A thesis report on

# DEVELOPMENT OF AN OPEN ARCHITECTURE CONTROLLER FOR VERTICAL 3-AXIS MILLING MACHINE.

**Submitted**

**In partial fulfillment of the requirements for the award of degree of**

**Master of Engineering**

in

**CAD/CAM & ROBOTICS ENGINEERING**

Submitted by:


**HARPREET SINGH NAGI**

**Roll No: 800981011**


Under the guidance of


**Mr. Ravinder K. Duvedi**
Assistant Professor, MED
TU, Patiala-147004

**Mr. Nirbhowjap Singh**
Assistant Professor, EIED
TU, Patiala-147004


**MECHANICAL ENGINNERING DEPARTMENT**

**THAPAR UNIVERSITY**

**PATIALA – 147004**

**2011**

## CERTIFICATE

This is to certify that the thesis report titled, **"Development of an open architecture controller for vertical 3-axis milling machine."**, being submitted by **Mr. Harpreet Singh Nagi**, Registration No. 800981011, in partial fulfillment of the requirements for the award of degree of Master of Engineering (CAD/CAM & ROBOTICS) submitted in Mechanical Engineering Department of Thapar University, Patiala, is a bonafide work carried out by him under our supervision and guidance and no part of this thesis work has been submitted to any other University or institute for award of any degree.


**(Mr. Ravinder K. Duvedi)**
Assistant Professor, MED
Thapar University
Patiala-147004

**(Mr. Nirbhowjap Singh)**
Assistant Professor, EIED
Thapar University
Patiala-147004

**(Dr. Ajay Batish)**
Professor and Head
Department of Mechanical Engineering
Thapar University
Patiala-147004

**(Dr. S. K. Mohaptra)**
Dean of Academic Affairs
Thapar University
Patiala-147004


04-10-11 9:45 am

## ACKNOWLEDGEMENT

I express my sincere gratitude to **Dr. Ajay Batish, Professor and Head of Mechanical Engineering Department, Mr. Ravinder K. Duvedi, Assistant Professor, Mr. A. S. Jawanda**, **Assistant Professor, Mechanical Engineering Department, and Mr. Nirbhowjap Singh, Assistant Professor, Electrical and Instrumentation Engineering Department, Thapar University**, **Patiala,** for their valuable guidance, proper advice and constant encouragement during my thesis work.

I do not find enough words with which I can express my feeling of thanks to the entire faculty and staff of **Mechanical Engineering Department and Electronics and Communication Engineering Department**, **Thapar university, Patiala,** for their help, inspiration and moral support which went a long way for successful completion of this thesis.

**(HARPREET SINGH NAGI)**

## ABSTRACT


The trends in the NC machine tool development in the direction of open architecture control and "Direct Model Part" machining strategies. To bring the simplicity, portability, interoperability, adaptability and low cost necessitates the improvement and simplification of conventional NC machining processes. A major problem with the conventional numerical control is that the motion control in the NC machining operations are not efficient with the restrictive (ISO 6983) G and M code. G and M code were designed in the era when a paper strip was used to feed part program in NC machine, was designed to give the program in NC machine. Today the G-code programs are generated by computer-aided manufacturing (CAM) tools by the geometric data from computer aided design (CAD) tools as input. However, CNCs different manufacturers implement different versions of G-code and leads to lack any proprietary CAD-CAM-CNC integrated modules / software. An open architecture machine tools can be controlled directly from a computer, instead of having separate numerical control. As the technology advances in sensors, positioning methods and tool path programming techniques, the high-speed PC and controller or the drive in an open architecture controller technology is open to integrate all the new technologies in the future. This is possible due to the high processing speed and low cost of a modern computer. An open architecture NC tool paths are free of conventional ISO G and M-based NC tool paths, but it is a file with the previously calculated positions of the relative position of the cutter relative to the workpiece in a logical order.

The aim of present work is **fabrication of prototype of an Open Architecture NC controller for Personal Computer-controlled vertical 3-axis NC milling machine for machining of sculptured surfaces of wooden artistic design.** The sculptured surfaces are made by artisans on wood, clay or rocks manually which leads to lack in productivity and reproducibility of artisan's design. Some dedicated CNC machines are available for machining sculptured surfaces but because of heavy cost these are not affordable by an artesian. So present work is to fabricate low cost CNC machine by developing PC Based Open Architecture Controller. The aim is to break the monopoly of commercially available vendor depended CAD/CAM integration which hinders the customization of NC machine tool with new control strategies, sensors and interfaces. Also the open architecture based NC system will be better affordable than the commercial NC system for sculptured surface machining.

## NOMENCLATURE

---

| | |
|---|---|
| AWG | : American Wire gauge |
| API | : Application Programming Interface |
| ASCII | : American standard code of information interchange |
| BF | : Buffer Full |
| CTS | : Clear To Send |
| DBMS | : Data Base Management System |
| DSR | : Data Set Ready |
| DTR | : Data Terminal Ready |
| EEPROM | : Electrically Erasable Programmable Read Only Memory |
| EIA | : Electronic Industries Associations |
| GUI | : Graphical User Interface |
| GND | : Ground |
| HMI | : Human Machine Interface |
| IDE | : Integrated Development Environment |
| IC | : Integrated Circuit |
| ICD | : In Circuit Debugger |
| IGES | : Initial graphics exchange specification |
| IMM | : Intelligent Machining Module |
| IEEE | : Institute Of Electrical & Electronics Engineers |
| LSB | : Least Significant Bit |

| | |
|---|---|
| RXD | : Received Data |
| RTS | : Request To Send |
| SEN | : Send Enable |
| SCL | : Signal Clock |
| SPI | : Serial Peripheral Interface |
| STL | : Stereo lithography |
| STEP | : Standard for the exchange of product model data |
| SFR | : Special Function Register |
| TXD | : Transmit Data |
| USART | : Universal Synchronous Asynchronous Receiver Transmitter |

# INDEX

**CONTENTS** **Page No.**

CERTIFICATE ------------------------------------------------------------------------------------ 1
ACKNOWLEDGEMENT----------------------------------------------------------------------------- 2
ABSTRACT ----------------------------------------------------------------------------------------- 3
**CHAPTER 1: INTRODUCTION ---------------------------------------------------------------12-17**
1.1 The architecture of CNC Machines---------------------------------------------------------------13
1.2 Sculptured Surface Machining ---------------------------------------------------------- 14
1.4 Open Architecture Personal Computer-Based CNC ----------------------------------- 14
1.5 Objectives of Present Work----------------------------------------------------------------------16
1.6 Scope of the Research ---------------------------------------------------------------------------16
**CHAPTER 2: LITERATURE REVIEW-------------------------------------------------------18-21**
2.1 Open Architecture Machine Tool Control --------------------------------------------------------18
2.1.1 Machine Tool Open System Architecture for Intelligent Control (MOSAIC)---------19
2.1.2 Open Systems Architecture for Controls within Automation Systems (OSACA)---19
2.1.3 University of Michigan Open Architecture Controller (UMOAC)---------------------20
2.1.4 Hierarchical Open Architecture Multiprocessor Computer Numerical Controller (HOAM-CNC) -----------------------------------------------------------------20
2.1.5 University of British Columbia Open Architecture Controller -------------------------20
2.1.6 FA Open systems Promotion (FAOP) ---------------------------------------------------21
2.1.7 The Open Modular Architecture Controls (OMAC) ------------------------------------21
2.2 Tool Path Generation Methods---------------------------------------------------------------------21
**CHAPTER 3: FLAT BED 3-AXIS VERTICAL NC MILLING MACHINE DESIGN 22-27**
3.1 Design Criteria-------------------------------------------------------------------------------------23
3.2 Static Finite Element Analysis of the Milling Machine---------------------------------------23
3..2.1 Modeling of Machine Frame. ------------------------------------------------------------23
3.2.2 Pre-Processing of Machine Frame Model. ----------------------------------------------24
3.2.3 Solution of Machine Frame Model.------------------------------------------------------26
3.2.4 Conclusion. --------------------------------------------------------------------------------27
**CHAPTER 4: OPEN ARCHITECTURE CONTROLLER FOR PERSONAL COMPUTER CONTROLLED VERTICAL 3-AXIS NC MILLING MACHINE --------28-45**

4.1 Micro Controller-------------------------------------------------------------------------------29
4.2 PCB Designing Using OrCad--------------------------------------------------------------30
4.2.1 Layout Plus Module of OrCAD ----------------------------------------------------30
4.3 Communications Techniques ------------------------------------------------------------31
4.3.1 Serial Communication ----------------------------------------------------------------32
4.3.1.1 Basics of Serial Communication ---------------------------------------------33
4.3.1.2 Serial data communication uses two methods ---------------------------33
4.3.1.3 Asynchronous Serial Communication And Data Framing ---------------33
4.3.1.4 Start And Stop Bits -----------------------------------------------------------34
4.3.1.5 Data Transfer Rate ------------------------------------------------------------34
4.3.1.6 RS232 standards --------------------------------------------------------------34
4.4 USART Communication with PC in PIC 16f877 Microcontroller------------------------ 35
4.5 Master Synchronous Serial Port (MSSP) Module in the PIC 16f877A -----------------37
4.5.1 I2C Communication Master with Slaves ----------------------------------------------37
4.5.2 Transmission Control in Master/Slaves I2C Communication -----------------------38
4.6 Developed Open Architecture Controller for Vertical 3 Axis NC Machine. ------------- 40
4.7 Common Errors While Working With Electronics. ----------------------------------------43
4.7.1 Dry Solder -------------------------------------------------------------------------------- 43
4.7.2 Sharing of Ground Track ------------------------------------------------------------------43
4.7.3 Switch Bounce ------------------------------------------------------------------------------43
**CHAPTER 5: ELECTRICAL COMPONENTS FOR NC MILLING MACHINE -----46-58**
5.1 Rotary Stepper Motor------- -------------------------------------------------------------------46
5.2.1 Types of stepping motors----------------------------------------------------------------46
5.2.2 Unipolar Versus Bipolar-------------------------------------------------------------------47
5.2.3 Bipolar Basics-------------------------------------------------------------------------------47
5.2.4 Torque vs. Angle Characteristics--------------------------------------------------------49
5.2.5 Step Angle Accuracy-----------------------------------------------------------------------50
5.2.6 Torque vs. Speed Characteristics --------------------------------------------------------51
5.2.7 Single Step Response and Resonances -------------------------------------------------52
5.2.8 Mechanical Parameters --------------------------------------------------------------------53
5.2.9 Specification of Rotary Stepper Motor Used for OAC NC Milling Machine.-------53
5.3 Rotary Encoders ----------------------------------------------------------------------------------54
5.3.1 Incremental Encoder -----------------------------------------------------------------------54
5.3.2 Sensing Direction of Rotation with Incremental Encoder. ----------------------------55

5.3.3 Specification of Rotary Incremental Encoder Used for OAC NC Milling Machine 56
5.4 Power supply ---------------------------------------------------------------------------------58
**CHAPTER 6: GRAPHIACAL USER INTRFACE (GUI)---------------------------------59-72**
6.1 Microsoft Visual Basic ---------------------------------------------------------------------------59
6.1.1 Visual Basic Environment-----------------------------------------------------------------61
6.2. Development of GUI in Visual Basic 6.0 for CNC Machine ------------------------------ 64
6.2.1. Key Features of GUI ---------------------------------------------------------------------64
6.2.2 The Visual Basic Front End------------------------------------------------------------------65
6.2.1 The Welcome Splash screen -----------------------------------------------------------65
6.2.2 The User Login Module -----------------------------------------------------------------66
6.2.3 The CNC Control Module Main Screen --------------------------------------------------66
6.2.4 Loading the tool path ---------------------------------------------------------------------68
6.2.5 The Preset Material Module ------------------------------------------------------------69
6.2.6 The Material Library Management Module ---------------------------------------------70
6.2.6.1 Addition of New Material in the Library------------------------------------------70
6.2.6.2 Updating the available materials in the Library ----------------------------------71
**CHAPTER 7: DISCUSSION AND VALIDATION OF DEVELOPED 3-AXIS OPEN ARCHITECTURE CONTROLLER ON SELF FABRICATED 3-AXIS NC MILLING MACHINE ---------------------------------------------------73-81**
7.1 Features of Developed Open Architecture controller (OAC) --------------------------------73
7.2 Testing of Developed OAC on Fabricated 3-Axis NC Milling machine. -------------------74
7.3 Limitation of Developed Open Architecture Controller for 3-Axis NC Milling Machine 80
7.4 Future Scope ---------------------------------------------------------------------------------------81
**Appendix A: Flow Charts of Front End GUI --------------------------------------------------82-87**
**REFERENCES ---------------------------------------------------------------------------------------88-91**

# List of figures

Figure 1.1: Open Architecture Control System.------------------------------------------------------15
Figure 1.2: Conventional Method of Tool Machining. ---------------------------------------------16
Figure 1.3: Open Architecture Controller.-----------------------------------------------------------17
Figure 3.1: Design of 3 Axis Vertical Milling Machine. -------------------------------------------22
Figure 3.2: CAD Model of CNC Machine Structure.----------------------------------------------24
Figure 3.3: Idealization of Bolts, Frame Members and holes.---------------------------------------25
Figure 3.4: Mesh Generation of CNC Machine Frame. ---------------------------------------------25
Figure: 3.5 Deflections in Machine Frame.-------------------------------------------------------26
Figure: 3.6 Deflections in Machine Frame Scaled at 1000X.---------------------------------------26
Figure 3.7: Stresses Generated in Machine Frame.-------------------------------------------------27
Figure 4.1: A Decomposition of Machine Controller Functions. ----------------------------------28
Figure 4.2: Pin Diagram of PIC Microcontroller. -----------------------------------------------29
Figure 4.3: PCB of OAC Designed in OrCAD Layout. ----------------------------------------------31
Figure 4.4: Shows Serial vs. Parallel Data Transfer.----------------------------------------------32
Figure 4.5: Shows RS 232 Connections with PIC Microcontroller and PC by Using MAX 232 IC.----------------------------------------------------------------------------35
Figure 4.6: (a) Connection to PC. (b) Signal at PIC Port.------------------------------------------37
Figure 4.7: Inter-IC Bus (I2C) (a) I2C Connections; (b) I2C Signals. ------------------------------38
Figure 4.8: Interfacing 24c02 to PIC via I2C Lines. -----------------------------------------------39
Figure 4.9: Fabricated 3-Axis NC Machine to Validate Developed OAC. ------------------------40
Figure 4.10: Developed Controller Circuit of OAC for 3 Axis Vertical Milling Machine. ------41
Figure 4.11: Signal Flow Lines among Machine, Microcontrollers and PC.----------------------42
Figure 4.12: Contact Bounce upon Closure of Microswitch. ----------------------------------------44
Figure 4.13: Hardware Debounce. ------------------------------------------------------------44
Figure 5.1: Conceptual Model of Bipolar Stepper Motor.----------------------------------------48
Figure 5.2: Torque vs. Rotor Angular Position.--------------------------------------------------49
Figure 5.3: Positional Accuracy of a Stepper Motor. ----------------------------------------------50
Figure 5.4: Torque vs. Speed Characteristics of Stepper Motor.-------------------------------------51
Figure 5.5: Single Step Response vs. Time.-------------------------------------------------------52
Figure 5.6: Incremental Encoder Output Wave Form.-------------------------------------------------55
Figure 5.7: Connection Diagram for the Rotary-Encoder. ----------------------------------------56

Figure 5.8: Phase Shift in Output Signal from Two Wires. ----------------------------------------56
Figure 5.9: DELTA™ Rotary Optical Encoder Model ES3-10LN6541. ---------------------------57
Figure 5.10: Output Wave Form and Output Signal Wires.-------------------------------------------57
Figure 6.1: Programs Must Respond to Some Events and Ignore Others. --------------------------60
Figure 6.2: Visual Basic Environment.---------------------------------------------------------------61
Figure 6.3: General Tool Palette and Additional Component Dialog.--------------------------------62
Figure 6.4: The VB Code Editor and Its Intellisense Feature. ----------------------------------------63
Figure 6.5: Combo Boxes for Control and Their Events in Text Editor.------------------------------64
Figure 6.6: Welcome Screen of the CNC Machine Control Software. ------------------------------65
Figure 6.7: Incorrect Password Prompt.--------------------------------------------------------------66
Figure 6.8: CNC Milling Machine Front End GUI. --------------------------------------------------67
Figure 6.9: Loading the Commands from *.tp file to List Box. ------------------------------------68
Figure 6.10: Loading the Material, Feed Rate and Speed from the Preset Library. ---------------69
Figure 6.11: The Update Material form during Design Time. ----------------------------------------70
Figure 6.12: Message Box Prompt on Addition of Material in the Library.-------------------------70
Figure 6.13: Error Message When Parameters Are Not Entered By User. -------------------------71
Figure 6.14: “Material already exists” Error Prompt.------------------------------------------------ 71
Figure 6.15: Selecting the Material from the List and Prompt to Enter the Parameters.----------72
Figure 6.16: Message Box Prompt on Successful Update of the Library. --------------------------72
Figure 7.1: Fabricated 3-Axis NC Milling Machine.--------------------------------------------------74
Figure 7.2: 8 Cutting Points on Circumference of 25 mm Circle.-----------------------------------75
Figure 7.3: Loading of Tool Path file of 8 Points.------------------------------------------------------76
Figure 7.4: Fixing of Wax Brick on NC Machine. -----------------------------------------------------77
Figure 7.5: First Cut Located By Machine Cutting Tool. --------------------------------------------77
Figure 7.6: Sequence of Cutting on Wax Brick According To Tool Path File Loaded. ----------78
Figure 7.7: Distance between Two Points along Diameter of Circle. -------------------------------79
Figure 7.8: Distance Measured of a Point Lie in First Quadrant from Vertical Center Line of Circle.------------------------------------------------------------------------------------79
Figure 7.9: Distance Measured of a Point Lie in First Quadrant from Horizontal Diameter of Circle.------------------------------------------------------------------------------------ 80
Figure A.1: Loading of Main Form Subroutine.-------------------------------------------------------82
Figure A.2: Load click command button.---------------------------------------------------------- 83
Figure A.3: To Delete Selected Command from Loaded Tool Path File By Pressing Del Key on Keyboard. ---------------------------------------------------------------------------------83

Figure A.4: Connect Button to Open Serial Port.----------------------------------------------------84
Figure A.5: Click Command Button to Start Machining. ---------------------------------------------84
Figure A.6: Click Command Button to Start “Start At” Machining. ---------------------------------85
Figure A.7: Material Library Management Subroutines.----------------------------------------------86
Figure A.8: Update Already Stored Material Value In Library.---------------------------------------87
Figure A.9: Exit Front End GUI Form. -----------------------------------------------------------------87

# CHAPTER 1

# INTRODUCTION

---

The term Numerical Control is a generally accepted term in machine tool industry. Numerical Control (NC) makes possible for an operator to communicate with machine tools by sequences of numbers and symbols. Numerical control, that soon become computer numerical control (CNC) brought tremendous changes at the manufacturing industry. New machine tools in CNC made industry possible to produce parts through out to the accuracies which are fromly some years ago undreamed. The parts can be reproduced to the higher degree at accuracy each possible number of times, only if the CNC program were prepared correctly and the correctly programmed computers. The operating instructions, which control the machine tool, are accomplished automatically with amazing speed, accuracy, efficiency and repeatability. Nearly everything, which can be produced on a conventional machine tool, can be produced on a computer controlled machine, with its many advantages.

NC or simply Numerical Control was built in the late 1940s and early 1950s by John T. Parsons in collaboration with MIT (Massachusetts Institute of Technology) [37]. It was developed, in order to help in the postwar manufacture effort. The airplane parts were required a level of the precision more complicated, which human operators could not obtain.

At the first machines hard-wired and then instructions were given over the punched tape, in 1952. Five years later, NC machines were attached completely in metal work production environment over the United States. Up to the middle 1960s, NC technology played numerical control technology a dominating role in the industry. Most coded programs were noted on a punched paper or aluminum tape until approximately 1980. In the 1970s and in the 80s the growth of the microprocessor technology made it possible, so that computers control the numerical control machines, therefore the expression CNC.

Basically, numerical control is to enter encoded part program in the form of numerical data into a computer to control the movement of machine components. CNC machines were formily developed for machining metal. They were subsequently modified for other industries such as wood, fabric, foam, and plastic etc.

## 1.1 The Architecture of CNC Machines

The architecture of CNC machines is generally divided into four layers

i. User Interface
ii. Machine controller
iii. Drive unit
iv. Machine structure

The user interface of the CNC machine is the consoles that permits the operator to communicate with the system of CNC devices. The user surface shows the in process machine events and the part programs. It permit the operator to revise and to attest the actual prepared part programs to prevent any mistake in the tool path. The part programs are produced in general manually or with computer (CAM) software packages supported through using of the cutting parameters and through computer aided part design (CAD) model data. This part programm contains the machine movement directions usually in a normal machine tool language, like for example G code. The initial machine settings of the tool compensation and the workpiece origin made through the user interrface. Simulations of the prepared part prorgram could be shown also on the user interface by some modern CNC machines.

The CNC controllers as the brain of CNC machines that drives and senses the entire machine movements. The machine controller is monitoring in-process machine status and the machine coordinate systems to find the real time tool and part locations by taking feedback from sensors, limit switches and encoders devices.

Drive unit is having the electric power motors, routers and high precision motion convertor mechanism. Sufficient rated torque of electric drive motors, with having zero play in motion converter mechanisms and high rpm router ensures the high productivity in manufacturing industry.

Structure design of CNC machine ensures the smooth running of machine with no vibration in machine structure. Design of fixture and bed such that which protects the drive unit from chips, and also which provide adequate cooling to electrical and electronic components.

## 1.2 Sculptured Surface Machining

Complex surfaces can be catograsied into two types namely Ruled surfaces and Sculptured surfaces. Ruled surfaces are used in turbine blades and compressor disks and Sculptured surfaces. Free-form surfaces are basically used in moulds and dies. NC machining is the important technology to get a precise and accurate shape of dies, punches, moulds, blades or artistic design [1]. The presentt NC machining technology is extensively used for three axes NC milling using ball-end milling tools, or five axes NC milling using ball or flat-end mills. The current requirement is to produce precise free-forms on different materials, with least tolerances and best economical performance. The latter aspect is important because of low-wage countries are new competitors in the manufacturing of dies and moulds in market. Thats why, several aspects must be taken into consideration: the use of 3 or 5-axis CNC machines and powerful machine tool path generation methods in system. The complex sculptured surface machining systems are NC tool path planning and generation modules which is integrated with CAD packages & dedicated NC machining tools, which is a complicated and expensive process.

## 1.3 Open Architecture Personal Computer-Based CNC

In present state of time has capability to integrate all the level of manufacturing operation with dedicated software packages and hardware used with the help of computer technology. Industrilist can now be able to reduce manufacturing cost, get high product quality, decrease product development time and can introduce competitiveness in local and international market place. With the development of computer graphics and computational power of personal computer extended to Computer Aided Design (CAD), Computer Aided Manufacturing (CAM) and then with the development in Data Base Management System (DBMS) ultimately bring Computer Integrated System (CIM) into study.

The main difficulty in CIM is integration of computer sysytems, machine tools and control systems, even though in automated industry the CNC machines, various robots and different controlled equipments in production unit are programmable, but they are still control by their own specific control method. Goodness of CIM system is decreased when uncommon communication associate among various computer controlled machine. This is the drawback which decrease the flexibility in the system because of incompatibility of varoius machine and leads to inefficient automated production system.

Time is leading towards the next generation of computer numerically controlled (CNC) machines which has to be portable, interoperable and adaptable. From many years, G-codes (ISO 6983) have been greatly used by the NC machine tools for part programming and are now taken as a bottleneck for developing of CNC machines with new paradigm [40]. An open architecture system, as defined by Institute of Electrical & Electronics Engineers, IEEE, provides capabilities of applications to execute on a variety of platforms environments, inter-operation with other systems. An open CNC system is a hardware, software architecture that reflects capabilities such as interoperability, interchangeability, scalability and portability.

The main characteristic of OAC (open architecture controller) numerical control is to establish a kind of software and hardware architecture that can run with a general computer as shown in figure: 1.1. However the open CNC system needs a communication software and hardware standard among computer system, operating system, application software. Using the great advantages of open CNC system, users or buyer can quickly build an advanced capability controller which contains diffrent functions based on a general computer and operation system environment.

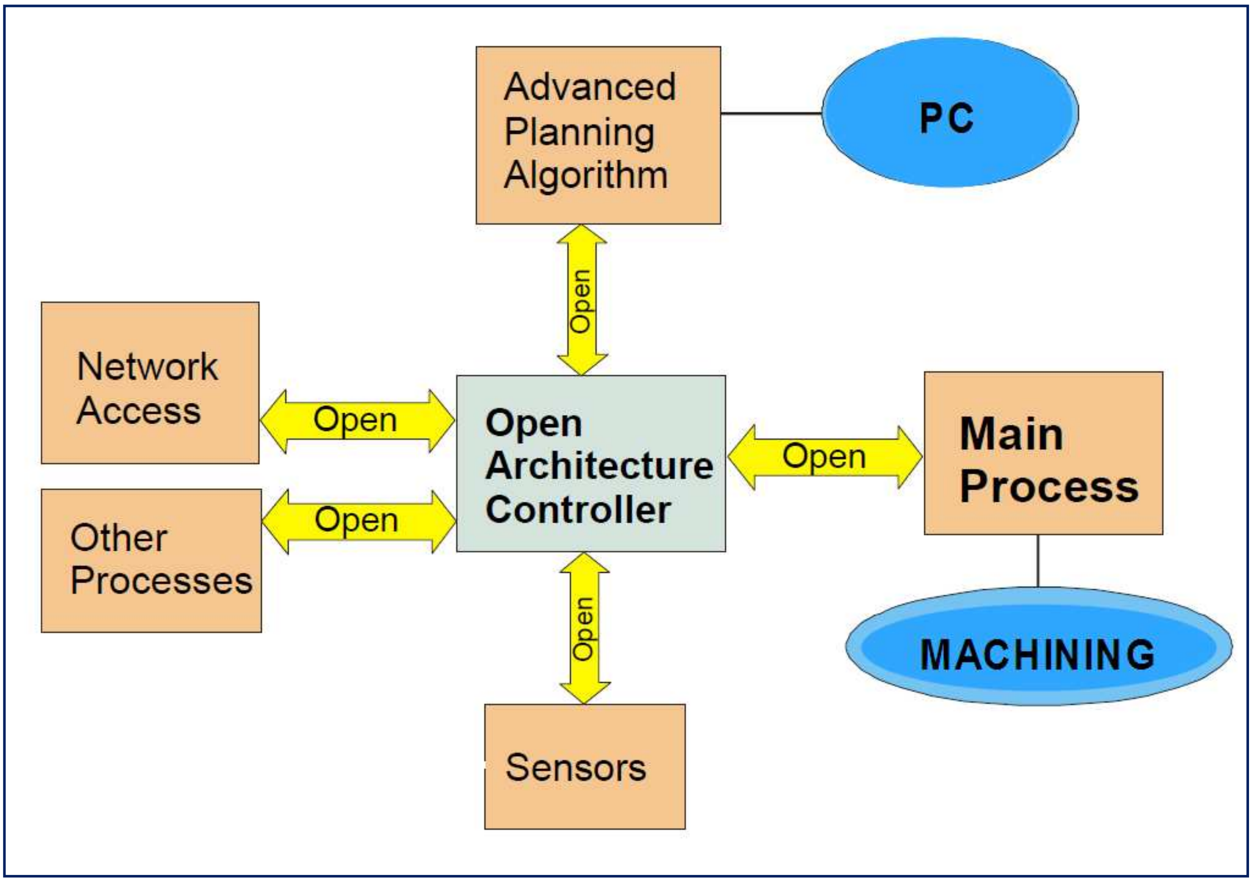



**Figure 1.1: Open Architecture Control System**

## 1.4 Objectives of Present Work

The main objectives of the proposed work are:

i. To Design and fabricate a prototype of an Open Architecture Personal Computer-based Vertical 3-axis NC Milling Machine for machining wooden sculptured surface.
ii. To develop PC side Front End software in Visual Basic that read the tool path file, control serial port, send instruction to machine and receive feedback to generate error report.
iii. To develop a driver and an interfacing circuit between machine and PC.
iv. To develop controller side program to receive instruction from PC, control the motors movements, checks the encoder's feedback sends the error value to PC.

The proposed NC machine architecture will include the development of following components:

i. PC interface program.
ii. Controller suite (hardware and software) to integrate PC and motor / actuators.
iii. Feedback suite for motion control (hardware and software) to communicate the errors with respect to commanded and achieved tool position.
iv. Development of prototype of physical structure of NC machine tool.

## 1.5 Scope of the Research

As in conventional method of tool machining as shown in figure1.2, the generation of G & M codes for particular machine for complex sculptured surface is difficult and inefficient.

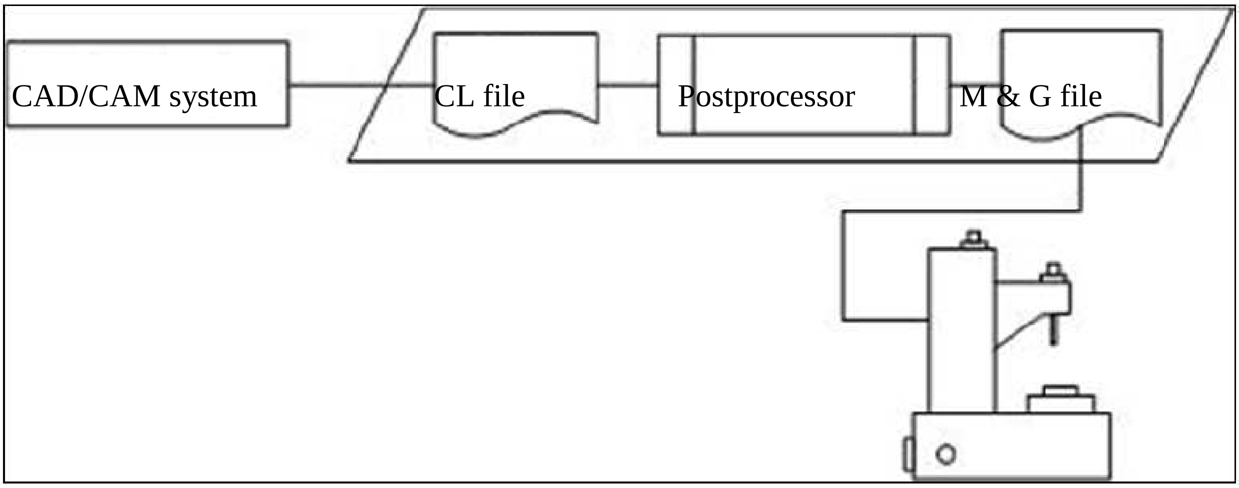



**Figure 1.2: Conventional Method of Tool Machining**

A **new open architecture controller** will be developed as shown in figure1.3. Development of controller is for self designed and fabricated prototype of 3 – axis vertical milling machine for machining wooden sculptured surface having key features low cost, portability, interoperability and adaptability.

