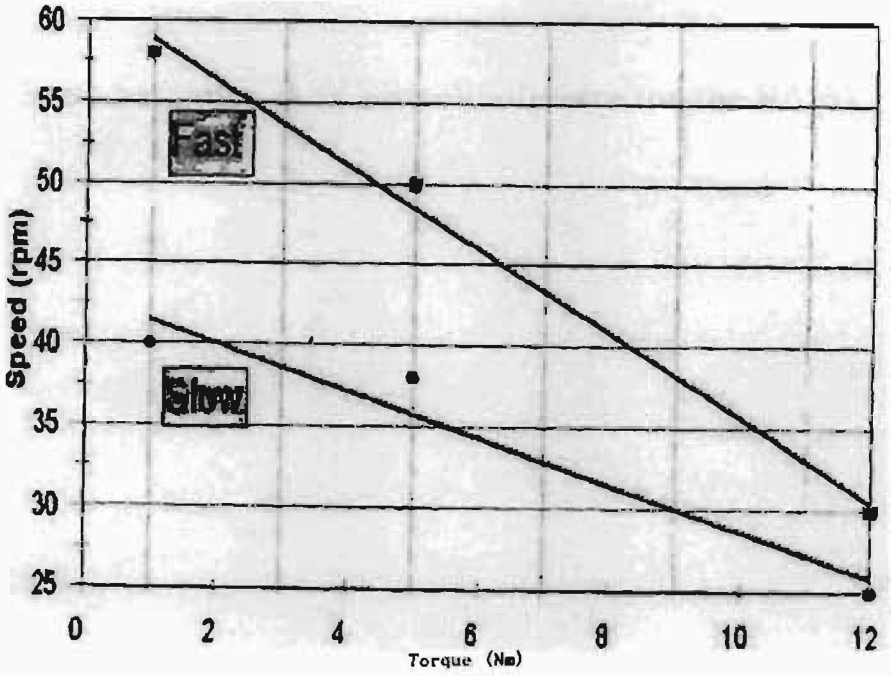
(2) **Two speed** - **Fast** ± 70 r.p.m.
Slow ± 50 r.p.m.

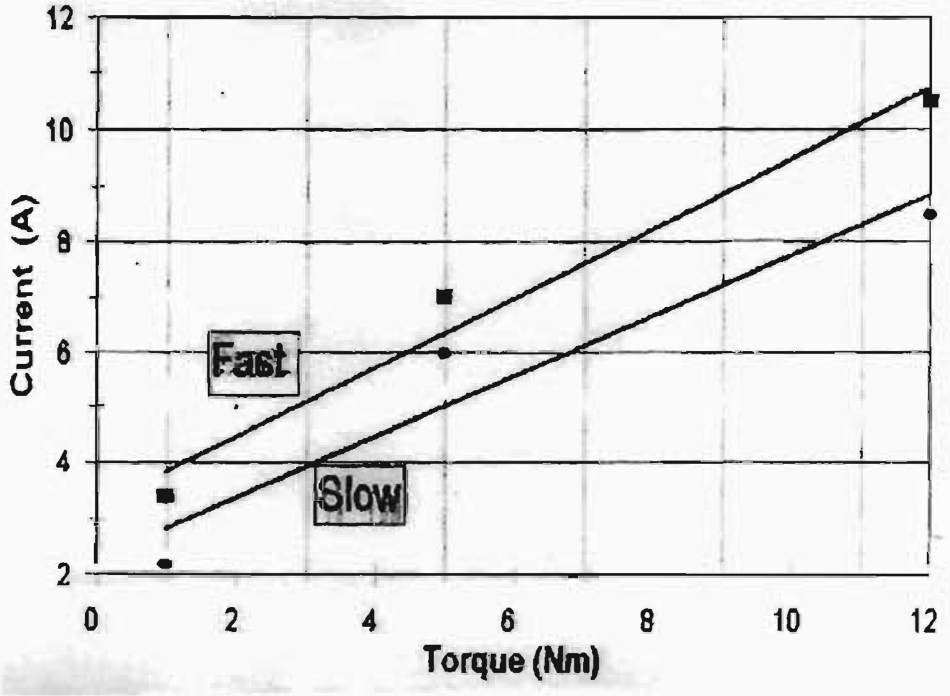
SPECIFICATIONS	DRIVE SHAFT DIAMETER	TORQUE/AMPS			RPM TORQUE/SPEED			STALL TORQUE
1) MS269-03/MS270-03	12 mm	1Nm	FAST SLOW	2.5 3.5	1Nm	FAST SLOW	70 50	18Nm/18 amps
		3Nm	FAST SLOW	6.5 5.3	3Nm	FAST SLOW	45 35	4.5
		12Nm	FAST SLOW	8.8 11.0	12Nm	FAST SLOW	20 15	3.8
2) MS 269-04/MS270-04	11 mm	1Nm	FAST SLOW	3.4 2.2	1Nm	FAST SLOW	46 40	20Nm/17 amps
		5Nm	FAST SLOW	7.5 5.2	5Nm	FAST SLOW	40 31	
		12Nm	FAST SLOW	12.0 10.9	12Nm	FAST SLOW	20 17	

V.9T

ALL DATA QUOTED AT OPERATIONAL VOLTAGE OF 13.5 VOLTS

Black Wire = EARTH
Red " = FAST SPEED
Green " = SLOW SPEED





Current Characteristic

Current vs Torque

Appendix C

Code Listing for the Control Software of the RMM

This appendix list presents a listing of the source code of the RMM control software that was developed using Microsoft Visual Basic.

Components of the control software package

The software for the RMM was developed as a VB 6.0 project, RMM control software.vbp. the forms and modules or the project are listed below.

Forms:

- FrmAbout;
- FrmAutomated;
- FrmExitControl;
- FrmAPTS;
- FrmAPC/RS;
- FrmAPLS;
- FrmATCS;
- FrmMain.

The graphic user interfaces for the various form objects have been include in this appendix. The following sections list the source code for the all the forms.

```
'Eagle Technology
'===============
'Copyright (c) 1999
'EDRE Digital I/O Example
'Option Explicit
Dim A, B, counter As Integer
Dim SerialNo As Long
Dim Major As Long
Dim Minor As Long
Dim bld As Long
Dim opsys As Long
Dim nd As Long

Private Sub cmdDown_Click()
EDREDioX1.Write 2, 63
End Sub

Private Sub cmdEntireSyst_Click()

Dim A As Integer
'G is the value counting the gripper motor rotations

For G=0 To 16 Step 1
EDREDioX1.Write 0, 15
txtG=Val(G)
If G=16 Then
EDREDioX1.Write 0, 0
End If
```

```
Next G

For V=0 To 3000 Step 1
EDREDioX1.Write 2, 15
If V=3000 Then
EDREDioX1.Write 2, 0
End If
Next V

For H=0 To 5000 Step 1
EDREDioX1.Write 1, 63
If H=5000 Then
EDREDioX1.Write 1, 63
End If
Next H

For V1=0 to 1000 Step 1
EDREDioX1.Write 2, 63
If V1=1000 Then
EDREDioX1.Write 2, 0
End If
Next V1

For G1=0 To 32 Step 1
EDREDioX1.Write 0, 63
If G1=32 Then
EDREDioX1.Write 0, 0
End If
```

```
Next G

For V2=0 To 1000 Step
EDREDioX1.Write 2, 63
If V2=500 Then
EDREDioX1.Write 2, 0
End If
Next V2

For H=0 To 2000 Step 1
EDREDioX1.Write 1, 15
If H=2000 Then
EDREDioX1.Write 1, 15
End If
Next H

End Sub

Private Sub cmdInit1_Click()
'G is the value counting the gripper motor rotations
For G=0 To 18 Step 1
EDREDioX1.Write 0, 63
txtG=Val(G)
If G=18 Then
EDREDioX1.Write 0, 0
End If
Next G
End Sub
```

```
Private Sub cmdInit2_Click()

'G is the value counting the gripper motor rotations

For G=0 To 16 Step 1

EDREDioX1.Write 0, 15

txtG=Val(G)

If G=16 Then

EDREDioX1.Write 0, 0

End If

Next G

End Sub


Private Sub cmdInit3_Click()

EDREDioX1.Write 2, 15

End Sub


Private Sub cmdInit4_Click()


EDREDioX1.Write 1, 63

End Sub


Private Sub cmdInit5_Click()

EDREDioX1.Write 1, 15

End Sub


Private Sub cmdReset_Click()

EDREDioX1.Write 0, 0

EDREDioX1.Write 1, 0

EDREDioX1.Write 2, 0
```

```
End Sub

Private Sub Form_Load()
SerialNo=EDREUtlX1.SelectDialog
EDREDioX1.SerialNumber=SerialNo
txtSerial.Text=Format(SerialNo)
BTBox.Text=EDREUtlX1.BoardType
BNBox.Text=EDREUtlX1.DriverVersion
Check Board
End Sub

Private Sub CheckBoard()
If EDREUtlX1.DIOPorts=0 Then
MsgBox “This board has got no Digital I/O ports”, vbExclamation
Unload Me
End If
End Sub

Option Explicit
Dim SerialNo As Long
Dim Major As Long
Dim Minor As Long
Dim bld As Long
Dim opsys As Long
Dim nd As Long

Private Sub btnread_Click()
Txtnumrd.Text=Format(EDREDioX1.Read(Val(txtportrd.Text)))
```

```
End Sub

Private Sub btnwrite_Click()
EDREDioX1.Write Val(txtport.Text), Val(txtnum.Text)
End Sub

Private Sub cndEmergencyStop_Click()
EDREDioX1.Write 6, 0
EDREDioX1.Write 7, 0
EDREDioX1.Write 8, 0
End Sub

Private Sub cndGripperClamping_Click()
EDREDioX1.Write 6, 1                    ‘Gripper Clamping
End Sub

Private Sub cmdGripperLoosing_Click()
EDREDioX1.Write 6, 3                    ‘Gripper Loosing
End Sub

Private Sub cmdGripperStop_Click()
EDREDioX1.Write 6, 0                    ‘Gripper Stop
End Sub

Private Sub cmdHORbackward_Click()
EDREDioX1.Write 7, 3                    ‘Horizontal Backward
End Sub
```

