
USING ROBOT TO SERVE THE NC LATHE



Bachelor's thesis

Mechanical Engineering & Production Technology

Riihimäki 25.11.2011

Pablo, John Paul D. & Rahman, Mohammad Ziaur



Riihimäki
Mechanical Engineering and Production Technology

Author	Pablo, John Paul D.	Year 2011
	Rahman, Mohammad Ziaur	
Subject of Bachelor's thesis	Using robot to serve the NC lathe	

In 2007, there was a project in HAMK, Riihimäki to integrate the robotic arm to the NC lathe machine. During that project, one company successfully integrated them. In spring 2011 a topic was introduced to us to make a demonstration to automatically load and unload the CNC lathe, using the robot arm, and machines the work pieces.

Basically, the objective of this thesis is to make a program for the robotic arm so it can load the CNC machine and retrieve it from the lathe after its being machined for six cycles. Six small aluminum cylindrical pieces were kept on a work table with same distance between them and the robotic arm was taught about their positions precisely. A simple part programming was required for the NC as the main objective was to show a demonstration of a semi-automated production cell. The process could not be made fully automated as the door of the CNC lathe is a manual one.

A very sophisticated method was chosen to make a single robotic program for all the six pieces. And, the speed of the robot movements has been frequently changed to establish it as a fast process. All information presented to the reader has been gathered from literature used at HAMK production lab, interviews with the lab employees, discussion with the thesis supervisor, observations made at the lab, books and internet sources.

The final result of the work was quite satisfactory as we were able to overcome all the barriers that came during this practical work. Further improvement of this project is still possible which could take a long time but the outcome should be bright.

Keywords programming, work envelope, integration, turning

Pages 48 p. + appendices 3 p.

CONTENTS

1	INTRODUCTION	1
2	ROBOT ARM	1
2.1	Work Envelope.....	1
2.2	Types	1
2.2.1	Cartesian	2
2.2.2	SCARA.....	2
2.2.3	Cylindrical	3
2.2.4	Delta	3
2.2.5	Polar.....	3
2.2.6	Articulated	4
2.3	Axis	4
2.4	End-Effectors	5
2.5	Digital Input / Digital Output	6
2.6	Programming.....	7
2.7	Motion Instruction.....	9
2.7.1	Motion Type	9
2.7.1.1	Linear Movement	9
2.7.1.2	Joint Movement	10
2.7.1.3	Circular Movement.....	10
2.7.2	Positional Information	11
2.7.3	Speed	11
2.7.4	Termination Type	11
2.7.4.1	Fine	11
2.7.4.2	Continuous.....	12
2.7.5	Motion Options.....	12
2.7.5.1	Incremental (INC).....	12
3	CNC MACHINES	13
3.1	Literature search for NC operations	13
3.2	Numerical control.....	16
3.2.1	Short history of numerical control.....	16
3.2.2	Basic components of a NC system	16
3.2.3	NC Coordinate system.....	18
3.2.4	Motion Control System	19
3.3	Programming for NC part	19
3.4	Turning Operations	20
3.4.1	Operating conditions	21
3.4.2	Cutting Speed	21
3.4.3	Depth of cut	21
3.4.4	Feed rate	22
3.4.5	Cutting time	22
4	AUTOMATED MANUFACTURING SYSTEM.....	22
4.1	Fixed automation.....	23

4.2	Programmable automation	23
4.3	Flexible automation.....	23
4.4	Reasons for automating.....	24
5	CELL DESCRIPTION	25
5.1	Fanuc robot arc mate 100i.....	25
5.1.1	Axis.....	25
5.1.2	Work envelope.....	25
5.1.3	Configuration.....	26
5.2	Puma 6-HS	27
6	INTEGRATING FANUC ARC MATE 100I TO PUMA 6 – HS.....	28
6.1	Wait Command	29
7	PROGRAMMING THE REQUIRED PART	29
7.1	FANUC ARC MATE 100i Robot Arm	30
7.2	Programming the required part for NC lathe machine.....	39
7.2.1	General precautions for the lathe.....	44
8	CONCLUSION	45
	SOURCES	47

Appendix 1 Absolute locations and orientations of the FANUC ARC MATE 100i Robot Arm used to help manufacture six work pieces in the PUMA 6-HS NC Lathe Machine.

Appendix 2 Incremental positions of the FANUC ARC MATE 100i Robot Arm used to help manufacture six work pieces in the PUMA 6-HS NC Lathe Machine.

Appendix 3 2-D drawing for the work part

1 INTRODUCTION

With the revolution of industrial automation, robots are slowly outnumbering humans. In this era of production, proficiency and faster production rate is the key to survive in the market, so the need for automated production facility comes in place. Speaking of automated systems, industrial robots are contributing widely in this sector. Industrial robots are used to make people's lives easier. Most of them were built to do tasks faster, precised and most importantly to do task which are impossible for people.

The purpose of this thesis is to establish a model semi-automated production system inside a production cell by using a robotic arm and a CNC lathe machine. The challenge is to make them communicate according to the instructions and to manage all the operations in a limited working space. A demonstration was made to machine six aluminum pieces where they were loaded and unloaded to the CNC by the robotic arm.

2 ROBOT ARM

A robot arm, widely known as an industrial robot, is a reprogrammable, automated machine which is used to execute tasks that are impossible or difficult to perform for a human being. According to ISO, industrial robots are automatically controlled, reprogrammable, multipurpose manipulator programmable in three or more axes (Industrial Robot, 2011). All industrial robots have a limited working area that is called a work envelope.

2.1 Work Envelope

Every industrial robot has its own allowable working area. This is called a work envelope. A work envelope is the range of the robot's allowable movement to perform its tasks. A robot's work envelope depends on its configuration (length of the arm, number and design of axis). A shape will be created when all possible movements of the robot are known.

Not all robots have the same work envelope. That is why it is good practice to have the work envelope of the robot first in mind before assigning that robot to perform a certain task. Although robots have a limited work envelope, it can be changed by modifying and putting adaptors or extensions on it.

2.2 Types

There are many ways to categorize industrial robots. They are commonly categorized according to their configuration or assembly. There are six main types of industrial robots: Cartesian, SCARA, cylindrical, delta, polar and vertically articulated (Robot Types, 2011). In addition to these,

there are more types of industrial robots but most of these originated from the six main types of robot configuration.

2.2.1 Cartesian

Cartesian configuration has only three prime axes and does not rotate. This configuration is probably the simplest configuration of the industrial robot. They only tend to move in a straight line (for example, up and down, left and right and forward and backward). Due to these all three axes are in a right angle all the time. Cartesian robots are usually big in size. An example a Cartesian configuration with its work envelope is shown in Figure 1.

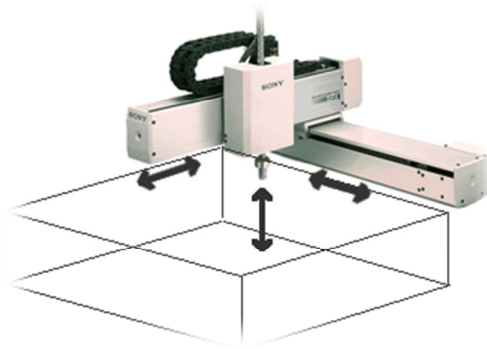


Figure 1 Cartesian configuration (Cartesian/Gantry Robot, 2011)

2.2.2 SCARA

SCARA also known as Selective Compliant Assembly Robot Arm or Selective Compliant Articulated Robot Arm configuration is known to have one main axis (Z axis). This configuration has its arm fixed to one of its axes (Z axis) and has an adjustable X and Y axis. It can execute a task faster than a Cartesian robot. All the joints on a SCARA robot are located at the end of the arm (SCARA vs. Cartesian Robots, 2011). Due to this, SCARA robots have a limited payload capacity which is why they are commonly used in handling small parts and best used in assembly. A example of a SCARA configuration with its work envelope is shown in Figure 2.

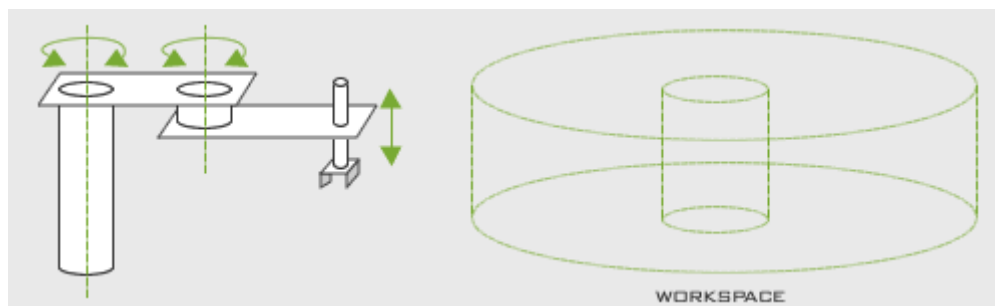


Figure 2 SCARA configuration (Types of Robots, 2009)

2.2.3 Cylindrical

Cylindrical types of robots have a rotary base that can rotate at a certain degree. The name cylindrical is derived from its work envelope that has the shape of a cylinder. A cylindrical type robot is shown in Figure 3 with its work envelope.