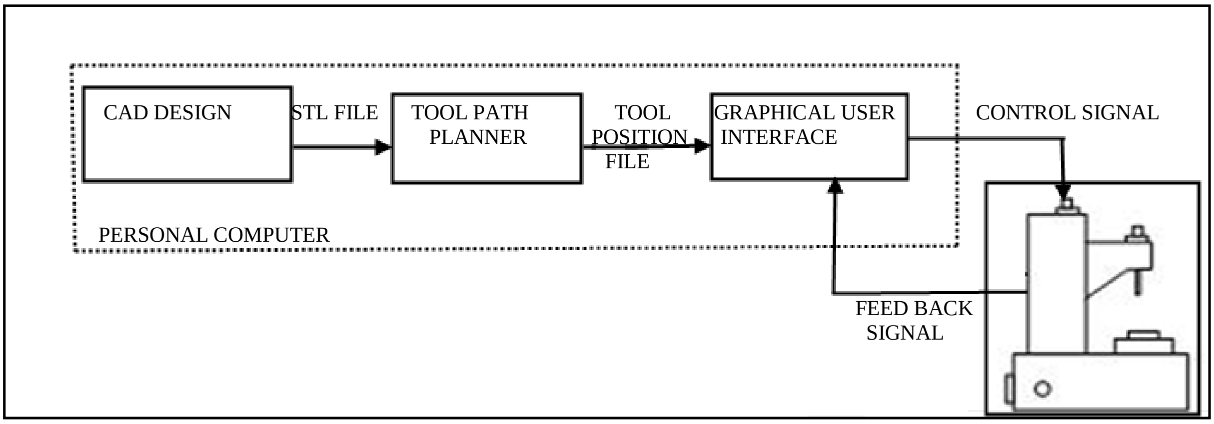



**Figure 1.3: Open Architecture Controller**

# CHAPTER 2

# Literature Review

The computer numeric control machine has become indispensible for the modern manufacturing industries in every type of manufacturing process due its **production capacity with superior accuracy.** The inbuilt tool positioning strategies are utilized by the commercial CAM packages to generate tool path plans in the ISO G-code programming language or some machine specific proprietary languages. But the vendor specific CAM packages and their dedicated CNC controllers are difficult to reconfigure and update to newer technology, thus there is a requisite to develop a new controller which should free from G-code instructions. The continuous decline in the price of electronic circuitry and enhancement in the computational capability of new generation microprocessors and microcontrollers has made it possible to produce low cost CNC controllers which possess features like portability, configurability and interoperability.

In this chapter, a literature survey is presented which supports the present work. The first part of this chapter discusses the Open Architecture Controller (OAC). The next presents Personal Computer based Intelligent Machining, sculptured surface machining and lastly, tool path generation and positioning methods have been briefly reviewed.

## 2.1 Open Architecture Machine Tool Control

Open architecture machine tool control is an 'empowering' technology for the application of new generation controls in manufacturing industry. The restraints and difficulties faced by practitioners and researchers, when advanced manufacturing control techniques were experimented on current 'closed' architecture controllers, have been the driving force for designing the ‘open’ architecture systems. Personal computer (PC) based machine controllers offer the flexibility to add peripherals without rebuilding the complete machine. The vision of developing PC-based CNC controllers to replace the traditional proprietary CNC controllers is not new [17][18] [19][20].

Various commercial and academic researches have focused on developing some form of open architecture for machine tool controllers and manufacturing automation systems. Fundamentally, each approach has been concerned with one or more of the following:

i. Integrating physical hardware devices.
ii. Communication interfaces between devices.
iii. Models of interaction between computing resources motion control and I/O devices.

These items provide a reference with which a comparison can be made between the different OAC systems. The following survey outlines the major accomplishments in this area.

### 2.1.1 Machine tool Open System Architecture for Intelligent Control (MOSAIC)

The MOSIAC system developed at New York University is one of the most recognized researches related to open architecture machine tool control [4]. Originally, The Machine Open Systems Architecture for Intelligent Control was developed in order to facilitate research related to expert planning systems and quality related sensor data. Conventional controllers at that time were not competent enough to support the information flow needed for these tasks. The MOSAIC architecture with memory mapped backplane architecture (VMEbus) possessed specialized hardware for axis control, machine input / output and a general purpose processor running a real time POSIX compliant operating system.

A library of C language routines was used to access the controller functions. Later work related to the MOSAIC system was centered on the development of these higher level functions. Several published works have included a wide range of applications such as adaptive control strategies [7] and an Internet based machining center [10].

### 2.1.2 Open Systems Architecture for Controls within Automation Systems (OSACA)

The OSACA project developed in Europe, emphasized on the formulation of open communication / messaging services within the OAC [15]. At the top level of the OSACA model were architecture objects that used these information models to implement actual high end functions such as:

i. Human / machine control.
ii. Logic control.
iii. Motion control.
iv. Axis control.
v. Process control.

Sperling and Lutz, presented the research regarding the OSACA project [25]. In their paper, they recapitulated the OSACA design approach and presented an example of integrating a human / machine interface (HMI) architecture object with OSACA infrastructure in which the Internet Protocol communications were used to interlink the hardware platforms for the HMI and motion control sub systems.

### 2.1.3 University of Michigan Open Architecture Controller (UMOAC)

UMOAC is a test bed developed at the University of Michigan using a variety of platforms [21] such as:

i. PC based
ii. VME bus (backplane)
iii. CANbus (distributed)

Birla et al, presented the work done on UMOAC. An elaborate software hierarchy enabled an object oriented application interface with abstractions of the machine functions. Custom software libraries allowed the generation of control and monitoring systems within the environment of a real time operating system.

### 2.1.4 Hierarchical Open Architecture Multiprocessor Computer Numerical Controller (HOAM-CNC) Controller

Altintas et al, in his research described a hierarchical, OA multi processor computer numerical controller (HOAM-CNC) consisting of multiple processors on a PC bus [3]. It has individual axis control processors for loop closure and master processor for executing functions for axis interpolation, adaptive control, and process monitoring. A script like language enables the master (PC) to handle the execution of activities on the Intelligent Machining Module (IMM), such as adaptive force control and tool breakage monitoring. The IMM employed a special form of the Open Real Time Operating System (ORTOS) using which the combination of processor modules could be serially connected and signal processing network could be formed. The system also included special ORTOS functions to support the interactions between modules. The Open Modular Architecture Controls (OMAC) users group was formed in 1994 to collaborate with industrial companies world-wide in developing common international application programming interface (API) standard for motion and machining operations [13] [32] [15][16].

### 2.1.5 FA Open systems Promotion (FAOP)

The FA Open Systems Promotion (FAOP) group, organized by Manufacturing Science and Technology Center (MSTC) in Japan is dedicated in the research and development of a standard distributed manufacturing open communication system for information exchange between the shop-floor and the management department [12].

## 2.2 Tool Path Generation Methods

The manufacturing industry employs the conventional tool path generation method in which the cutter contact point on the part surface is specified and then that point is offset to yield the cutter location. The cutter contact point (CC) is the location where the tool touches the part surface. The cutter location (CL) is the location of the centre of the tool. There are a number of tool path generation methods that are popular in industry. Some of the common ones are illustrated by Jasra [42]:

i. The Isoparametric Method
ii. The Cartesian Method
iii. The Offset Surface Method
iv. Feed Forward Method
v. Side Step Method

# CHAPTER 3

# FLAT BED 3- AXIS VERTICAL NC MILLING MACHINE DESIGN

---

The design of 3-axis CNC milling centre has been developed. A gantry styled CNC machine structure would be simple to implement. The idea is of moving the milling cutter over the material rather than the material under the cutter. This is because the machine so built will have smaller size for the given traverse of the tool relative to the work piece material. Considering only 300 X 200 mm horizontal traverse, a gantry machine would be most suitable. The approximate machine size would be about 400 mm long and 600mm wide. Figure 3.1 shows the flat bed 3- axis vertical NC milling machine fabricated for open architecture controller.

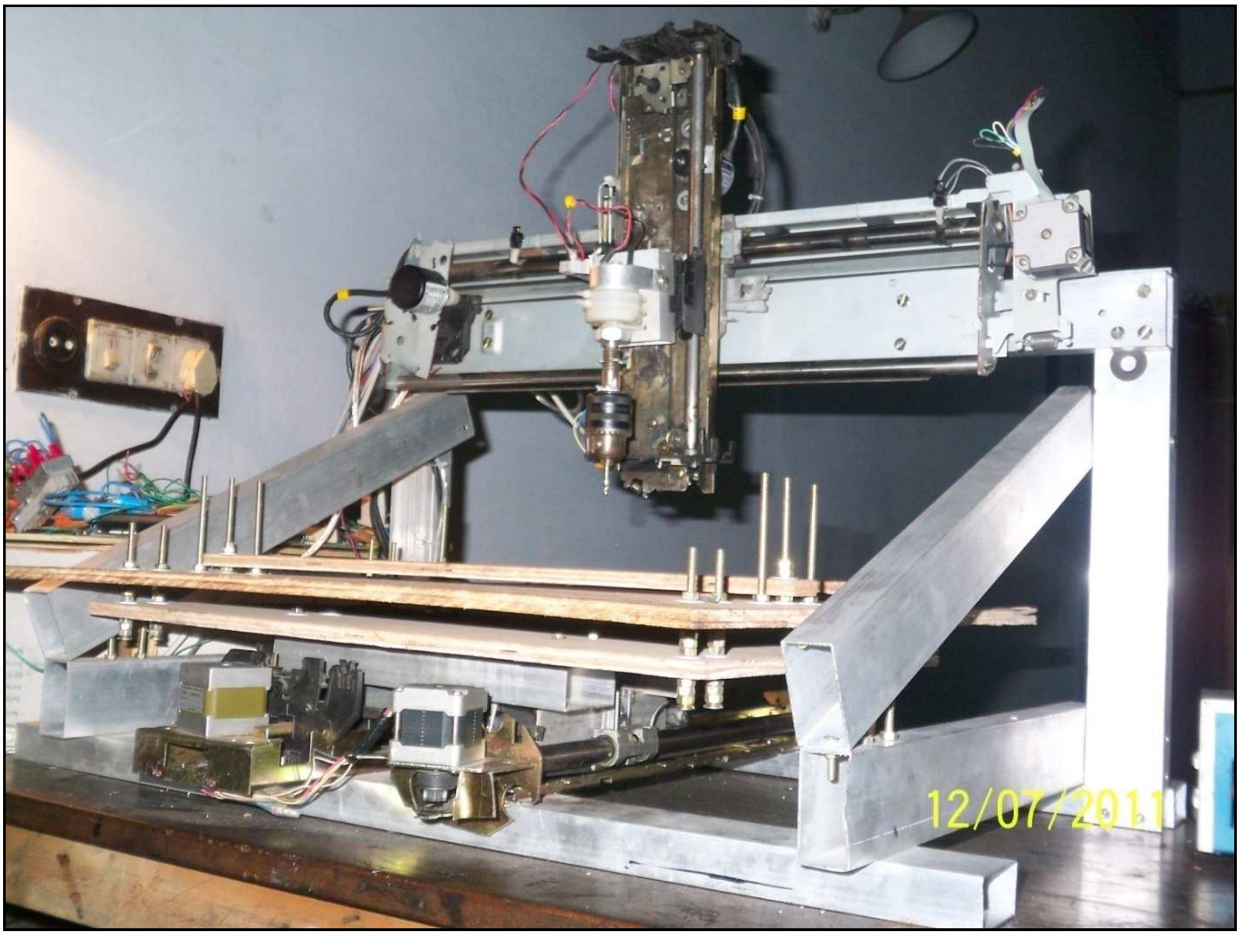



**Figure 3.1: Design of 3 Axis Vertical Milling Machine**

The said NC machine tool structure is specially designed for compact shape and larger span of moment of tool relative to work piece. Also the new design of machine does not allow the chips to come on the drives as they are fully protected from chips.

### 3.1 Design Criteria

The machine structure to be built is to demonstrate the concept of mechatronic interface of PC with a NC machine via customized NC controller. The main objective is to develop the PC controlled NC Mill, where the main focus is on the development of PC interface and electronic interface for milling centre. Machine structure is over designed against static loading. Machine structure also face minimal vibrations as drill motor speed is 150 rpm and to slow down jerky motion of stepper motor half step sequence is used.

### 3.2 Static Finite Element Analysis of the Milling Machine.

To validate machine design Finite Element Analysis is done of machine structure under static loading. Powerful CAD tool ProE$^{TM}$ is used to model the frame of machine. Altair HyperMesh$^{TM}$ software is used for preprocessing the model. Finally Ansys$^{TM}$ is used to done the analysis of preprocessed model of machine structure. The loading on the machine structure is only due to z-axis mechanism and y-axis mechanism. Z-axis mechanism is residing on y-axis mechanism. The frame elements fixed to each other with nuts and bolts are the media to transfer force to other members of frame. Failure of structure would be due to tearing of frame element where bolts are assembled or due to bending or buckling of frame member. The machine structure considers safe if maximum stress occurring in machine structure will less than yield strength of material.

#### 3.2.1 Modeling of Machine Frame.

ProE$^{TM}$ CAD tool is used to model frame of CNC machine. Figure 3.2 shows model of CNC machine frame. All drilled holes in frame elements are of 4mm. Frame element in green color shown in figure 3.2 is carrying both y-axis and z-axis mechanism. This element transfers load to other frame members through nuts and bolts. In the frame where nuts and bolts are assembled there drilled hole are shown.

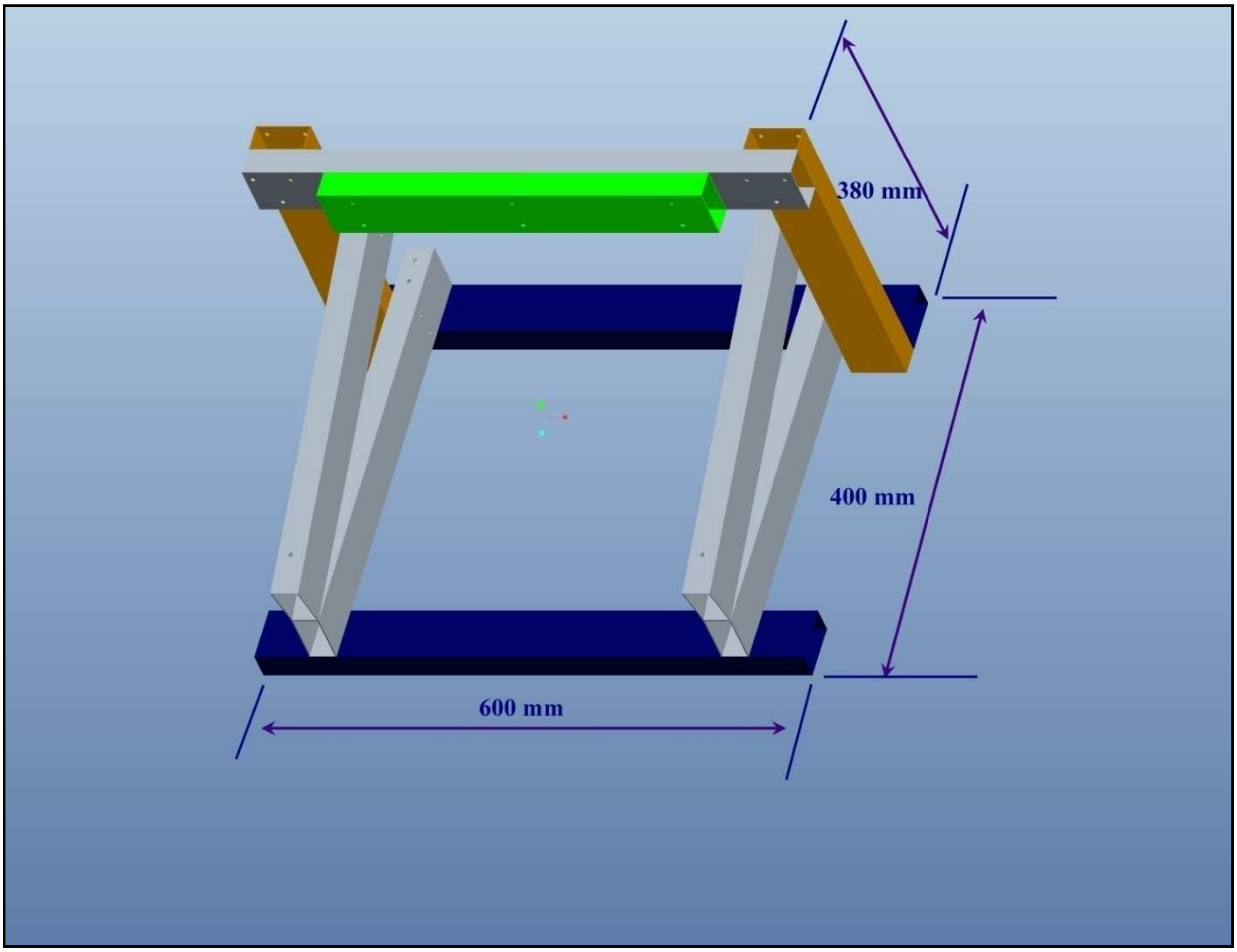



**Figure 3.2: CAD Model of CNC Machine Structure**

CAD model is exported to Altair HyperMesh™ software in IGES file format.

### 3.2.2 Pre-Processing of Machine Frame Model.

Altair HyperMesh™ software was used for pre-processing work. Bolts were considered as 1-D beam elements. FE20 material was taken for bolts. Frame elements of machine structure are of hollow rectangular pipe. Idealization of hollow rectangular pipe was done by finding its mid surface and thickness of 1mm was assigned. Drill weight is of approx 200 gram as it is geared DC motor. Total load that have to bear by member carrying y-axis and z-axis mechanisms is not more than approx. 1.2 kg. 2 Newton load is applied vertically down on each bolts those are carrying both y-axis and z-axis mechanisms. Base frame members were taken as fixed to ground. Figure 3.3 shows idealization of bolts, frame member and holes. Hollow rectangular pipe are of aluminum material. Al2014 material is assigned to frame. Al2014 has 70 MPa yield strength, elastic modulus 70-80 GPa and Poisson's ratio 0.33.

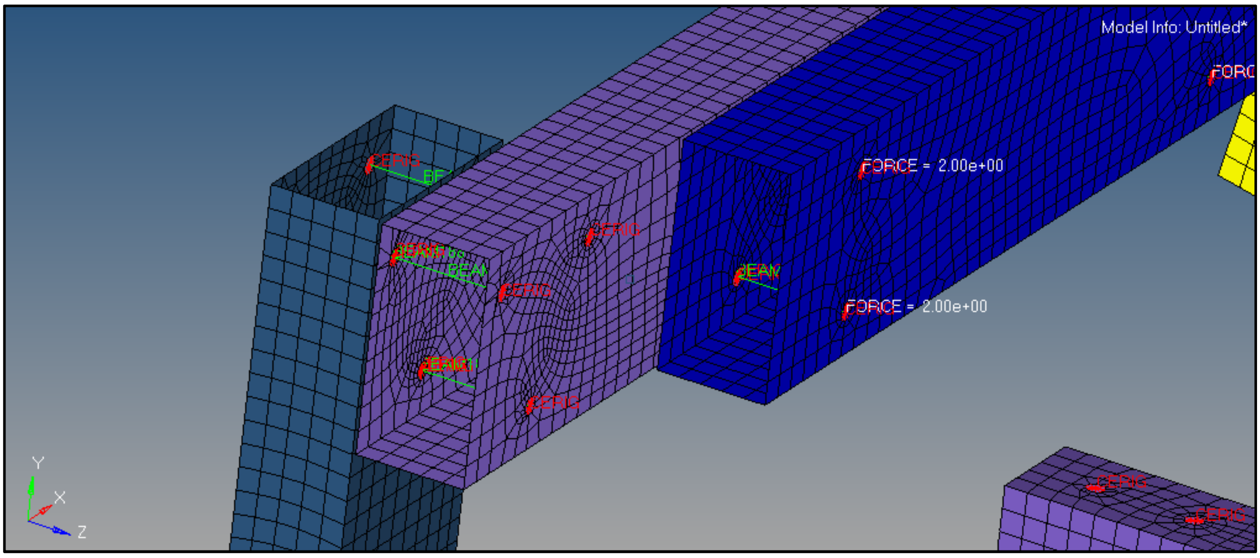



**Figure 3.3: Idealization of Bolts, Frame Members and holes.**

After completing pre-processing work save the database file. Hyper Mesh (.hm) file is generated after completing the mesh generation. Figure 3.4 is shown meshing of CNC frame.

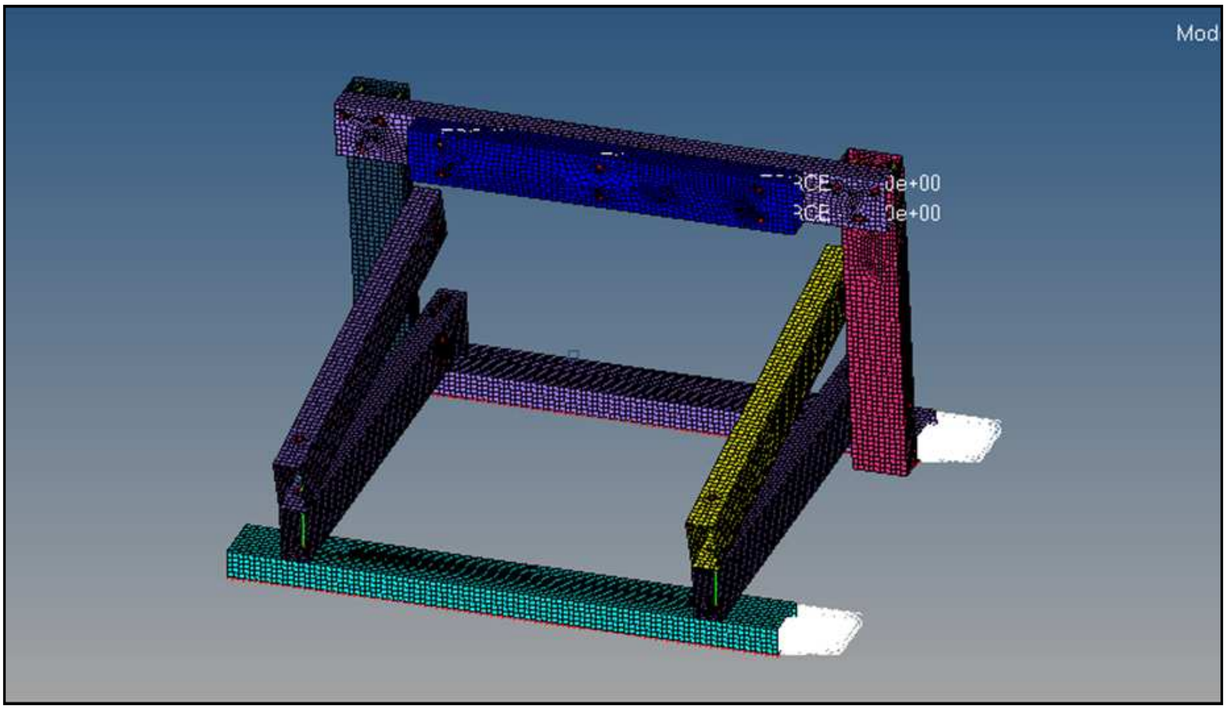


**Figure 3.4: Mesh Generation of CNC Machine Frame.**

### 3.2.3 Solution of Machine Frame Model.

Ansys$^{TM}$ software was used for solving pre-processing work. Under analysis type static Ansys gave the results are shown in figure 3.5 to figure 3.7. Maximum deflection occurred in machine frame was 0.223E-03 mm and Maximum stress induced in machine frame was 6.84959 MPa.

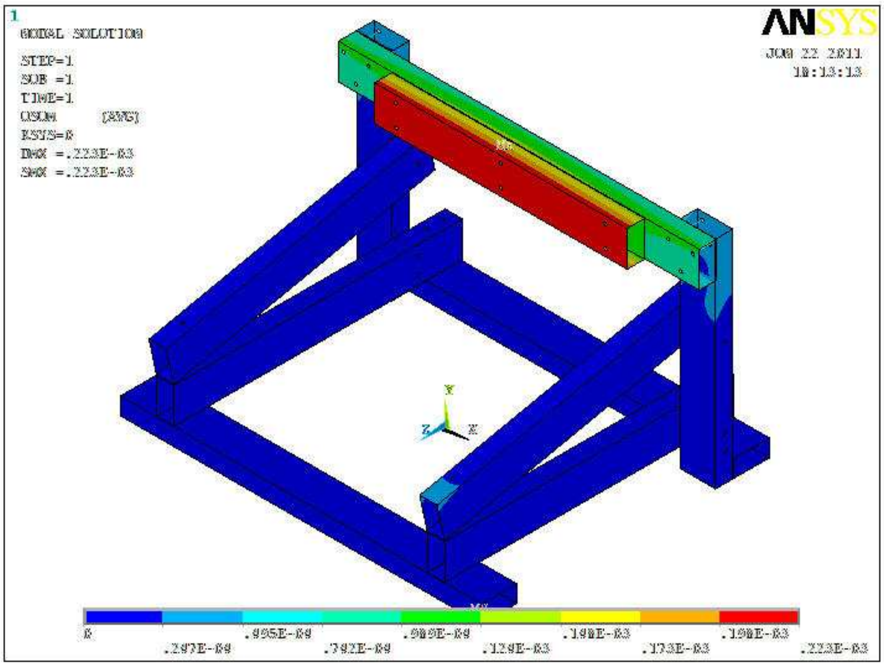



**Figure: 3.5 Deflections in Machine Frame.**

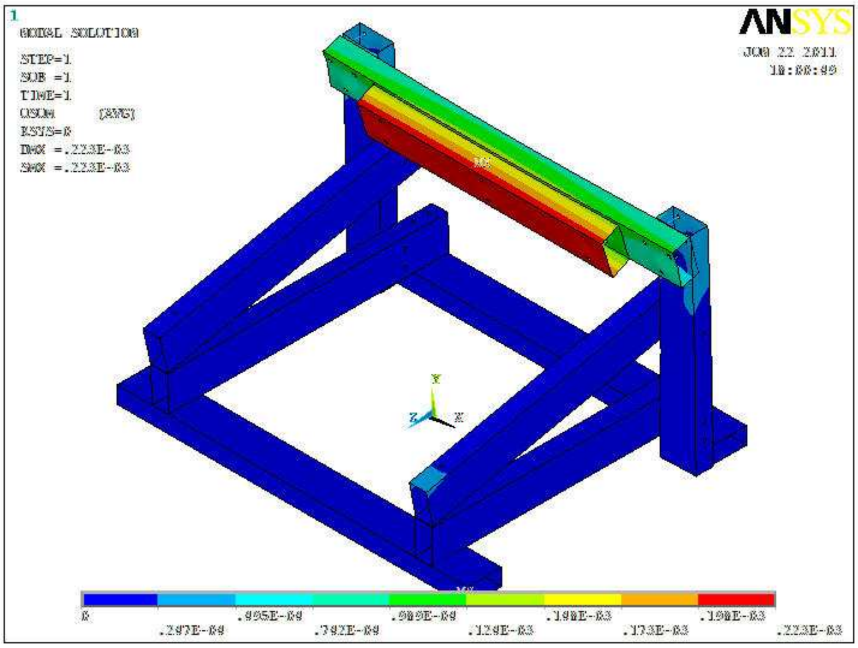



**Figure: 3.6 Deflections in Machine Frame Scaled at 1000X.**

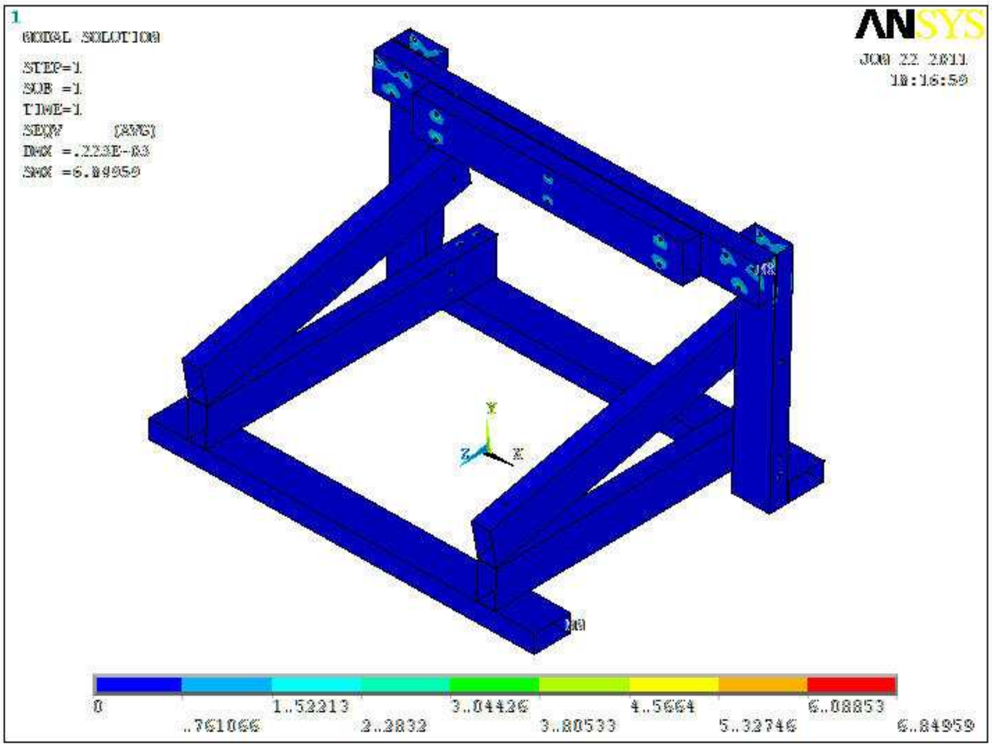



**Figure 3.7: Stresses Generated in Machine Frame.**

### 3.2.4 Conclusion.

Maximum stress induced in machine structure 6.84959 Mpa as shown in figure 3.7 is far less than yield strength 70 MPa of material aluminum al2014. This concludes that the machine frame is over designed and safe under static loading condition. Deflection occurred in machine frame is very negligible 0.223E-03 mm which is visible in figure 3.6 after scaling 1000 times of true deflection.

## CHAPTER 4

## OPEN ARCHITECTURE CONTROLLER FOR PERSONAL COMPUTER CONTROLLED VERTICAL 3-AXIS NC MILLING MACHINE.

Generally, OAC are defined in terms of their functional elements using a functional model. The functional model is a combination of several functional elements where each element represents a particular activity performed by the controller machine and describes the interactions between these elements. Figure 4.1 depicts a functional break down of a typical CNC machine tool controller.

There are two main functional models that have been introduced in published OAC designs:

i. Master / Slave,
ii. Peer to Peer.