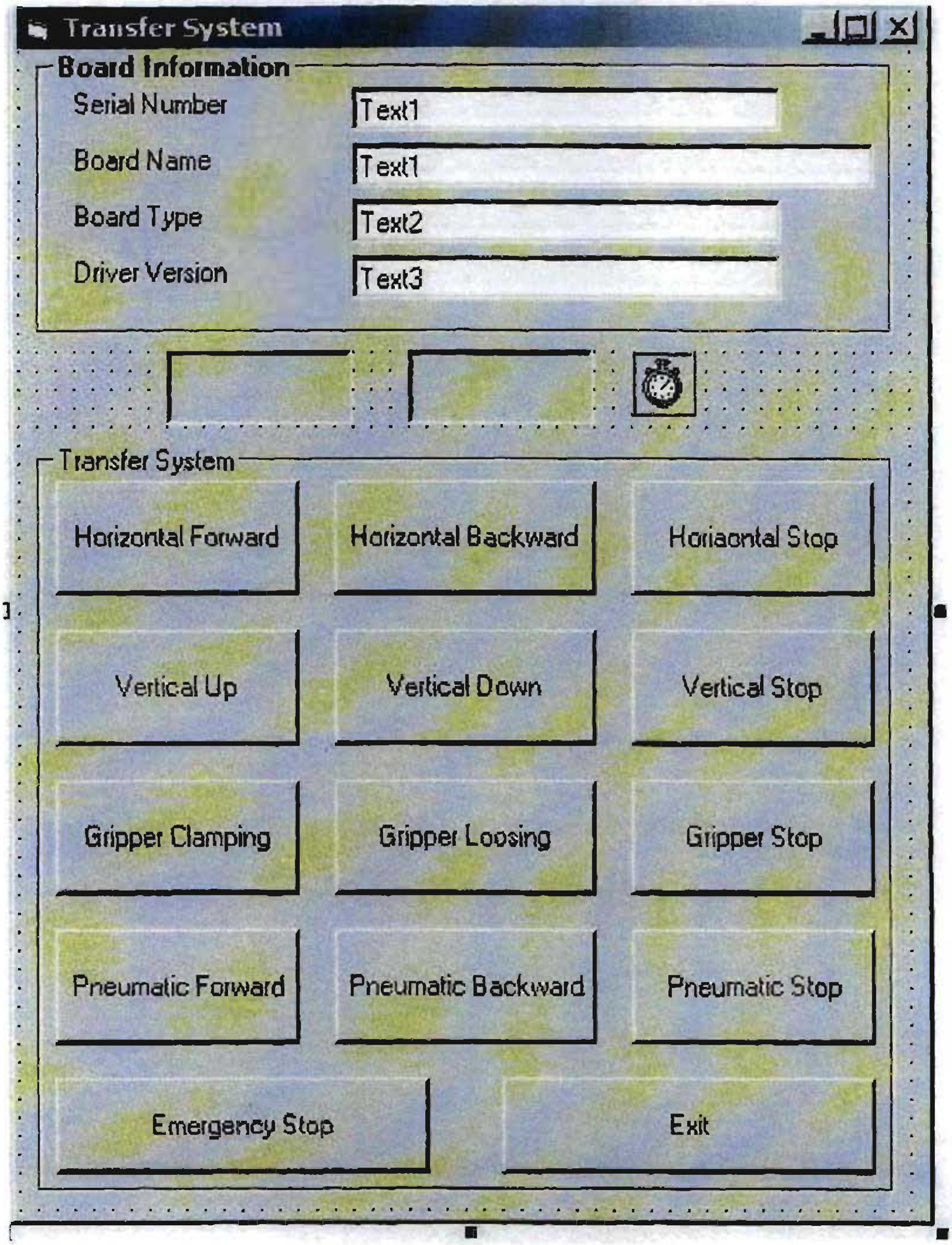
```
Private Sub cmdHORforward_Click()
EDREDioX1.Write 7, 1                          'Horizontal Forward
End Sub

Private Sub cmdHORstop_Click
EDREDioX1.Write 7, 0                           'Horizontal Stop
End Sub

Private Sub cmdPneumaticForward_Click()
EDREDioX1.Write
End Sub

Private Sub cmdVERdown_Click()
EDREDioX1.Write 7, 3                            'Vertical Down
End Sub

Private Sub cmdVERsto_Click()
EDREDioX1.Write 7, 0                            'Vertical Stop
End Sub

Private Sub cmdVERup_Click()
EDREDioX1.Write 7, 1                            'Vertical Up
End Sub

Private Sub Form_Load()
SerialNo=EDREDUtlX1.SelectDialog
EDREDioX1.SerialNumber=SerialNo
Txtserial.Text=Format(SerialNo)
```

```
BTBox.Text=EDREUtlX1.BoardType
BNBox.Text=EDREUtlX1.BoardName
DVBox.Text=EDREUtlX1.DriverVersion
CheckBoard
End Sub

Private Sub CheckBoard()
If EDREUtlX1.DIPorts=0 Then
MsgBox”This board has got no Digital I/O ports”, vbExclamation
Unload Me
End If
End Sub

Option Explicit
Dim SerialNo As Long
Dim Major As Long
Dim Minor As Long
Dim bld As Long
Dim opsys As Long
Dim nd As Long

Private Sub btnread_Click()
Txtnumrd.Text=Format(EDREDioX1.Read(Val(txtportrd.Text)))
End Sub

Private Sub btnwrite_Click()
EDREDioX1.Write Val(txtport.Text), Val(txtnum.Text)
End Sub
```

```
Private Sub cmdemergencystop_Click()

EDREDioX1.Write 3, 0

EDREDioX1.Write 4, 0

EDREDioX1.Write 5,0

End Sub


Private Sub cmdXbackward_Click()

EDREDioX1.Write 3, 3

End Sub


Private Sub cmdxforward_Click()

EDREDioX1.Write 3, 1                    'X-aixs Forward

End Sub


Private Sub cmdXsotp_Click()

EDREDioX1.Write 3, 0                    'X-axis Stop

End Sub


Private Sub cmdybackward_Click()

EDREDioX1.Write 4, 3                    'Y-axis Forward

End Sub


Private Sub cmdYstop_Click()

EDREDioX1.Write 4, 0                    'Y-axis Stop

End Sub


Private Sub cmdZbackward_Click()

EDREDioX1.Write 5, 3                    'Z-axis Forward
```