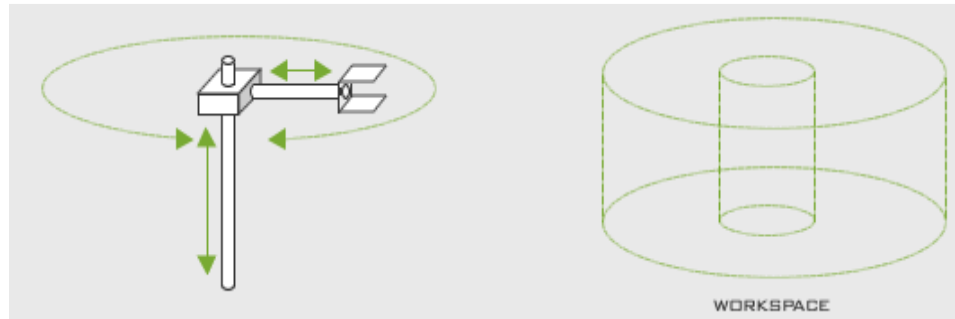


Figure 3 Cylindrical type robot (Types of Robots, 2009)

2.2.4 Delta

Delta types have spider-like legs. Each of these legs is linked and meets at one point where the end-effectors are attached. It has a dome-shaped work envelope that can be seen in Figure 4.



Figure 4 Delta type robot (Types of Robots, 2009)

2.2.5 Polar

Polar or commonly known as spherical robots consist of rotary joints to twist the torso and arm of the robot and at least one linear joint to have a reciprocating motion. Figure 5 shows an example of a Polar robot with its work envelope.

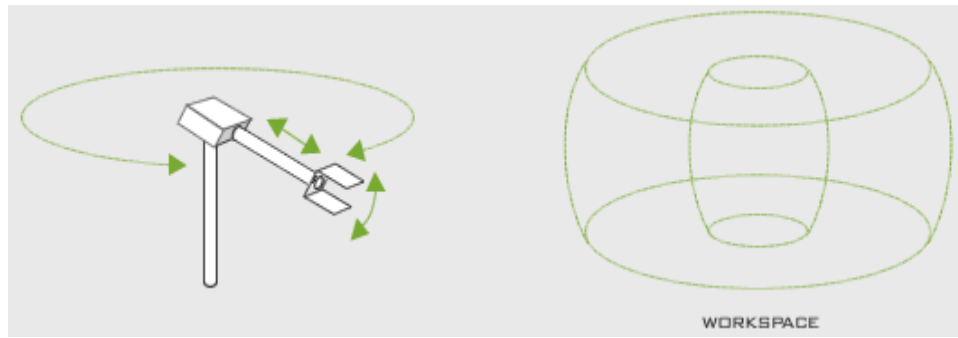


Figure 5 Polar/Cylindrical robot (Types of Robots, 2009)

2.2.6 Articulated

Articulated robots are robots that have rotary joints consisting of a body/trunk, shoulder, arm, elbow, forearm and wrist. The configuration of these robots is quite similar to a human being's arm that can reach a certain point with a limited work envelope. With this configuration, the robot can move more freely and reaches areas that Gantry and SCARA configurations cannot. Articulated robots are commonly used as a welding robot or as a handling robot. An Articulated robot type is shown in Figure 6.

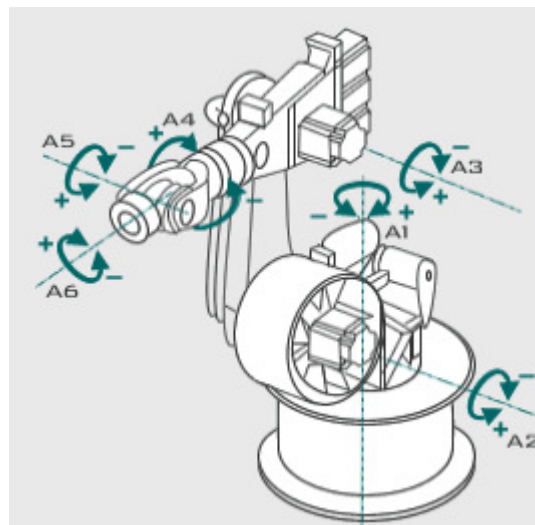


Figure 6 Articulated robot (Types of Robots, 2009)

2.3 Axis

Robots have a limited movement or limited task to perform. These limitations are due to how they are manufactured. There are many reasons why robots have a limited work envelope. One aspect that affects this is the axis used in the robot.

Axes are put in the robot to have a better and a more flexible movement. With these robots tend to run or perform tasks smoothly. The more axes a robot has the better for it to do tasks when compared to a robot that has fewer axes. Figure 7 illustrates all of the axes of an articulated type robot.

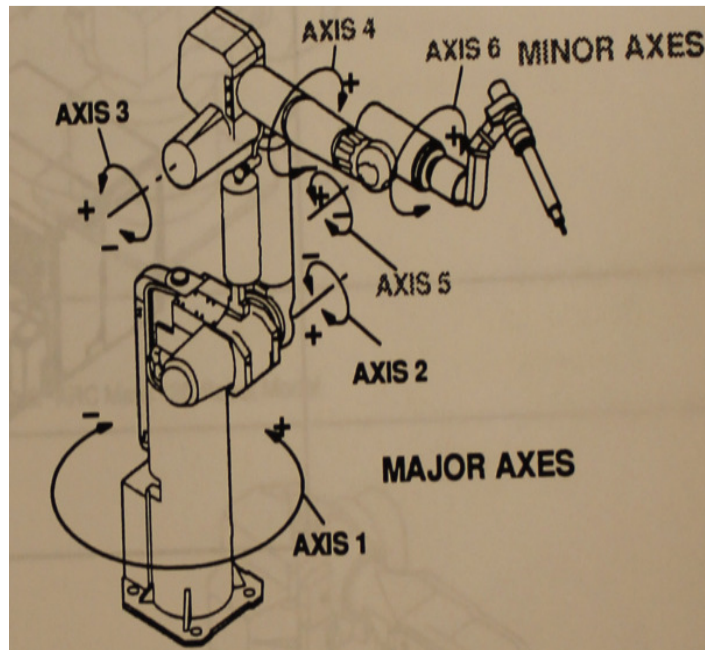


Figure 7 Axes of an articulated type robot (Fanuc Robot Series 1996, P1-3)

Most industrial articulated robots consist of six rotary joints or axes to be flexible enough. These axes are briefly described below:

- Axis 1 – is located at the base of the robot. It swings the robot from side to side.
- Axis 2 – is connecting the robot arm to the torso of the robot that is responsible for the back and forth movement. It is also referred to as the shoulder.
- Axis 3 – is called the elbow of the robot that provides up and down movements.
- Axis 4 – known as the forearm of the robot that pivots the arm up and down.
- Axis 5 – is the wrist of the robot that oscillates the arm from left to right or up to down depending on its orientation.
- Axis 6 – is located at the wrist of the robot that connects the end-effectors to the arm. It is responsible for rotating the tool attached to it.

2.4 End-Effectors

Robots need hands to perform a certain task. Most robots cannot complete its task without a hand attached at tip of its arm. They can move from point-to-point but that is just the only thing they could do. By attaching an end-effect to the end of the robot arm, it can perform tasks that humans usually do. The end-effector is the part of the robot that interacts with the environment (End effector, 2008). Different kinds of end-effectors can be attached to the robot depending on its use. You could attach a spray gun if

you want the robot to paint or a gripper to hold or carry parts. Here are some examples of end-effectors:



Figure 8 End-effectors (from top left clockwise) welding gun (End-Effectors, 2011), vacuum cup (End-Effectors, 2011), cutting tool (End-Effectors, 2011), magnet (Magnet type end effector, 2011), gripper, drill (End Effectors, 2011), and spray gun (End-Effectors, 2011)

2.5 Digital Input / Digital Output

Digital inputs and outputs allow the robot to send and receive signals. They can be configure, monitor, and manually control (with the exception of Digital Inputs). Digital outputs can be configured to control a device depending on its use.

Digital output can be configured independently so it will send one signal. Digital output could also be configure in a way that it is paired to a digital input while it sends a signal to a device connected to it. This configuration is called complimentary pair. When a command switches the output ON, this configuration switches the input, opposite to it, OFF and vice

versa. Figure 9 illustrates a simple explanation on how complimentary output signal works.

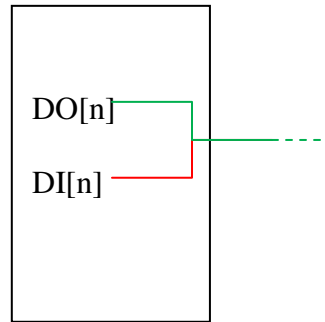


Figure 9 DO[n] is ON and switches DI[n] OFF

Digital input (DI) and digital output (DO) signals are user-controlled input and output signals (Fanuc Robot Series 1996, P7-66). DO can be manually switched ON or OFF by the user using the R-J2 Teach Pendant controller. The number 5 indicates the location or address of the input/output that needs to be used. Figure 10 indicates a more detailed explanation of digital input and output in the program.