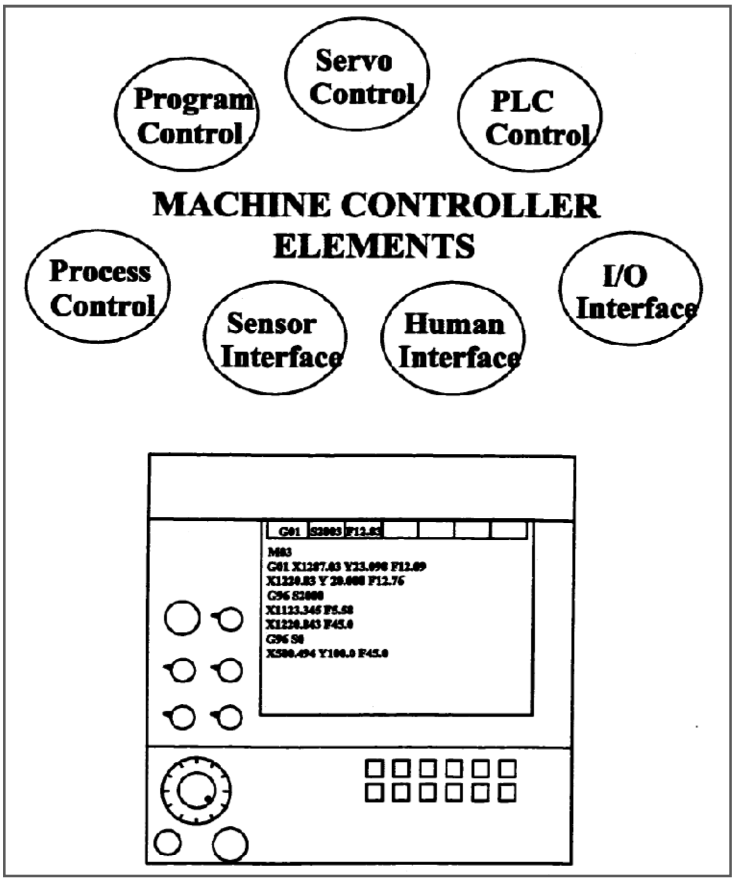



**Figure 4.1: A Decomposition of Machine Controller Functions.**

In the Master / slave type systems the system components are dependent on some central administrative entity. Altintas [8] and Park et al, [23] include the examples of such approaches. Theoretically, such systems may suffer from certain performance problems if the master is heavily loaded with demands. This may be manifested as a limit on system scalability (i.e.: the master as a performance 'bottleneck'), or on a reliability of the entire system integrity on the master.

Peer to Peer type systems are quite opposite to the master / slave type systems. In these systems, the system modules designed as independent agents, and are assigned individual responsibilities. Peer modules can act as service providers and consumers as well. Examples of such paradigms are presented by Haynes, [13] and Pritchow and Junghans, [15]. In fact, the peer to peer concept is the predominant model used in established open computing systems because of its improved reliability.

### 4.1 Micro Controller

For driver circuit Micro Chip 40-Pin 8-Bit CMOS FLASH Microcontroller PIC16F877 will be used. Figure 4.2 shows Pin description of pic microcontroller.

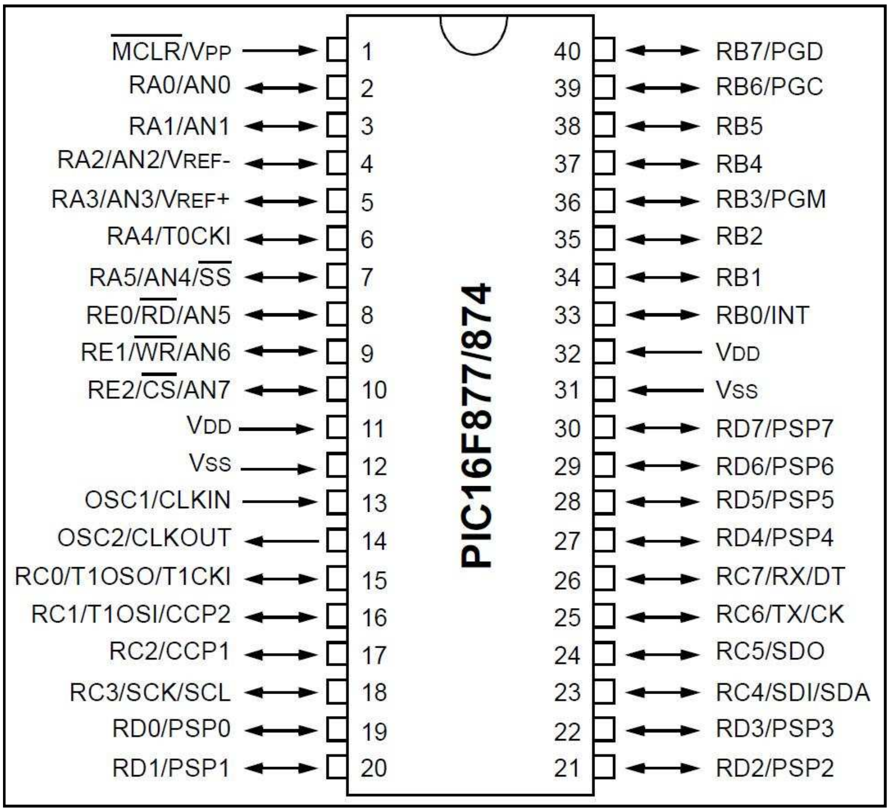



**Figure 4.2: Pin Diagram of PIC Microcontroller.**

Following are the features of PIC16F877 microcontroller:

i. High speed 20 MHz

ii. Enough memory  Up to 8K x 14 words of FLASH Program Memory, 368 x 8 bytes of Data Memory (RAM), 256 x 8 bytes of EEPROM Data Memory

iii. Power-on Reset (POR).

iv. Programmable code protection.

v. Power saving SLEEP mode.

vi. Low power, high speed CMOS FLASH/EEPROM technology.

vii. In-Circuit Serial Programming (ICSP) via two pins.

viii. In-Circuit Debugging via two pins

ix. Processor read/write access to program memory

x. Wide operating voltage range: 2.0V to 5.5V

xi. Commercial, Industrial and Extended temperature ranges.

xii. Low-power consumption: - < 0.6 mA typical @ 3V, 4 MHz
-<20 μA typical @ 3V, 32 kHz
- < 1 μA typical standby current

### 4.2 PCB Designing Using OrCAD

OrCAD is an electronic package, which is used to make the schematic Diagrams, the layout of PCB's and PSpice Simulation of the circuits as shown in figure: 4.3. There are many subprograms of OrCad e.g. Capture, Layout Plus, PSpice for diffrent tasks etc.

#### 4.2.1 Layout Plus Module of OrCAD

Layout plus is used to make PCB layouts. OrCad Layout is a powerful circuit board layout tool that has all the automated functions to quickly complete the PCB. Using a schematic capture tool, layout can be created which would be compatible with Netlist. The Netlist contains much of the design information that Layout uses to produce the board. The main part is to place the components on the board. Selecting a track in layout highlights the corresponding net. In layout, set up of an environment is needed to create a new design board, e.g., selecting the units of measurement, create the board outline, defining the layer stack, setting the system grids, defining the manufacturing technology, defining pad stack, defining via and setting has to be done. Figure 4.3 has shown the PCB design of Open Architecture NC controller for personal computer controlled vertical 3-axis NC milling machine.

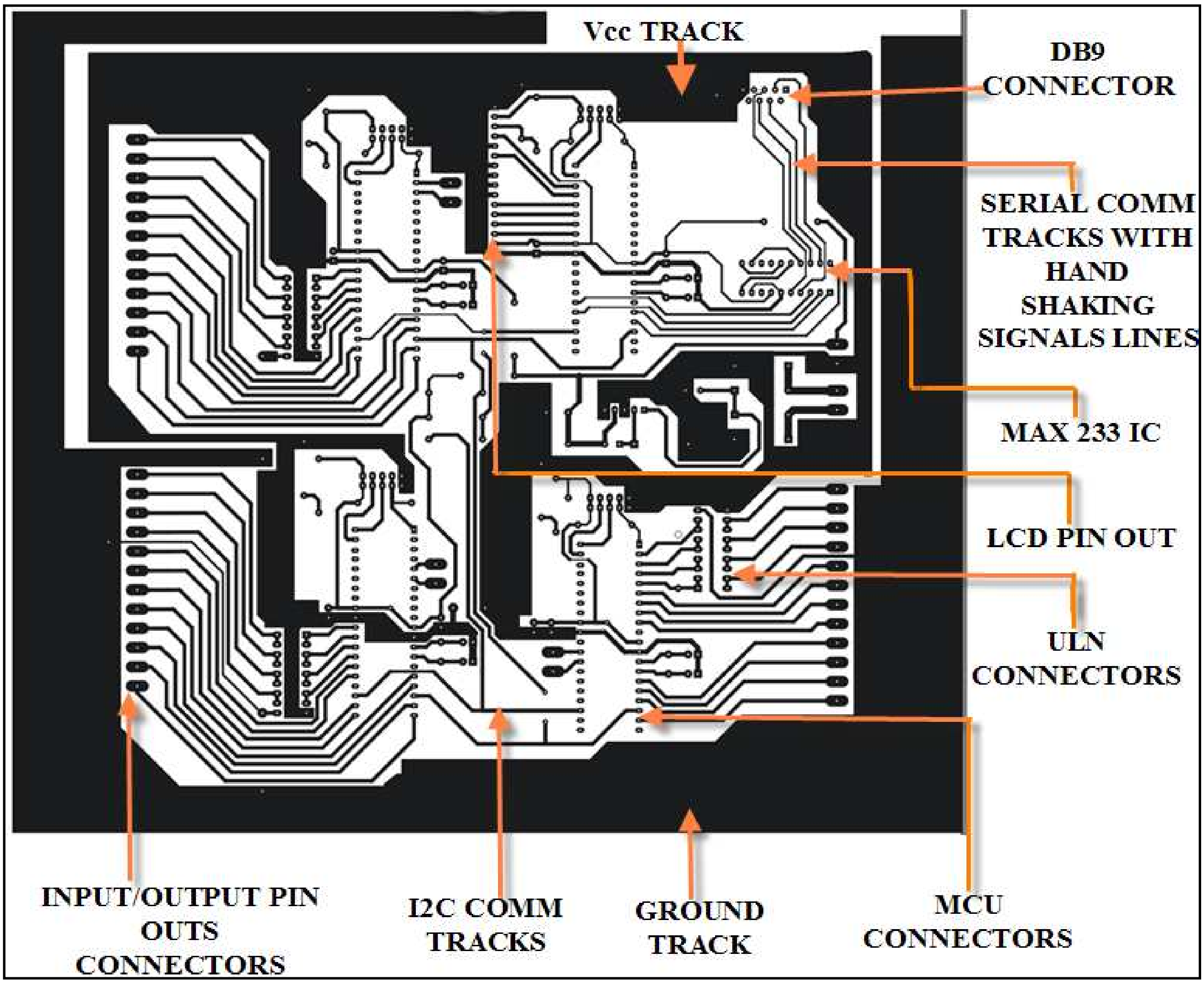



**Figure 4.3: PCB of OAC Designed in OrCAD Layout.**

## 4.3 COMMUNICATIONS TECHNIQUES

There are several communications techniques available for transfer of data between OAC system modules, including:

i. Shared memory,
ii. Bus communications,
iii. Standard communications protocols.

The method used is largely decided by the kind of topology of the communicating modules; the choice in any given situation depending on the time requirements of the particular modules involved. In the case of shared memory and bus based communications, modules which need to be interfaced with other modules must use hardware addresses. These addresses are typically

‘hard’ coded into module source code, and if modified due to changes in hardware configuration, require minor source code modifications and recompilation.
Alternatively, most standard communications protocols use 'soft' addressing schemes. In this case, the system resolves communication addresses at run time. Changes to hardware configuration require only changes in addressing the data bases. Application code remains unchanged.

### 4.3.1 Serial Communication

The personal computers offer various data and command transfer protocols, such as serial port, parallel port, Universal Serial Bus (USB) communication port, and others, the serial port was retained as the method for communicating with the controller circuit. The main advantages of using the serial port are simplicity of the method and ease of programming the serial port via the computer’s operating system. While to transfer data only a few feet away, parallel data transfer scheme utilizing often 8 or more lines (wire conductors) is used, to transfer data to a device located at any distance, the serial transmission is employed as it offers transfer of data with minimal hardware requirements. In serial communication, the data is sent one bit at a time. In parallel communication data is sent a byte or more at a time. Figure 4.4 shows the main difference in serial and parallel data transfer scheme.

The PIC microcontroller has inbuilt ready to use serial communication module. The module works independently to communicate with other devices, So the microcontroller can simultaneously control the machine control operations and communicate with other devices by USART.

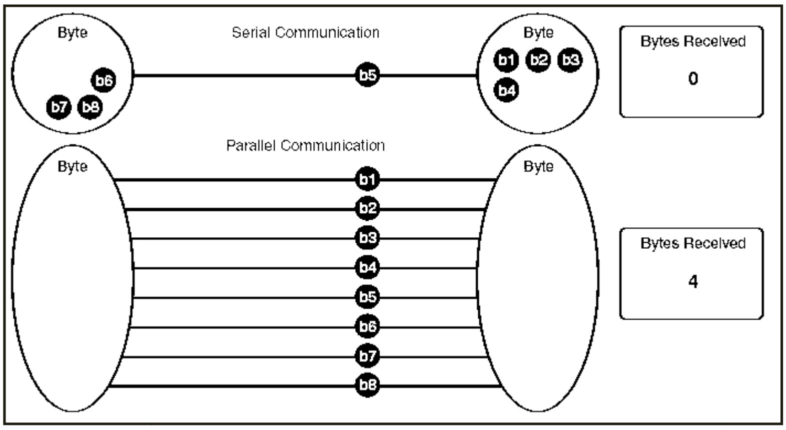



**Figure 4.4: Shows Serial vs. Parallel Data Transfer.**

#### 4.3.1.1 Basics of Serial Communication

When a microprocessor communicates with other devices or modules, it provides the byte-sized data. As already discussed, the serial communication is used for transferring data between two systems located at distances of hundreds of feet. The fact that, in serial communication one data line for transmission and one for reception is used, (instead of the 8-bit data line in parallel communication) makes it economical as well as makes it possible for two devices at different location to communicate over the telephone line.

For serial data communication, initially, the byte of data is converted to serial bits using a parallel-in-serial-out-shift register. Then it is transferred to the receiver via a single data line. This necessitates the receiver end to include a serial-in-parallel-out shift register to receive the serial data and convert them into a byte (original data).

#### 4.3.1.2 Serial Data Communication Uses Two Methods

i. Asynchronous
ii. Synchronous

The **Synchronous** method transfers a block of data at a time while the **Asynchronous** transfers a single byte at a time enclosed between start and stop bits. These two methods can be used independently by programming the PIC separately for each of them. Asynchronous serial data communication is mainly used for character-oriented data, while Block-Oriented data transfer is done using the Synchronous method. There are special I.C. chips made by many manufacturers for serial data communication. These chips are commonly known as UART (Universal Asynchronous Receiver-Transmitter) and USART (Universal Synchronous-Asynchronous Receiver-Transmitter).

#### 4.3.1.3 Asynchronous Serial Communication and Data Framing

The receiver and sending modules transfer data on the basis of some well defined rules known as the communication protocol. This Protocol defines the way the data is packed, the number of bits in a character, and also specifies the beginning and end of the data.

#### 4.3.1.4 Start and Stop Bits:

In the Asynchronous method, each character is framed by placing it between the Start and Stop bits. In Data Framing, for Asynchronous communications, the data, such as ASCII characters, are enclosed between a Start bit and a Stop bit. The Start bit is always one bit but a Stop bit can be 1 or 2 bits. The Start bit is always a 0(low) and stop bits is l(high).

#### 4.3.1.5 Data Transfer Rate

The rate of data transfer in serial data communication is stated in baud rate. This is different from the conventional unit bits per second (bps). The baud rate is defined as the number of signal changes per second while bps is number of bits transferred in one second. In modems, there are occasions when a single change of signal transfer several bits of data but in a conductor wire, the baud rate and the bps are the same (as a signal is only one bit-0 or 1).

#### 4.3.1.6 RS232 Standards

To ensure fast and reliable data transmission between two devices, the data transfer must be coordinated. An interfacing standard called RS-232 was defined by the Electronics Industries Association in 1960 to make communication equipments made by various manufacturers compatible with each other. Today, RS232 is the most widely used serial input-output interfacing standard. Many of the pins of DB-9 (RS-232 connector) are used for handshaking signals. The pins of DB-9 connector are described in table 4.1.

**Table 4.1: IBM PC DB-9 Signals**

| Pin | Pin Description |
|---|---|
| 1 | Data carrier detect (DCD) |
| 2 | Received data (RxD) |
| 3 | Transmitted data (TxD) |
| 4 | Data terminal ready (DTR) |
| 5 | Signal ground (GND) |
| 6 | Data set ready (DSR) |
| 7 | Request to send (RTS) |
| 8 | Clear to send (CTS) |
| 9 | Ring indicator (RI) |

This standard is used in PCs and many other types of equipment. DB-9 Connectors are used for serial data communication at the PC end while at the micro-controller end, DB-9 has to be interfaced via a line driver chip. The details of the serial transmission using PIC 16F877 are presented in the following section.

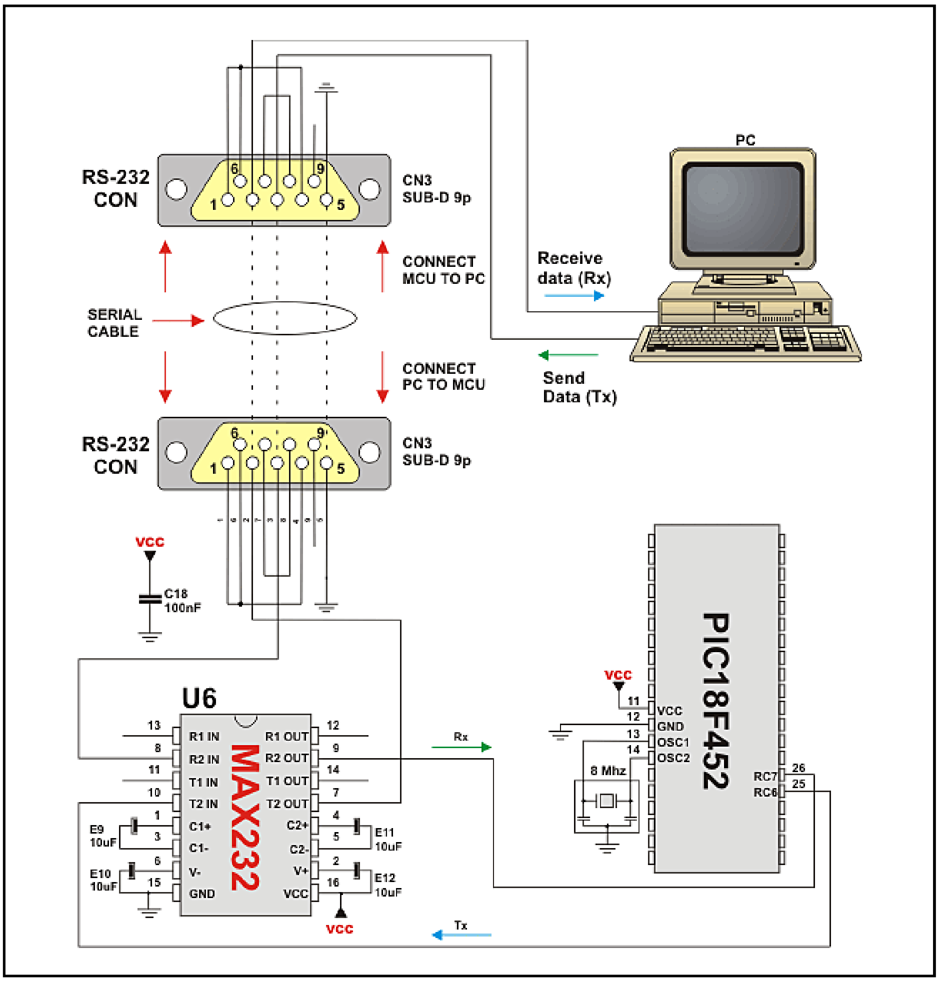



**Figure 4.5: Shows RS 232 Connections with PIC Microcontroller and PC by Using MAX 232 IC.**

### 4.4 USART Communication with PC in PIC 16f877 Microcontroller

PIC 16F877 USART provides two modes of operation, synchronous (using a separate clock signal) and asynchronous (no clock connection). In the present CNC machine design, the PIC 16F877's USART has been configured as a full-duplex asynchronous system so that it can communicate with PC.

The USART Asynchronous module consists of the following important elements:

• Baud Rate Generator

• Sampling Circuit

• Asynchronous Transmitter

• Asynchronous Receiver

In asynchronous mode, RB6 pin acts as a data transmit (TX) output and RB7 pin as data receive input (RX) and 8-bit or 9-bit data can be transmitted. The transmitter and receiver are functionally independent but use the same data format and baud rate. The USART transmits and receives the LSB first. The 9600 baud is used here with standard 8-bit data format, meaning that the bits are transmitted at about 10k bits/s. Separate transmit and receive lines are used, so it is possible for these operations to be carried out simultaneously.
In the figure 4.5, the PIC is connected to a PC via a MAX232 line driver chip. The PIC USART pins are TTL compatible and thus require an external serial line driver to convert its output into a higher symmetrical line voltage so that the signals are not attenuated during the transfer of data due to the distributed resistance and capacitance of the cabling. The standard RS232 interface has a higher line voltage levels so that the signal can be transmitted further without being hampered by line interference. RS232 can be used for distances up to 10 m with symmetrical voltages of up to +/-25V. The signal is also inverted with respect to the TTL version as shown in figure 4.6 (b).

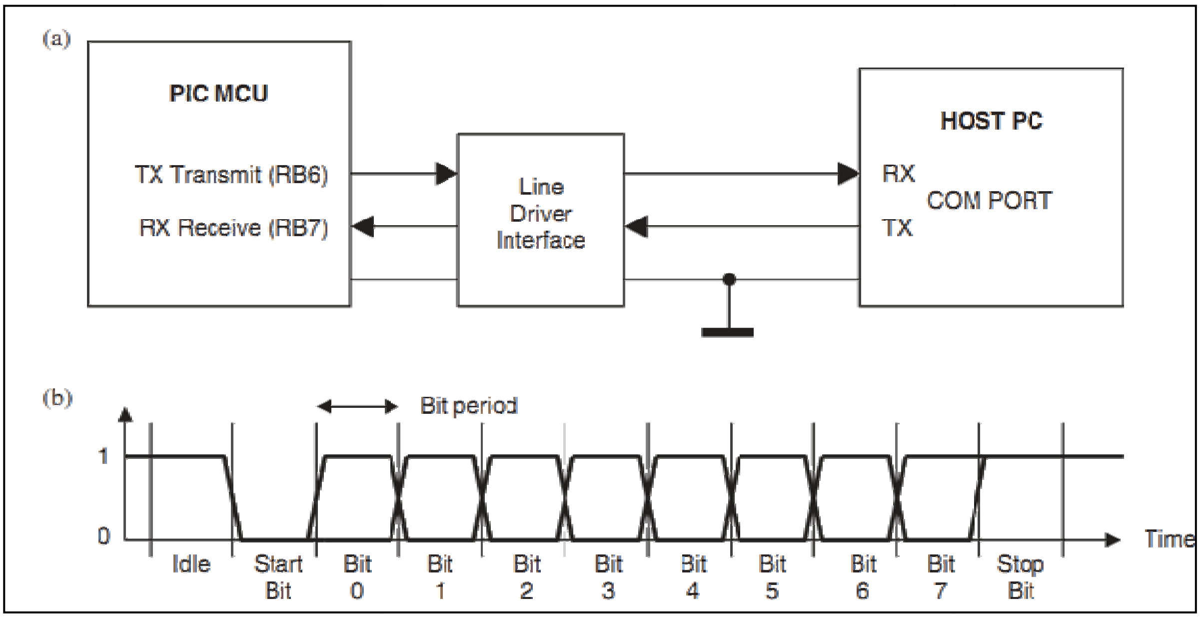



**Figure 4.6: (a) Connection to PC. (b) Signal at PIC Port.**

The transmit (TX) output is high when no data is being transmitted and when receiving the data on the RC7 pin, it is sampled three times by a majority detect circuit to determine if a high or a low level is present at the RX pin.

The following steps are used to transmit a byte from the USART.

i. The SPBRG register is initialized with value 0Fh so as to set the baud rate to 9600.
ii. The asynchronous serial port is enabled by clearing bit SYNC and setting bit SPEN of the TXSTA register.
iii. To enable the transmission, bit TXEN is set, which will also set bit TXIF.
iv. The TXREG register is loaded with the data byte to start the transmission of data.
v. The TXIF bit is monitored for high so as to ensure that the byte has been completely transmitted and the TXREG register is not overloaded

To asynchronously receive the data, the following sequence of steps is followed:

i. To set a baud rate of 9600, the SPBRG register is initialized to 0Fh.
ii. To enable the asynchronous serial port, bit SYNC is cleared and bit SPEN is set. (The first two steps have been already been done when transmission was started)
iii. The reception is enabled by setting bit CREN.
iv. Flag bit RCIF will be set when reception is complete
v. The 8-bit received data is read by reading the contents of RCREG register.

### 4.5 Master Synchronous Serial Port (MSSP) Module in the PIC 16f877A

The Master Synchronous Serial Port (MSSP) module is used communicate with slower peripheral devices, such as A/D or D/A integrated circuits, serial EEPROMs, etc. In the PIC micro controller, the MSSP provides two main types of communication: SPI (Serial Peripheral Interface) and I2C (usually pronounced I squared C). A serial peripheral interface (SPI) bus is a system for serial communication at the system level and utilizes four lines. One conductor is used for data receiving, one for data sending, one for synchronization and one alternatively for selecting a device to communicate with. Thus simultaneous transmission and reception is enabled (i.e. it a full duplex connection), making SPI faster and simpler. I2C is more complex, with software addressing, but needs only to two lines to communicate.

#### 4.5.1 I2C Communication: MSSP in Slave mode

I2C is a more versatile system level serial data transfer method.

Two pins are used for data transfer:

• Serial clock (SCL) – RC3

• Serial data (SDA) – RC4

A master controller can be connected to up to 1023 other devices such as MCUs, memory devices, analogue converters and so on. As observed from the figure 4.7, the signal lines have to be pulled up to 5 V so that any device connected to it can get the control of the line by pulling it down; hence slaves can acknowledge operations initiated by the master. Clock speeds are specified using the baud rate generator, preloading it with a suitable value enabling generation of speeds of up to 1 MHz. The transmitted byte has a start bit, which is a low bit, and an 8-bit address or data byte with MSB transmitted first, and ends with an acknowledge signal from the slave.

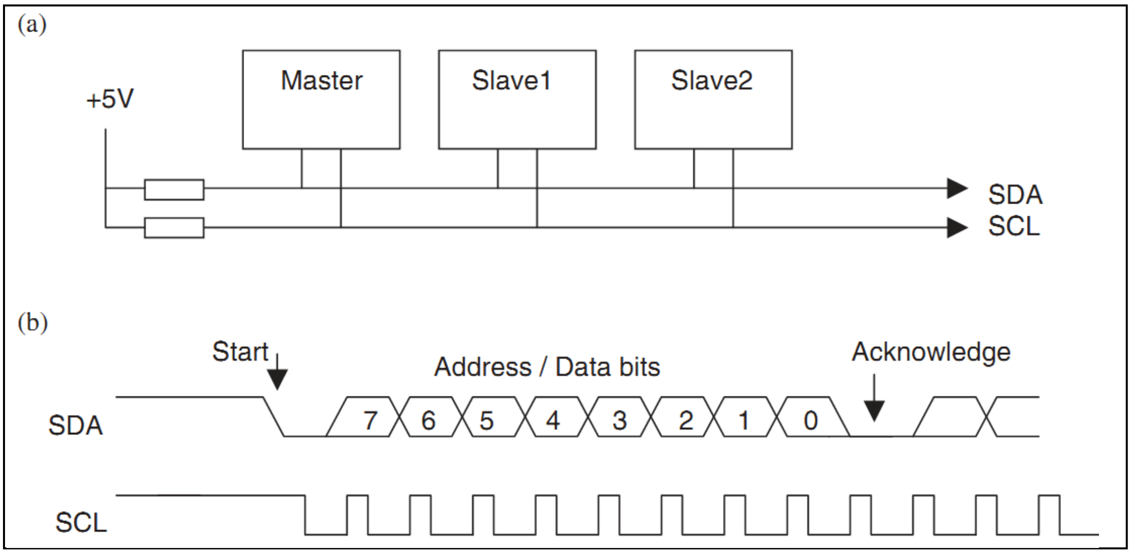



**Figure 4.7: Inter-IC Bus (I2C) (a) I2C Connections; (b) I2C Signals.**

### 4.5.2 Transmission Control in Master/Slaves I2C Communication

The master device generates all of the serial clock pulses and the Start and Stop conditions. The SDA and SCL are both kept high when idle.

In the master transmission mode (memory write mode) , serial data is output through SDA while serial clock is transmitted by SCL. The beginning of transmission is indicated by the low status on the data line (SDA), the clock then starts. The first byte transmitted contains the slave address of the receiving device (7 bits) and the Read/Write (R/W) bit (R/W bit will be logic '0'). Serial data is transmitted one byte at a time.

During each byte transfer, each bit (address or data) is output during the clock high period, and on the falling clock edge, is latched into the slave receive shift register. After the eighth bit, the master (MCU) relinquishes the data line and allows the slave to hold the line low to acknowledge that the bits have been received. At the end of the next (ninth) clock high period, the slave releases the control of the data line and then the master can transmit the first bit of the next byte. The Start and Stop conditions are output to indicate the beginning and the end of a serial transfer.

In Master Receive mode (memory read mode), the first byte transmitted is a 7-bit slave address followed by a ‘1’ to indicate the receive bit. As in memory write, serial data is received 8 bits at a time. In the memory read sequence, the master stops after the address write has been sent, and restarts in order to send a read control byte. It will then read the eight data bits returned by the memory chip, but does not generate acknowledge. The master then stops, and the lines go idle again. Figure 4.8 shows interfacing 24c02 IC to PIC via I2C lines.

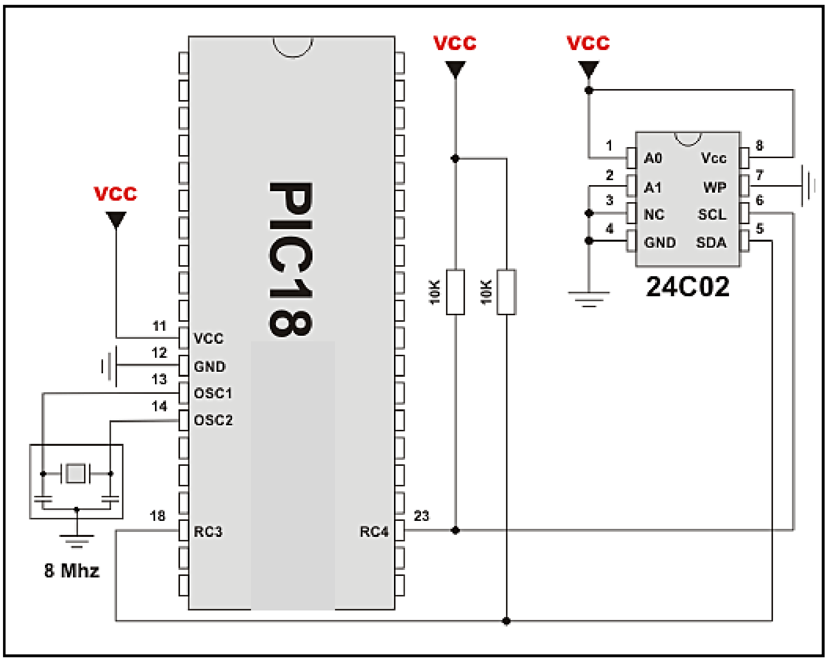



**Figure 4.8: Interfacing 24c02 to PIC via I2C Lines**

The shift register of micro controller used to send and receive the data bits cannot be directly accessed. The buffer register SSPBUF represents its contents. The SSPBUF holds the data until the shift register is ready to send it (transmit mode), or receives it when the shift is finished in receive mode. Setting the Send Enable (SEN) bit triggers the send operation, and the status of the Buffer Full (BF) flag indicates that the data have been loaded. The interrupt flag (SSPIF) is automatically set to indicate the beginning of transmission, and must be cleared in software when required. The Acknowledge Status (ACKSTAT) bit is cleared after acknowledge is received from the slave, to indicate that the byte has been received. SSPIF is then set again to indicate the end of the byte transmission, and then the next byte is written into the buffer.