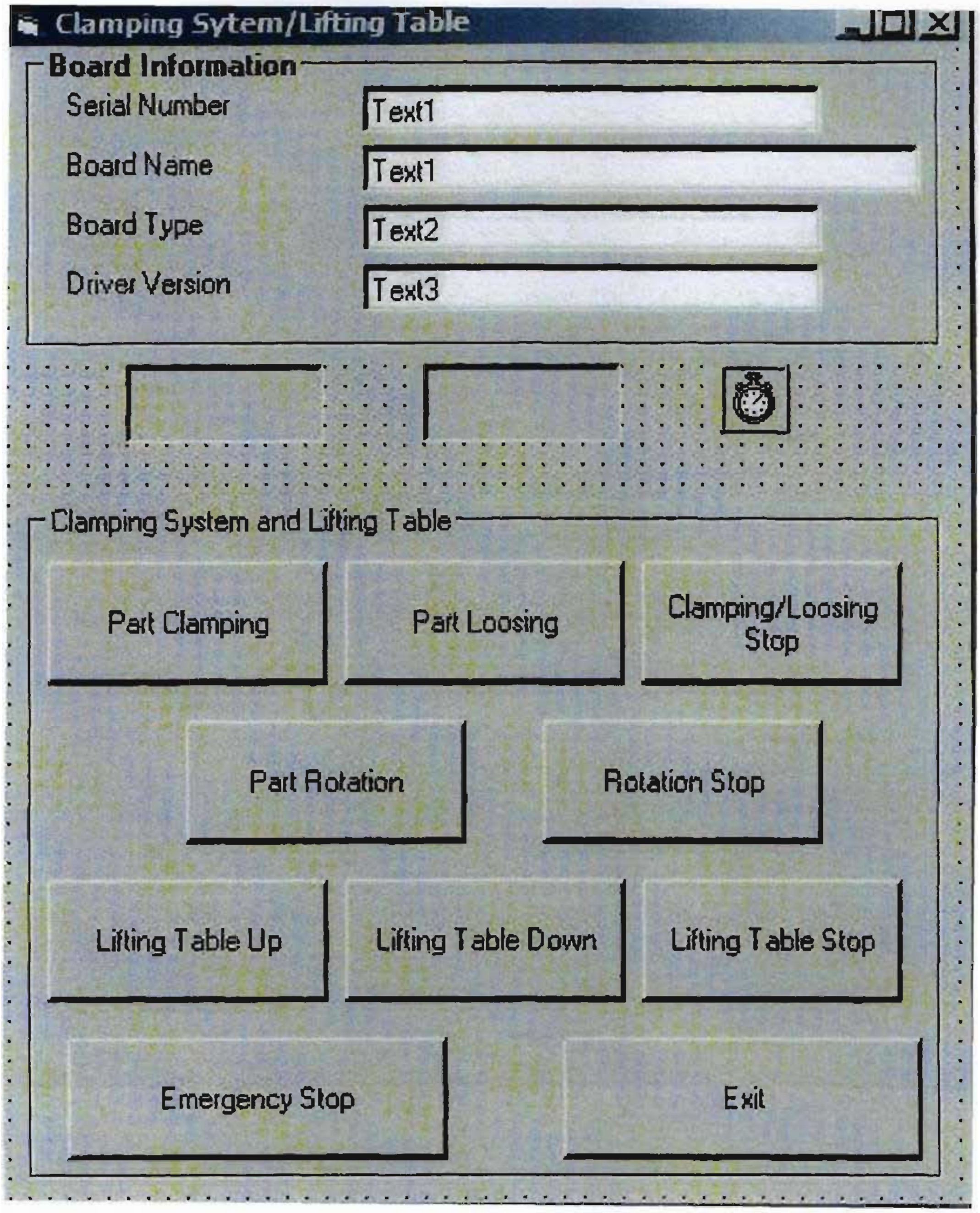
```
End Sub

Private Sub cmdZstop_Click()
EDREDioX1.Write 5, 0                              ‘Z-axis Stop
End Sub

Private Sub Form_Load()
SerialNo=EDREUtlX1.SelectDialog
EDREDioX1.SerialNumber=SerialNo
Txtserial.Text=Format(SerialNo)
BTBox.Text=EDREUtlX1.BoardType
BNBox.Text=EDREUtlX1.BoardName
DVBox.Text=EDREUtlX1.DriverVersion
CheckBoard
End Sub

Private Sub CheckBoard
If EDREUtlX1.DIOPorts=0 Then
MsgBox”This board has got no Digital I/O ports”, vbExclaimation
Unload Me
End If
End Sub

Private Sub Command1_Click()
TransferSytem.Show
End Sub

Private Sub cmdClampingSytem_Click()
```

```
ClampingSystem.Show

End Sub


Private Sub cmdExit_Click()

Unload Me

End Sub

Private Sub cmdToolChangingSytem_Click()

ToolChangingSystem.Show

End Sub


Private Sub cmdTransferSystem_Click()

TransferSystem.Show

End Sub


Private Sub FrameXYZSystem_Click()

FrameSystem.Show

End Sub
```



Serial Number: Text1
Board Name: Text1
Board Type: Text2
Driver Version: Text3

Figure C-1 APTS Control Interface.



Clamping Sytem/Lifting Table

Board Information

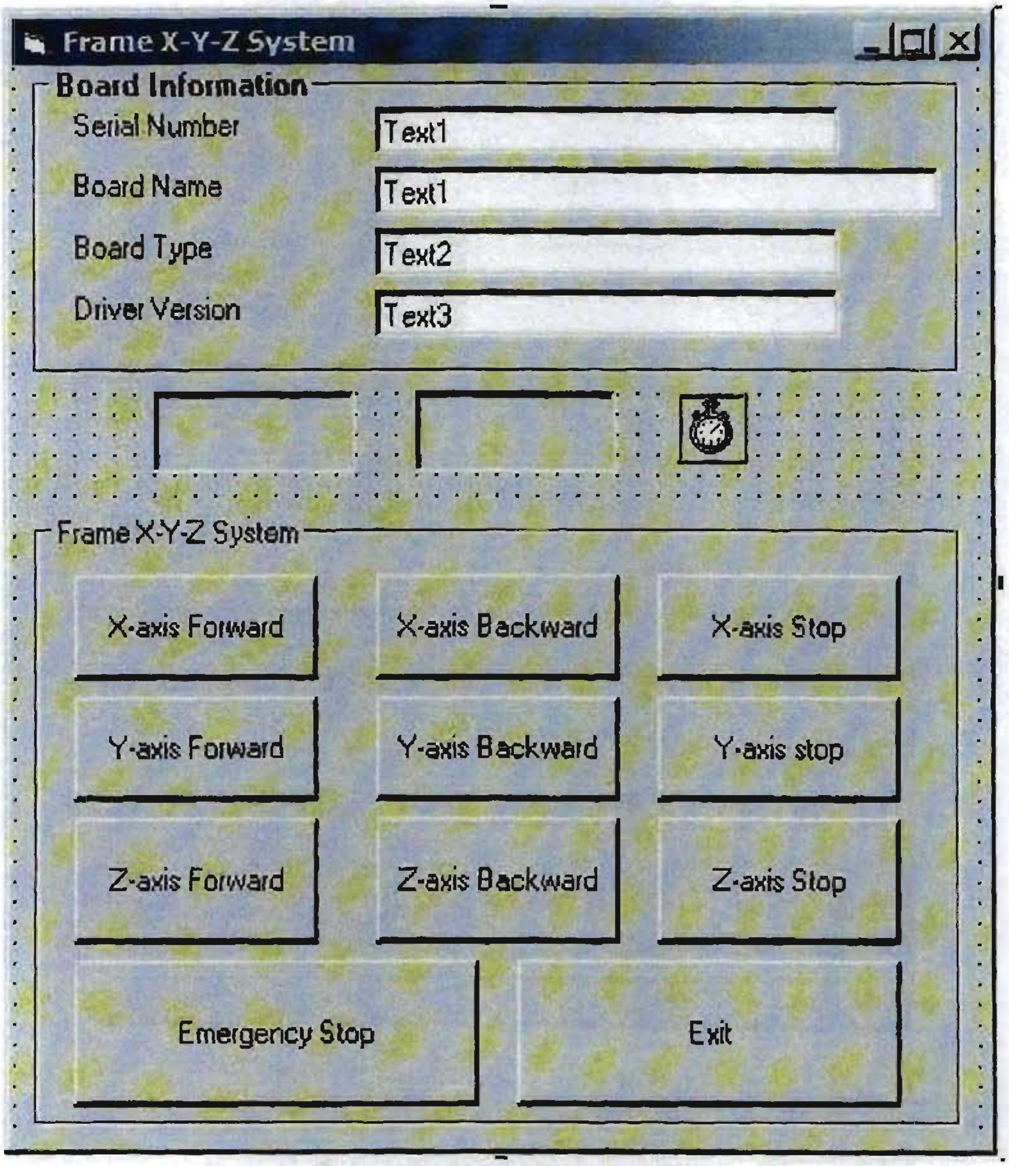
Serial Number: Text1

Board Name: Text1

Board Type: Text2

Driver Version: Text3

Figure C-2 APC/RS Control Interface.



Serial Number	Text1
Board Name	Text1
Board Type	Text2
Driver Version	Text3

Figure C-3 ATCS Control Interface.