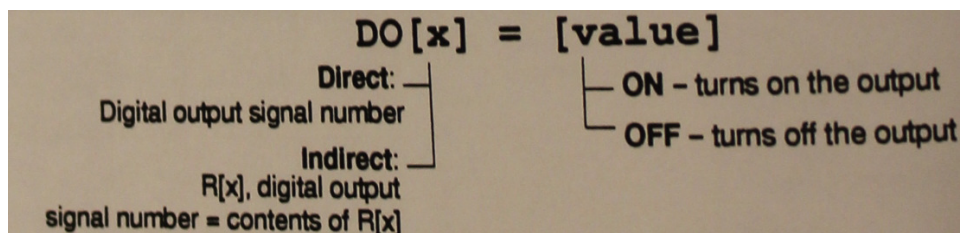


Figure 10 DO[X] = ON/OFF (Fanuc Robot Series 1996, P7-66)

2.6 Programming

Humans communicate with one another by talking, facial expressions and body language. They send signals to each other to let them know what they desire and feel. When it comes to robots, humans communicate to them through the use of robot language or programming language. Humans convert their commands to programming language so the robot may understand and perform the desired task.

The whole structure of a robot is connected by cables to a digital computer called Programmable Logic Controller (PLC) that gives a command to the robot. In general, PLC receives the signal that was received from the controller and sends the signal to the robot to perform the command.

All robots have different programming languages. Usually, all robots coming from the same manufacturer may have the same language. Although programming languages of the robots are different from one to another, there are similarities. An example of a programming language is shown below:

PROGRAM PICKPLACE (Robot Software, 2011)

1. MOVE P1
2. MOVE P2
3. MOVE P3
4. CLOSEI 0.00
5. MOVE P4
6. MOVE P5
7. OPENI 0.00
8. MOVE P1

END

This is an example of a robot programming language called VAL (Variable Assembly Language).

Fanuc Arc Mate 100i uses RJ2 Teach Pendant to write and gives commands to the robot. It reads and computes lines and equations quickly giving you a fast and accurate result. The RJ2 controller runs ArcTool software.

When translating the previous program to the programming language Fanuc Arc Mate 100i has it will look like:

Program Pickplace

1. J P[1] 50% CNT 50
2. L P[2] 50mm/s CNT 50
3. L P[3] 50mm/s CNT 50
4. CALL CLOSE
5. L P[4] 50mm/s CNT 50
6. L P[5] 50mm/s CNT 50
7. CALL OPEN
8. J P[1] 50% CNT 50

END

Both of these robot programming languages are different but you can definitely see the similarities between the two. Both of them will perform the task exactly even though they have different programming language. The type of robot programming language will vary depending on its manufacturer or developer.

In line 1 of Fanuc Arc Mate 100i programming language, this line commands the robot to move to point 1 using joint movement having 50% speed with 50 continuous termination. Figure 11 describes more clearly the command movements that can be written in RJ2 controller.

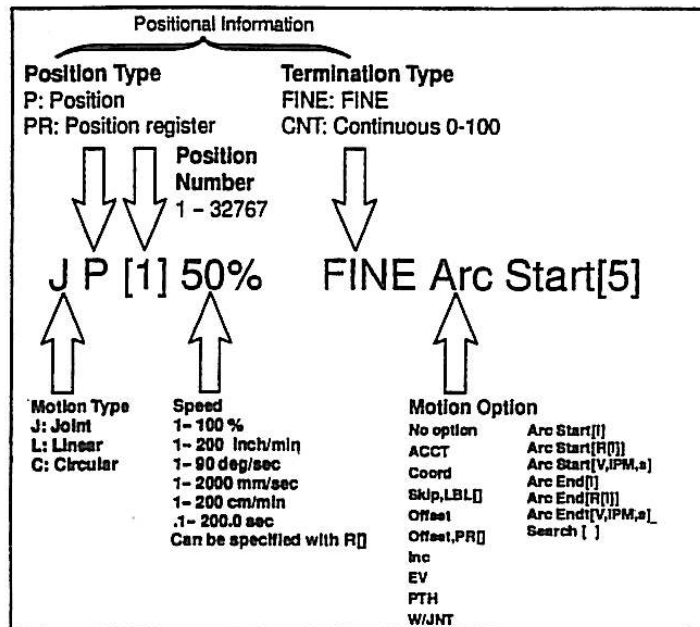


Figure 11 A Typical Motion Instruction (Fanuc Robot Series 1996, P7-9)

2.7 Motion Instruction

In order for the robot arm to move from one point to another, the user must instruct the robot to do so. With the use of the motion instructions, the robot can move to a certain location and perform its task. A motion instruction contains the motion type, positional information, speed, termination type, and some optional motion instruction for the robot.

2.7.1 Motion Type

A motion type is a kind of movement the robot will perform when moving to the desired location. There are three types of motion type the robot arm can perform. These are:

2.7.1.1 Linear Movement

A linear movement is performed when the tool center point moves in a straight line from the starting point to the desired position. When using this movement, the speed of the robot arm can be seen as millimeters per second, centimeters per second, inches per minute, and degrees per second depending on the unit setting. Figure 12 shows a sample movement of a robot arm using a linear movement.

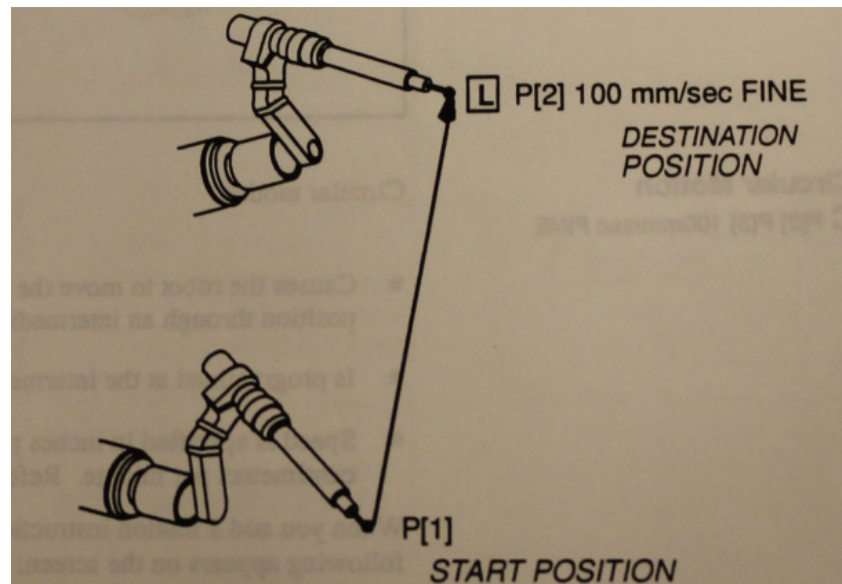


Figure 12 Linear Motion (Fanuc Robot Series 1996, P7-11)

2.7.1.2 Joint Movement

Joint movement allows the robot arm to move from one point to another using its joints. The speed of the movement is expressed in percentage (%). Figure 13 shows a sample of joint movement.

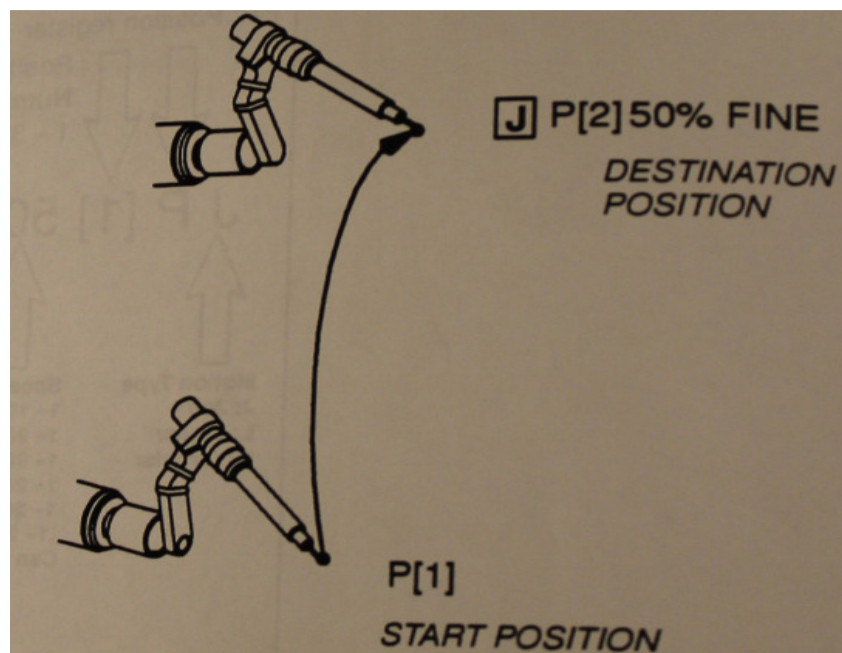


Figure 13 Joint Motion (Fanuc Robot Series 1996, P7-10)

2.7.1.3 Circular Movement

Circular movement allows the robot to move in an arc or a curve line. Circular movement is mainly used in welding. It also shows the speeds in

millimeters per second, centimeters per second, inches per minute, and degrees per second.