### 4.6 Open Architecture Controller for Vertical 3 Axis NC Machine

To control the fabricated three axis of NC machine as shown in figure 4.9, three dedicated PIC16f877A micro controllers have been used. Fourth micro controller called as Master micro controller guides these three Slaves micro controllers. All three Slaves micro controllers communicate with fourth Master micro controller via I2C lines. Master micro controller communicates with PC via USART communication.

Three motors are driven by dedicated slave microcontroller. The three Slave independent microcontrollers are solely busy for controlling their corresponding motor’s motion and position. Slave microcontrollers control the position by taking feedback from encoders. Figure 4.10 shows the developed controller circuit of OAC for 3 axis vertical milling machine. New kind of sensors and other peripheral devices can be easily added to this OAC (Open Architecture Controller). Most of the pins on each microcontroller are idle. These idle pins can be used as input or output of controller from real physical world. Figure 4.11 shows the flow of signals among NC machine drives, controller and personal computer.

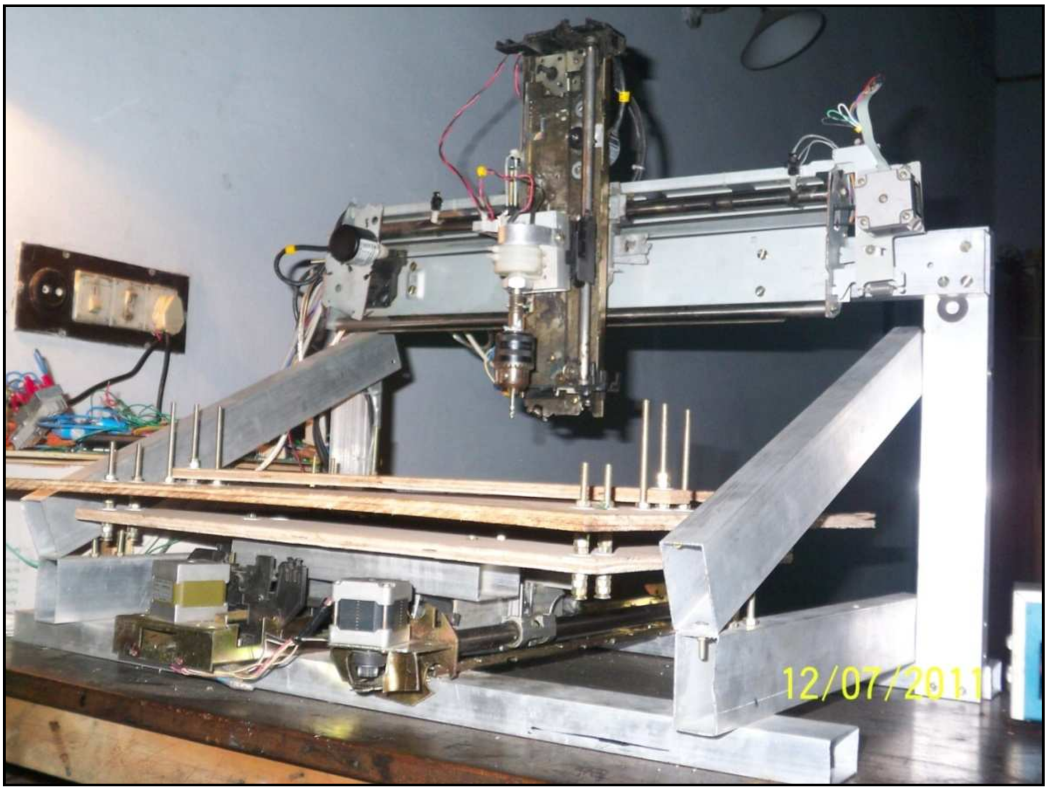



**Figure 4.9: Fabricated 3-Axis NC Machine to Validate Developed OAC.**

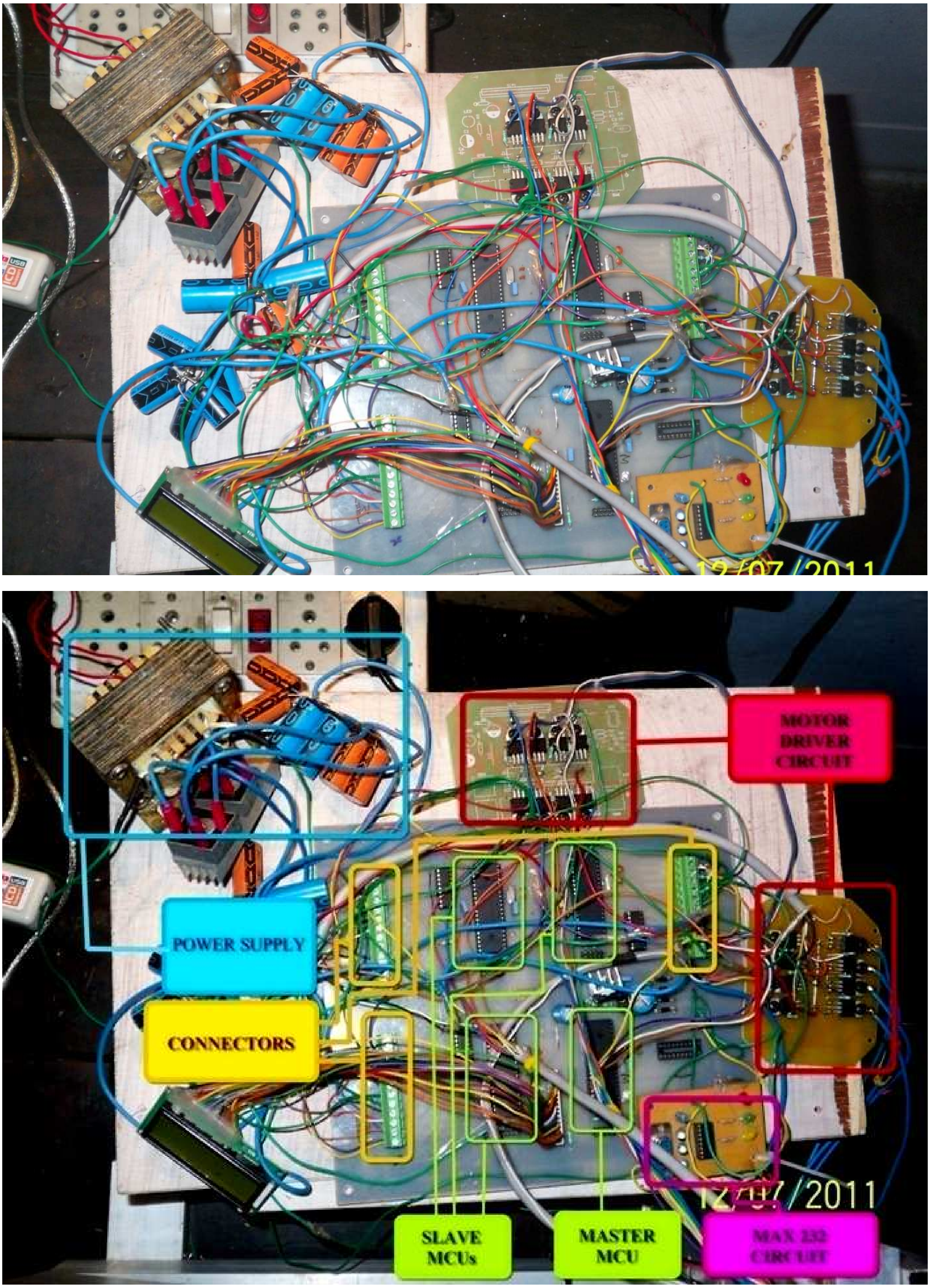



**Figure 4.10: Developed Controller Circuit of OAC for 3 Axis Vertical Milling Machine**

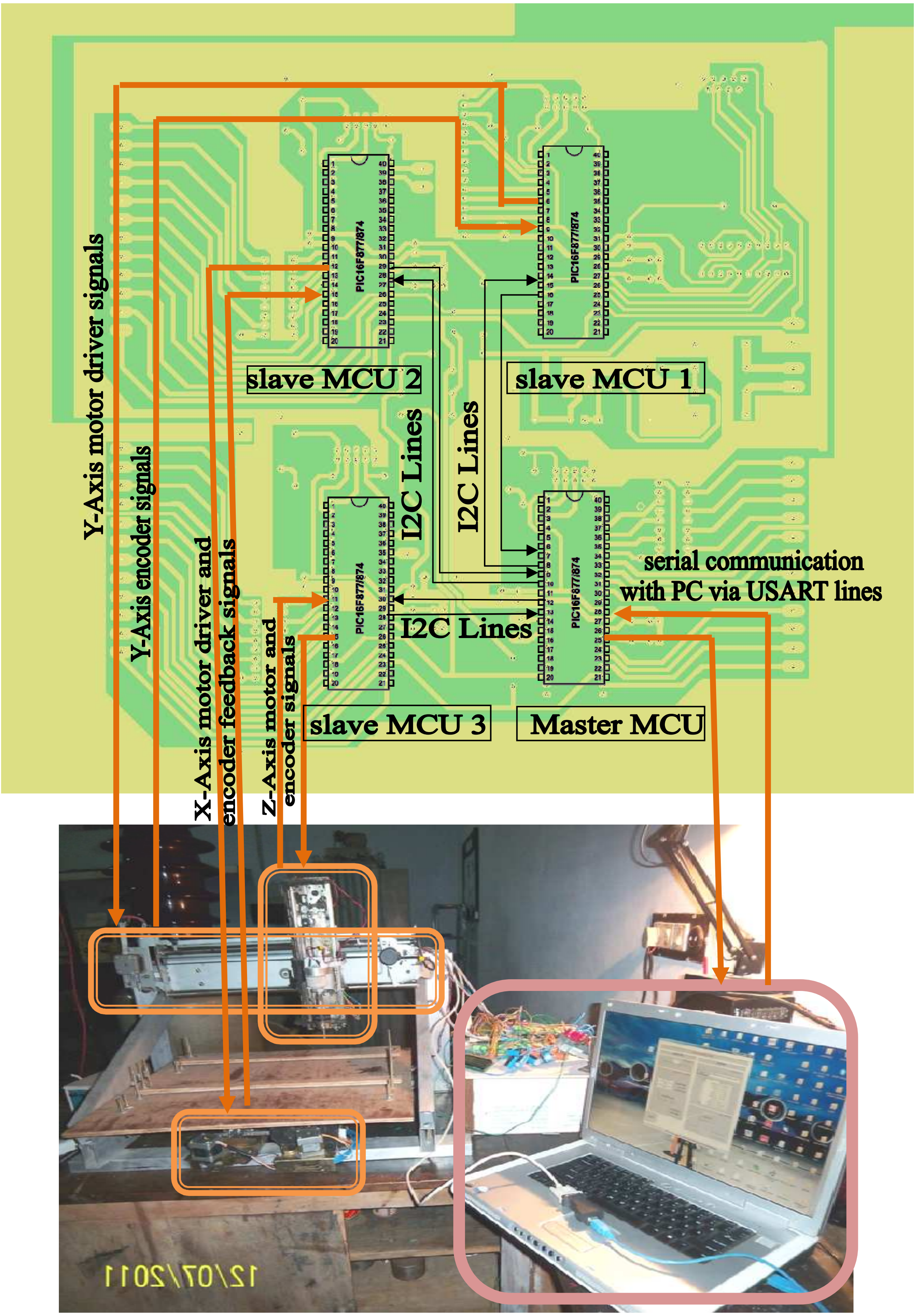



Figure 4.11: Signal Flow Lines among Machine, Microcontrollers and PC.

### 4.7 Common Errors While Working With Electronics.

Design always goes through iteration process. It is very rare a design is finalized in first trial. In designing of PCB for any controller is completely modified after its first trail. While testing initial designed PCB many changes need to be done in circuit such as adding or removing of new electronic components and routing and deleting tracks on PCB. Common errors or mistake while assembling components on PCB and testing controller circuit needs to be avoided. Some are discussed below.

#### 4.7.1 Dry Solder

Dry soldering is common error while assembling electronic components on PCB. In dry solder joint appear to be fine but due to improper soldering the joint is not complete. While testing PCB sometimes it may connect with circuit and circuit appears to be work fine then in next test joint may disconnect and circuit starts misbehaving. This error is critical as it does not locate so easily because everything appears soldered fine and cause big trouble while testing.

#### 4.7.2 Sharing of Ground Track

Microcontroller, limit switches and encoder works on 5V and 500mA current. While collective load of motors is 5V and 8Amp current. No doubt voltage rating is same but current rating is different. If ground of limit switch or encoder circuit shared with stepper motor ground then drastic drop of current seen in motor which leads to drop in motor torque.

#### 4.7.3 Switch Bounce

When a mechanical switch operates, “contact bounce” phenomena come into picture. In most switches, the contact separation operation (“break”) is relatively clean, but the contact closure (“make”) exhibits multiple bounce events. A figure 4.12 show nearly 1.5 ms is required to reach a steady state closed condition in the microswitch switch being tested. This bouncing of contact is enough to generate interrupt to microcontroller which shows multiple responses while pressing key single time. Debouncing of switch is needed and done by two ways, Software debouncing and Hardware debouncing.

Software debouncing is done in programming level were microcontroller has to wait for few millisecond after getting first rising edge when was being pressed so that bouncing signal get settle down then look for second interrupt.

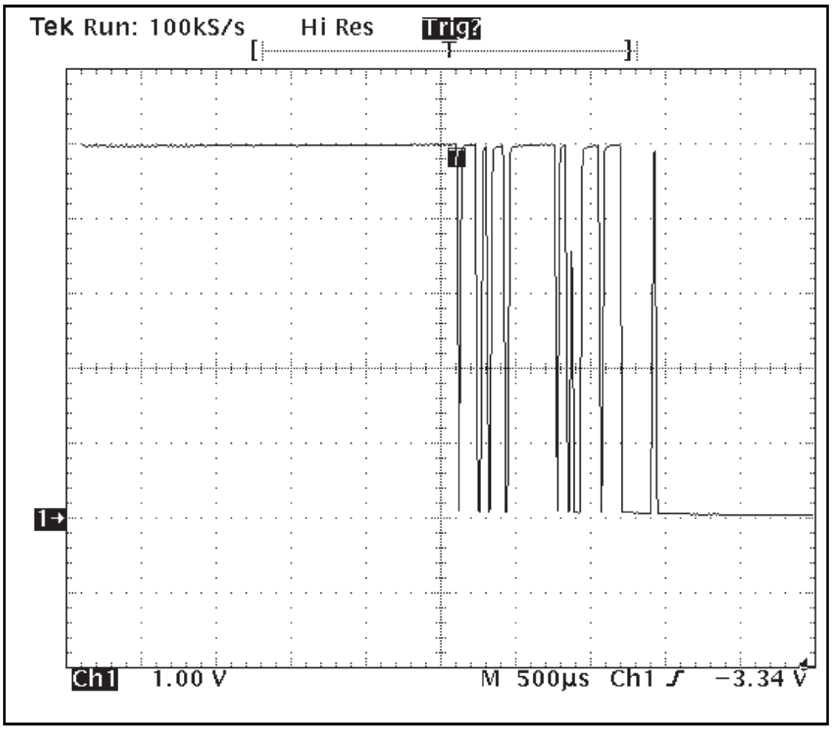



**Figure 4.12: Contact Bounce upon Closure of Microswitch.**

Hardware debouncing is done by external circuit as shown in figure 4.13. When the switch is open, C1 charges to VDD and RC0 is read as high. When the switch makes, C1 is discharged through R2 (D1 is reversed biased and may be neglected) and will be read as low when the voltage across C1drops below high-to-low transition voltage, approximately 1.8 V.

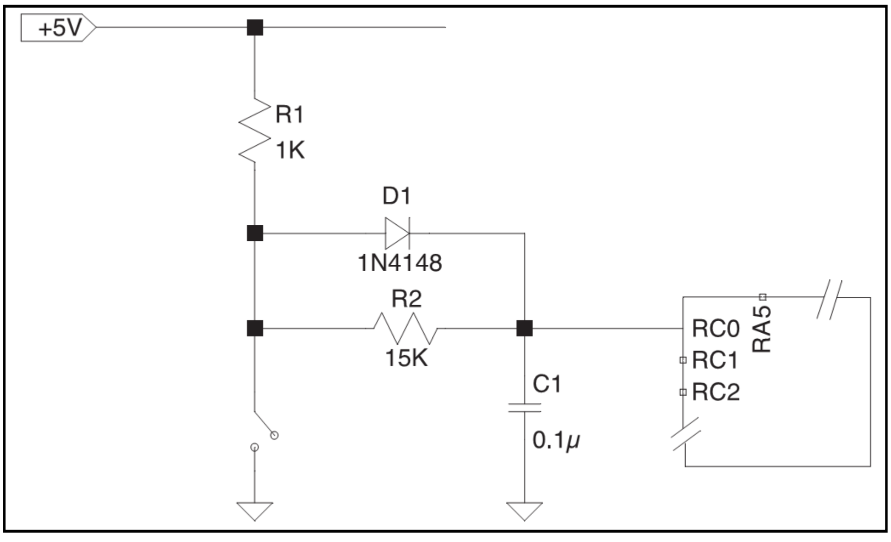



**Figure 4.13: Hardware Debounce**

If the time constant of R2-C1 is long compared with the individual bounce intervals, the decay will be smooth and only one transition through $V_{THRESHOLD}$ will occur. But, even if spikes of several hundred millivolts occur at RC0 the output will stay low, as in order to change its read state, RC0 must see the low-to-high transition voltage of approximately 3.1 V. When the switch is closed, the input pin is connected to ground through R2. The leakage current from the PIC input pin is rated not to exceed 1µA, so in the worst case with R2 at 15K ohm, the input pin will be at 0.015 V, well within the logical low range. When the switch is opened after closure, C1 charges through R1 and R2 in parallel with D1. Until the voltage across C1 reaches 0.7 V from VDD, C1 charges mostly through R1 and D1. Hence, the charge cycle is significantly shorter than the discharge cycle. This design assumes that the switch has bounce problems only on make and therefore little or no anti-bounce effect is required on break.

# CHAPTER 5

# ELECTRICAL COMPONENTS FOR NC MILLING MACHINE

---

To control motor motion from electronic digital signals precisely there must be use of electronically commutated driven motors. In present work, stepper motor has been used, as linear and precise motion is needed which BLDC (Brushless DC Motor) motor cannot achieve as BLDC motor has 30 degree minimum step angle. For getting feedback of position rotary increment encoders with 1000 pulses per revolution have been used. To run whole machine and components specific power supply was needed which was designed according to machine load.

## 5.1 Rotary Stepper Motor

Stepping motors fill a unique niche in the motor control world. These motors are commonly used in measurement and control applications. Sample applications include ink jet printers, CNC machines and volumetric pumps. Several features common to all stepper motors make them ideally suited for these types of applications. These features are as follows:

i. **Brushless –** Stepper motors are brushless. The commutator and brushes of conventional motors are some of the most failure-prone components, and they create electrical arcs that are undesirable or dangerous in some environments.
ii. **Load Independent –** Stepper motors will turn at a set speed regardless of load as long as the load does not exceed the torque rating for the motor.
iii. **Open Loop Positioning –** Stepper motors move in quantified increments or steps. As long as the motor runs within its torque specification, the position of the shaft is known at all times without the need for a feedback mechanism.
iv. **Holding Torque –** Stepper motors are able to hold the shaft stationary.
v. **Excellent response** to start-up, stopping and reverse.

### 5.2.1 Types of stepping motors

There are three basic types of stepping motors:

i. Permanent Magnet,
ii. Variable Reluctance

iii. Hybrid Stepper Motor.

Permanent magnet motors have a magnetized rotor, while variable reluctance motors have toothed soft-iron rotors. Hybrid stepping motors combine aspects of both permanent magnet and variable reluctance technology. The stator or stationary part of the stepping motor holds multiple windings. The arrangement of these windings is the primary factor that distinguishes different types of stepping motors from an electrical point of view. From the electrical and control system perspective, variable reluctance motors are distant from the other types. Both permanent magnet and hybrid motors may be wound using either unipolar windings, bipolar windings or bifilar windings.

### 5.2.2 Unipolar Versus Bipolar

Permanent magnet and hybrid stepping motors are available with either unipolar, bipolar or bifilar windings; the latter can be used in either unipolar or bipolar configurations. The choice between using a unipolar or bipolar drive system rests on issues of drive simplicity and power to weight ratio. Bipolar motors have approximately 30% more torque than an equivalent unipolar motor of the same volume. The reason for this is that only one half of a winding is energized at any given time in a unipolar motor. A bipolar motor utilizes the whole of a winding when energized. The higher torque generated by a bipolar motor does not come without a price. Bipolar motors require more complex control circuitry than unipolar motors this will have an impact on the cost of an application. If in doubt, a unipolar motor or bifilar motor are good choices. These motors can be configured as a unipolar or bipolar motor and the application tested with the motors operating in either mode

### 5.2.3 Bipolar Basics

The Bipolar Stepper motor shown in figure 5.1 has 2 coils. The coils are identical and are not electrically connected. The separate coils can be identified by touching the terminal wires together. If the terminals of a coil are connected, the shaft becomes harder to turn.

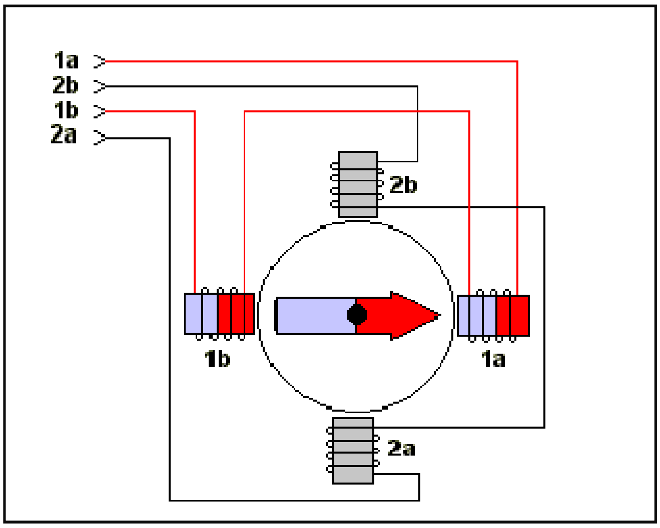


**Figure 5.1: Conceptual Model of Bipolar Stepper Motor.**

The coils are activated, in sequence, to attract the rotor, which is indicated by the arrow in the picture. (Remember that a current through a coil produces a magnetic field.) This conceptual diagram depicts a 90 degree step per phase. Assuming Terminal 1a is positive and 1b is negative, the rotor points to the East in this diagram. If these two terminals were reversed in polarity the rotor would point to the West. Coil 2 is entirely de-activated in the diagram. In a basic "Wave Drive" clockwise sequence, winding 1 is de-activated and winding 2 activated to advance to the next phase. The rotor is guided in this manner from one winding to the next, producing a continuous cycle. Note that if two adjacent windings are activated, the rotor is attracted mid-way between the two windings. The table 5.1 describes 3 useful stepping sequences and their relative merits. The polarity of terminals is indicated with +/-. After the last step in each sequence the sequence repeats. Stepping backwards through the sequence reverses the direction of the motor.

**Table 5.1: 3 Useful stepping sequences**

| Sequence | Polarity | Name | Description |
|---|---|---|---|
| 0001<br>0010<br>0100<br>1000 | ---+<br>--+-<br>-+--<br>+--- | Wave Drive, One-Phase | Consumes the least power. Only one phase is energized at a time. Assures positional accuracy regardless of any winding imbalance in the motor. |
| | | | Hi Torque - This sequence energizes two adjacent phases, |

| | | | |
|---|---|---|---|
| 0011<br>0110 | --++<br>-++- | Hi-Torque, Two-Phase | which offers an improved |
| 1100<br>1001 | ++--<br>+--+ | | torque-speed product and greater holding torque. |
| 0001<br>0011<br>0010<br>0110<br>0100<br>1100<br>1000<br>1001 | ---+<br>--++<br>--+-<br>-++-<br>-+--<br>++--<br>+---<br>+--+ | Half-Step | Half Step - Effectively doubles the stepping resolution of the motor, but the torque is not uniform for each step. This sequence reduces motor resonance which can sometimes cause a motor to stall at a particular resonant frequency. Note that this sequence is 8 steps. |

### 5.2.4 Torque vs. Angle Characteristics

The torque vs angle characteristics of a stepper motor are the relationship between the displacement of the rotor and the torque which applied to the rotor shaft when the stepper motor is energized at its rated voltage. An ideal stepper motor has a sinusoidal torque vs displacement characteristic as shown in figure 5.2.

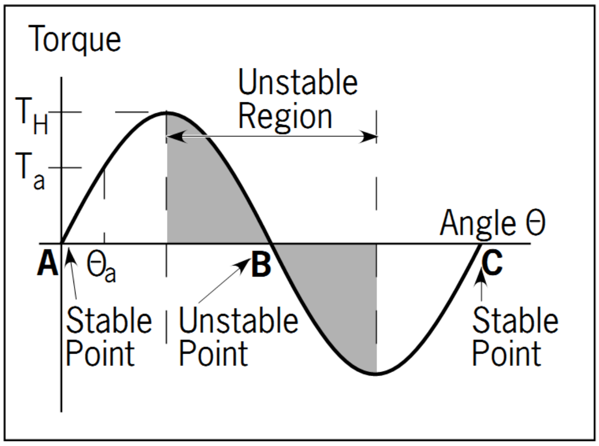



**Figure 5.2: Torque vs. Rotor Angular Position.**

Positions A and C represent stable equilibrium points when no external force or load is applied to the rotor shaft. When external force $T_a$ applied to the motor shaft then an angular displacement is generated, Θa. This angular displacement, Θa, is referred to as a lead or lag angle depending on wether the motor is actively accelerating or decelerating. When the rotor stops with an applied load it will come to rest at the position defined by this displacement angle. The motor develops a torque, $T_a$, in opposition to the applied external force in order to balance the load. As the load is increased the displacement angle also increases until it reaches the maximum holding torque, $T_h$, of the motor. Once $T_h$ is exceeded the motor enters an unstable region. In this region a torque is the opposite direction is created and the rotor jumps over the unstable point to the next stable point.

### 5.2.5 Step Angle Accuracy

One reason why the stepper motor has achieved such popularity as a positioning device is its accuracy and repeatability. Typically stepper motors will have a step angle accuracy of 3 – 5% of one step. This error is also noncumulative from step to step. The accuracy of the stepper motor is mainly a function of the mechanical precision of its parts and assembly. Figure 5.3 shows a typical plot of the positional accuracy of a stepper motor.

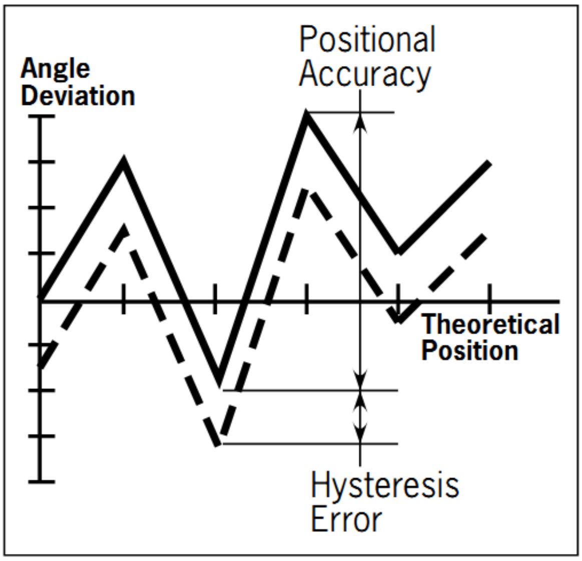



**Figure 5.3: Positional Accuracy of a Stepper Motor.**

Step Position Error The maximum positive or negative position error caused when the motor has rotated one step from the previous holding position.

Step position error = measured step angle - theoretical angle

### 5.2.6 Torque vs. Speed Characteristics

The torque vs. speed characteristics are the key to selecting the right motor and drive method for a specific application. These characteristics are dependent upon (change with) the motor, excitation mode and type of driver or drive method. A typical "speed – torque curve" is shown in figure 5.4. To get a better understanding of this curve it is useful to define the different aspect of this curve.

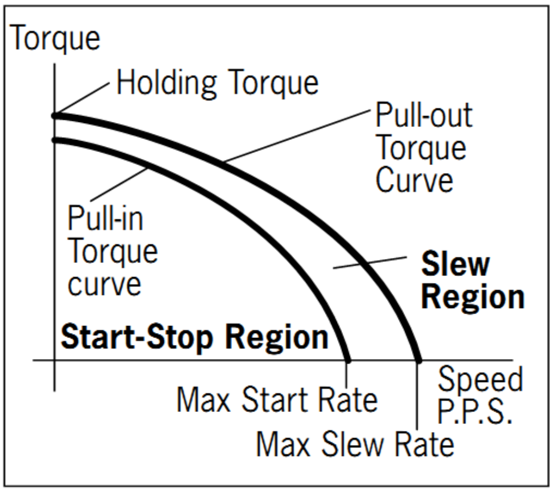



**Figure 5.4: Torque vs. Speed Characteristics of Stepper Motor.**

i. Holding torque The maximum torque produced by the motor at standstill.
ii. Pull-In Curve The pull-in curve defines an area referred to as the start stop region. This is the maximum frequency at which the motor can start/stop instantaneously, with a load applied, without loss of synchronism.
iii. Maximum Start Rate The maximum starting step frequency with no load applied.
iv. Pull-Out Curve The pull-out curve defines an area refered to as the slew region. It defines the maximum frequency at which the motor can operate without losing synchronism. Since this region is outside the pull-in area the motor must ramped (accelerated or decelerated) into this region.
v. Maximum Slew Rate The maximum operating frequency of the motor with no load applied. The pull-in characteristics vary also depending on the load. The larger the load inertia the smaller is the pull-in area. The shape of the curve has shown that the step rate affects the torque output capability of stepper motor The decreasing torque output as the speed increases is caused by the fact that at high speeds the inductance of the motor is the dominant circuit element.

### 5.2.7 Single Step Response and Resonances

The single-step response characteristic of a stepper motor is shown in figure 5.5. When one step pulse is applied to a stepper motor the rotor behaves in a manner as defined by the above curve. The step time t is the time it takes the motor shaft to rotate one step angle once the first step pulse is applied. This step time is highly dependent on the ratio of torque to inertia (load) as well as the type of driver used. Since the torque is a function of the displacement it follows that the acceleration will also be. Therefore, when moving in large step increments a high torque is developed and consequently a high acceleration comes into picture. This can cause overshoots and ringing as shown. The settling time T is the time it takes these oscillations or ringing to cease. In certain applications this phenomena can be undesirable. It is possible to reduce or eliminate this behavior by microstepping the stepper motor. For more information on microstepping please consult the microstepping note.