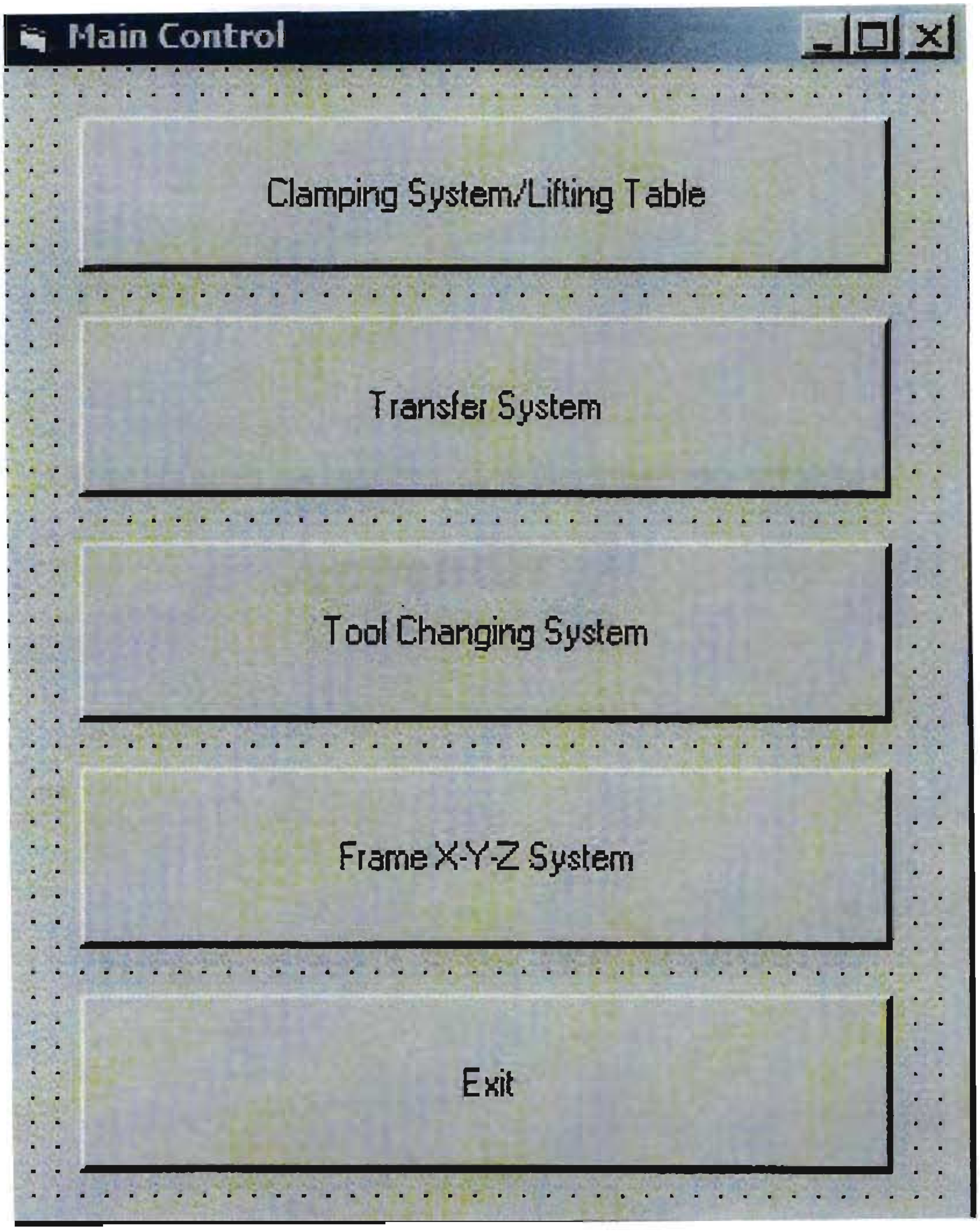


Figure C-4 RMM Main Control Interface.

Appendix D:

Interfacing using the standard printer adapter

This appendix details the development of a RMM modular motion control system using the standard printer available on all computers. This section first details the structure of the printer adapter and reviews the techniques required to interact with the printer adapter. Finally the application of the printer adapter in a RMM modular motion control system is presented.

The standard printer, or parallel port, can be used to effectively implement a PC-based control system. The parallel port hardware system is able to transfer information from the transducers to the CPU. All input and output signals are digital signals based on the Transistor-Transistor-Logic (TTL) format (see Figure D-1).

When one initially approaches the field of parallel interfacing, it is important to consider that the parallel interfacing system was designed specifically to enable interaction between the computer and a parallel printer. This allows one to easily comprehend the reasons for the very structure of the standard printer adapter and its corresponding software addresses.

When the port is used to control a printer, the primary function of the printer is to produce a hardcopy of selected information in the CPU's memory. The CPU sends this information to the printer character by character. Each character is described uniquely by an eight digit binary number, termed a byte, determined from the American Standard Code for information standard TTL voltage levels for high (1) and low (0) states.

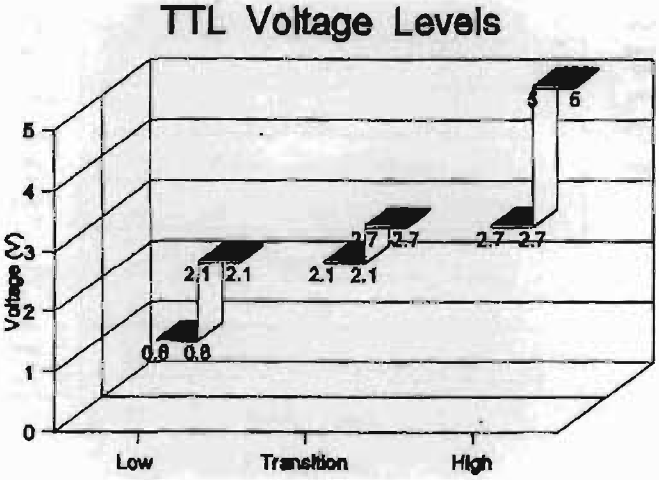


Figure D-1 Graph showing the voltage levels of the two binary states, high and low, of the standard TTL format.

Before these signals can be sent to the printer, the computer needs to ascertain information regarding the status of the printer. The printer must be able to present this information to the CPU via the parallel port.

The parallel port achieves the highest information transfer rate of all available interfacing systems by using eight dedicated data lines to send the eight signals corresponding to one character of byte to the printer simultaneously, hence the name parallel interfacing. Specific software blocks termed printer drivers are developed that control the printer and load the output data into the latched 8-bit output port. The strobe line is activated to coordinate the flow of information to the printer and two input ports can be read to monitor printer status (see Figure D-2).

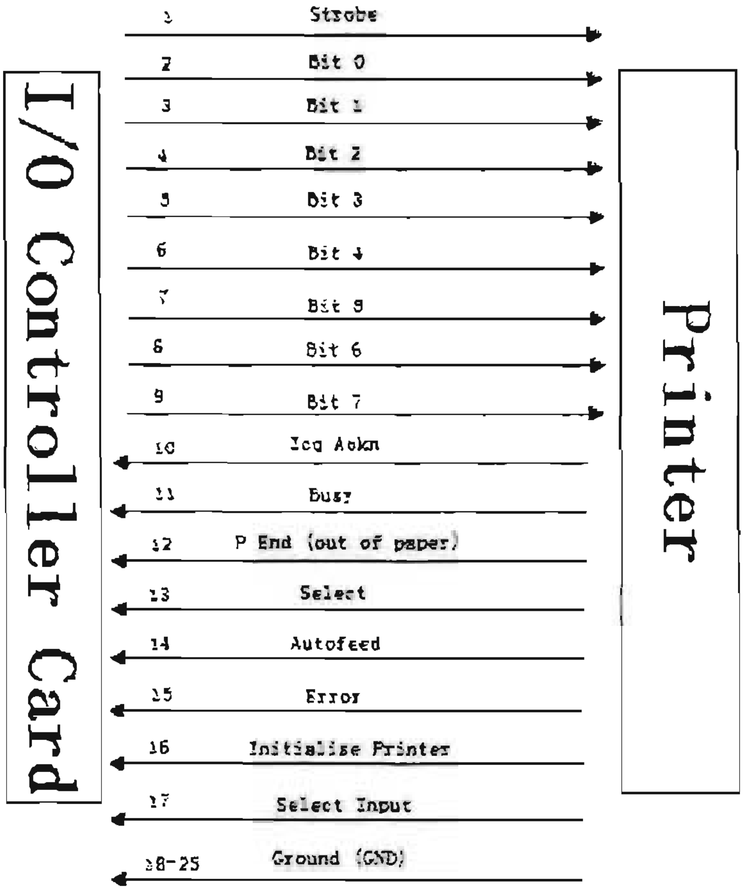


Figure D-2 The standard computer-printer interface protocol. The typical signal names and corresponding pin numbers are indicated.

In parallel interfacing systems the digital signals are present to the CPU via standard parallel port hardware comprising of the Input/Output (I/O) controller card. The I/O card transfers the data presented at the pins of the D-type connector plug to the computer data bus (see Figure D-3).