2.7.2 Positional Information

The position information indicates the name of the desired location for the robot to be. The exact coordinates of the position can be seen by highlighting the position number and pressing the “position” button.

2.7.3 Speed

The speed indicates how fast the robot arm will move to the desired position. It can be presented in millimeters per second, centimeters per second, inches per minute, degrees per second, and in percentage depending on the motion type used.

2.7.4 Termination Type

A termination type is a kind of movement the robot will perform towards the end of a command.

2.7.4.1 Fine

This type of termination makes the robot stop at one point before continuing to the next point. Figure 14 illustrates a detailed transition of the robot arm having a fine termination.

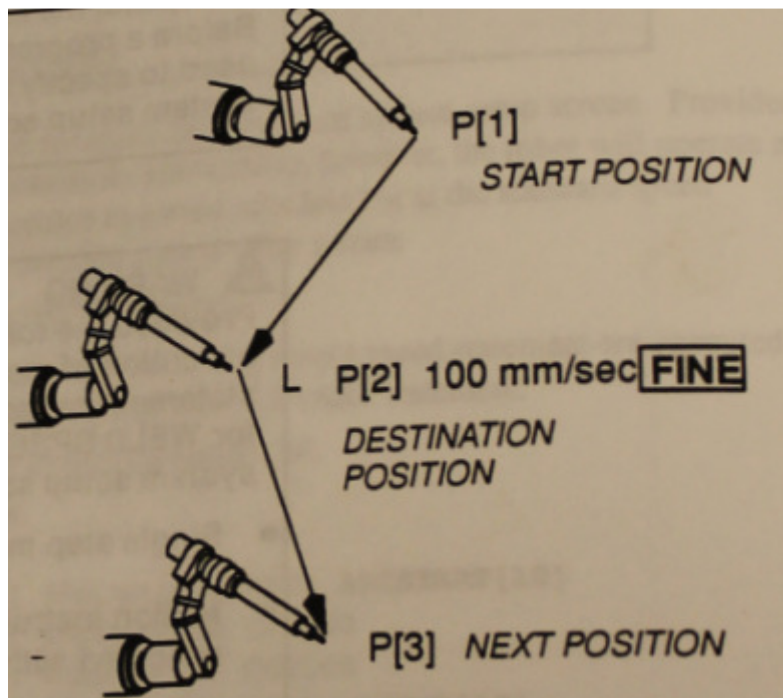


Figure 14 A Linear Robot Movement with a Fine Termination (Fanuc Robot Series 1996, P7-32)

2.7.4.2 Continuous

In continuous termination, the robot arm decelerates as it approaches a desired point but not stopping at it. It accelerates as it passes one point and continues to the next point. The robot arm may or may not reach its destination depending on the continuous value. The value ranges from 0 to 100 with 0 being closest to the desired point (also equivalent to Fine termination). Figure 15 illustrates the robots transition with different continuous values.

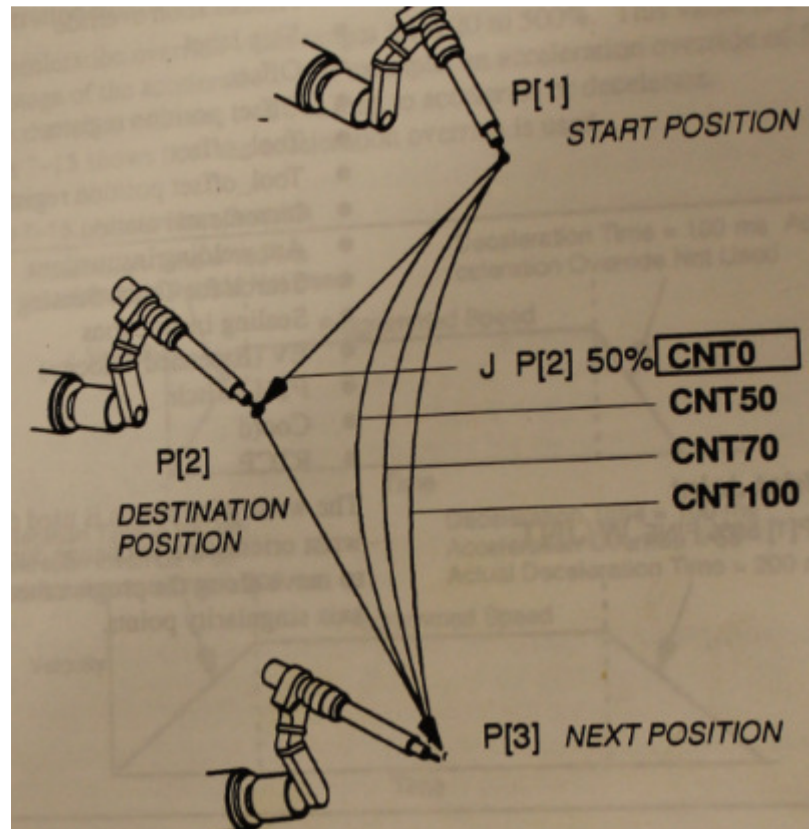


Figure 15 A Joint Movement with Continuous Terminations (Fanuc Robot Series 1996, P7-33)

2.7.5 Motion Options

Motion option indicates the additional instruction for the robot while performing the task. This includes some additional motion options for the robot like incremental movements.

2.7.5.1 Incremental (INC)

An Incremental motion allows the robot arm to move from one point to another with respect to the starting point. This movement makes the previous point its new origin and move to the next point. Figure 16 shows the difference between absolute and incremental motion coordinates.

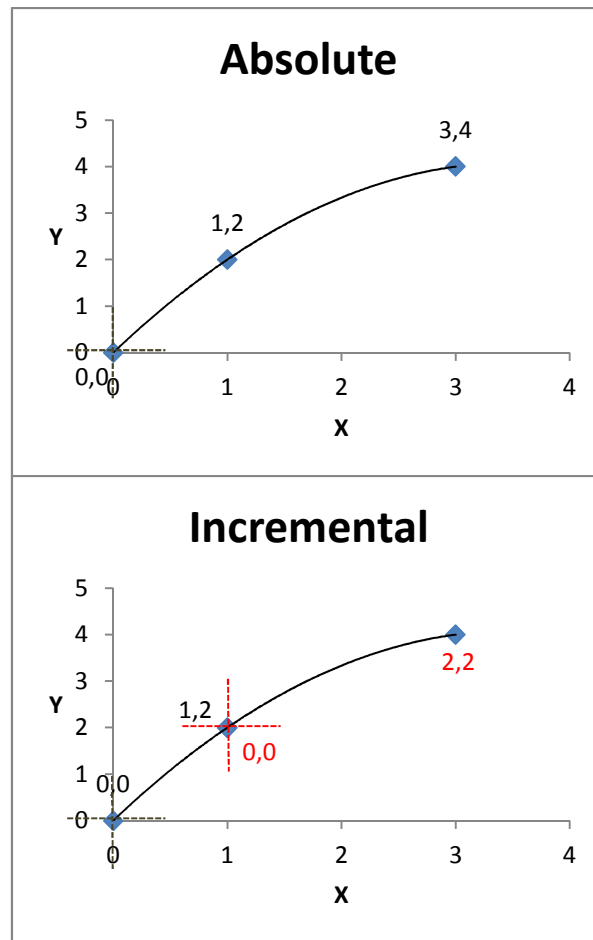


Figure 16 Absolute vs. Incremental. Robot arm's coordinates in absolute (top), robot arm's coordinates in incremental (bottom)

3 CNC MACHINES

Computerized numerical control machine, vastly known as CNC machines are sophisticated machine working tools required by modern technology which utilizes computer logic to control movements. It is a numerically controlled machine tool which is able to position a cutting tool with great accuracy without the need to make hard tooling. Computer NC systems include different additional features such as big storage for program memory, various form of program inputs, acceleration and deceleration calculations, communication interfaces etc. CNCs are run by a machine control unit (usually built-in) and CNC software.

3.1 Literature search for NC operations

Absolute coordinates: When programming with absolute coordinates, there is one point on the work piece that is called program zero. All positions that are programmed to move around the tool are depended on this zero point. The command: G00 X20. Z15. means the tool to move 20 mm

towards the right side of the zero point and 15mm above the program zero on Z axis.

Incremental coordinates: Incremental coordinate programming contains programs where tools are to travel in an incremental value every time with a new offset. In order to set the machine into incremental mood G91 code has to be used. In turning operations usually x & y letters are substituted with U & W, instead of commanding G91.

Circular interpolation: It is a type of motion in which a programmed command specifies the end point, radius of the circle and the feed rate. The control then takes the command and moves the tool from the current position to the end point in a circular motion with a given radius at a constant velocity. It can be performed in two axes.