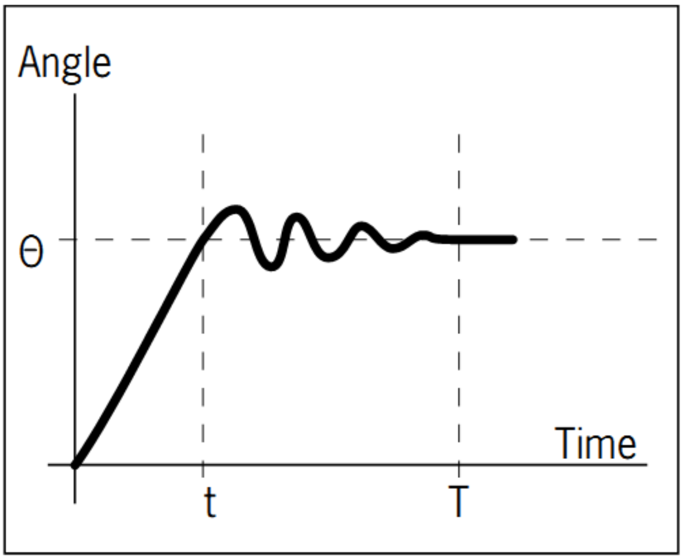



**Figure 5.5: Single Step Response vs. Time.**

Stepper motors can often exhibit a phenomena referred to as resonance at certain step rates. This can be seen as a sudden loss or drop in torque at certain speeds which can result in missed steps or loss of synchronism. It occurs when the input step pulse rate coincides with the natural oscillation frequency of the rotor. Often there is a resonance area around the 100 – 200 pps region and also one in the high step pulse rate region. The resonance phenomena of a stepper motor come from its basic construction and therefore it is not possible to eliminate it completely.

It is also dependent upon the load conditions. It can be reduced by driving the motor in half or microstepping modes.

### 5.2.8 Mechanical Parameters

The performance of a stepper motor system (driver and motor) is also highly dependent on the mechanical parameters of the load. The load is defined as what the motor drives. It is typically frictional, inertial or a combination of the two.

Friction is the resistance to motion due to the unevenness of surfaces which rub together. Friction is constant with velocity. A minimum torque level is required throughout the step in over to overcome this friction (at least equal to the friction). Increasing a frictional load lowers the top speed, lowers the acceleration and increases the positional error. The converse is true if the frictional load is lowered.

Inertia is the resistance to changes in speed. A high inertial load requires a high inertial starting torque and the same would apply for braking. Increasing an inertial load will increase speed stability, increase the amount of time it takes to reach a desired speed and decrease the maximum self start pulse rate. The converse is again true if the inertia is decreased.

The rotor oscillations of a stepper motor will vary with the amount of friction and inertia load. Because of this relationship unwanted rotor oscillations can be reduced by mechanical damping means however it is more often simpler to reduce these unwanted oscillations by electrical damping methods such as switch from full step drive to half step drive.

### 5.2.9 Specification of Rotary Stepper Motor Used for OAC NC Milling Machine.

Stepper motor of Minebea Corporation Limited has been installed in the machine. Before selecting stepper motor, designer has to know how much load will have to bear by motor. How much fine step of motor is needed? As resolution of machine directly depend on step angle. Which driver circuit will be used to run motor? In some cases heat generation by motor also taken in consideration where heat rejection has particular effect on any of machine component like on circuit, machine structure or mechanism or on machine installation environment.

Power levels for IC-driven stepper motors typically range from below a watt for very small motors up to 10 – 20 watts for larger motors. The maximum power dissipation level or thermal limits of the motor are seldom clearly stated in the motor manufacturer’s data. To determine this the relationship P =V ×I can be used. For example, a step motor may be rated at 6V and 1A per phase. Therefore, with two phases energized the motor has a rated power dissipation of 12 watts. It is normal practice to rate a stepper motor at the power dissipation level where the motor case rises 65°C above the ambient in still air. Therefore, if the motor can be mounted to a heat sink it is often possible to increase the allowable power dissipation level. This is important as the motor is designed to be and should be used at its maximum power dissipation ,to be efficient from a size/output power/cost point of view.

Motors used in OAC NC 3-axis milling machine is of model 17PM-KC03-P1L:

i. Power rating 5V × 2 Amp = 10 watt.
ii. Step angle 1.8 degree
iii. Holding torque 200 mNm.
iv. Unipolar.

## 5.3 Rotary Encoders

Rotary incremental encoder has been used to get feedback of motor motion. As stepper motors are uses which never travel beyond its step angle but there are chance missing steps as load increases. Secondly stepper when stepper motor is off then no winding is excited and rotar is not aligned with stator excitation. As the first pulse is come from microcontroller then any of three winding will excited according to signal then rotor immediately aligned to excited winding no matter how much angle rotor has rotate. So to insure step missing rotary incremental encoder has been used to develop 3-axis OAC NC milling machine.

### 5.3.1 Incremental Encoder

Incremental encoders provide a specific number of equally spaced pulses per revolution (PPR) or per inch or millimeter of linear motion. A single channel output as shown in figure 5.6 is used for applications where sensing the direction of movement is not important.

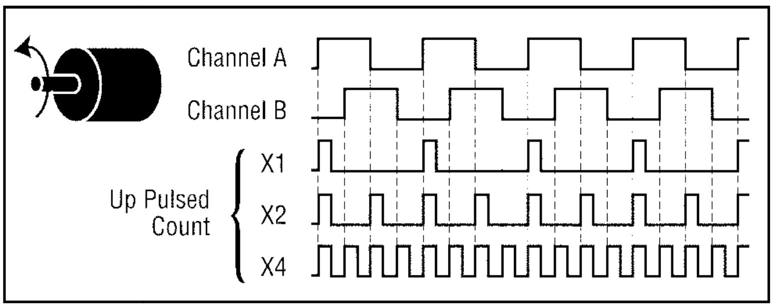



**Figure 5.6: Incremental Encoder Output Wave Form.**

Where direction sensing is required, quadrature output is used, with two channels 90 electrical degrees out of phase; circuitry determines direction of movement based on the phase relationship between them. This is useful for processes that can reverse, or must maintain net position when standing still or mechanically oscillating. For example, machine vibration while stopped could cause a unidirectional encoder to produce a stream of pulses that would be erroneously counted as motion. The controller would not be fooled when quadrature counting is used. When more resolution is needed, it's possible for the counter to count the leading and trailing edges of the pulse train from one channel, which doubles (×2) the number of pulses counted for one rotation or inch of motion. Counting both leading and trailing edges of both channels will give 4 × resolutions.

An incremental encoder's output indicates motion. To determine position, its pulses must be accumulated by a counter. The count is subject to loss during a power interruption or corruption by electrical transients. When starting up, the equipment must be driven to a reference or home position to initialize the position counters. Some incremental encoders also produce another signal known as the "marker," "index," or "Z channel." This signal, produced once per revolution of a shaft encoder or at precisely-known points on a linear scale, is often used to locate a specific position, especially during a homing sequence.

### 5.3.2 Sensing Direction of Rotation with Incremental Encoder

Commonly a simple incremental encoder has four to five output wires. Among of four wires two wires are for supply to encoder. One wire goes to Vcc other to ground as shown in figure 5.7.

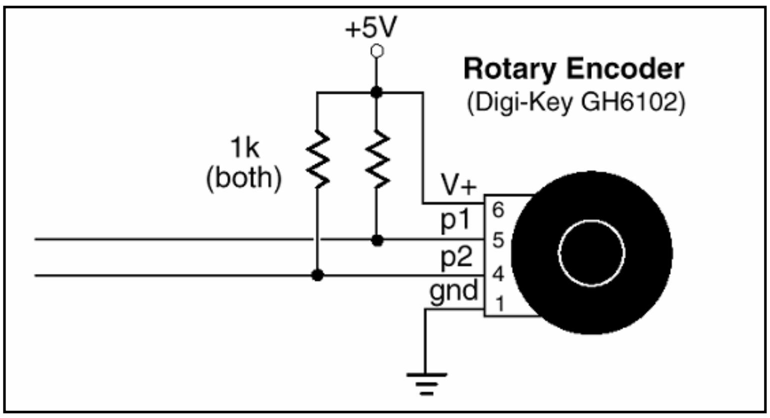



**Figure 5.7: Connection Diagram for the Rotary-Encoder**

Other two wires gives the output square wave form signals. There is phase shift between these two signals which would be helpful for detecting direction of rotation of motor. Figure 5.8 shows output form from P1(phase one) and from P2(phase two).

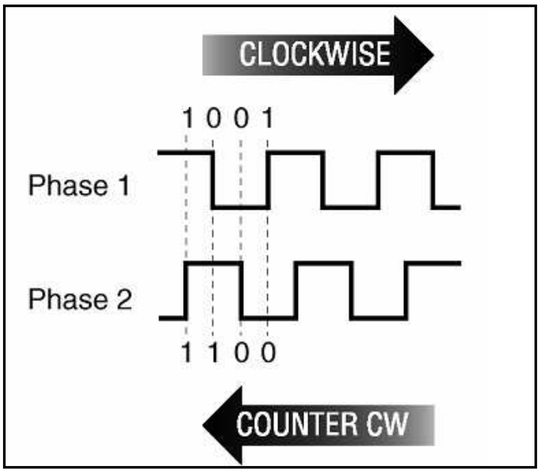



**Figure 5.8: Phase Shift in Output Signal from Two Wires.(courtsy** DELTA™**)**

To detect direction simple XOR (exclusive OR) operation gives the direction of rotation.

(Phase 1 ) XOR ( Phase 2) = Output

0 ^ 0 = 0

1 ^ 0 = 1

0 ^ 1 = 1

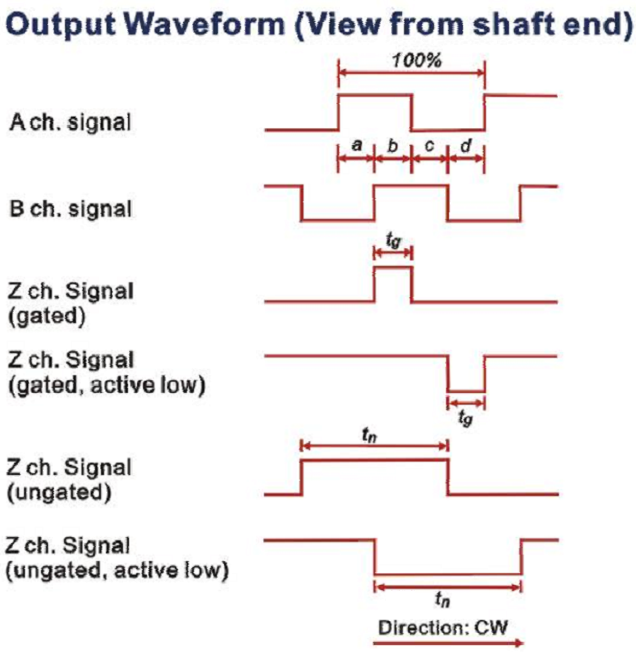
1 ^ 1 = 0

XOR has quite a few uses in programming. Any bit XORed with 1 is inverted but bits XORed with 0 are unchanged. If two bits are equal they XOR to 0; if not equal they XOR to 1. In the case of the encoder sequence, it turns out that for any given sequence, XORing the righthand bit of the old value with the lefthand bit of the new value tells the direction of rotation. For example, take the clockwise sequence 01 00: 1 XOR 0 = 1. Now the counter-clockwise sequence 01 11: 1 XOR 1 = 0. This relationship holds for any pair of numbers in either direction.

### 5.3.3 Specification of Rotary Incremental Encoder Used for OAC NC Milling Machine

DELTA™ rotary optical encoder model ES3-10LN6541 shown in figure 5.9 has been fitted in designed OAC NC machine. This model has operating voltage 5V and on output it gives 1000 pulses per revolution (PPR). This model has nine wires. The function of each wires and output wave form has shown in figure 5.10.

**Figure 5.9: DELTA$^{TM}$ Rotary Optical Encoder Model ES3-10LN6541**




- Vcc: It depends on model type (EXX-XXXXX5XX: 5V or EXX-XXXXX8XX: 5~12V or EXX-XXXXX9XX: 7~24V).
- Voltage Output: Internal 2KΩ pull up resistor is included.
- Line Driver: It is compatible with 26LS32 differential line receiver.

| Wire Color | Brown | Blue | Black | Black /Red | White | White /Red | Orange | Orange /Red |
|---|---|---|---|---|---|---|---|---|
| Function | Vcc | 0V | A | $\overline{A}$ | B | $\overline{B}$ | Z | $\overline{Z}$ |
| Voltage Output | ○ | ○ | ○ | - | ○ | - | ○ | - |
| Open Collector | ○ | ○ | ○ | - | ○ | - | ○ | - |
| Line Driver | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |
| Push Pull | ○ | ○ | ○ | - | ○ | - | ○ | - |

**Figure 5.10: Output Wave Form and Output Signal Wires.**

### 5.4 Power supply

Power supply for NC milling machine has been calculated from the maximum power rating of stepper motor and drill motor. Power consumption microcontrollers and encoders are far less than load carrying motors. This 3 axis OAC NC milling machine has four stepper motor and one geared DC motor. Each stepper motor has maximum rating of 5V and 2 Amp current hence one stepper motor load is 10 watt. Geared DC motor runs on 12V and draws max 0.5 Amp current hence load by DC motor is 6 watt. Microcontroller runs on 5V and draws 0.025 Amp current therefore micro controller has negligible load. Hence power supply for NC milling machine should be of 5V and capable of giving 8 Amp current on full load condition and there should no drop of voltage at maximum loading situation. For geared DC motor small 12V adapter has been used. DC motor speed and direction is not controlled by microcontroller hence drill motor has to start manually.

Customized transformer has been made of 8Amp current on 5 volts. This requires proper selection of number of turns in secondary winding of transformer and right selection of wire gauge in primary and secondary coils. Following equation gives relations between number of turns in primary and secondary windings, voltage requirement and current rating.

$$\frac{V_1}{V_2} = \frac{N_1}{N_2} = \frac{I_2}{I_1}$$

$V_1/V_2$ is called the voltage ratio and $N_1/N_2$ is the turn's ratio, or the 'transformation ratio' of the transformer. If $N_2$ is less than $N_1$ and $V_2$ is less than $V_1$ the device is termed a step-down transformer. The rating of a transformer is stated in terms of the volt-amperes that it can transform without overheating. The transformer rating is either $V_1$ $I_1$ or $V_2$ $I_2$, where $I_2$ is the fullload secondary current. Transformer made for NC machine has 46 turn ratio and wire size is 18 AWG (American Wire Gauge) shown in table 5.1.

**Table 5.1**

| WIRE GAUGE SELECTION TABLE | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Circuit Amperes | | Circuit Watts | | Wire gauge (for length in feet) | | | | |
| 6V | 12V | 6V | 12V | 3' | 5' | 7' | 10' | 15' |
| 5.0 | 10 | 30 | 60 | 18 | 18 | 18 | 18 | 16 |
| 5.5 | 11 | 33 | 66 | 18 | 18 | 18 | 18 | 16 |
| 6.0 | 12 | 36 | 72 | 18 | 18 | 18 | 18 | 16 |
| 7.5 | 15 | 45 | 90 | 18 | 18 | 18 | 18 | 14 |

# CHAPTER 6

# GRAPHIACAL USER INTRFACE (GUI)

In today's fast-changing world, program maintenance is more critical than ever before. The computer programs of today must be fluid and maintainable so that programmers can quickly change the program to meet the needs of a changing environment in which the programs are used. The use of today's graphical interface based operating systems puts tremendous strain on application programmers. They have to cope with all the stuff that is going on inside the operating system.

## 6.1 Visual Basic 6.0

Visual Basic (VB) is the third-generation event-driven programming language and integrated development environment (IDE) from Microsoft. Visual Basic (VB) is an ideal programming language for developing sophisticated professional applications for Microsoft Windows. It makes use of Graphical User Interface for creating robust and powerful applications. The Graphical User Interface as the name suggests, uses illustrations for text, which enable users to interact with an application.

The VB environment is called the integrated development environment (IDE) because it includes not only the programming language but also the tools needed to design forms and to test & debug the program. Visual Basic takes a different approach to programming for the GUI (Graphical User Interface) systems. Coding in GUI environment is quite a transition to traditional, linear programming methods where the user is guided through a linear path of execution and is limited to small set of operations.

Visual Basic was derived from BASIC and enables the rapid application development (RAD) of graphical user interface (GUI) applications, access to databases using Data Access Objects, Remote Data Objects, or ActiveX Data Objects, and creation of ActiveX controls and objects. Visual Basic can create executables (EXE files), ActiveX controls, or DLL files, but is primarily used to develop Windows applications and to interface database systems. Dialog boxes with less functionality can be used to provide pop-up capabilities.

Programming in VB is a combination of visually arranging components or controls on a form using drag-and-drop techniques. Controls have attributes and event handlers associated with them. Many attribute values can be modified during run time based on user actions or changes in the environment, providing a dynamic application.

Visual Basic (VB) is called an event-driven programming language because the programs respond to user-initiated events such as a mouse click or key press. Within a VB GUI program, an event Listener waits for an event to occur and then responds to it. The event listener is actually a procedure that contains VB code that executes when a particular event is fired.

Hence, in VB the code does not follow a predetermined path- it executes different code sections in response to events triggered by the user actions, by messages from the system or other applications or from objects within the application as shown in figure 6.1. Thus, Visual basic differs from traditional procedural approach in which the application itself controls the portions and sequence of the code to be executed, the code beginning with the first line of code and following a predefined path through the application, calling procedures as required.

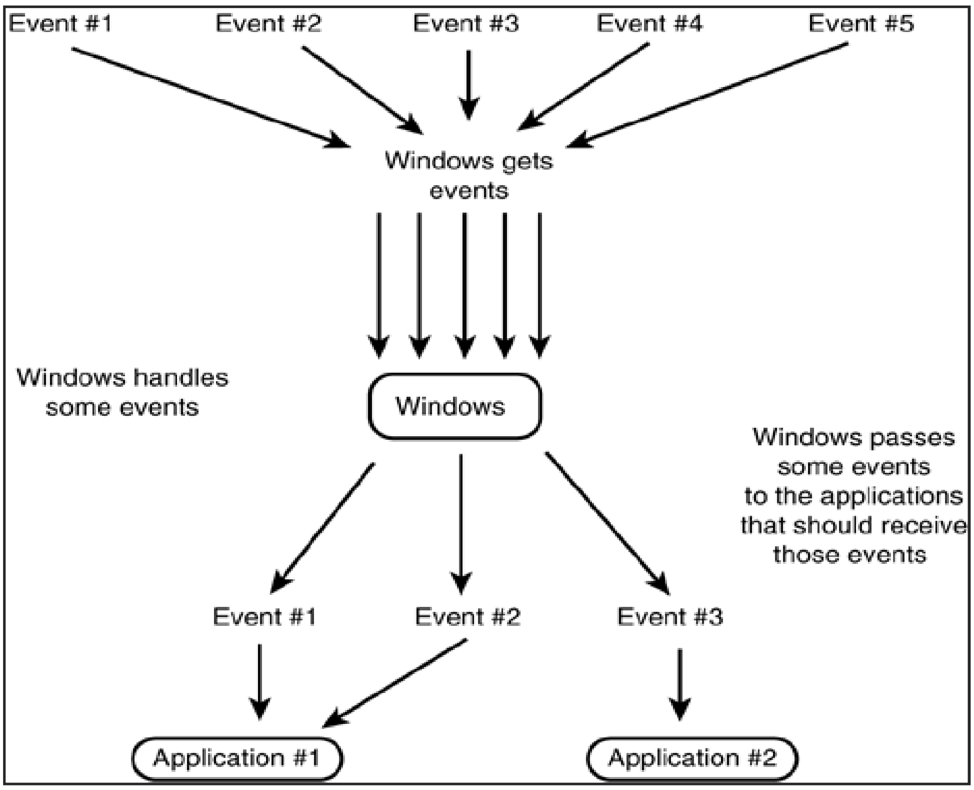



**Figure 6.1: Programs Must Respond to Some Events and Ignore Others.**

Features such as easier comprehension, user-friendliness, faster application development and many other aspects such as introduction to ActiveX technology and Internet features make Visual Basic 6.0 an interesting tool to work with.

### 6.1.1 The Visual Basic Environment

The Visual basic IDE shown in figure 6.2 neatly organizes a number of toolbars and information panels that assist in creating user interfaces and writing code. It manages and allows an easy access to the entire project.

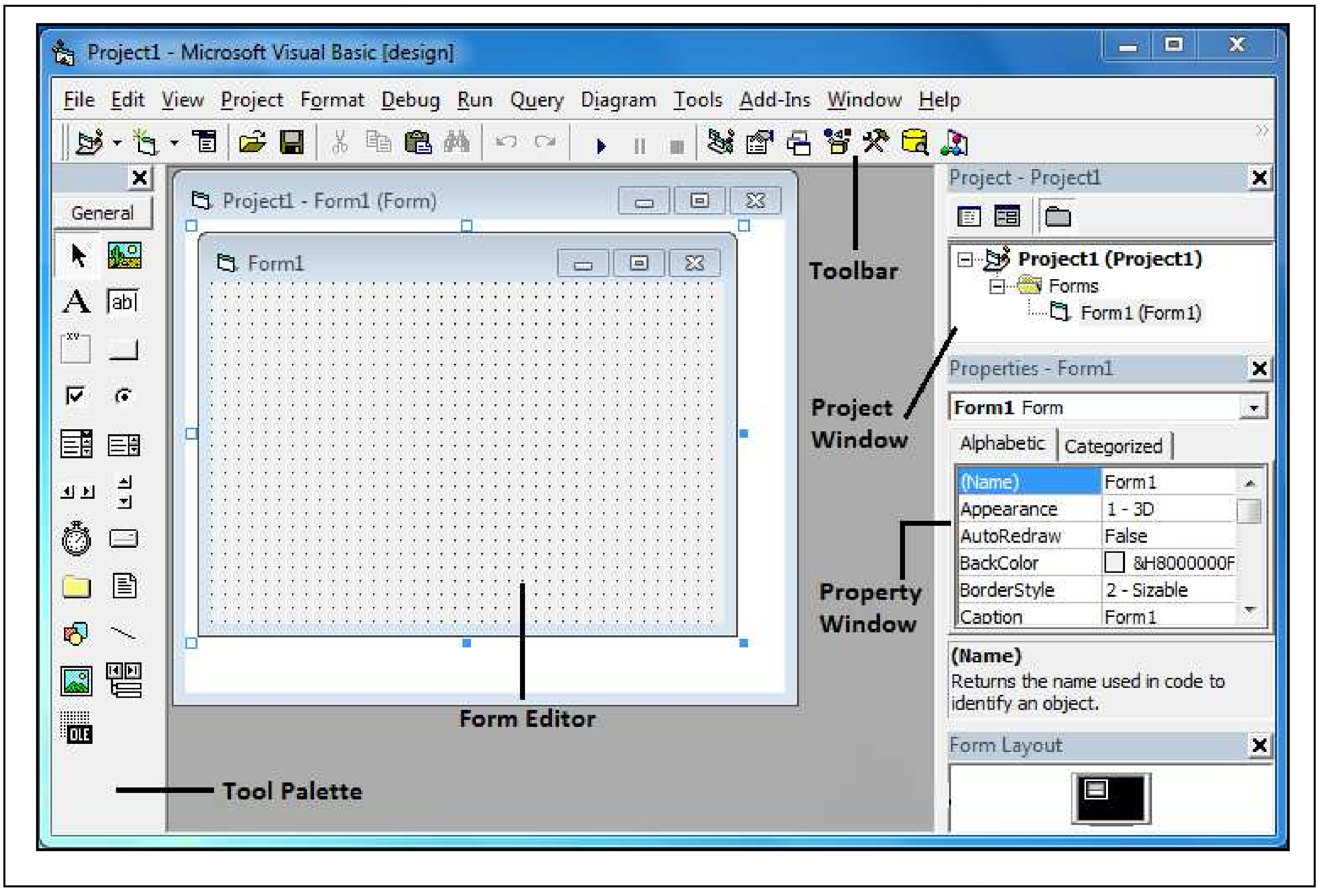



**Figure 6.2: Visual Basic Environment**

Development Environment in Visual Basic has several elements, such as:

i. **Visual Basic Standard Toolbar**

The Standard Toolbar contains buttons for most of the commands used in VB that make easier to edit the VB application. The button such Open Project, Save Project, Start, Break, and End are embedded in this toolbar along with buttons to display Project Explorer, the Properties Window, the Toolbar, and other elements of VB IDE.

### ii. Visual Basic Toolbox

The Visual Basic Toolbox contains the general objects and controls. There are forms to create the user interface of your VB applications as shown in figure 6.3. Additional controls such as MSComm, Common Dialog, ADODC can be added to the Toolbox by using Components command on the Project menu.

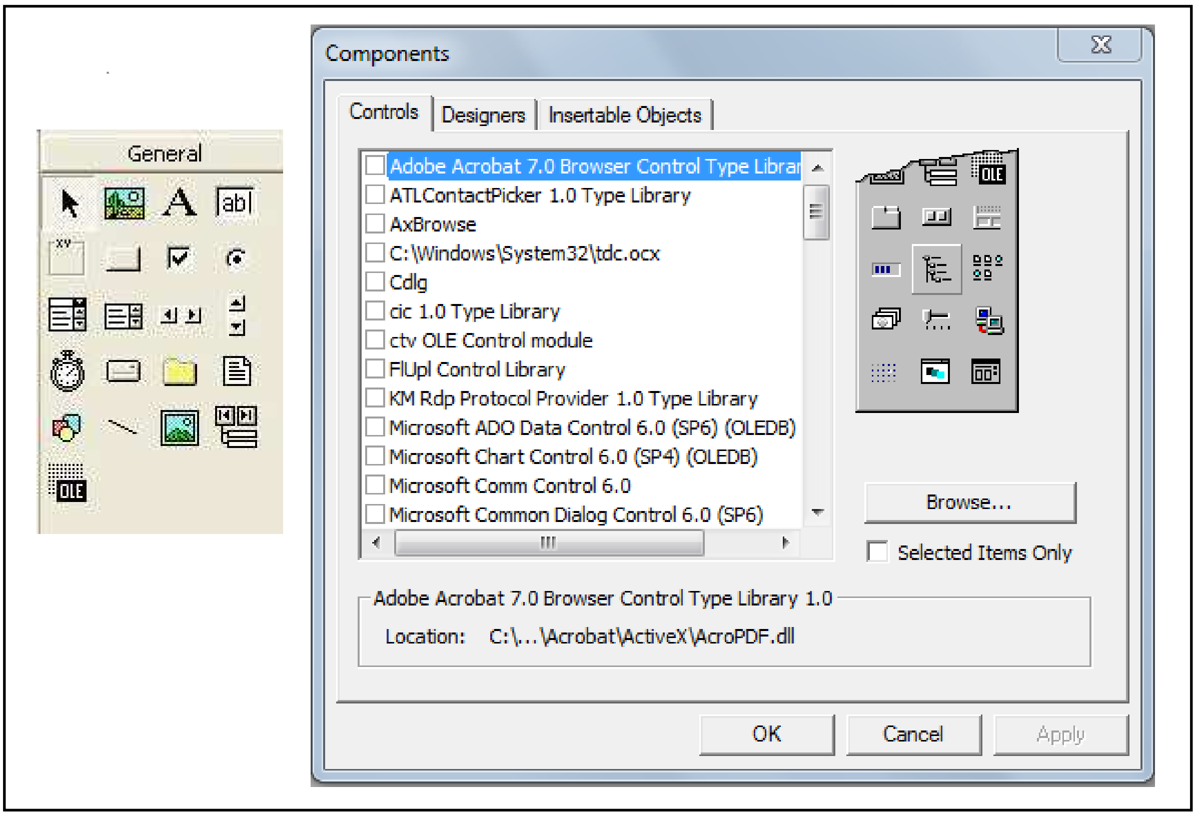



**Figure 6.3: General Tool Palette and Additional Component Dialog.**

### iv. Project Explorer Window

Project Explorer Window is one of element of Visual Basic Development Environment that contains list of all files that build a VB application. The collection of files in a VB application is called VB Project (with extension .vbp). Several file types may exists in VB project, for example Form (with extension .frm), Module (with extension .bas), Designer (with extension .dsr) etc.

### v. Properties Window

Properties of an object reflect the characteristic of the object (for example Width, Height, and so on). Each object in VB6 program has its own properties. For example properties of ***text box*** are Name, Alignment, Appearance, Locked, Text, etc. Properties of ***label*** are Name, Alignment, Appearance, Locked, Caption, etc. Each property of object has a default value and with Visual Basic Properties window, the properties of the objects can be set at ***design time***. Some of properties can be changed at run time also (when the program running).

**vi. Code Editor Window**

A very powerful feature of VB is its code editor as shown in figure 6.4. The VB code editor uses color coding, comments are in green, VB keywords are in blue and most other code is in black. Even more important is the fact that VB has a very smart text editor. Every time a line of code is entered, VB checks its syntax and immediately flags any errors. Besides, VB's Intellisense feature gives users lots of help in getting the syntax right.

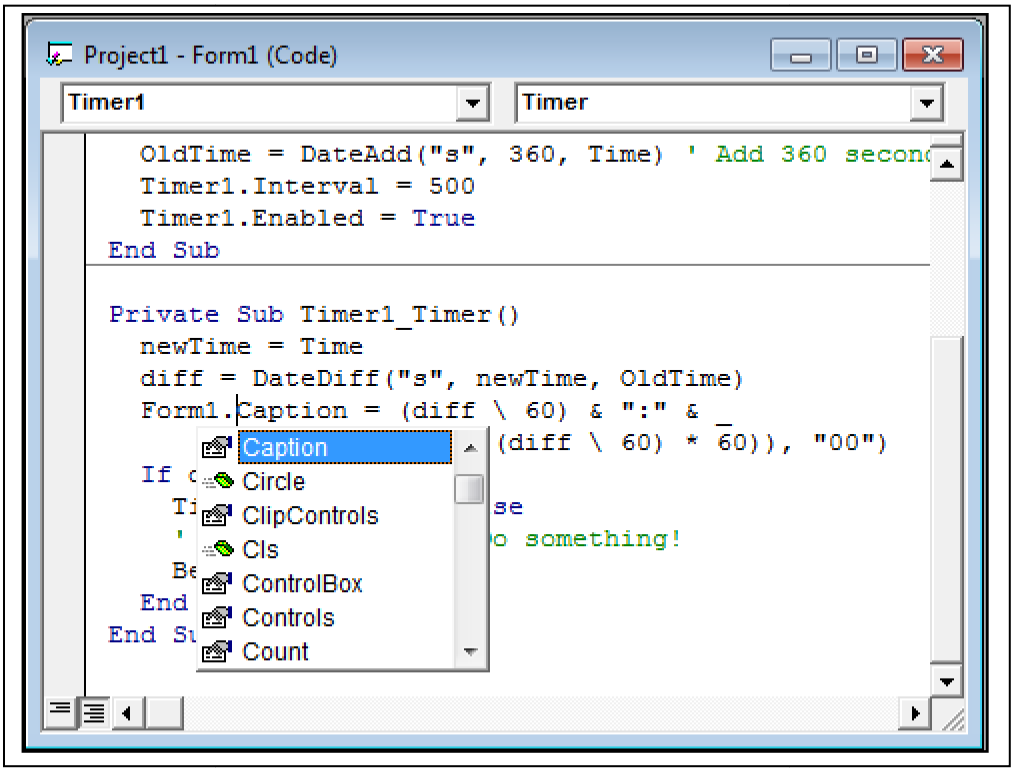



**Figure 6.4: The VB Code Editor and Its Intellisense Feature**

Not only that but on completion of the line it checks the syntax and, if correct, immediately fancy formats the line. Fancy formatting includes adding blank spaces around operators (like "+", "-", "*", etc), auto-indenting code blocks and capitalizing variable names that have been entered in all lowercase. And if a code line is not fancy-formatted, the user can easily catch misspellings.