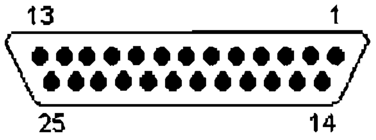


Figure D-3 The standard pin layout for the 25-pin, D-type connector.

D.1. PARALLEL PORT SUB-STRUCTURES

The I/O controller card or printer adapter is specifically designed to attach printers with a parallel port interface; however it can also be used as a general I/O for any device of application that matches the adapter's capabilities. The printer adapter comprises of 12 TTL-buffer, latched output points, 5 steady state input points and 7 ground connections. The 12 outputs are able to have information written to or information read from them using the CPU'S input or output instruction. The 5 steady state input points can also have commands read from them using the CPU's input command. The INPUT or OUTPUT commands can be evoked using any suitable software language, for example C++, and need not be directly coded into assembly language.

The structure of the output and input points of the parallel port can be subdivided into 3 sub-ports each having their own unique hexadecimal address. The printer adapter driver structure of each of the sub-ports and indicates the bit number and corresponding pin numbers of the 8 signals that can be written to or read from the particular sub-port.

The hexademical address of Port I provide the base address for the entire parallel and define the address of the remaining two sub-ports. The address is usually 378 or 3BC depending on the I/O card controlling the port. If the card is a standard printer adapter card the base port address is 378, however if the controller card is in the format of an IBM Monochrome display and printer adapter card, the address of the base port is 3BC. The home address of the parallel port normally is given on the system status display shown on the initial power up of the standard desktop computer.

Port I: Base Port

The base port can respond to both the IN and OUT assembler commands. The OUT instruction captures an 8-bit data word from the computer data bus and presents this data to the pins of the D-type connector plug (see Figure D-3). The pin-bit correspondence is shown in Table D-1. Each pin is capable of sourcing 2.6 mA, 5 V and sinking 24 mA to ground. It is imperative that the interface system design does not attempt to drive these pins to ground.

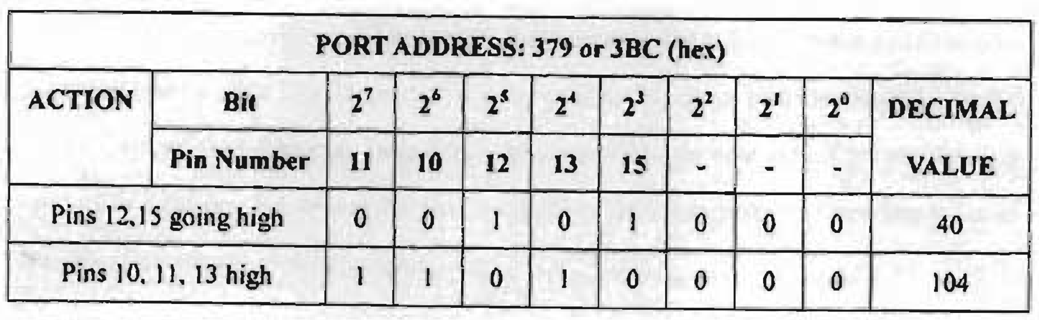
The IN presents the processor with an 8-bit word that represents the data present on the pins associate with the base port. This data should reflect the exact value that was last written to the base port latch. If this is not the case than a fault in the interface system has caused one or more of these pins to be externally driven. The input data from the IN command is ORed with the last value of the latch resulting in the fault pins remaining in the high state. The implementation of the IN command should only be used for fault diagnosis loops.

PORT ADDRESS: 378 or 3BC (hex)										
ACTION	Bit	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	DECIMAL VALUE
	Pin Number	9	8	7	6	5	4	3	2	
Pins 2,3,7 going high		0	0	1	0	0	0	1	1	35
Pins 3,4,5 high		0	0	0	0	1	1	1	0	14

Table D-1 The pin-bit correspondence of the base port 378 or 3BC.

Port II: Base+1

Port II contains five steady-state input points that can be used to monitor the real-time status of the five input pins given in Table D-2. This port is a uni-directional port that can only respond to the assembler IN command. The IN command presents the real time status of the five points to the processor. Note that the command captures an 8-bit word but only considers the five most significant digits.



PORT ADDRESS: 379 or 3BC (hex)										
ACTION	**Bit**	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	**DECIMAL VALUE**
	Pin Number	11	10	12	13	15	-	-	-	
Pins 12,15 going high		0	0	1	0	1	0	0	0	40
Pins 10, 11, 13 high		1	1	0	1	0	0	0	0	104

Table D-2 The pin-bit correspondence of port II.

The use of pin 10 must be carefully considered during the design stage of the interface system. The pin can be enabled as an interrupt acknowledge pin that interrupts the processor on every high-to-low transition of the data on pin 10. The interrupt enable command is set using port III, and is discussed in more detail under that section.

Port III: Base + 2

The base + 2 port is the only true bidirectional port of the standard printer adapter. The port can therefore respond to both the assembler IN and OUT commands. The OUT command is used to set or enable the interrupt request on pin 10. The corresponding pins shown in Table D-3 are driven by open collector drivers pulled to + 5 V dc through 4.7 $K\Omega$ resistors. The pins are individually capable of maintaining a 0.8 V dc down level and can sink in the region of 7 mA.

The OUT command causes the latch to capture the five least significant bits of an 8-bit work written to the data bus. The four least significant bits present their outputs or inverted outputs (as indicated by the double underlined pins in Table D-3), to the corresponding pins. The fifth least significant bit is used to set the interrupt request (IRQ) function of pin 10. If the bit is high, then the interrupt request is enabled and the processor will interrupt on every high-to-low transition of the signal applied at pin 10.

The IN command presents an 8-bit word to the processor that indicates the state of the five pins indicated in table D-3. If the port is being used as an output port, this function can be used to perform fault diagnosis loops in the same manner as the base port. If the port is being used as an input port the state of the pins is externally driven and the corresponding value of the pins is transferred to the processor.

The status of port III is reset whenever the processor is reset. The reset value presented to the port sets every pin to the low condition, as shown in Table D-3.

PORT ADDRESS: 37A or 3BC (hex)										
ACTION	**Bit**	**2^7**	**2^6**	**2^5**	**2^4**	**2^3**	**2^2**	**2^1**	**2^0**	**DECIMAL VALUE**
	Pin Number					**17**	**16**	**14**	**1**	
Pins 1, 14, 17 going high					0	0	0	0	0	11
Pins 1 high, IRQ enabled					1	0	0	1	0	17
Processor reset					0	0	0	0	0	

Table D-3 The pin-bit correspondence of port III. The double underlined pins indicate an inverting output. Note that pins 1, 14 and 17 are NOTed.

D.2. SOFTWARE CONSIDERATION

In the design of a parallel interfacing system it is important that the designer remembers that all outputs are software generated and that all inputs are real-time signals that are not latched.

Although ports I and III are in fact bidirectional ports, the designer should consider the designed direction and purpose of these ports. Port I, the base port, was designed to be used as the data port; control signals should be sent from this port and the devices that are being driven off this port should never attempt to pull any of these pins to ground. Port III was designed to returned information to the computer. Parallel interfaced systems should be designed in keeping with these direction standards; the direction of these ports should only be changed in extreme circumstances.

D.3. APPLICATION OF DRIVER SOFTWARE

This section details the development of the driver software code that can be used to control motors and decipher feedback information. The example software codes have been written in *Borland® Turbo Pascal* 7. Turbo Pascal is a clear, well structured language that incorporates an intelligent that allows the use of additional units that are separately compiled thereby optimizing the size of the application code and thus its compile and execution time [22]. Two demonstration programs have been included to illustrate the functioning of the driver software algorithms. The section first details the control of DC motors and then explains the deciphering of feedback signals.

The control of the motors

The power requirements of DC motors are far greater than that which can be supplied directly from the parallel port. The motors require a constant 12 Vdc supply providing between 7 A – 13 A. Specifically designed driver circuit are obtainable that are capable of controlling the power supplied to each of the motors using three standard TTL based signals to switch relays on the motor power circuit. A schematic diagram of a typical driver circuit, illustration the connection for the control circuit and the motor power circuit is presented in Figure D-4.

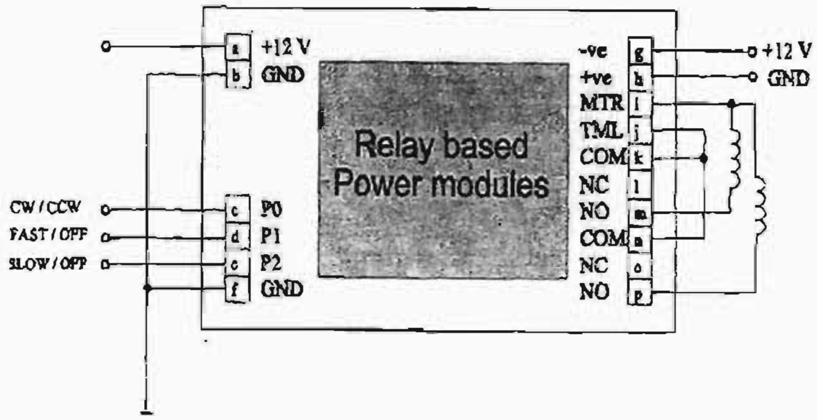


Figure D-4 Schematic diagram illustrating the connections for the control circuit and motor power circuit.