Linear interpolation: Linear interpolation is a type of motion in which a programmed command specifies the end point and rate of feed. The control then takes the command and moves the tool from its current position to the end point in a straight line in a constant velocity (feed rate). Linear interpolation can be performed in one or more axes (Chapman 2004, 225)

Home position: Each axis of a CNC machine needs a reference point which establishes a reference between the machine tool and its slides. It's not a standard but the home position is normally the farthest position that the axes can move in the plus direction for each axes. "Zero return" is a mode offered by the machine that allows the user to move the axes manually to the home position.

Offsets: An offset is used to shift the tool path to compensate for something that may vary from setup to setup (Chapman 2004, 226). CNCs have several types of offsets like tool length compensation or tool radius compensation which has to be done in the beginning of the machining process. Turning machines usually use geometrical offsets to establish a reference point between the tool and work piece.

Program zero: Program zero is an initial point on the work piece from where all the coordinates are referenced. The programmer should keep in mind that this program zero has to be set with absolute coordinate marking, not incremental.

Edit mode: In this mode it's possible to load and alter the CNC program. In the edit mode the user can make any change to the program such as altering codes, inserting new ones, or deletion of any. Program can also be loaded into the machine from an outer device. Usually personal computers are integrated to the CNC to upload new programs while it's in the edit mode. The information is transmitted electronically through the RS232 (serial) port of the machine via a cable that is connected to the personal computer (Chapman 2004, 224).

Jog mode: With this mode the machine can be operated manually which basically means moving the turret manually after switching on the function from the control panel.

MDI mode: MDI or manual data input allows the user to write NC codes and then execute. The command is retained by the control unit until it is executed by pressing the cycle start button, then it is forgotten (Chapman 2004, 225). The user can then reset and type a new command upon choice. It is usually used to clamp or unclamp the chuck or other manual interventions.

Cycle start: This is a button on the CNC used to start the machining cycle. To run a program, the machine must be in the automatic mode.

Single block: By turning on this button the user can assure to run a program line by line, not the whole program at once. But, it is not possible to jump between lines by this mode but basically it continues the next line when cycle start button is pressed. It allows the operator to take the full control of the process.

Machine lock: Machine lock assures that no axes movement during a cycle of operation. First, it was launched to check program errors by initiating the program without any tool movement thus no machining. This procedure should be completed all the way to avoid the risk of possible coordinate shifting and resulting in a crash.

Main program: main program is the highest order or the master program of the entire program. Existences of most of the program are as main programs although it can have many similar smaller parts which can be called sub-programs. These subprograms are called in the main program to run the cycle and also called off to terminate.

Emergency stop: This button is used as a panic button to stop all kind of machining operation such as axes motion, spindle, coolant, hydraulics in case something is not right. To reset this condition the emergency button has to be rotated 1/4 turn so that it can come back out. Another way is to pull back the switch. (Chapman 2004, 224.)

Control: The control is a computer which runs the CNC. Back in the days this control used to be a huge unit which occupied a lot of working space. But, in modern machining era CNC controls are smaller and they comfortably fits inside a chamber attached to the machine. The control must provide a means to operate both operate the machine manually during setup and to run the machining during automated cycle.

Turret: turret is the device in a turning center which is mounted by different machining tools which moves around an axis. It usually contains 8-12 tools and rotates when it is changed to the desired one.

3.2 Numerical control

Numerical control or vastly known as NC is a type of programmable automation. In NC the mechanical actions of a machine tool or other equipment are controlled by a program containing coded data. These coded data represents relative positions between a tool and the work part as well as other processing conditions. The work part is the object being processed. It is highly proficient for medium and low scale production as it is capable of changing the program. It is much easier to write new programs than to make major alternations of the processing equipment. Numerical control can be applied in various applications. It can be divided into two categories:

- Machine tool operations, such as drilling, turning, milling and other metal working.
- Non-machine tool applications, such as assembly, drafting and inspection.

3.2.1 Short history of numerical control

The first NC machine was a milling machine which was converted to NC. In late 50's there was a big demand of machining complex tooling fixtures. Conceptionally, the effort to create a numerically controlled machine tool was borne of the desire to be able to position a cutting tool with great accuracy without the need to make hard tooling, such as cams or other mechanical positioning devices (Chapman 2004, 221). The best contributor in this sector in the beginning was John Parsons who was a defense contractor for the US Air Force. In 1940s Air Force was experimenting with the idea of using coordinate position data stored on punched cards to define and machine the surface contours of airfoil shapes. After he presented his idea to the air force he was awarded the contract and then he made a sub contract with the Massachusetts Institute of Technology to: (1) perform a systems engineering study on machine tool controls and (2) develop a prototype machine tool based on the principle. In the beginning it became clear that the required data transfer rates between the controller and the machine tool could not be achieved using punched cards. Then the idea of punch paper tape or magnetic tape was proposed as a solution to store the numerical data. The name Numerical Control was adopted in March 1951. John Parsons arranged a naming contest among the MIT employees who were working on this project. (Groover 2001, 121.)

Mass production of NC which utilizes minicomputers began in 1970s. In a few years microchips replaced these minicomputers in NC machine tool controls. During that time, the name changed from Numerical control into Computerized Numerical Control.

3.2.2 Basic components of a NC system

A basic NC system consists of three main components: (1) a program of instruction, (2) a machine control unit, and (3) processing equipment. The general relationship between these three can be shown in following figure:

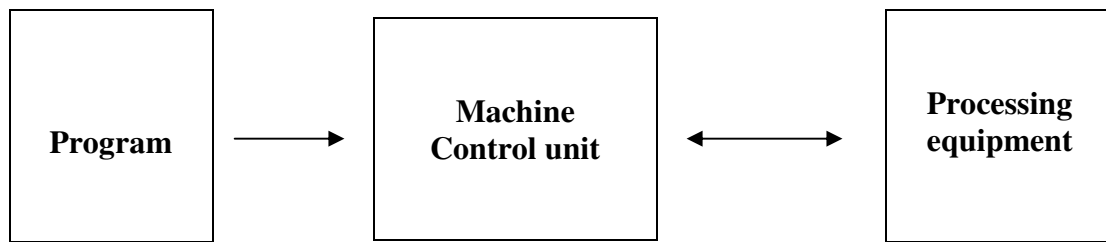


Figure 17 Basic components of an NC system The first component of NC system is the part programming which will be discussed broadly later in this thesis.

MCU or commonly known as Machine control unit includes a microcomputer in modern era of machining. This control unit is related to a control hardware that stores the program of instructions written by the programmer. Then it executes each command line by line into mechanical actions of the processing equipment. Usually the procedure is one command at a time to avoid any machine confusion. The related hardware of the MCU includes components to interface with the processing equipment and feedback control elements (Groover 2001, 123). The MCU also got to have one or more reading devices for entering part programs into the memory of the NC machine. The type of reader would depend on the storage media that is available. MCU also consists other software such as control system software, transition software and more importantly calculation algorithms to convert the NC part programs into a suitable format for the MCU and convert these signals into a mechanical action. MCU is basically a computer so the term computerized numerical control (CNC) is used to establish a type of NC. Nowadays all MCUs are based on computer technologies so that NC and CNC basically is the same thing.

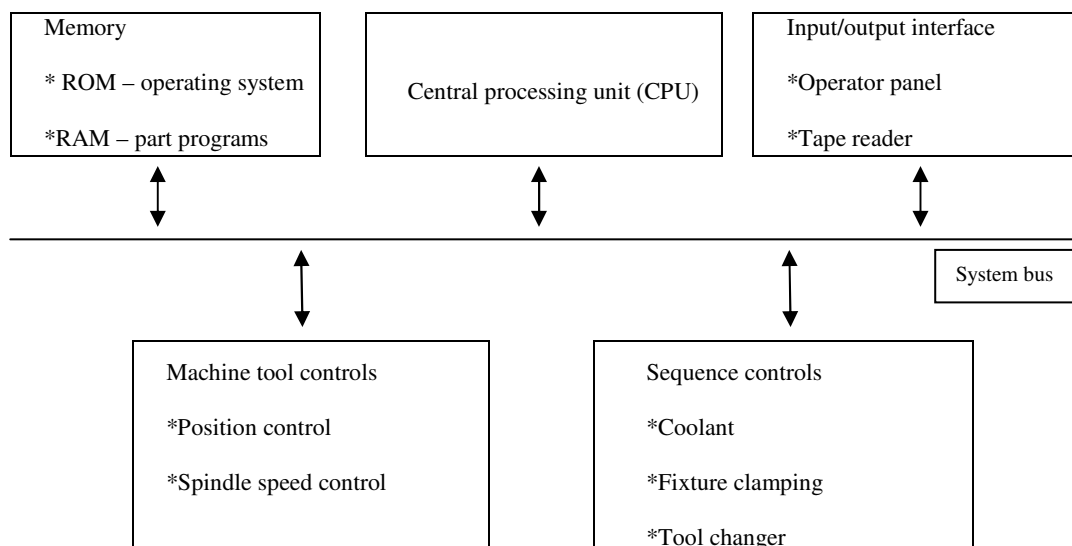


Figure 18 Configuration of CNC machine control unit (Groover 2001, 131)

The processing equipment in the NC system is another basic component which performs useful works. These equipments are responsible for processing a work piece into a ready part. The operation is directed from the MCU and its parts like spindle, chuck, tools does the rest according to the set of instructions which is written and coded as part program. The other parts of this equipment are motors, worktable, coolant machine etc.