The two combo box coding aids also add a lot of convenience into VB's text editor. The first combo box lists all the controls used in a form. The second combo lists all the events that a specific control can respond to a form as shown in the figure 6.5. This aids coding because only the correct events are shown in the list and if an event is selected, VB automatically writes the subroutine header and closer.

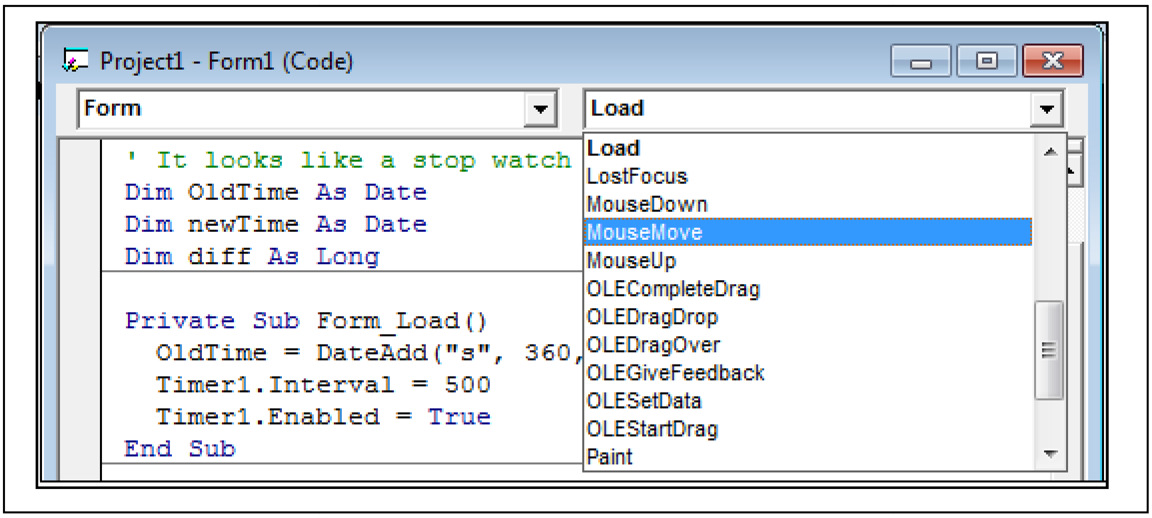



**Figure 6.5: Combo Boxes for Control and Their Events in Text Editor**

## 6.2. Development of GUI in Visual Basic 6.0 for CNC Machine

Front end of Open Architecture Controller was developed in Visual Basic 6.0. This Graphical User Interface ( GUI) of Open Architecture Controller controls the machine motions by sending the commands to CNC controller and receives feedback from it to show update on front end GUI.

### 6.2.1. Key Features of GUI

The front end GUI software has the following features:

i. Password Authenticated Access
ii. Hand shaking capability with controller memory.
iii. Proper file extension filter.
iv. Preset definition of feed and speed according to material and tool selection.

v. User can update preset library by entering new material type.
vi. Custom start from any location from loaded tool position file.
vii. Progress bars to indicate the percentage of task completed.
viii. Report the successful completion of task.
ix. User friendly easy- to-operate interactive Interface.
x. Computationally fast.
xi. Requires lesser memory.
xii. If system crashes while running, the software generates error report easy to debug.

### 6.2.2 The Visual Basic Front End

The Graphical User Interface developed in Visual basic has been optimized to be user friendly and easy to operate. It consists of the following modules:

i. Welcome Screen
ii. User Login Module
iii. CNC Controller Main Module
iv. Preset Library Module
v. Material Library Management Module

### 6.2.1 The Welcome Splash screen

The Welcome screen has been designed as shown figure 6.6 using the standard Splash screen dialog of the VB environment. The login module and the main module are loaded in the memory simultaneously which is indicated by the progress bar incorporated using the label control.

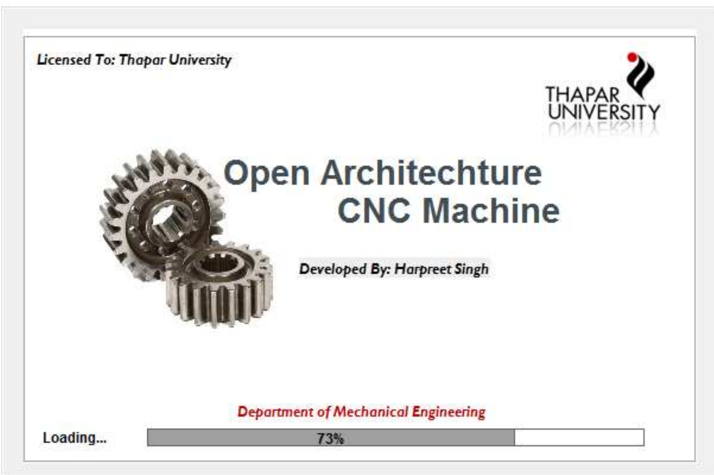



**Figure 6.6: Welcome Screen of the CNC Machine Control Software**

### 6.2.2 The User Login Module

The User must enter the authorized password before entering the main module of the CNC machine controller software. In case, the correct password is not provided, the access to the main module is restricted. The run time view of the user Login and password module is shown in Figure 6.7.

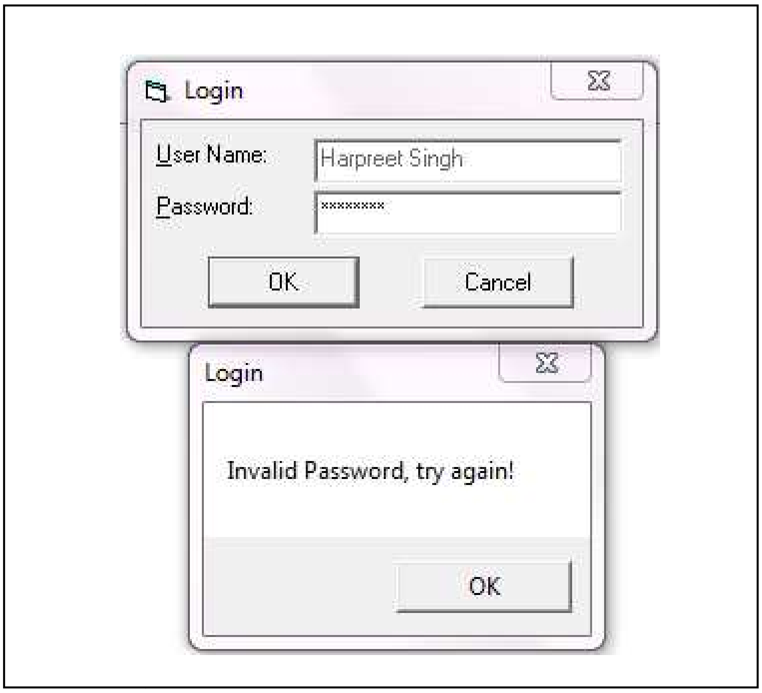



**Figure 6.7: Incorrect Password Prompt**

### 6.2.3 The CNC Control Module Main Screen

The main Control module provides the following capabilities:

i. Loading the tool path from the command text file.
ii. Serial data connection with the microcontroller.
iii. Sending the commands to the Controller serially through Cycle Start/ Start at command buttons.
iv. Progress indicator bar showing the percentage of the commands completed.
v. Preset Library and update library control buttons.
vi. Displaying current material selected and its feed rate and speed.

Figure 6.8 shows the GUI for 3-axis vertical open architecture controller CNC milling machine.

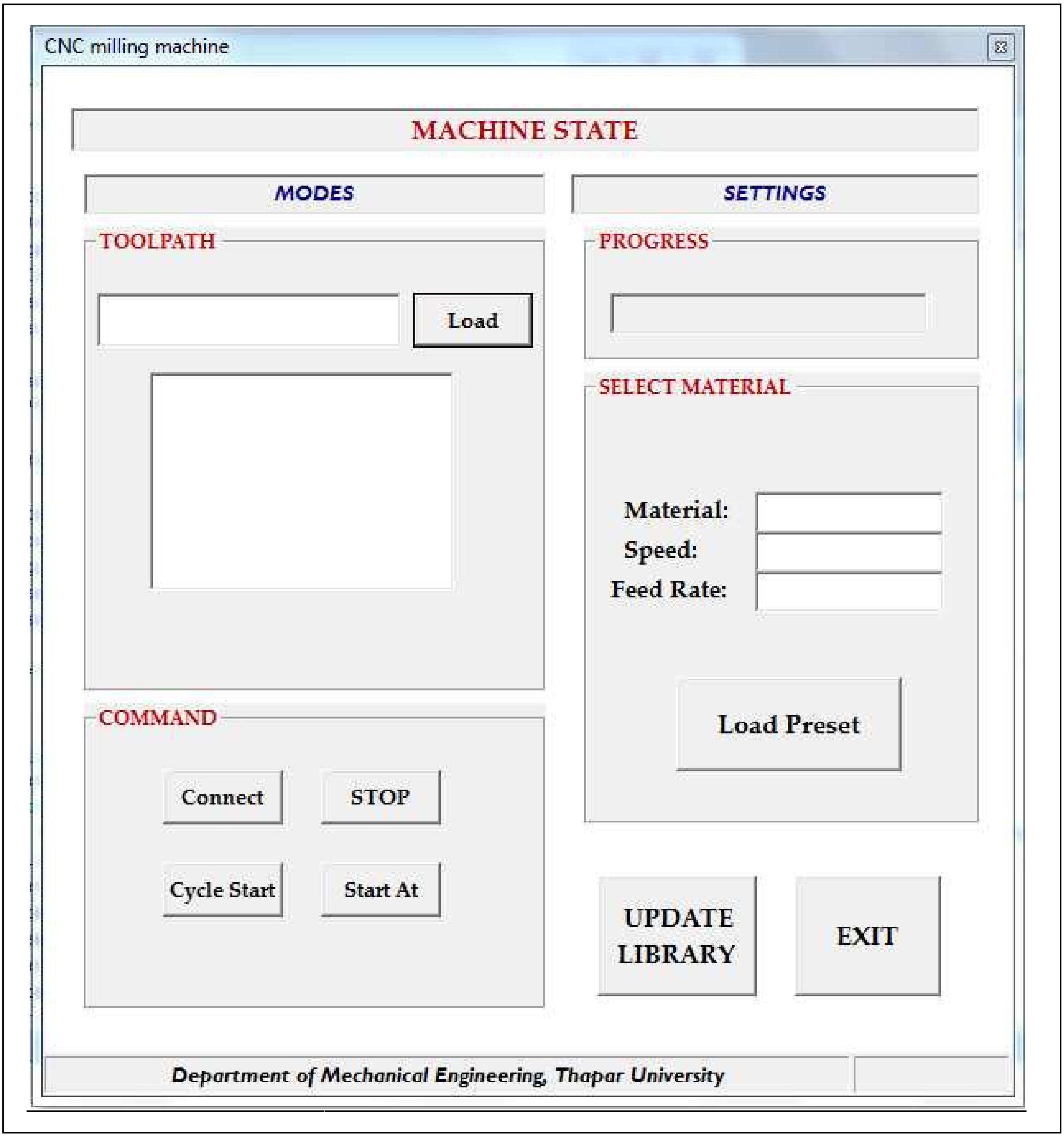



**Figure 6.8: CNC Milling Machine Front End GUI**

### 6.2.4 Loading the tool path

The Load command Button opens an 'Open' Dialog as shown in figure 6.9 which is used to load the tool path *.tp file into the list provided. The text file contains the values to be serially transmitted to the controller for the movement of 3 motors viz. X-axis, Y-axis, Z-axis motors.

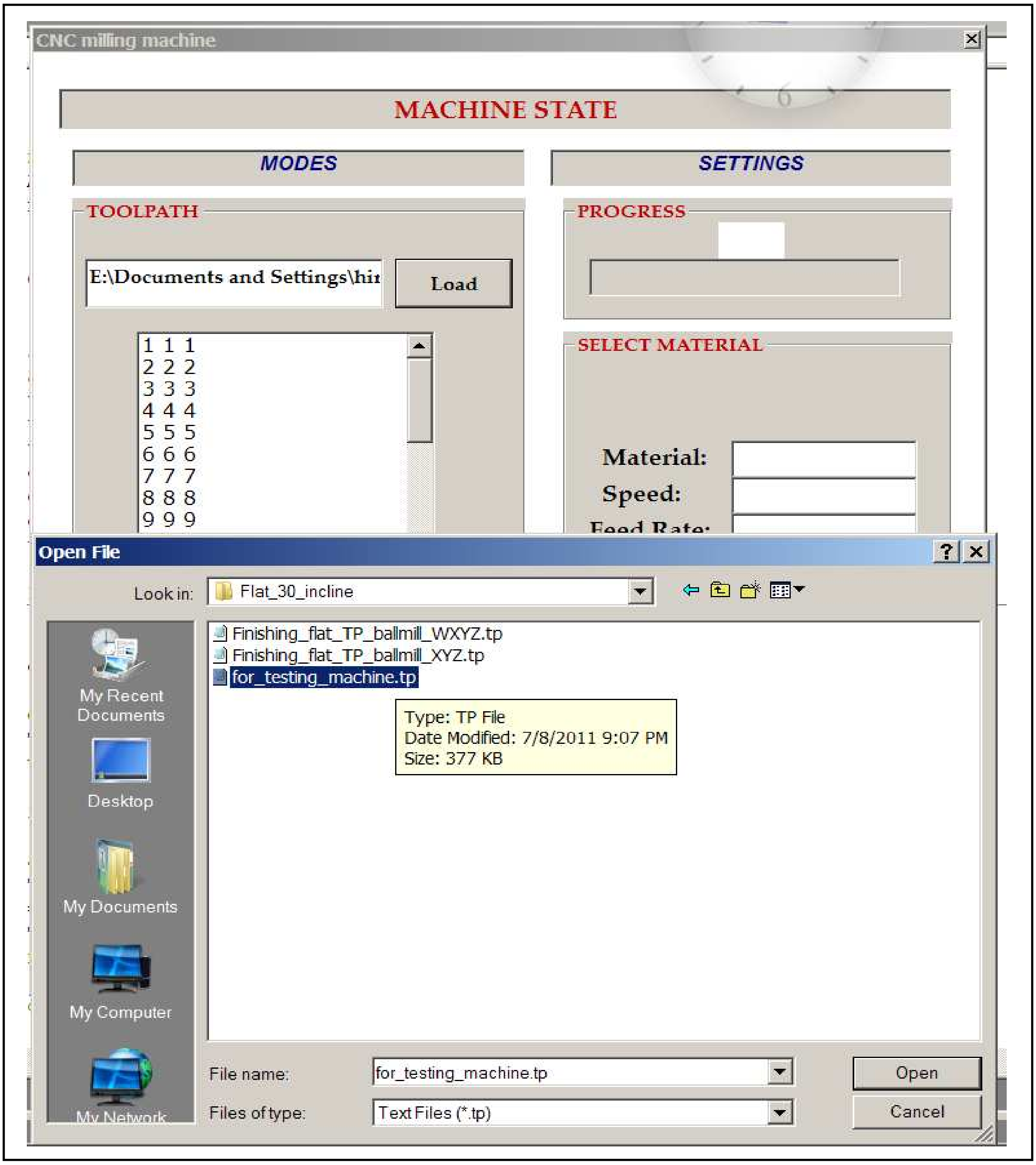



**Figure 6.9: Loading the Commands from *.tp file to List Box**

### 6.2.5 The Preset Material Module

In this module, the user can select the material from the material library already stored in the MS access database. The MS access databases can be accessed using the data control of Visual basic. The Form has been designed with the combo box control which is populated with the materials stored in the database library during the run time. When the user selects a material, the control returns to the main module and the material name, speed and feed rate is displayed in the labels in the selected material frame as shown in figure 6.10.

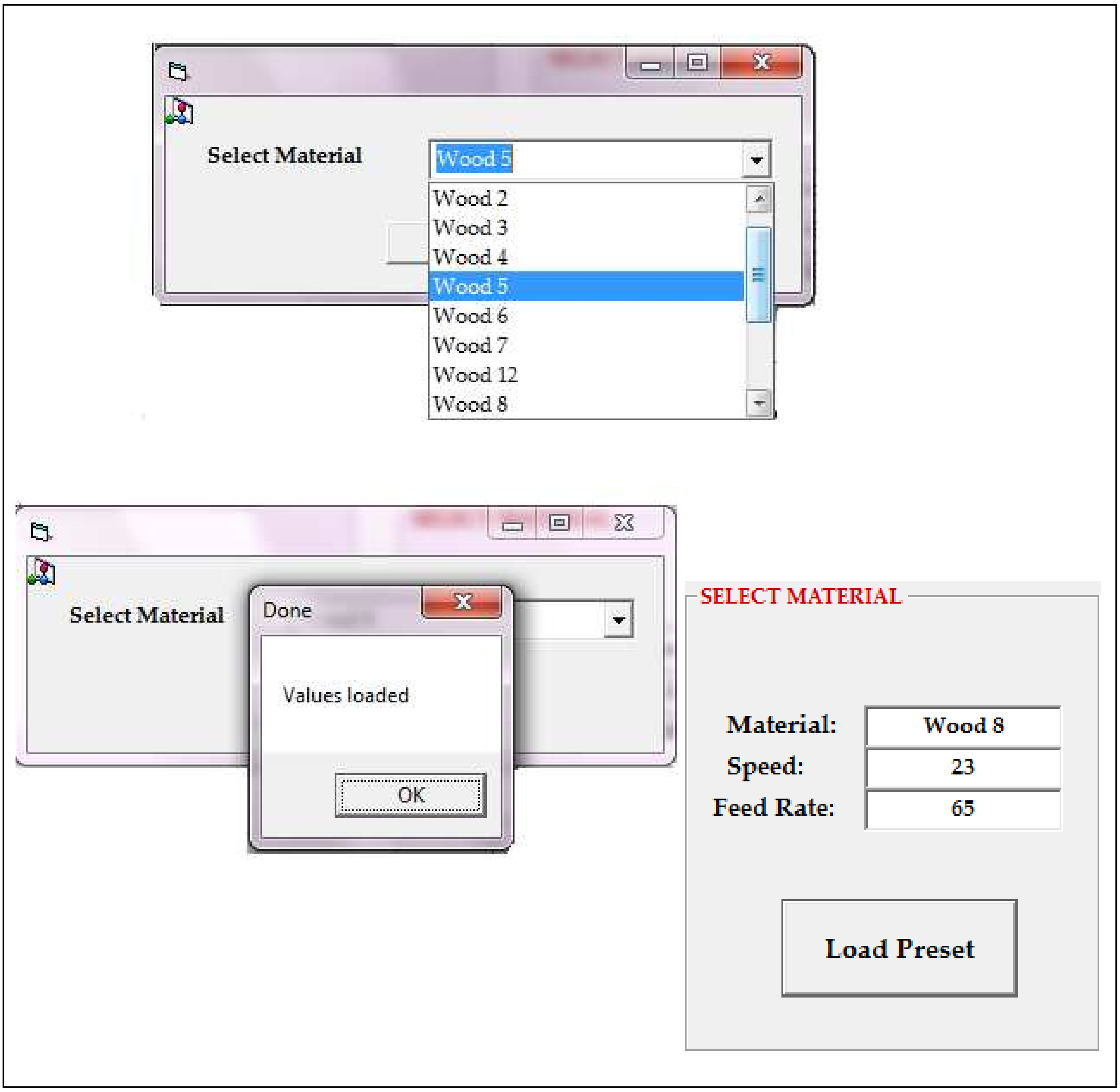



**Figure 6.10 Loading the Material, Feed Rate and Speed from the Preset Library**

### 6.2.6 The Material Library Management Module

The update module equips the user with the tools to change the feed rate and speed of the existing materials in the library and also to add new materials to the library. Various functions of the update module are described in the following text as shown in figure 6.11.

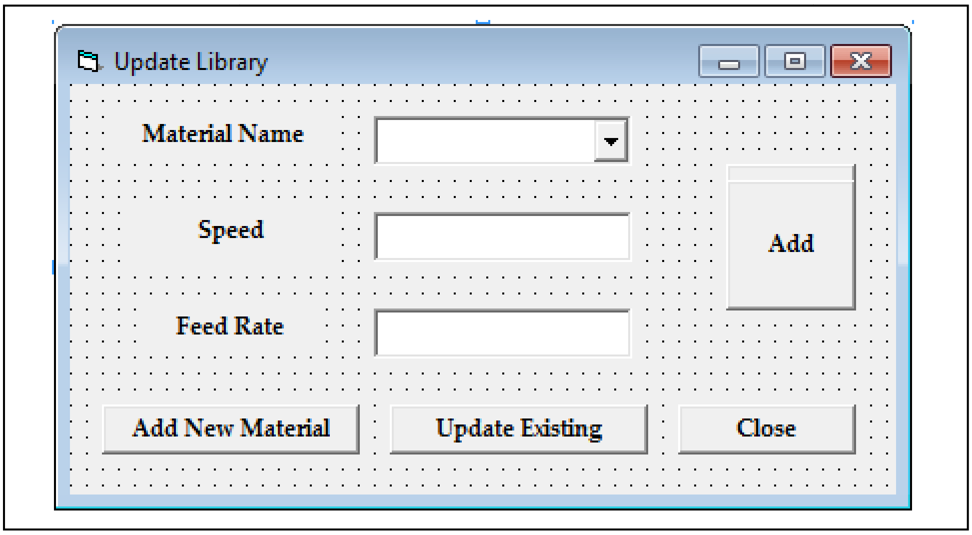



**Figure 6.11: The Update Material form during Design Time**

#### 6.2.6.1 Addition of New Material in the Library

When the Add New material Command button is clicked, the Add button is enabled and user can enter the material name and corresponding parameters in the provided text boxes. If Material name, Speed or Feed Rate is missing when the add button is clicked, the error is generated. Illustrations are shown in figure 6.12 and figure 6.13.


**Figure 6.12: Message Box Prompt on Addition of Material in the Library**

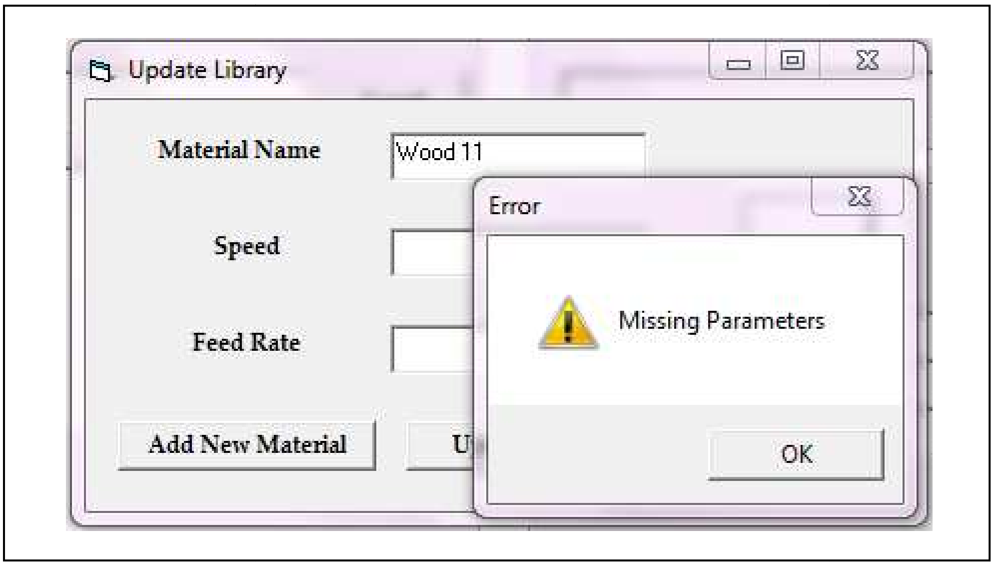



**Figure 6.13: Error Message When Parameters Are Not Entered By User**

In case, the user attempts to add an existing material again in the Library, the software generates an error message as shown in Figure 6.14.

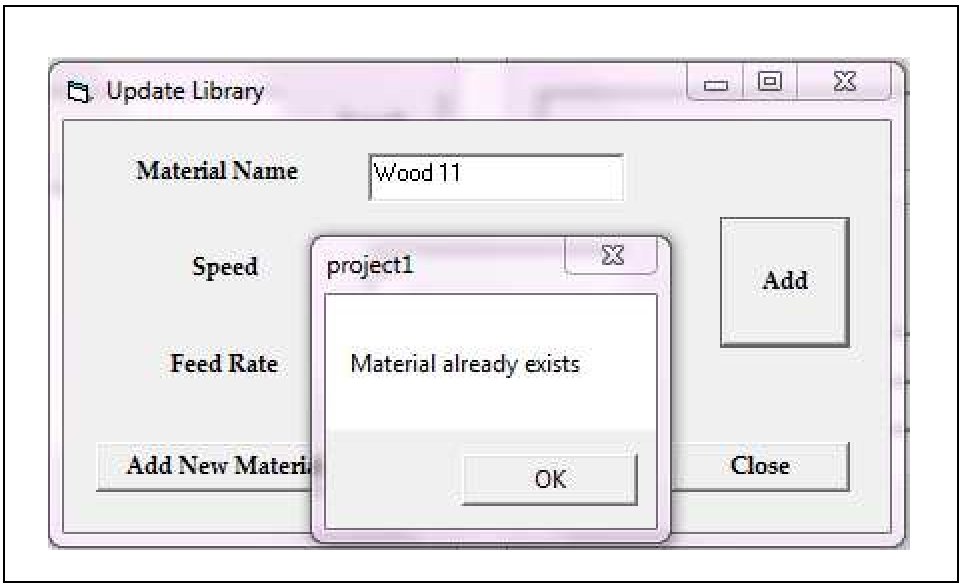



**Figure 6.14: "Material already exists" Error Prompt**

### 6.2.6.2 Updating the available materials in the Library

The user can choose from the list of the materials which are already stored in the library. The Speed and Feed rate can be changed and the corresponding changes are updated in the MS Access database Material Library. The screenshots are shown in figure 6.15 and figure 6.16.

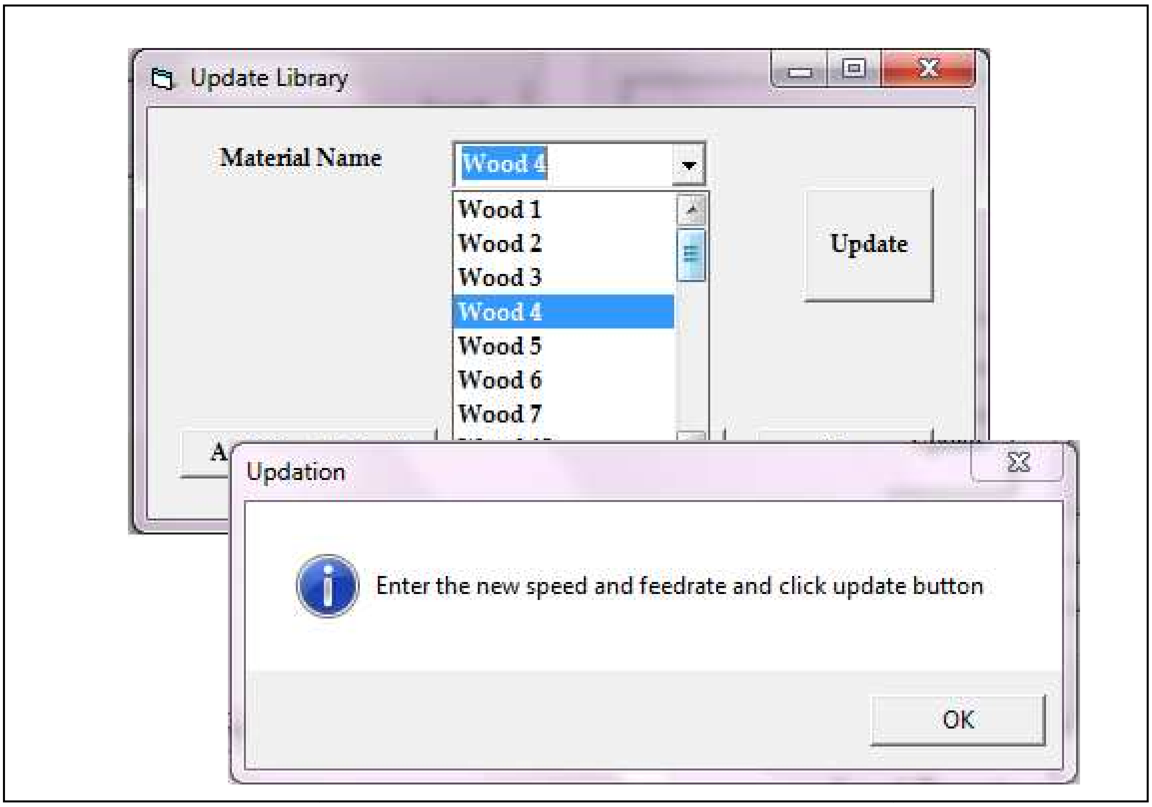



**Figure 6.15: Selecting the Material from the List and Prompt to Enter the Parameters**

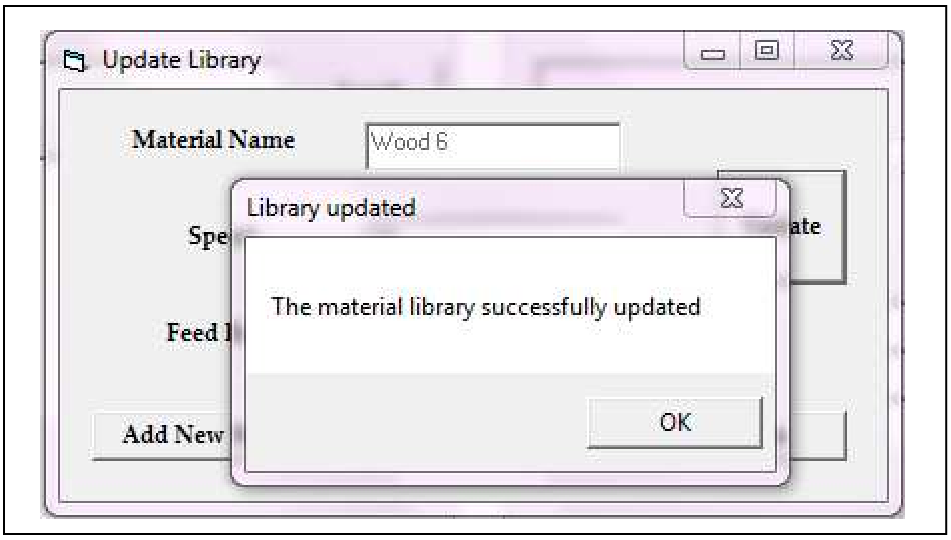



**Figure 6.16: Message Box Prompt on Successful Update of the Library**

Flow charts corresponding to each command button and event of front end GUI presented in appendix A.