A demonstration program has been included to show the software algorithm used in the main program to control the motors. However, the main program will not require prompting to perform the control action on the motor but will perform these actions in accordance with the control logic defined in the software code.

```
Program MOTORV
        uses crt, docs;
        var stop: boolean;
            mcont: char;
            pinset: integer;
        begin
            port[$378]:=0;
            repeat
                 clrscr;
                 stop:=false;
                 writeln;
                 writeln(“1”Start motor turning’);
                 writeln(“2”Stop motor turning’);
                 writeln(“3”Set clockwise rotation’);
                 writeln(“4”Start anticlockwise rotation’);
                 writeln(“5”Exit the demonstration’);
                 writeln;
                 writeln;
                 write(‘ENTER selection ->’);
                 mcont:=readkey;
                 case mcont of
                           ‘1’:pinset:=1; {drives pin 2 high: #00000001b}
                           ‘2’:pinset:=1; {drives pin 2 low: #00000000b}
                           ‘1’:pinset:=1; {drives pin 2,3 high:#00000011b}
                           ‘4’:pinset:=1; {drives pin 2 high:#00000001b}
                           ‘5’:stop:=true;
                 end;
                 port[$378]:=pinset; {378 is address of base port}
                 until stop=true;
         wirteln(‘Press Enter to Continue’);
         readln;
        End.
```

The program begins by initializing the base by driving all pins low, “*port [$378]:=0*”, and then defines the user interface. The program then uses the ease statement, (*case mcont of ... end*), to set the values of the pins for the different control action. The value of the variable “*pinset*” is determined using Table D-1, as shown below I Table D-4.

<table>
<tr><th colspan="11">PORT ADDRESS: 378 or 3BC (hex)</th></tr>
<tr><th rowspan="2">ACTION</th><th>Bit</th><th>2^7</th><th>2^6</th><th>2^5</th><th>2^4</th><th>2^3</th><th>2^2</th><th>2^1</th><th>2^0</th><th rowspan="2">DECIMAL VALUE</th></tr>
<tr><th>Pin Number</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th></tr>
<tr><td colspan="2">Set pin 2 high</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td colspan="2">Set pin 2 low</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="2">Set pin 2 high, 3 low</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td colspan="2">Set pin 2,3 high</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>3</td></tr>
</table>

Table D-4 Table showing the calculation of the decimal values which need to be written to the parallel port to initiate the desired control action.

Pins 2 and 3 are hard wired to terminals P1 and P0 respectively on the DC servomotor driver circuit. The program then writes the decimal value to the port, in accordance with the user defined control action, using the command "*port [$378]:=pinset*". The appropriate transistor-relay circuits are energized by the control signals from the port and the motor power circuit is able to drive and control the servomotor.

The algorithm used to read in information through the parallel port is also based on the "*port*" command. The implementation of the command to read information from the input ports is best described using a demonstration program.

Position feedback information

The algorithm used to read in information through the parallel port is again best described using a demonstration program.

```
program RDSWITCH;

uses crt, docs;

var pinchk, pin1.count: integer;
    stop: boolean;
    x: char;
begin
 clrscr;
 writeln('Press any key to begin switch count');
 readln;
 stop:=false;
 writeln('Press "s" to stop counting');
    repeat
      pinchk:=port[$37A];          {read the value of the pin 1 in port III}
      pin1:=pinchk and 1;          {check if pin 1 is low}
      count:=0;
      writeln;
      if pin1=1 then begin
         count:=count+1;
         repeat
            pinchk:=port[$37A];    {keep checking the port until}
            pin1:=pinchk and 1;    {pin 1 goes low};
         until pin1=0
         end;
      x:=readkey;
      if x= 's' then stop:=true;
until stop=true
writeln('the switch was closed', count, 'times');
write('Press and key to exit');
readln;


End.
```

The program checks the status of the port by assigning its corresponding decimal value to the variable *"pinchk"*. The decimal value read from the port 37A is a decimal representation of the binary number that indicates the states the state of the individual port pins. As illustration example showing the pin state significance of the decimal value read from the port using the command *"pinchk:=port[$37A]"* is presented in Table D-5.

PORT ADDRESS: 37A or 3BC (hex)										
ACTION	**Bit**	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	**DECIMAL VALUE**
	Pin Number	-	-	-	-	$\overline{17}$	16	$\overline{14}$	$\overline{1}$	
Pin 1 low		-	-	-	-	0	0	0	0	1

Table D-5 Illustrative example showing the decimal value read from the parallel port if the switch is closed. Note that pins 1, 14 and 17 are NOTed.

The two techniques described above are used to implement the control algorithms of the individual motions systems of the RMM modular motion controller.

Appendix E

Linear Bearing Specifications

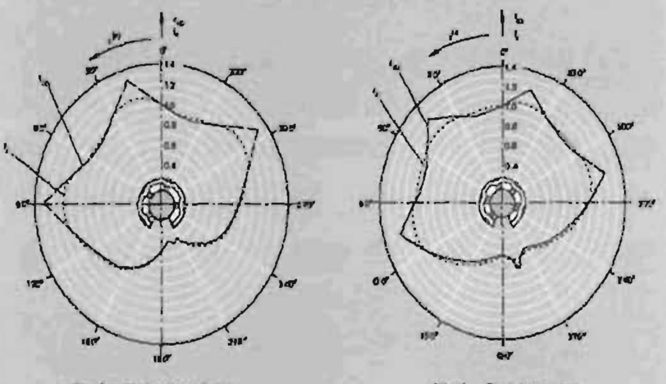
The specifications of linear bearing used in the RMM have been included in this appendix [47].

Linear Sets, R1067
open type

Linear Sets, R1068
open type, adjustable

Structural design
- Precision housing (spheroidal graphite cast iron)
- Retention by means of locating screw
- Standard Linear Bushings with seals

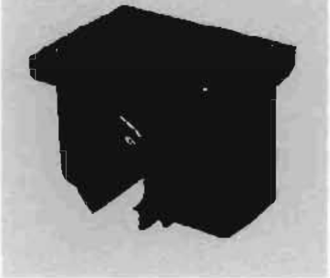
Load capacity factors



Shafts Ø d 20 and 25

Shafts Ø d 30 to 80

open type

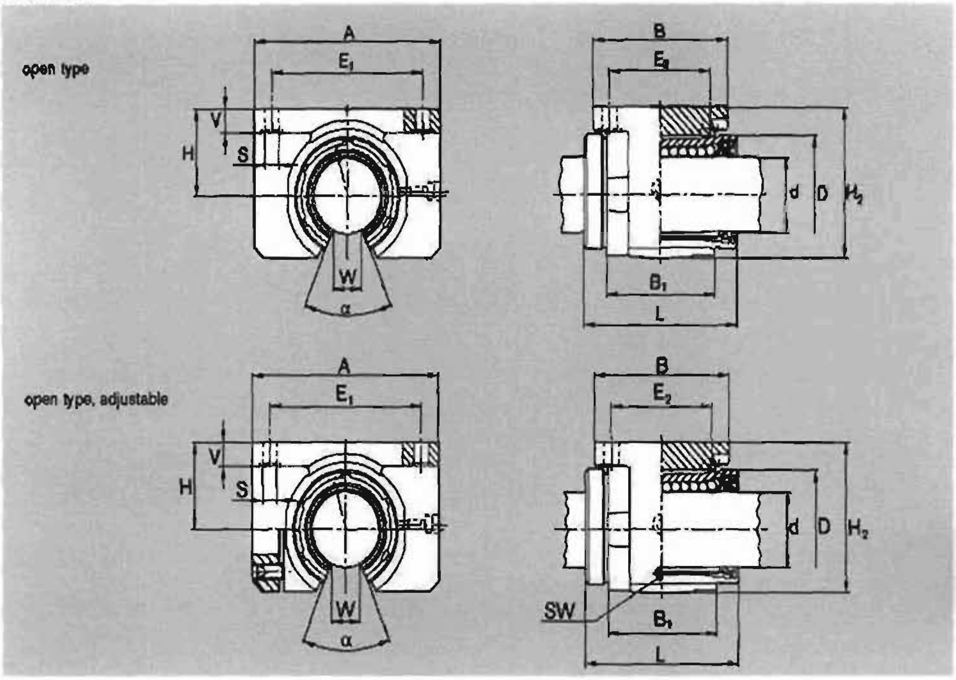


Shaft Ø d (mm)	Part numbers with two seals	Mass (kg)
20	R1067 220 00	0.39
25	R1067 225 00	0.74
30	R1067 230 00	1.14
40	R1067 240 00	2.25
50	R1067 250 00	3.13
60	R1067 260 00	5.76
80	R1067 280 00	13.15

open type, adjustable

Shaft Ø d (mm)	Part numbers with two seals	Mass (kg)
20	R1068 220 00	0.38
25	R1068 225 00	0.74
30	R1068 230 00	1.12
40	R1068 240 00	2.20
50	R1068 250 00	3.11
60	R1068 260 00	5.72
80	R1068 280 00	13.09