3.2.3 NC Coordinate system

In order to program the NC processing equipments a standard coordinate system should be established to perform every operation. It's basically the relative distance and movement between the work head and the processing part. There are two axis systems used in NC, one for flat and prismatic work parts and the other for rotational parts. Both axis systems are based on the Cartesian coordinate system. (Groover 2001, 124.)

The axis system for a simple flat part includes three linear axes which are labeled as x, y & z axis and three rotational axis around them which could be labeled as a, b & c axis. In most CNCs x and y axis are used to move and position the worktable that part is attached. The z axis is used to control the vertical position of the cutting tool. This kind of positioning system could be applied to simple machining processes such as milling or punching of flat sheet metals.

The a, b & c axis represents the rotational axis for x, y & z axis respectively (shown in Figure 19). To define the angel right hand screw rule is applied. This rotational axis can be used in the orientation of the work part to present different surfaces for machining or the tool at some angel related to the part. Advanced machining centers usually have either four or five axes which consists three linear axes plus one or two rotational axes.

The coordinate system for a rotational NC system has been shown in Figure 19. These systems are related to NC lathes and turning machines. In turning operations the y-axis is not used. The path of the cutting according to the rotating work piece is defined in the x-z plane.

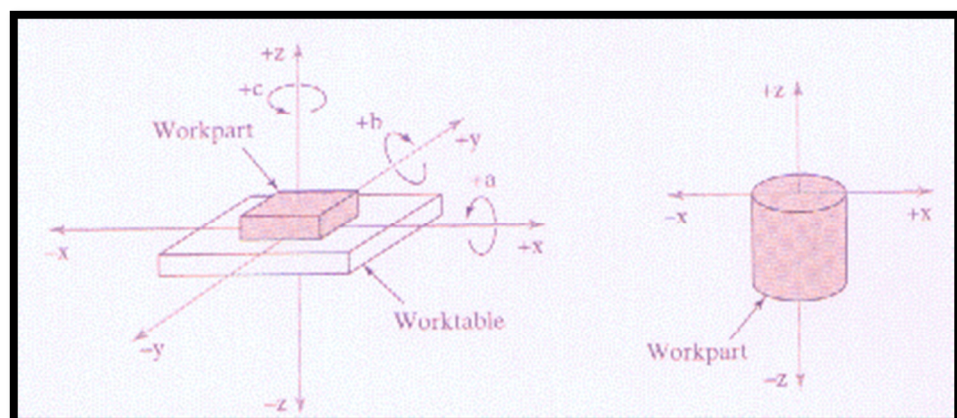


Figure 19 Coordinate systems used in NC: (a) for flat and prismatic work and (b) for rotational work. (On most turning machines, the z-axis is horizontal rather than vertical as shown (Groover 2001, 134).

The programmer must figure out where the origin of the coordinate system should be located. That would be called the zero point of the programming. It always depends on the shape of the material and the type of the process. In most turning operation the origin of the coordinate or the zero point is selected in the right side of the part touching the plane. In milling it could be anywhere and multiple zero positions depending on the sequence of tooling operation. If the part is symmetrical then the programmer should keep in mind to put the zero point right in the middle of the symmetry. (Groover 2001, 124-125.)

3.2.4 Motion Control System

Some NC processes are performed at a certain point of the work part such as drilling and spot welding. Others are done while tool is moving. If the work head or tool is moving it might require directions to follow to perform the procedure. This direction could be a straight line, circular path or it might include a curve. These different kinds of movements are accomplished by the motion control system.

Motion control systems in NC could be divided into two parts: (1) point to point and (2) continuous path. Point to point system which is also known as positioning system moves the worktable to a programmed location without regard for the path taken to get to that location. (Groover 2001, 125.) When this move is completed some processing operation is done by the tool at that location. So, the program consists of a series of point locations at which operations are performed.

In continuous path systems the tool can move along the worktable to perform an operation. So, basically it's an operation which consist movement of two different axes in the same time. This provides control of the tool trajectory relative to the work part. Generating angular surfaces, two dimensional curves or three dimensional contours in the work part are enabled in this mode. This control mode is necessary in many milling and turning operation.

3.3 Programming for NC part

CNC part programming consists of planning and documenting the sequence of processing step by step which has to be performed in the NC machine. The programmer must have a sound knowledge of part programming along with geometry, trigonometry and other basics. The documentation portion of part programming involves the input medium used to transmit the program of instruction to the NC machine control unit (MCU) (Groover 2001, 145). The programming can be done using three methods: manually, conversationally and using CAD/CAM software. Manual programming is very common in manufacturing simple parts. This method allows the programmer to write the program line by line. To be a good manual programmer one has to keep good knowledge of the control, the machine, the tooling, mathematics and the actual codes used in the

program. The programmer must decide first which process or processes he is going to operate and in which sequence. Then he has to figure out the right tools for those operations and the other parameters. Then another important part to define is the feed rates and spindle speed values which frequently changes depending on the operations and materials used in the process. Manual programs are usually written on the control panel of the CNC directly if they are small enough otherwise programmers write them into a notepad format and then upload to the corresponding pc with the CNC. These computers have a CAM program which interprets and uploads the program to the CNC and also shows if there is any programming mistake.

As the complexity of the part increases the total time to manually program a part requires a lot of extra time. And then there could be missing dimensions which also requires complex math such as geometry and trigonometry. In this kind of situation programming a part through CAD/CAM software becomes essential. Because, even an experienced and proficient programmer can spend a big amount of time figuring these problems out and change the calculations according to the coordinates. This method of programming reduces critical calculations for the programmer to perform thus saves time. The programmer has to design the part into the CAD software first and then transfer the data into the CAM file. In CAM or computer aided manufacturing software the programmer has to decide the sequence of the processes and select the right tools and feed rates according to operation. He also has to keep in mind other parameters such as options for coolant, cutting depths etc. once all these steps are done the CAM program shows a simulation of the actual process and also can provide the G codes for it. In the modern machining era programming through CAD/CAM method would be ideal. However, it is good to know that there could be unintentional mistakes might happen by entering incorrect data or having wrong drawings which could be a deal breaker. Sometimes, programs provided by CAM software intends to take a longer way of machining thus it might take a lot of extra time then required for the actual machining. In mass production cells engineers cannot afford to lose any production time. Availability of CAD/CAM software could also be a problem in many production facilities.

Conversational programming method is not that common in most of the production units as it only refers to the special capability of some controls. This kind of programming usually takes place in the production floor so different kind distraction can cause while the programming which results into many errors and production time loss.

3.4 Turning Operations

Turning in CNC is a type of operation where a single point tool is used to remove the material from a rotating cylindrical work piece. The main purpose of turning is to create a circular shape. Usually the work piece rotates in a spindle with high speed and a tool is fed into it radially or axially or both ways simultaneously to achieve the required surface (Schneider Jr, G, 2010). Early turning operations involve gear driven manual lathes. But,

in modern days all turning machines are computerized numerical controlled. With the respect to turret numbers turning centers can be divided into two categories: (a) single turret turning machines which have two axes; vertical and horizontal and (b) double turret turning machines with four axes.

3.4.1 Operating conditions

There are three significant metal cutting variables which affects the operating conditions in turning operations. They are: (a) material removal rate, (b) tool life and (c) surface finish. If these variables can be balanced perfectly then it's easier to achieve the maximum production rate with minimum cost and with the best relative surface finish. The success of many machining operation depends on the cutting tool. Sometimes few set-ups have to done in order to make sure the tool reaches the area to be machined. These set-ups may involve extending the tool length. Nose radius of the tool affects the surface roughness of the machined material. A large nose radius contributes a smooth surface at lower feed rates and higher cutting speeds. On the other hand, cutting with a small nose radius prolongs tool life (A comprehensive study of operational condition, 2010). Balancing the piece in the chuck should be considered very important. When material is removed from the work piece surface, there is a possibility that the balance may change. If it happens during the early rough cutting cycles, then it is balanced again during the finishing of the rough cut (Schneider Jr, G, 2010).

3.4.2 Cutting Speed

Since, turning is a combination of linear and rotational movements; the cutting speed is defined as the rotational distance traveled in one minute by a point on the part surface (Chapman 2004, 380). The following equation and is used to calculate the cutting speed:

- $V = \pi * D * N$
Where,
V = cutting speed.
D= work piece diameter.
N= rotation per minute

3.4.3 Depth of cut

The depth of cut defines the depth of tool cutting the edge in every cutting cycle. It determines one linear dimension of the area of the cut. For an example, to reduce the outer diameter of a cylinder by 5mm the cutting depth could be 2.5 mm which will take one cycles to remove the existing material in turning operations. It does not have any effect on the surface roughness (A comprehensive study of operational condition, 2010).