# CHAPTER 7

# DISCUSSION AND VALIDATION OF DEVELOPED 3-AXIS OPEN ARCHITECTURE CONTROLLER ON SELF FABRICATED 3-AXIS NC MILLING MACHINE

NC machines are now becoming more intelligent by incorporating new emerging technologies in it. Image processing techniques and more fine algorithms are being developed by researchers. G codes and M codes which were bottle neck for NC machine development are now a part of past. New tool path strategies are being developed by researchers. To test or run new tool path conventional machine are not suitable and CAM package is needed to convert new tool path into vendor controlled CNC machine code.

Problem in existing NC machines are these are having controller which are vendor or manufacturer controlled. Researchers could not do modification in vendor controlled machine.

In this thesis work development of new architecture of controller has been done which are open and have capabilities of parallel processing.

## 7.1 Features of Developed Open Architecture controller (OAC)

This thesis work presents new architecture controller which is fully designed to meet new need of tool path programmers those are working on “Direct Model to Part” machining. Power of personal computer which is having great computational speed has been utilized to do complex calculation and to run decision making algorithms by taking feedback from controller.

Following are the key feature of developed open architecture NC controller for personal computer-controlled vertical 3-axis NC milling machine for machining of sculptured surfaces of wooden artistic design.

i. Developed controller has capability to incorporate new sensors and machine vision system on later part to make intelligent machining system.
ii. Developed controller is free for adding new algorithms and controller programs according to upgradation of existing system.
iii. Developed controller has one master and three slave controllers. Slave controllers are simultaneously controlling and real time monitoring the motion of individual axis and master controller keep on communicating with PC and slaves microcontrollers.

iv. All four microcontrollers are running in parallel. This presents the application of parallel processing which makes system faster.
v. Controller is programmed to interpolate points in 3D space within its work volume in incremental mode.
vi. Handshaking signal is enabled in serial communication between microcontroller and PC which ensures the no loss of data while transmission from PC to MCU at high speed.
vii. 3 slaves MCUs controlled by Master MCU via I2C lines. With I2C communication Master MCU can handle up to 1024 Slave MCUs which make OAC expandable.

## 7.2 Testing of Developed OAC on Fabricated 3-Axis NC Milling machine.

The self fabricated 3-axis NC machine shown in figure 7.1 has 0.22 mm resolution in Y and Z axis of motion. Both axis are having stepper motor of 1.8 degree and pulley diameter of 14 mm. X-axis has minimum travel in one step of stepper motor is 0.27mm. This makes the machine sufficient precise to interpolate tool path in 3D space.

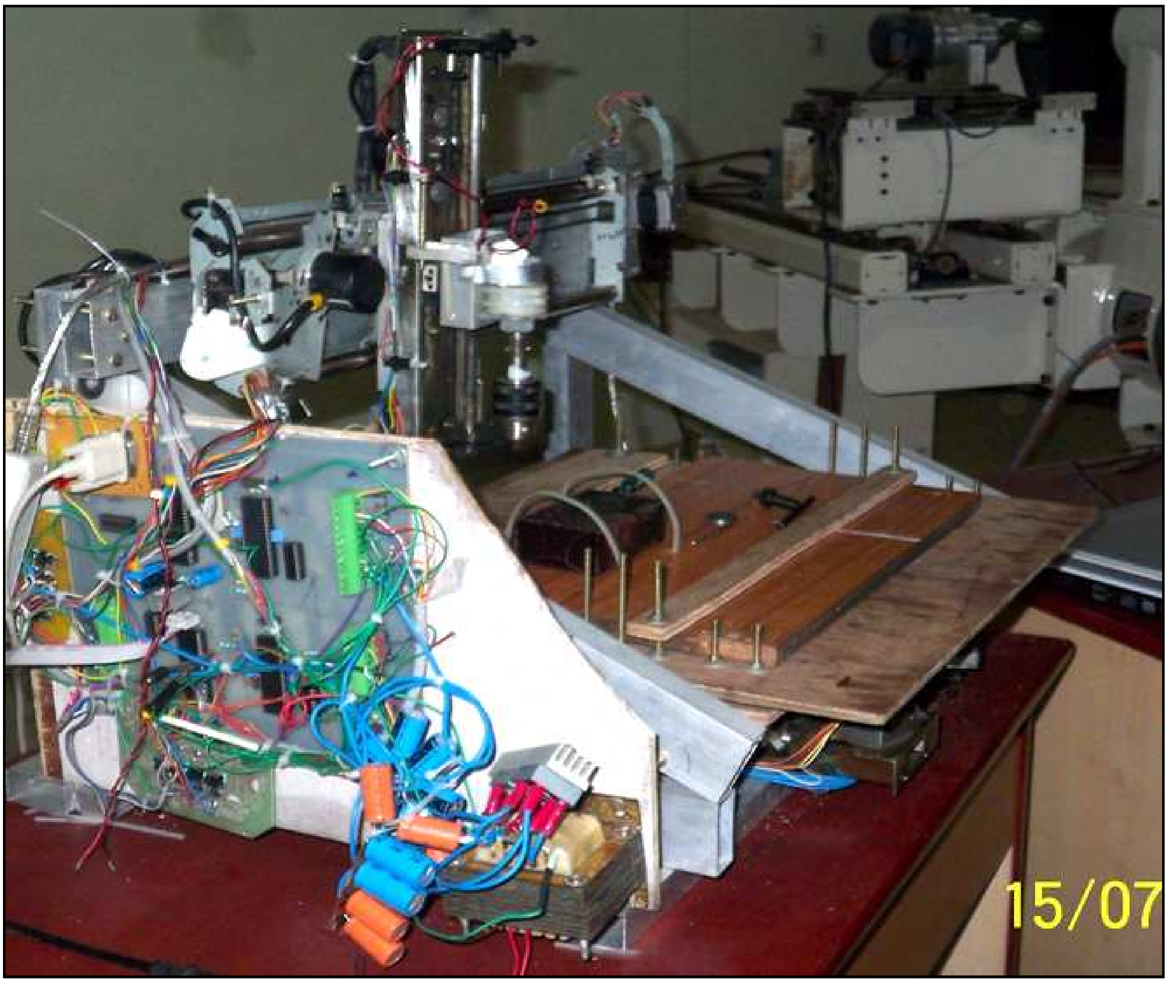



**Figure 7.1: Fabricated 3-Axis NC Milling Machine.**

To validate OAC on 3-axis NC milling machine, controller has to interpolate 8 points in X-Y plane on circumference of 25mm circle as shown in figure 7.2.  Z axis position validation has not done here.

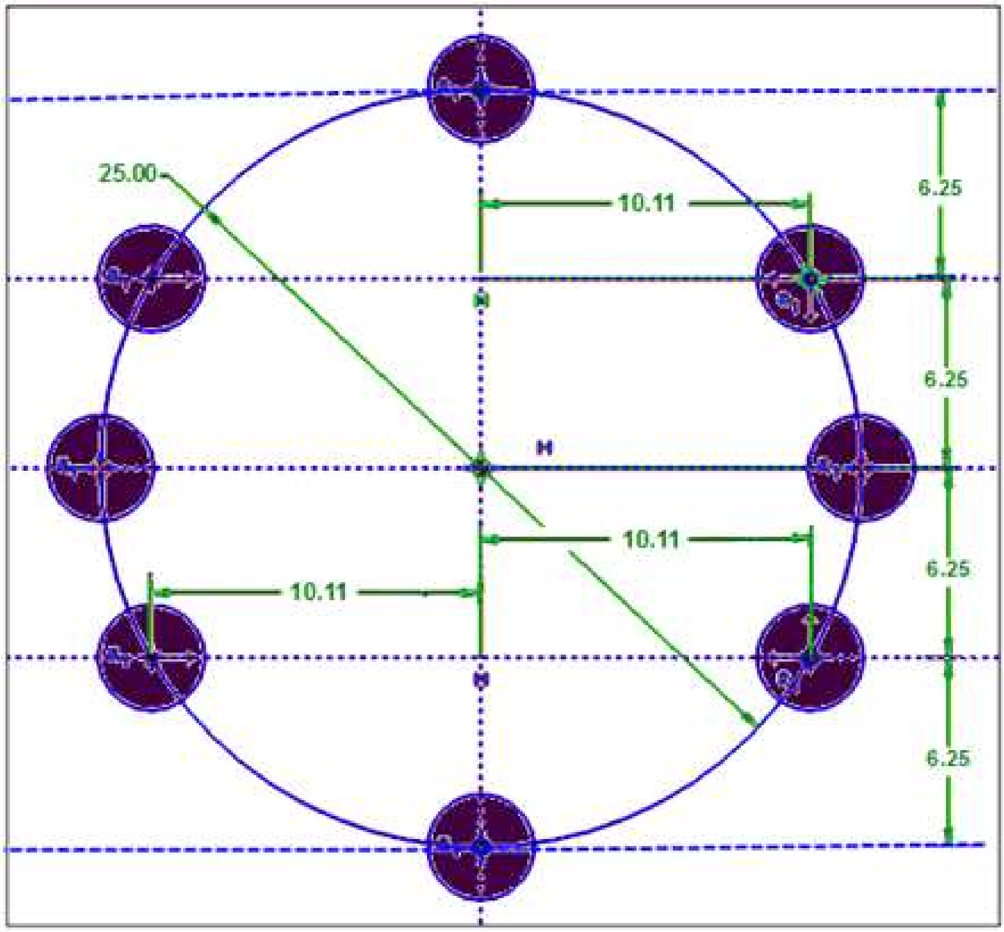



**Figure 7.2: 8 Cutting Points on Circumference of 25 mm Circle.**

Circle is divided into four equal parts along vertical line hence all horizontal lines will be on 6.25mm distance from adjacent one, where these lines cutting the circle, co-ordinate of those points have been calculated and tool path file was made. Simple wood drill of 3.17mm diameter was used as cutting tool.

For validation of controller, cutting on wax brick has done. Figures 7.3 - figure 7.6 shows the machining of 8 points on brick of wax.  Tool path file for these eight points was executed by front end GUI application in PC. GUI kept communication with machine and machine interpolated the points.

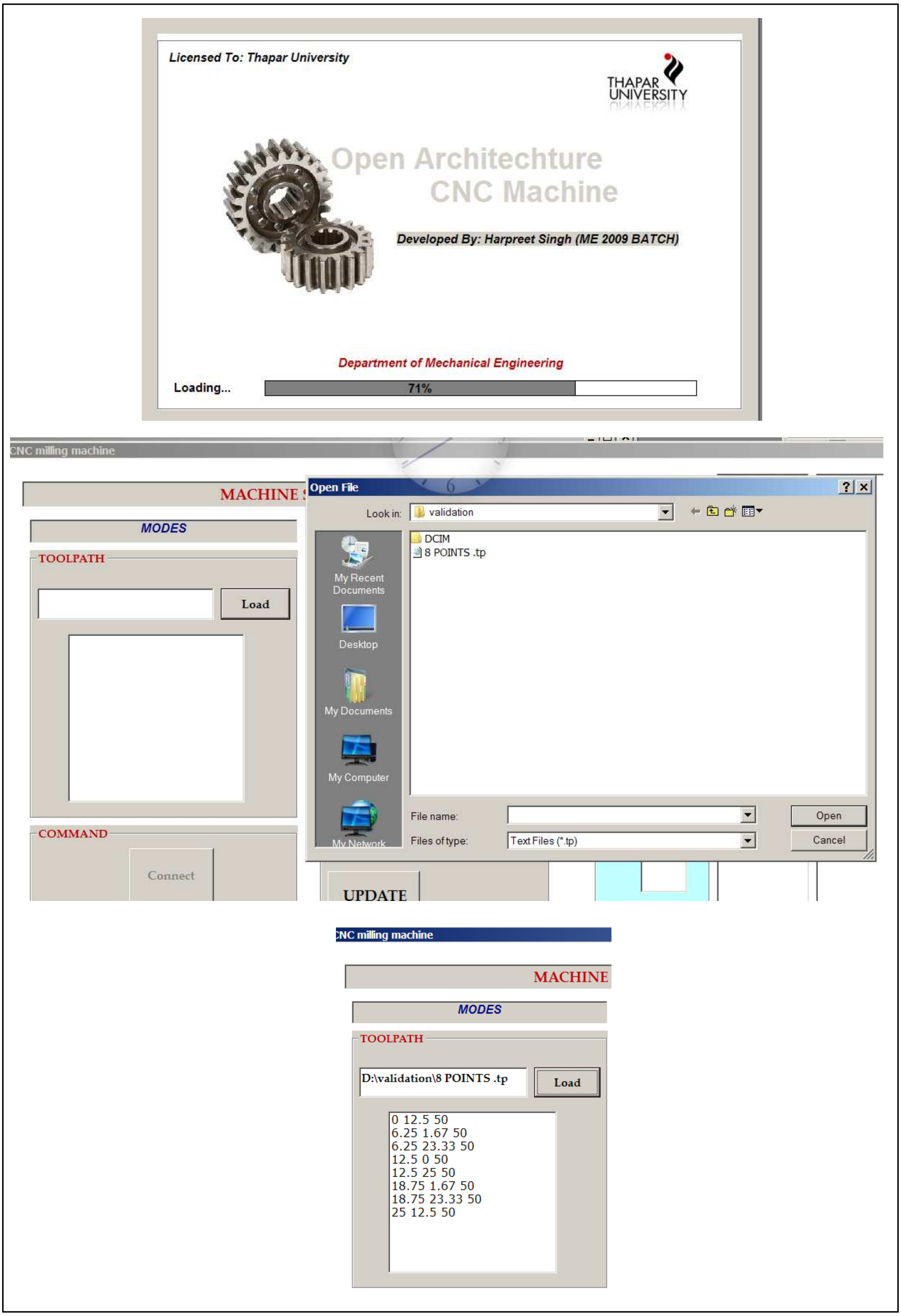



**Figure 7.3: Loading of Tool Path file of 8 Points.**

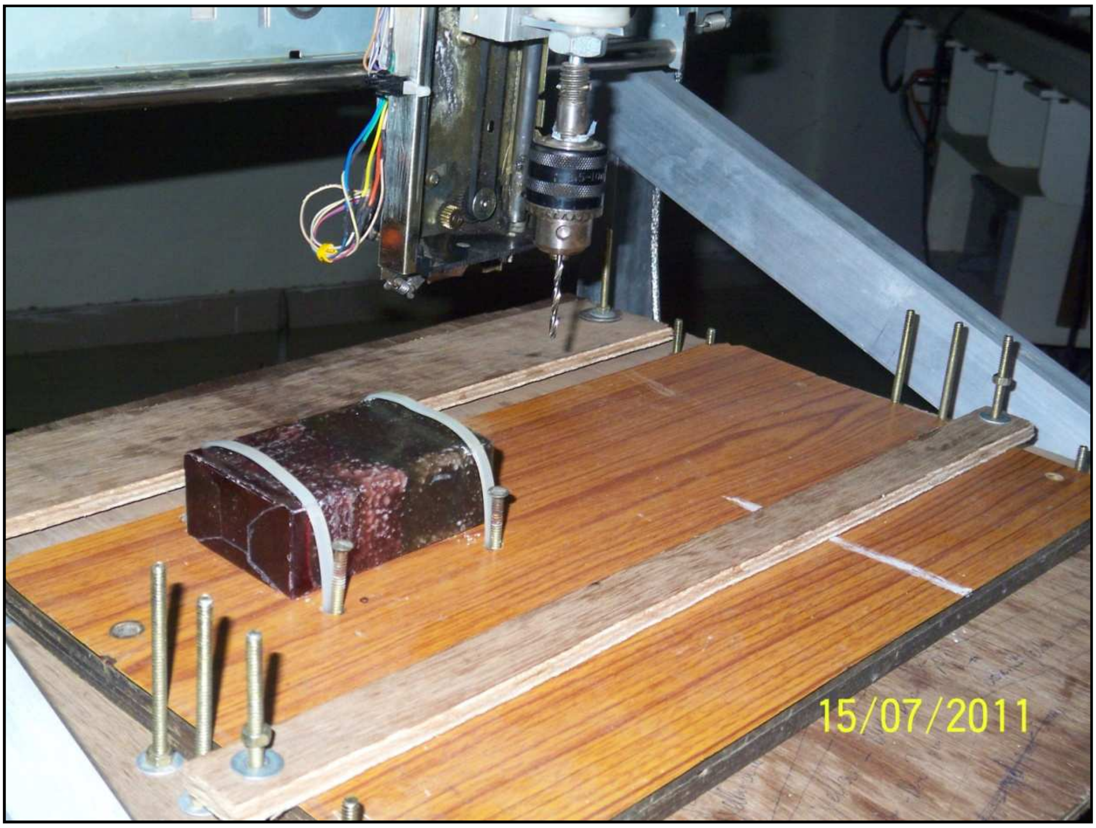


**Figure 7.4: Fixing of Wax Brick on NC Machine.**

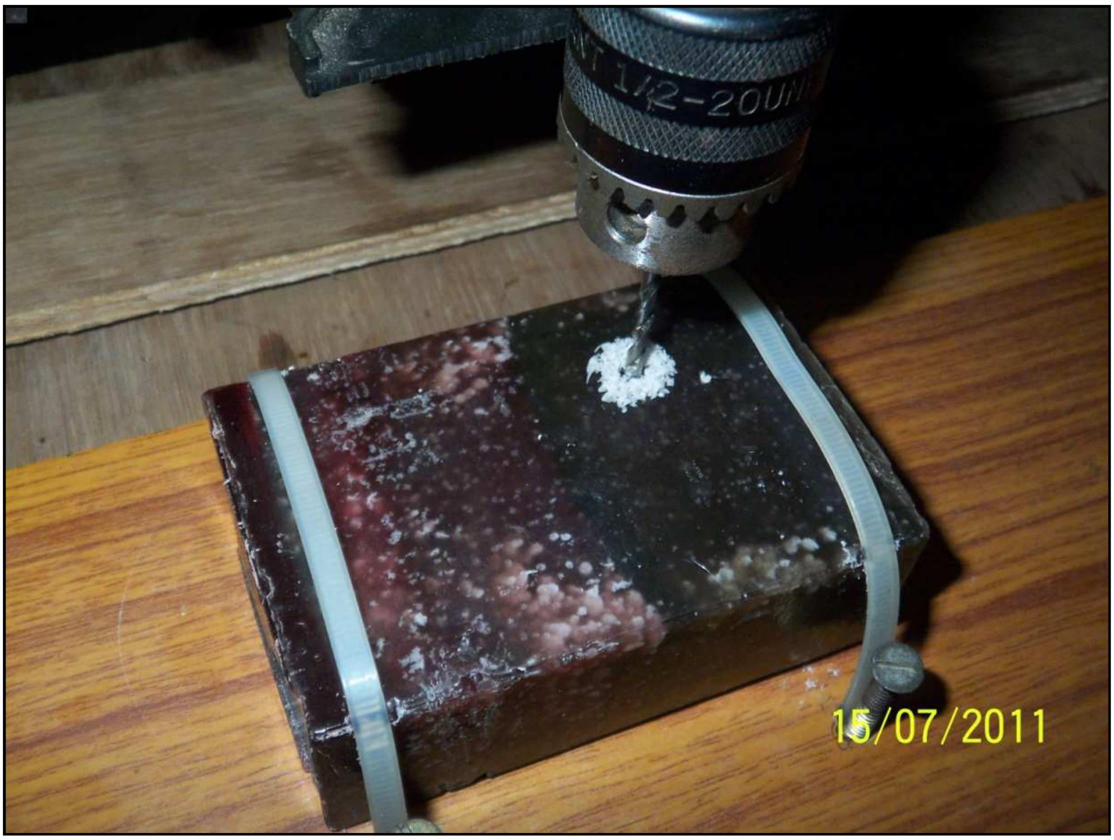


**Figure 7.5: First Cut Located By Machine Cutting Tool.**


**Figure 7.6: Sequence of Cutting on Wax Brick According To Tool Path File Loaded.**

After completing the machining on wax brick the distance between points has been checked with digital caliper for verification as shown in figure 7.7 – figure 7.9.


**Figure 7.7: Distance between Two Points along Diameter of Circle.**


**Figure 7.8: Distance Measured of a Point Lie in First Quadrant from Vertical Center Line of Circle.**


**Figure 7.9: Distance Measured of a Point Lie in First Quadrant from Horizontal Diameter of Circle.**

Developed open architecture NC controller for personal computer-controlled vertical 3-axis NC milling machine has been successfully tested on self fabricated 3-axis NC milling machine with cutting on wax brick. The deviation in reading by digital caliper after machining part is due to error in placement of hand held digital caliper in center of drilled points.

### 7.3 Limitation of Developed Open Architecture Controller for 3-Axis NC Milling Machine

The developed open architecture controller for 3-Axis NC milling machine has gone through its first trial. After the successful testing of controller there is need to redesign PCB. As in existing PCB some of controller pins are not routed out because of that there is no possibility as such to add new hardware in machine.

Slave microcontroller does not having LCD pins out. This makes debugging difficult while programming the slaves MCUs. In redesigned, PCB should have 4 bit LCD lines corresponding to each microcontroller.

Encoders fitted in NC machine are of 1000 PPR. Slave microcontrollers are getting feedback from encoders. If there is change in speed of stepper motor the rate of pulse generation counted by microcontroller is different. But in actual it should be same for any speed. This error needs to be rectified. Because of this error now motors are running on same speed.

### 7.4 Future Scope

There is enough scope for the integration and further automation of the CAD and CAM fields to make their usage cost effective and simple. The 3D CAD data from customized CAD packages can be used to generate the appropriate tool path for machining operation without using ISO G and M codes. The appropriate tool path planning strategy along with the correct geometry of the cutting tool could be used for machining of sculptured surfaces to get the surface finish with in specified tolerance ranges. The machine tools which work on open architecture control (OAC) for sculptured surface machining are still under development. The cost effective strategy of direct model to part manufacturing can only be realised by using the web based machining system for which the OAC based machine tools have to be developed to overcome the deficiencies of conventional NC controllers and conventional NC part programming.

The least skill person in art and craft can produce best 3D design only with the use of web based machining system is simply to know how to browse the web, and access the commands on the specified web page. Now days the internet is easily accessible around the world, and the graphics based application software make it further convenient to learn and use the technology, which even the unskilled artisan would be able to use after a short formal training. The scope of exploiting the web based machining technology is wide as it is easily affordable with minimum expenditure. The only expenditure would be for the purchase of the initial customized NC machining system which would be very simple to operate, similar to operating a printer connected to a PC. Further engineering assistance required for the CAD modeling and tool path generation will be freely available through the specified web page of the company supplying the customized NC machine tools. Thus the design and manufacturing activity could be accomplished in minutes.

# Appendix A

## Flow Charts of Front End GUI

Visual Basic is event driven programming. Particular event is fired when triggered by user by any mean. Following flow charts from figure A.1 to figure A.10 shows the flow of program corresponding to buttons on GUI.