Dimensions



Dimensions (mm)														Angle [°]	Radial clearance [µm]			Tolerance [µm] for dimension	Load capacities[5] [N]	
Ød	D	H	H_2[2]	L	A[2]	B[2]	B_1	E_1	E_2	S	V[2]	W[3]	SW	α	R1067 with Shaft h6	h7	R1068	H[4]	dyn. C	stat. C_0
20[1]	32	25	42	45	60	42	28	45±0.15	32±0.15	4.5	8	10	2.5	60	+36 +4	+42 +6	Adjusted to zero-clearance on h6 shaft (lower limit) prior to delivery when screwed down	+5 -19	1280	970
25[1]	40	30	51	58	74	54	40	60±0.15	40±0.15	5.5	9	12.5	3	60	+39 +4	+44 +5		+5 -19	2270	1750
30	47	35	60	68	84	60	48	68±0.20	45±0,20	6,6	10	12,5	3	50	+38 +4	+44 +6		+5 -19	2690	2390
40	62	45	77	80	108	78	56	86±0.20	58±0,20	9	12	16.8	4	50	+45 +5	+52 +7		+4 -21	5280	4000
50	75	50	88	100	130	70	72	108±0.20	50±0.20	9	14	21	5	50	+45 +5	+52 +7		+8 -25	8470	6900
60	90	60	105	125	160	92	95	132±0.25	65±0.25	11	15	27.2	6	54	+50 +5	+59 +7		+8 -26	11800	9780
80	120	80	140	165	200	122	125	170±0.50	90±0.25	13.5	22	36.3	8	54	+54 +6	+62 +9		+7 -28	21600	17400

[1] Contrary to the illustration, the locating screw is on the adjusting side in these sizes.
[2] Tolerance DIN 1685-GTB 15.
[3] Minimum dimension based on shaft diameter d.
[4] When screwed down, relative to shaft nominal dimension d.
[5] The load capacities apply when the load is acting along the line $\varrho = 0°$.

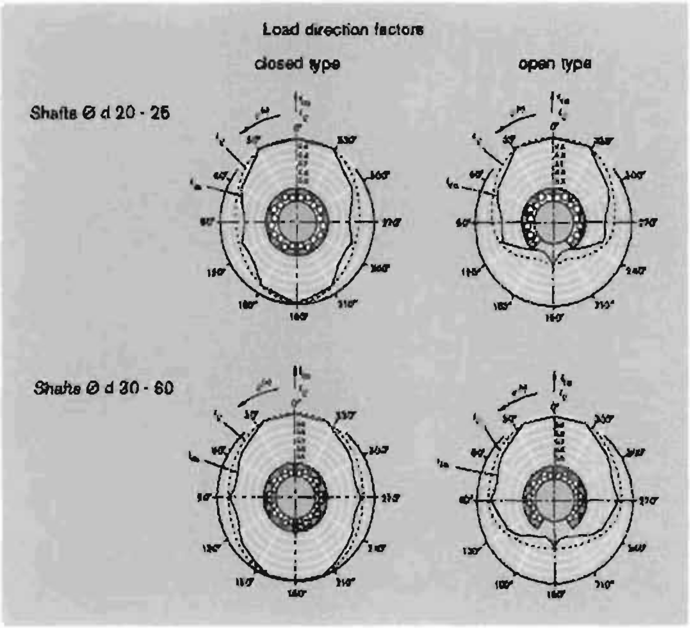
The figures for dynamic load-carrying capacity have been calculated assuming a nominal travel of 100,000 m. For a nominal travel of 50.000 m, the "C" figures in the table must be multiplied by a factor of 1.26.

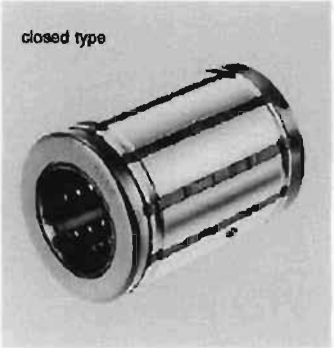
Super Linear Bushing, R0730
closed type

Super Linear Bushing, R0731
open type

Structural design
- Polyacetal ball retainer
- Hardened steel segmental load bearing plates with ground ball tracks and ground outer surfaces
- Two metal holding rings
- With or without twin-lip seal rings
- With or without axial seal strip.

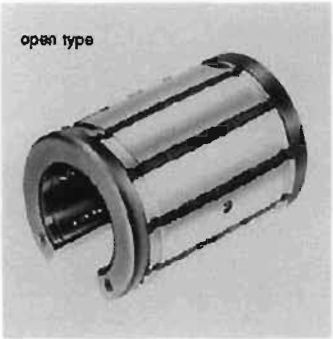
For precise values for the 4 main directions of load see "Technical Data - Load capacity factors".

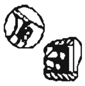



closed type

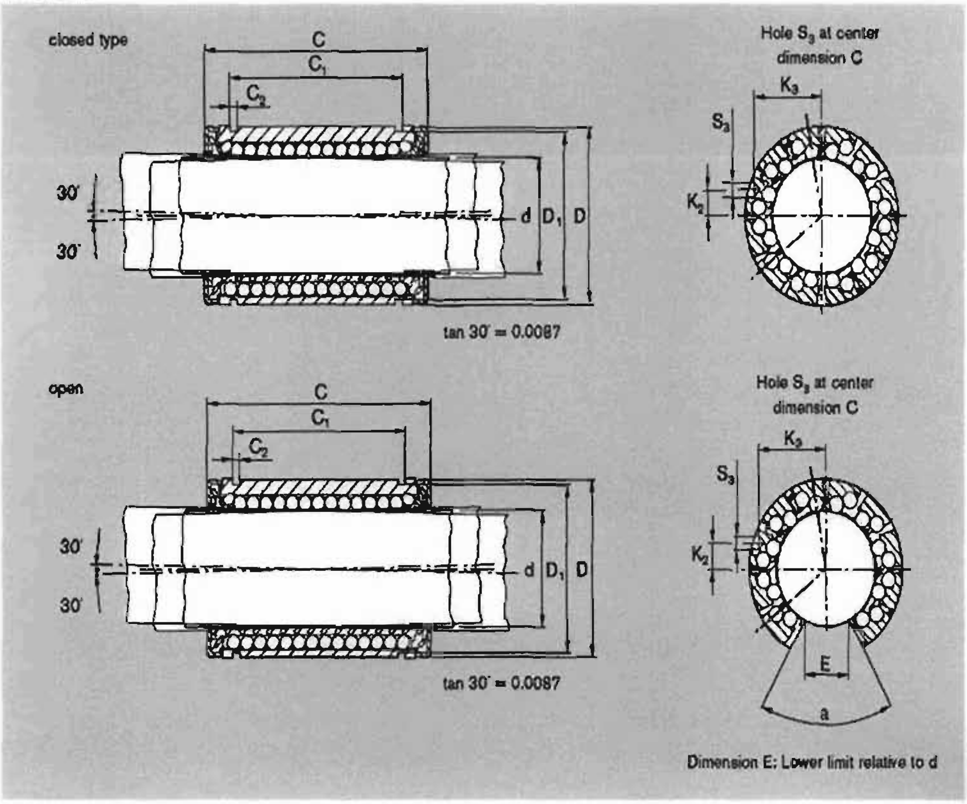
Shaft Ø d [mm]	Part numbers without seal	 with two seals	Mass [kg]
20	R0730 020 00	R0730 220 40	0.090
25	R0730 025 00	R0730 225 40	0.190
30	R0730 030 00	R0730 230 40	0.300
40	R0730 040 00	R0730 240 40	0.600
50	R0730 050 00	R0730 250 40	1.050
			

With one seal: R0730 1.. 40 oder R0731 1.. 40.


open type

Shaft Ø d [mm]	Part numbers without seal	 with two seals	 fully sealed	Mass [kg]
20	R0731 020 00	R0731 220 40	R0731 220 45	0.075
25	R0731 025 00	R0731 225 40	R0731 225 45	0.160
30	R0731 030 00	R0731 230 40	R0731 230 45	0.250
40	R0731 040 00	R0731 240 40	R0731 240 45	0.500
50	R0731 050 00	R0731 250 40	R0731 250 45	0.900
				