3.4.4 Feed rate

The feed rate is the linear distance that the tool travels along or across the part surface which is expressed usually in millimeters per revolution units. It is also expressed as a distance traveled by a single minute by the tool. It can be determined by the following formula:

Feed rate = cutting feed * spindle speed.

Where, cutting feed is the distance that cutting tool advances for every revolution of the spindle.

3.4.5 Cutting time

The cutting time can be calculated by the following formula:

$T_m = \text{distance} / \text{rate}$

Where,

Distance = length of cut, L

Rate = feed rate.

4 AUTOMATED MANUFACTURING SYSTEM

Physical products are operated in the factories by automated manufacturing system. Operations such as processing, assembly, inspection, or material handling are performed by this kind of system. And more importantly in some cases accomplishing more than one of these operations in the same time. Such operations are called automated because these operations are performed with a reduced rate of human participation compared with the existing manual manufacturing systems. No human participation can be observed in some highly automated facilities (Groover 2001, 10). Examples of automated manufacturing systems include:

- Automated machine tools for processing parts.
- A series of machining operations performed by transfer lines
- Automated assembly systems
- Manufacturing systems where industrial robots are used to perform processing or assembly operations
- Automatic material handling and storage systems to integrate manufacturing operations
- Automatic inspection systems for quality check.

Automated manufacturing systems can be divided into three kinds:

- (a) Fixed automation
- (b) Programmable automation
- (c) Flexible automation

4.1 Fixed automation

Fixed automation is a kind of system where the sequence of processing operations or assembly operations is fixed by the equipment configuration. The operations existed in the sequences existing in these operations are quite simple, involving maybe a joint or rotational motion or a simple combination of the two. It is the integration and coordination of many such procedures into one piece of equipment which makes the system complex. Some features of fixed automation rate:

- Higher production rate
- Relatively inflexible in accommodating product variation
- High initial cost for custom made engineering equipment

Products that are produced in large quantities and at high production rates only have the economic advantages for fixed automation. The high initial investment of the equipment can be spread over a very large number of unit, thus making ht unit cost attractive compared with alternative methods of production (Groover 2001, 10). Automated assembly equipments and machining transfer lines are good examples of fixed automation.

4.2 Programmable automation

Here production machineries are designed with the capability to alternate the sequence of operations to provide different product configurations. The operation sequence is controlled by a program. A program is a set of instructions coded into the machine language and interpreted by the system. Some significant features which define programmable automation include:

- High investment in general equipments
- Lower productivity than fixed automation
- Extremely suitable for batch production
- Flexible while dealing with variations and changes in product design.

Usually in industries programmable automation systems are used in low and medium volume production. The parts or products are usually made in batches. To produce each new batch of a different product, the system must be reprogrammed with the set of machine instructions that correspond to the new product (Groover 2001, 11).

4.3 Flexible automation

Flexible automation could be considered as an extension of programmable automation. A flexible automation system is capable of producing a variety of parts or products with virtually no time lost for changeovers from

one part style to the next. There is no lost production time while reprogramming the system and altering the physical setup (tooling, fixtures, machine settings). Flexible automation systems exist from the late 1960s (Groover 2001, 11.) The thing that makes flexible automation possible is that the differences between parts processed by the system are not significant. It is a case of soft variety, so that the amount of changeover required between styles is minimal. The features could be gathered as below:

- High investment cost for a custom-engineered system
- Continuous production of variable mixtures of products
- Production rate is usually medium or less
- Flexible while dealing with design variations

4.4 Reasons for automating

Companies adopt automated production technologies for a lot of good reasons. First purpose is to increase labor productivity. If a manufacturing operation is automated then production rate goes higher and also increases labor productivity. It also reduces labor cost. As machines will replace human, there will be no additional labor cost. Machines are slowly substituting human in the industries to reduce unit product cost. Another thing is to compensate the shortage of labor. In a lot of countries there are shortages of labor, to be precised efficient human power. So, automated production facility could be a major solution in most of the advance countries. Then another purpose is to reduce boring regular tasks. Working environment and productivity could be improved by using automated systems for boring regular repeating jobs. Automated system can improve safety for employees. By automating a given operation or sequence and transferring the worker from active participation in the process to a supervisory role, the work is made safer. The safety and physical well-being of the workers is a very important part to monitor for every employer. And also we get improved product quality. Automated system not only increases the production rate but also perform the process with greater precision and perfection. It has been seen in quality control greatly that automated systems are much reliable and safe. Reduction of fraction defect rate is one of the important benefits of automation. Reducing the whole process time could be another major reason for automating. Automated systems help to reduce the process time elapsed in manufacturing delivery and customer order. It is a great advantage for future orders thus business expansion. There are processes that are impossible to run to do manually. There are some operations which cannot be performed without help of machines. These processes have requirements for precision, miniaturization, or complexity of geometry that cannot be achieved manually. Example might be integrated circuit fabrication operations, rapid prototyping processes based on computer graphics models, and the machining of complex, mathematically defined surfaces using computer numerical control. That's where the need of automation comes again. And last of all to avoid high cost for not automating. There are significant advantages of automating a manufacturing facility. Such as improved quality, higher sales, better labor relations and better company image.

5 CELL DESCRIPTION

The production cell used for this demonstration is the laser welding cell. The Fanuc Robot Arc Mate 100i and the Puma 6-HS are located inside this cell. The Fanuc robot arm is a welding robot that can be used to lift small objects while the NC lathe machine is used for turning operation purposes. Both of these machines are integrated to work together. A working table was installed inside where the work pieces were placed during the demonstration.

5.1 Fanuc robot arc mate 100i

Fanuc Robot Arc Mate 100i model is a six-axis articulated type industrial robot with an extremely large work envelope that can be used to perform complicated tasks. The Fanuc Arc Mate 100i is an electric servo-driven robot designed for precise, high-speed welding and cutting (Fanuc ArcMate 100i Robot, 2011). It is a welding robot that has a payload of 6 kg.

5.1.1 Axis

Arc Mate 100i has six axes that make the robot perform tasks a lot easier. The manufacturer of Fanuc has their own way of naming these axes. They are named by the letter “J”, which is short for joint, followed by a number that corresponds to the number of its axis (for example J1). Each of these axes has their own purpose and function. These axes are described below.

- J1 (axis 1) Joint 1 is located at the base of the robot that swings the robot from side to side. It can rotate the robot 330°.
- J2 (axis 2) Joint 2 can be rotated 210°.
- J3 (axis 3) Joint 3 can be rotated 299°.
- J4 (axis 4) Joint 4 can be rotated 380°.
- J5 (axis 5) Joint 5 can be rotated 280°.
- J6 (axis 6) Joint 6 can be rotated 640°.

5.1.2 Work envelope

Because of the limitation of the axes, the work envelope of the Arc Mate 100i can be determined. Figure 20 is the side view orientation and Figure 21 is the top view orientation of the work envelope of Fanuc Arc Mate 100i.

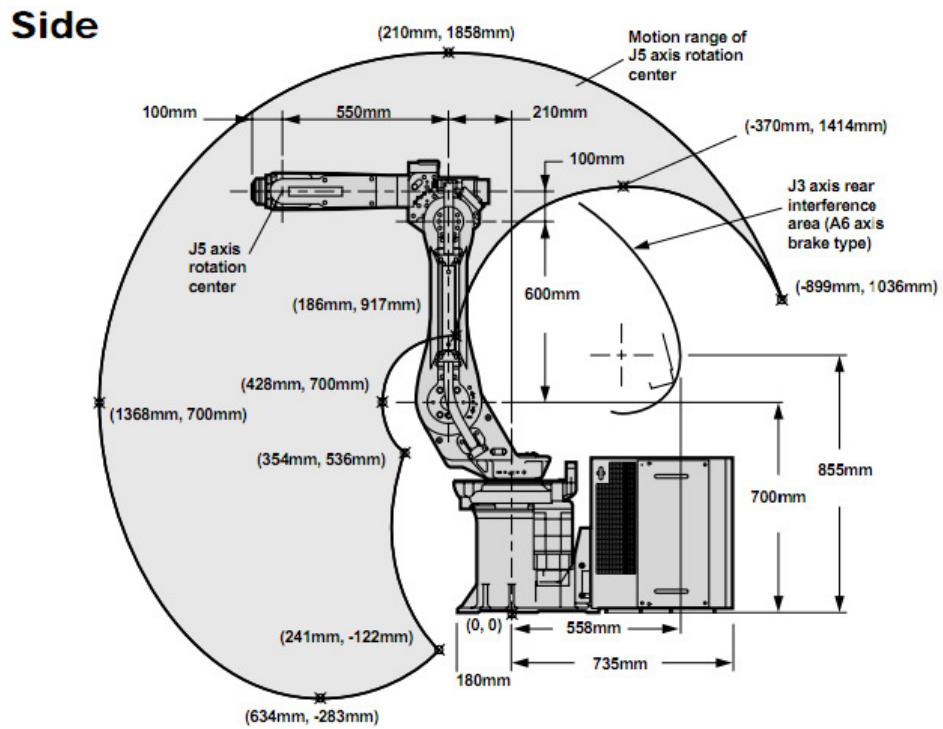


Figure 20 Side view (Fanuc Robotics, 1999)