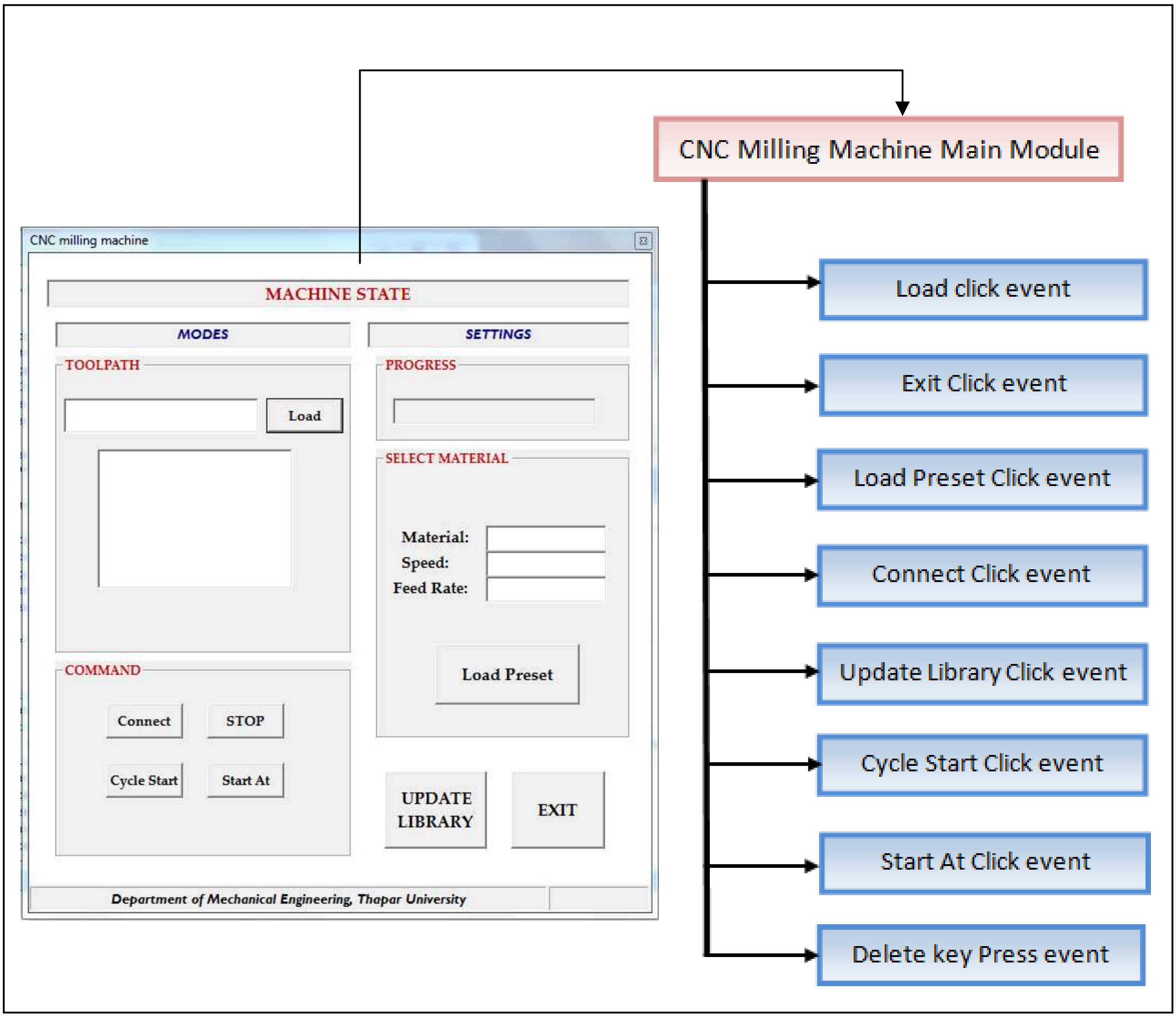



**Figure A.1: Loading of Main Form Subroutine.**

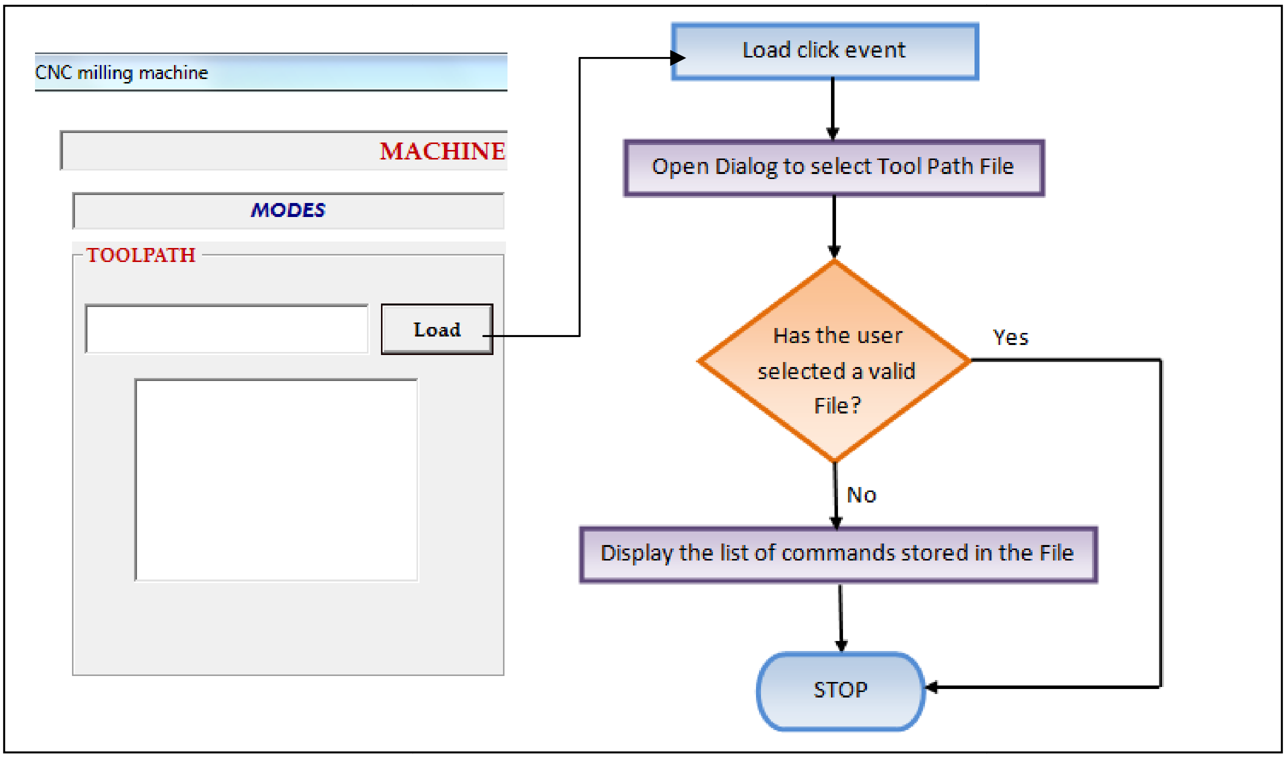



**Figure A.2: Load Click Command Button.**

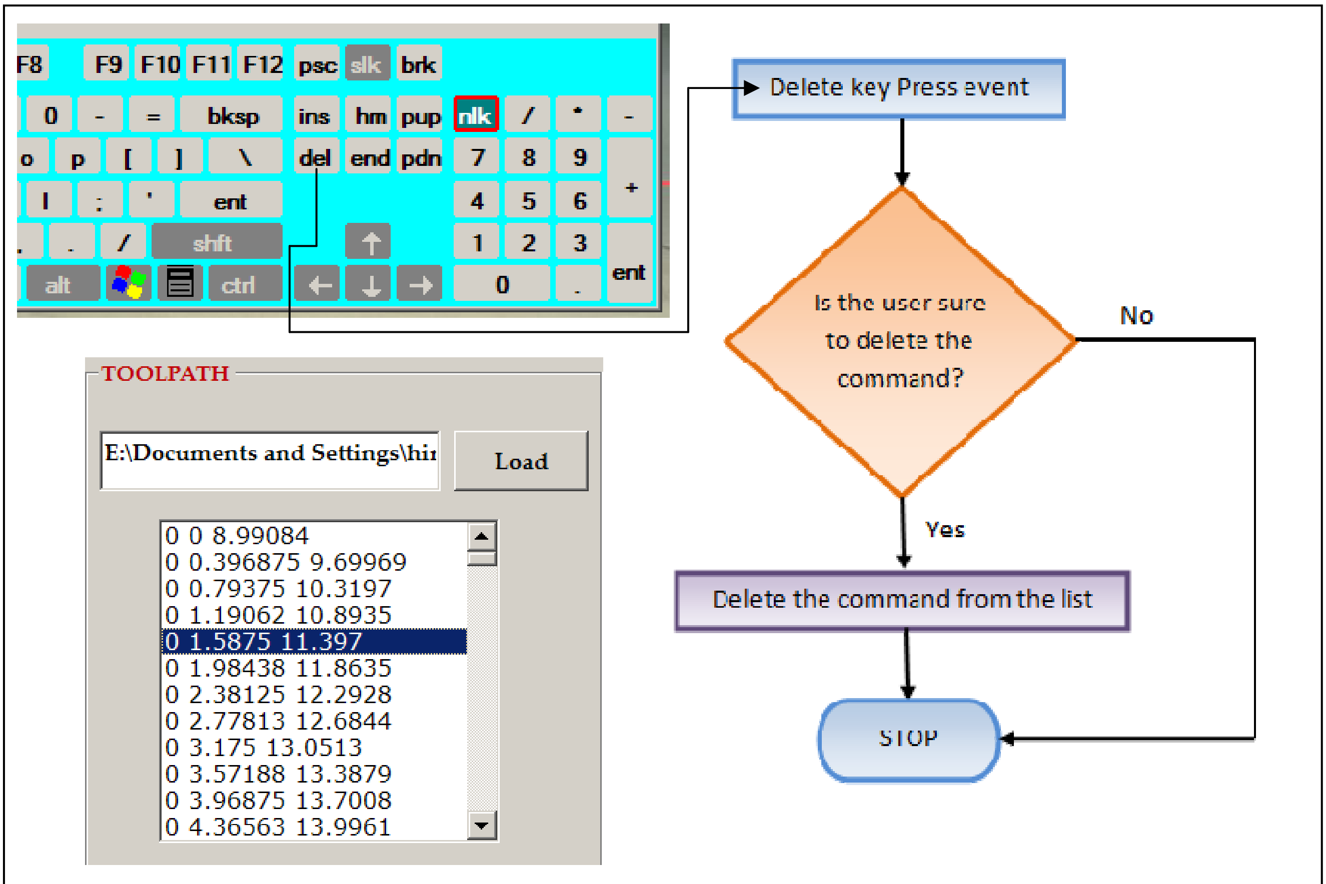



**Figure A.3: To Delete Selected Command from Loaded Tool Path File By Pressing Del Key on Keyboard.**

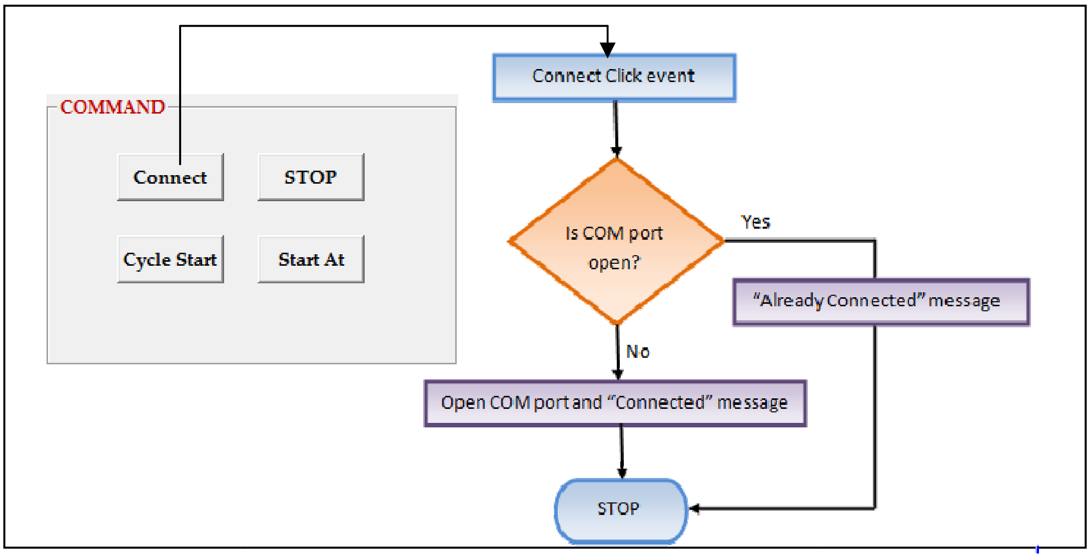



**Figure A.4: Connect Button to Open Serial Port.**

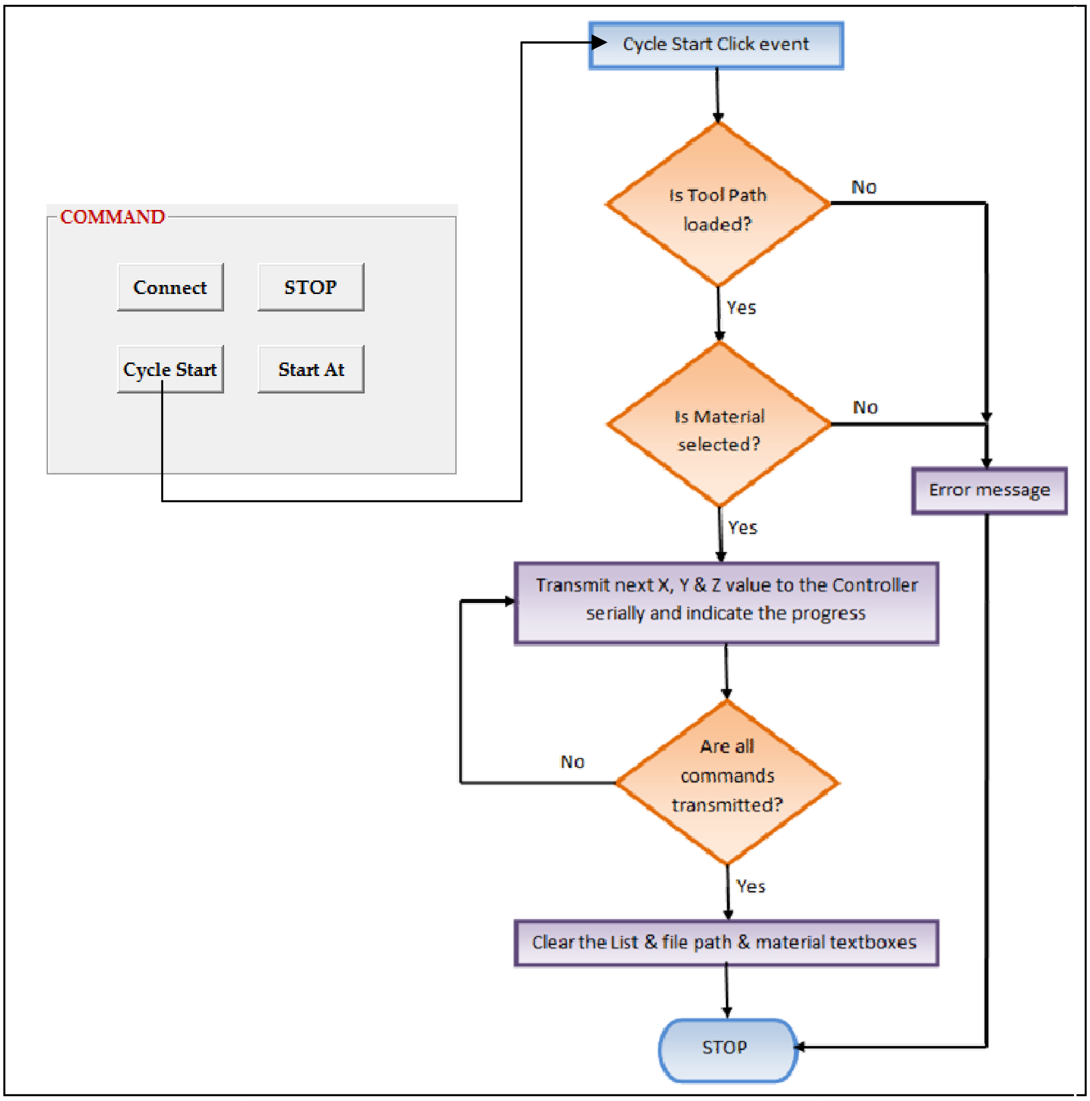



**Figure A.5: Click Command Button to Start Machining.**

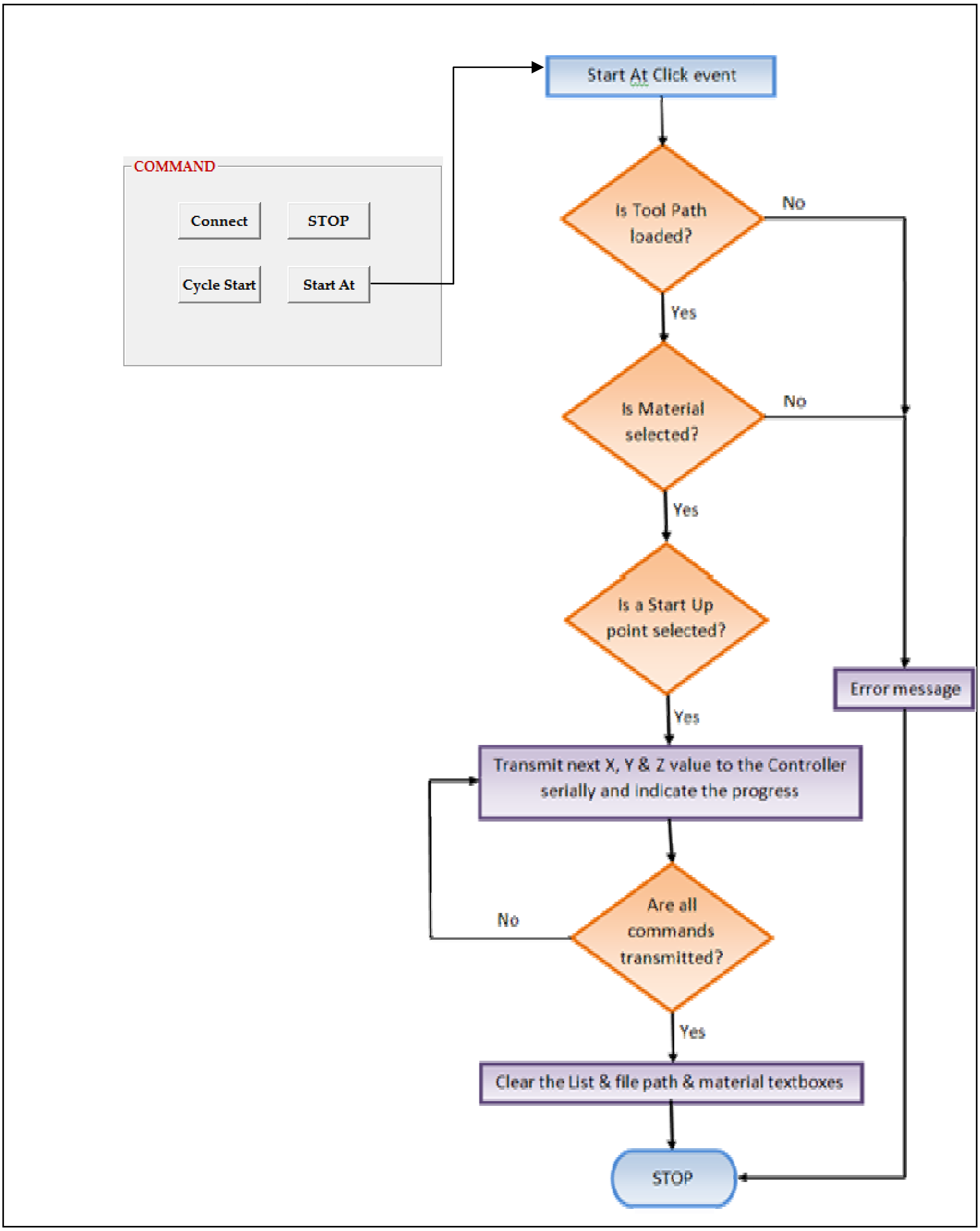



**Figure A.6: Click Command Button to Start "Start At" Machining.**

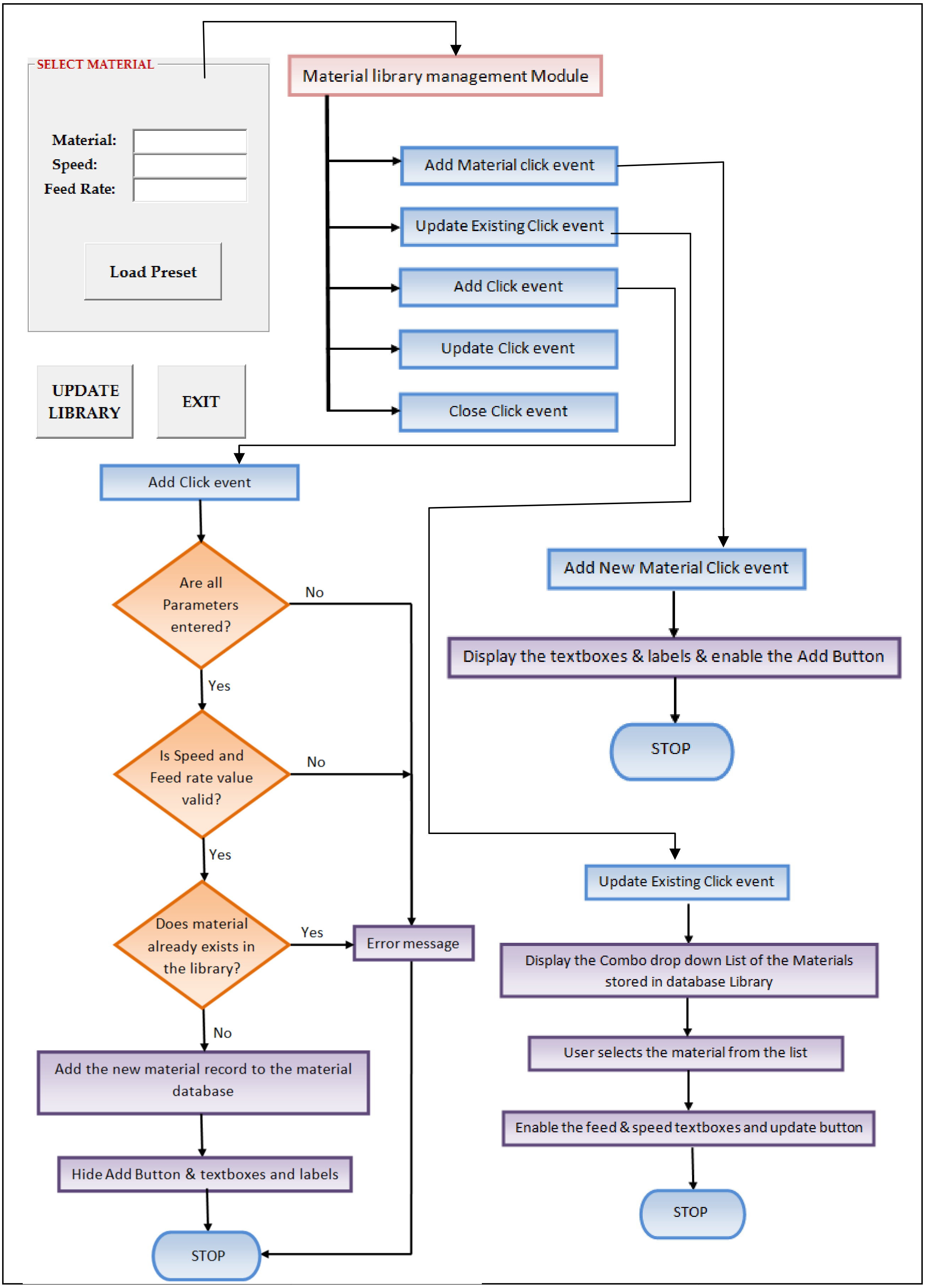



**Figure A.7: Material Library Management Subroutines.**

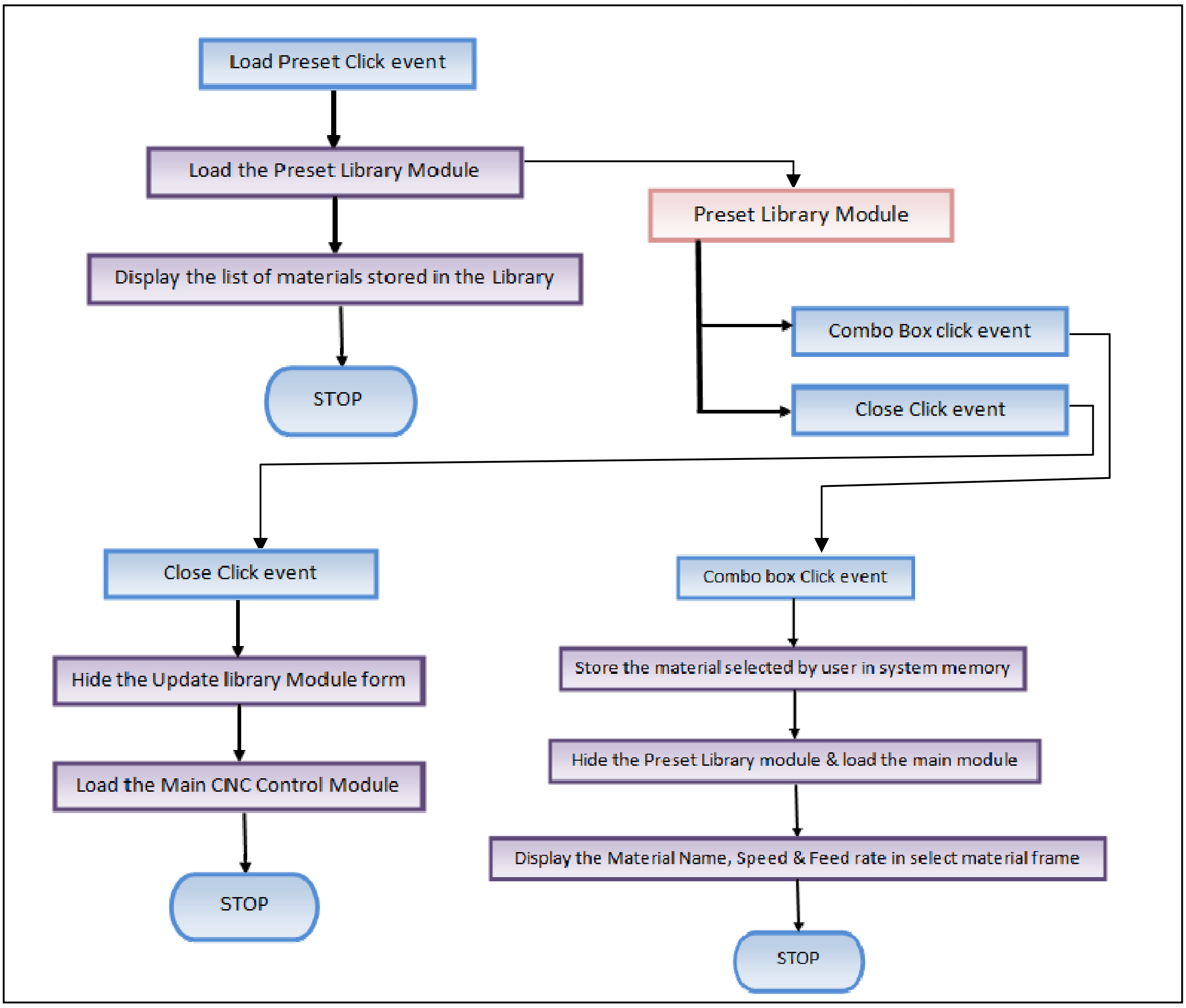



**Figure A.8: Update Already Stored Material Value in Library.**

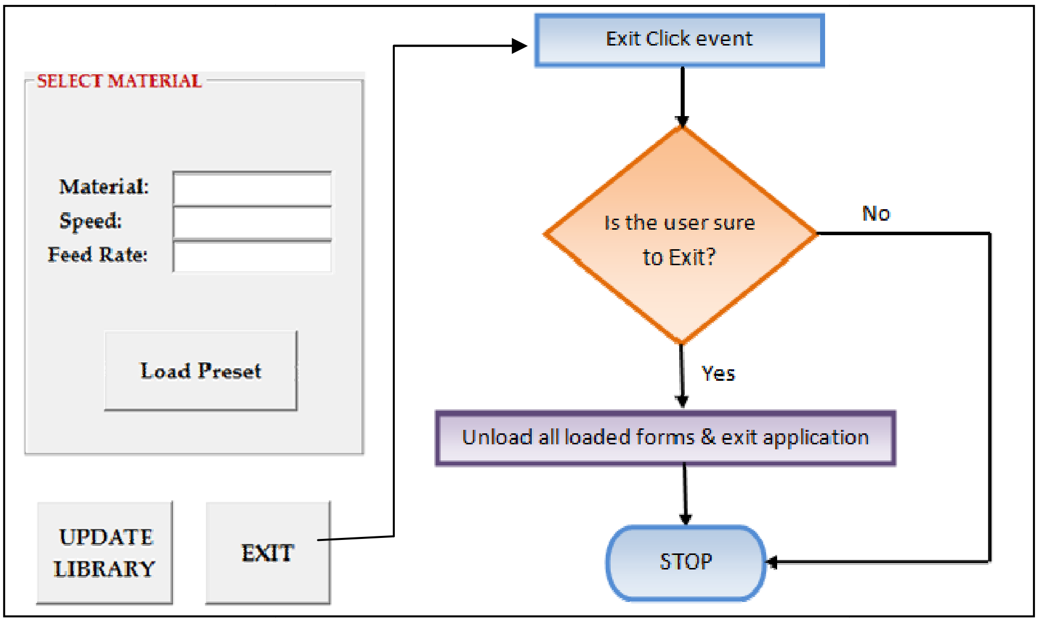



**Figure A.9: Exit Front End GUI Form.**

## REFERENCES


[1] L. Norberto López de Lacalle, A. Lamikiz,"Sculptured Surface Machining" Machining Fundamentals and Recent Advances,University of Aveiro, vol no. 100, pp 225-248, 1985.

[2] Chryssolouris G., Guillot, M., Domroese, M, "A Decision Making Approach to Machine Control", Trans. ASME Journal of Eng. for Jud., Vol 110, No 4, pp 397-398, 1988.

[3] Albus, J., H.G. McCain, R. Lumia, "NASA / NBS Standard Reference Model forTelerobot Control System Architecture (NASREM), Technical Report 1235, National Bureau of Standards, Gaithersburg, MD, USA, 1989.

[4] Pritchow, G., "Automation Technology - On the way to an Open System Architecture", Robotics and CIM, Vol 7, No.1-2, pp 103-111, 1990.

[5] O'hare,G.M.,"Distributed Artificial Intelligence: An Invaluable Technique for Development of Intelligent Manufacturing Systems", Annals of the CIRP, Vol 39, No 1, pp 485-488, 1990.

[6] Pritchow, G., "Automation Technology - On the way to an Open System Architecture", Robotics and CIM, Vol 7, No.1-2, pp 103-111, 1990.

[7] Wright, P.K., E. Pavlakos, F. Hansen, "Controlling the Physics of Machining on a New Open Architecture Manufacturing System", ASME PED-Vol.53, pp 93-100,1991.

[8] Altintas, Y., N. Newel, M. Ito, "A Hierarchical Open Architecture Multi-Processor CNC System for Motion and Process Control", ASME International Mechanical Engineering Congress, PED-Vol 64, pp 195-201,1993.

[9] C. G. Jensen, S. H. Mullins and D. C. Anderson, "Scallop elimination based on precise 5-axis tool placement, orientation and step-over calculations", ASME Advances in Design Automation, DE 65(2), pp 535-544, 1993.

[10] Hansen, F., E. Pavlakos, E. Haffman, T. Kanade, R. Reddy, P. Wright, "PARES: A Prototyping and Reverse Engineering System for Mechanical Parts on Demand on the National Networks", Journal of Manufacturing systems. Vol12, No 4, pp 269-281, 1993.

[11] Takata, S. "Generation of a Machining Scenario and its Applications to Intelligent Machining", Annals of the CIRP, Vol.42, No.l, pp 531-534,1993.

[12] Teltz, R., and Elbestawi, M.A. "Hierarchical, Knowledge Based Control in Turning", Trans. ASME Journal of Dyn. Sys. Meas. Control, Vol.122, pp 122-131, 1993.

[13] Haynes, T., "The NCMS Low End Controller Project", Proc. of the International Workshop on Open Architecture Controllers for Automation, Univ. of Michigan, pp 143-160,1994.

[14] OMAC, "Requirements of Open, Modular Architecture Controllers for Applications in the Automotive Industry", Version 1.1, 1994.

[15] Pritchow, G., G. Junghans, "General Overview of the OSACA Project and The OSACA Architecture", Proceedings of the International Workshop on Open Architecture Controllers for Automation, Ann Arbor MI, April, 12, pp 15-70,1994.

[16] Yellowley, I. P. Pottier,"The Integration of Process and Geometry within an Open Architecture Machine Tool Control System", International Journal of Machine Tools and Manufacture, Vol 34, No 2, pp 277-293, 1994.

[17] Chiang, S.T., D.I. Liu, A.C. lee, W.H. Chang, "Adaptive Control Optimization in End Milling using Neural Networks", International Journal of Machine Tools and Manufacturing, Vol 35, No 4, pp 637-660,1995.

[18] Hatamura, Y., T. Nagao, M. Mitsuishi, M. Nakao, "Actual Concept Design Process for Intelligent Machining Centers", Annals of the CIRP, Vol 44, No 1, pp 123-128, 1995.

[19] S.H. Suh, S.K. Noh, Y.J. Choi, "A PC-based retrofitting toward CAD/CAM/CNC integration", Computers & Industrial Engineering, Vol. 28, Issue 1, pp. 133-146, January 1995.

[20] S.J. Rober, Y.C. Shin, "Modeling and control of CNC machines using a PC-based open architecture controller", Mechatronics, Vol. 5, Issue 4, pp. 401-420, June 1995.

[21] Birla. S., J. Park, Z. Pasek, A.G. Ulsoy, Y. Shan, Y. Koren, "An Open Architecture Testbed for Real Time Monitoring and Control of Machining Processes", Proceedings of the American Control Conference, Seattle Washington, pp 200-204, June 1996.

[22] N. Rao, S. Bedi and R. Buchal"Implementation of the Principal-Axis Method for Machining of Complex Surfaces",Int J Adv Manuf Technol, pp249-257, 1996.

[23] Park, J., S. Birla, K.G.Shin, Z.J. Pasek, G. Ulsoy, Y. Koren, "An open architecture testbed for real time monitoring and control of machining processes", Proc. of the American Control Conference, pp 200-204, 1996.

[24] Zadeh, L.,, In "Intelligent Control Systems, Theory and Applications", edited by MM. Gupta and N.K. Sinha, IEEE Press, New York, pp xix-xx, 1996.

[25] Sperling, W., and P. Lutz, "Designing Applications for an OSACA Control", ASME International Mechanical Engineering Congress, MED Vol 6-1, pp 91-95, 1997.

[26] FA Open Systems Promotion Group, "Outline of MSTC/JOP", Manufacturing Science and Technology Center (MSTC), Japan, May 17, 1999

[27] A Warkentin, Ismail F, Bedi S" Multi-point tool positioning strategy for 5-axis machining of sculptured surfaces." Comput Aided Geom Des 17(1),pp 83–100, 2000.

[28] R. Jerard, B. Fussel, M. Ecran, J. Hemmett, "Integration of Geometric and Mechanistic Models of NC Machining into an Open-Architecture Machine Tool Controller", Proc. of Intl. Mech. Eng. Congress and Exposition, Orlando, Florida, 2000

[29] Entivity Inc., "A White Paper: Integrating PC-Based Logic and Motion Control", Ann Arbor, MI, 48108.

[30] Bedi, S. Gravelle and Y. H. Chen, "Principle curvature K alignment technique for machining complex surfaces", ASME Journal of Engineering for Industry, 2001.

[31] D. Roth, S. Bedi, and F. Ismail, "Surface Swept by a Toroidal Cutter during 5-Axis Machining of Curved Surfaces", Computer-Aided Design, vol. 33, no. 1, pp. 57-63, 2001.

[32] J. Nacsa, "Comparison of Three Different Open Architecture Controllers", Proc. of IFAC MIM, Prague, pp. 134-138, , 2001.

[33] M. Mori, K. Yamazaki, M. Fujishima, J. Liu, N. Furukawa, "A Study on Development of an Open Servo System for Intelligent Control of a CNC Machine Tool", Annals of the CIRP, Vol. 50, Issue 1, pp. 247-250, 2001.

[34] Z. Gong, S. Huang, H. Zeng, "Development of PC-based Adaptive CNC Control Syste" , SIMTech Technical Report (AT/01/043/AMP), Singapore Institute of Manufacturing Technology, 2001.

[35] C. M. Chuang, C. Y Chen, and H. T. Yau, "A reverse engineering approach to generating interference-free tool paths in three-axis machining from scanned data of physical models", International Journal of Advances in Manufacturing Technology, vol. 19, pp. 23-31, 2002.

[36] A. Lamikiz , L.N. Lo´pez de Lacalle, J.A. Sa´nchez, M.A. Salgado, "Cutting force estimation in sculptured surface milling", International Journal of Machine Tools & Manufacture, Vol 44, 1511–1526, 2004.

[37] MIT Libraries: The Institute Archives, "History of the MIT Servomechanisms Laboratory", Massachusetts Institute of Technology, September, 2004.

[38] Seamus Gordon 1, Michael T. Hillery, "Development of a High-Speed CNC Cutting Machine Using Linear Motors", Journal of Materials Processing Technology, Vol 166, pp:21–329, 2005.

[39] R. Salami, M.H. Sadeghi, B. Motake, "Feed Rate Optimization For 3-Axis Ball-End Milling Of Sculptured Surfaces", International Journal of Machine Tools & Manufacture, Vol 47, pp: 760–767, 2006.

[40] X.W. Xu, S.T. Newman, "Making CNC machine tools more open, interoperable and intelligent—a review of the technologies.", Computer In Industry, Vol. 57, pp. 141-152, 2006.

[41] Y. Hosseinkhani, J. Akbari, A. Vafaeesefat" Penetration–elimination method for five-axis CNC machining of sculptured surfaces", International Journal of Machine Tools & Manufacture vol. 47,pp 1625–1635,2007.

[42] Jasra P. M., "Generalized Tool Path Generation Algorithm for Sculptured Pseudo Symmetric Surface Machining", M.E. Thesis, Mechanical Engineering Department, Thapar University, Patiala, 2009.

[43] Yuwen Sun , FeiRen, DongmingGuo, Zhenyuan Jia, "Estimation and experimental validation of cutting forces in ball-end milling of sculptured surfaces", International Journal of Machine Tools and Manufacture, Vol 49, Issue 15, pp 1238-1244, 2009.