Dimensions



Dimensions [mm]										No. of ball circuits		Angle	Radial clearance [µm]				Load capacity[1] [N]	
Ø d	D	C	C_1	C_2	D_1	S_3	K_2	K_3	E			α	Shaft/bore					
		h13	H13			+0.1				○	◌	[°]	h7/H7	h7/JS7	h6/JS6	h6/K6	dyn. C	stat. C_0
20	32	45	31.2	1.6	30.5	2.8	1.3	14.7	9.5	10	8	60	+49 +13	+37 0	+28 +1	+23 -4	3530	2530
25	40	58	43.7	1.85	38.5	2.8	2	18.5	12	10	8	60	+49 +13	+37 0	+28 +1	+23 -4	6190	4530
30	47	68	51.7	1.85	44.5	3.6	7	21	12.8	12	10	60	+49 +13	+37 0	+28 +1	+23 -4	8800	7180
40	62	80	60.3	2.15	59	3.6	9.5	27.5	16.8	12	10	60	+57 +14	+42 -1	+31 +1	+26 -4	13500	10400
50	75	100	77.8	2.65	72	4.6	10	33.5	22.1	12	10	60	+57 +14	+42 -1	+31 +1	+26 -4	22300	16800

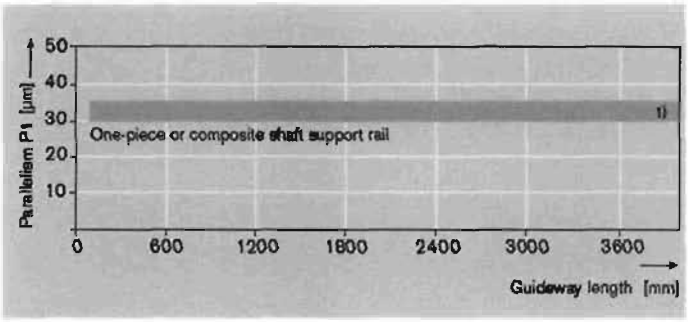
[1] The figures given for load capacity are maximum values as the position and load direction can be precisely defined.

The figures for dynamic load-carrying capacity have been calculated assuming a nominal travel of 100,000 m.
For a nominal travel of 50,000 m, the 'C' figures in the table must be multiplied by a factor of 1.26.

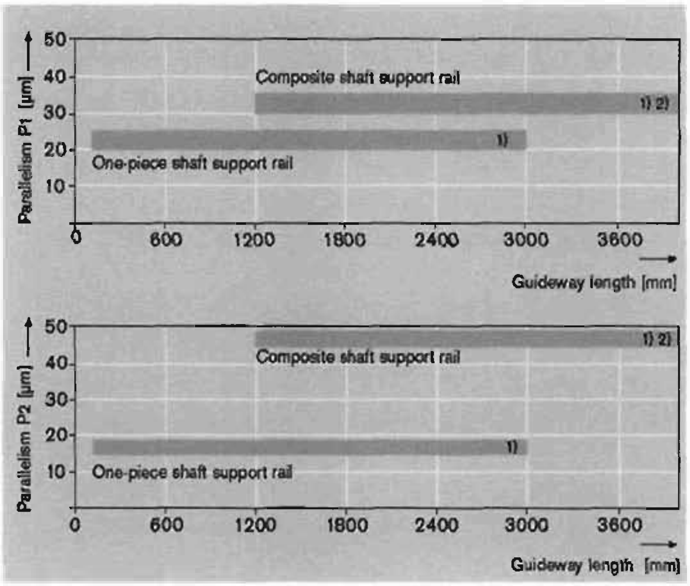
Technical Data

Tolerances, parallelism of guideway in service

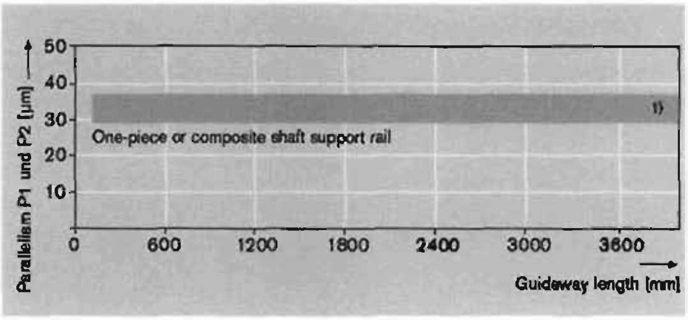
Linear sets R1703, R1704 with shaft support rail R1014 and shaft.



Linear sets R1703, R1704 with shaft support rail R1016 and shaft.

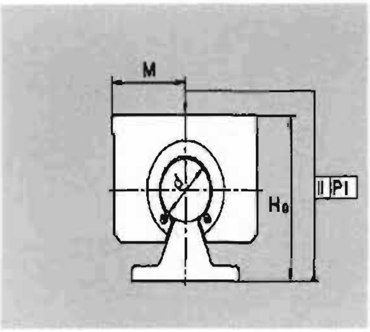


Linear sets R1706 with shaft support rail R1015 and shaft

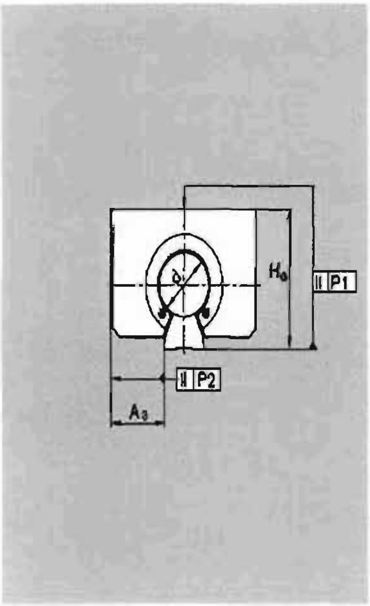


[1] For precise values see "Tolerances" table

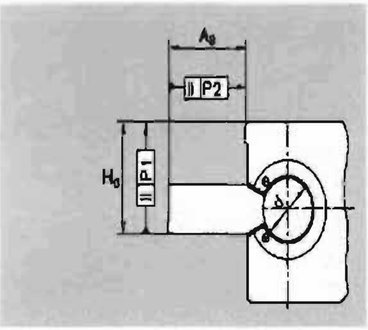
[2] Composite shaft support rail made up of several sections of the same grade



		Tolerance zone of shaft	ShaftØ d [mm]			
			20 25	30	40 50	60
Tolerances[5] [µm]	Dimension H_0[3][6]	h6	+18 -39	+18 -39	+18 -42	+18 -45
		h7	+18 -47	+18 -47	+18 -51	+18 -56
	Parallelism P1[4][6]	h6	30	30	32	33
		h7	32	32	35	35



		Tolerance zone of shaft	ShaftØ d [mm]				
			20	25	30	40	50
Tolerances[5] [µm]	Dimension H_0[3] Several support rails	h6	+28 -69	+28 -69	+28 -69	+28 -72	+28 -72
		h7	+28 -77	+28 -77	+28 -77	+28 -81	+28 -81
	Dimension H_0[3] Single support rail	h6	57	57	57	60	60
		h7	65	66	65	67	69
	Parallelism P1[4] Support rails in a row	h6	30	30	30	32	32
		h7	32	32	32	35	35
	Parallelism P1[4] Single support rail	h6	20	20	20	22	22
		h7	22	22	22	25	25
	Parallelism P2[4] Support rails in a row	h6	45	45	45	46	46
		h7	46	46	46	48	48
	Parallelism P2[4] Single support rail	h6	15	15	15	16	16
		h7	16	16	16	18	18
	Dimension A_3[4]	h6	+30 -37	+30 -37	+30 -37	+30 -38	+30 -38
		h7	+30 -41	+30 -41	+30 -41	+30 -43	+30 -43



		Tolerance zone of shaft	ShaftØ d [mm]				
			20	25	30	40	50
Tolerances[5] [µm]	Dimension H_0[3][6]	h6	+20 -35	+20 -35	+20 -35	+20 -36	+20 -36
		h7	+20 -39	+20 -39	+20 -39	+20 -41	+20 -41
	Dimension A_3[3]	h6	+20 -33	+20 -33	+20 -33	+21 -37	+21 -37
		h7	+20 -41	+20 -41	+20 -41	+21 -46	+21 -46
	Parallelism P1[4][6]	h6	29	29	29	30	30
		h7	30	30	30	32	32
	Parallelism P2[4][6]	h6	29	20	29	34	34
		h7	31	31	31	37	37

[3] Measured at center of housing

[4] When screwed to base mounting surface

[5] Tolerances valid for set with shaft and shaft support rail

[6] Shaft support single, several or in a row