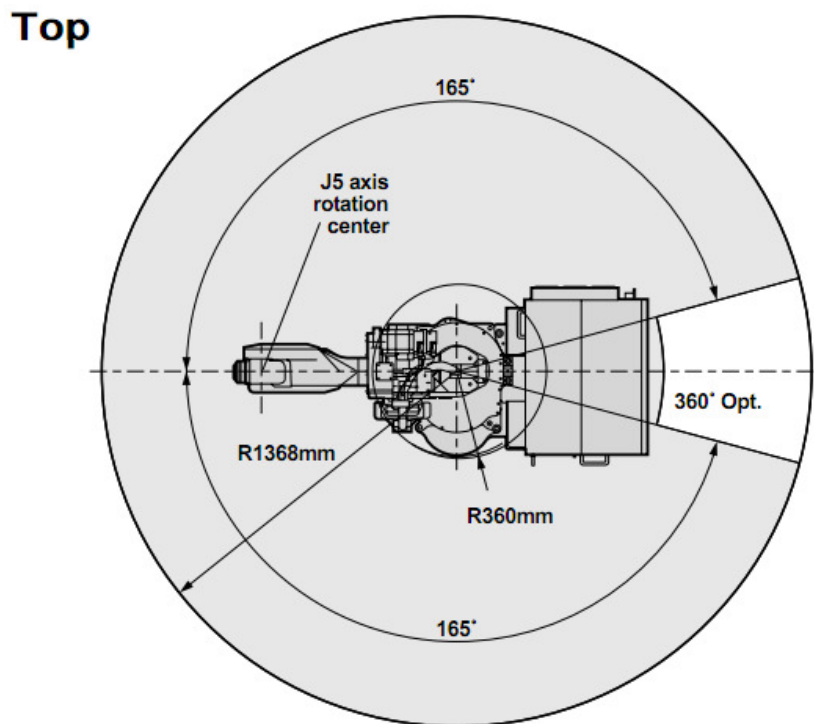


Figure 21 Top view (Fanuc Robotics, 1999)

5.1.3 Configuration

All of the joints of the robot arm can be twisted or rotated more than 180° which means there are many ways to reach a certain point. Sometimes

this could create a problem when it comes to the operation of the robot. Joints must be in the right position in order for the robot to go to a certain point.

In the case of J4, when it is rotated 180°, it flips the forearm making the rest of the robot arm and joints to be in the different position. When the forearm is flipped, the movement of the robot becomes more limited which could result in a smaller work envelope when compared to a non-flipped J4. Due to this, Fanuc has its own configuration that lets you easily monitor the position of the joints. They have used letters to represent the orientation of the joints of the robot. Here are the letters used to identify its orientation:

N/F – N stands for Non-flipped position and F stands for Flipped position of J4

U/D – U stands for Up position and D stands for Down position of J3 which the elbow of the robot.

T/B – T stands for front position and B stands for Back position of the torso of the robot. J1 is responsible for this configuration.

Flipping the forearm of the robot could result in problems. It is best to keep J4 in N position if possible to avoid conflicts when running the program.

Figure 22 shows where the configuration of the robot can be seen after pressing the “position” button from the teach pendant.

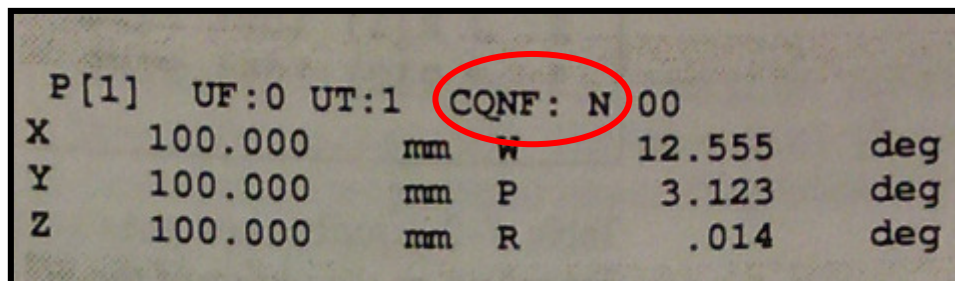


Figure 22 Location where the configuration can be seen (Fanuc Robot Series 1996, P7-17)

5.2 Puma 6-HS

This machine is a CNC lathe with the Rapid No-lift index turret. It is designed so that chip disposals and handling the work piece and tools can be performed easily. The rapid no-lift turret system precludes the danger of chips being jammed behind the disk. Cartridge type superb precision spindle eliminates vibrations and assures superior surface finish. Powerful AC spindle drive motor is service-free since troublesome motor brushes are no longer required, which provides heavy duty machining. Separated transmission gear box minimizes vibrations and heat influences transferred to the main spindle. Center work with tail stock and bar work with hydraulic chuck or collect are flexible. Chip disposal and handling the work piece

and tools have become more convenient owing to 45° slant bed construction which is a unique feature in this model (Puma lathe instruction manual, 2-1).

6 INTEGRATING FANUC ARC MATE 100I TO PUMA 6 – HS

Knowing the principle behind the robot arm and the NC machine is an important matter when it comes to integrating both of them. In this case, the robot arm was made by one company specialized in robotic automation called FANUC and the NC was made by another company called Daewoo. The programming language of these two machines was made the FANUC. Even though both programming languages were made by one company, the robot arm and the NC machine have different programming language.

When it comes to integrating two or more machines, they do not have to have the same programming languages. People who are specialized in the field of mechatronics could easily link two separate machines by just modifying their programs and system.

In 2007, there was a project linking the robot and the NC machine together to work simultaneously. During that project, one company, Numerical Controls Fin OY, successfully integrated Fanuc Arc Mate 100i and Puma 6-HS together. One cable was installed from the robot arm leading to the NC machine and modified their programs. DO[5] (Digital Output [5]) and DI[5] (Digital Input [5]) was added to the robot arm and M50 for the NC machine.

Inputs/Outputs commands give an instruction to the program to be switched on or off. Inputs receive signals coming from other devices or programs. Outputs send signals to devices, programs or in this case other machines to process the data.

DO[5] = ON – indicates that digital output number 5 gives a signal to the NC machine to start running the program and cancels the M50 command of the NC machine. It also switches DI[5] OFF.

DI[5] = ON – gives information to the robot arm that the NC machine stops running its program and will not continue running it until the robot arm sends back a DO[5] = ON signal to it.

M50 – is an M-code of the Puma 6-HS NC machine that causes the NC machine to temporarily hold the running program. It also switch DI[5] of the robot ON. M50 can only be cancelled by turning DO[5] of the robot arm ON.

The two figures below illustrate how the signals are transmitted with one cable attached to both machines:

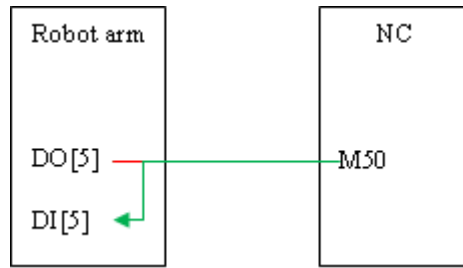


Figure 23 Illustrates how the signal is transmitted when M50 was read from the NC.

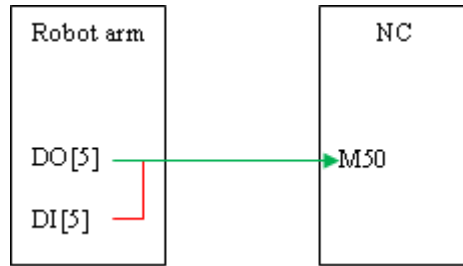


Figure 24 Illustrates how the signal is transmitted when DO[5] is ON.

One important thing regarding these commands is that when DO[5] is switched ON, M50 sends a signal to the robot arm but cannot switch DI[5] to ON. DO[5] and DI[5] cannot be switched on at the same time although they can be switched off at the same time. It is best to put a command DO[5] = OFF to switch off DO[5] before the NC machine sends the M50 command that switches DI[5] ON.

6.1 Wait Command

Right after the robot arm executes the DO[5] = ON command line, it will not stop reading the next line. The robot arm will continue reading lines unless you put a wait command that will temporarily halt the robot from reading command lines until the robot satisfies the wait command. There are wide varieties of wait commands to choose from. It is best to choose a statement wait command rather than choosing a timed wait command. Using a statement wait command allows the robot to read lines safely knowing that everything is where they should be. Below are examples of wait commands:

- WAIT 3 SECS - a timed wait command
- WAIT DI[5] = ON - a statement wait command

7 PROGRAMMING THE REQUIRED PART

Programs are used to communicate with robots and machines. The efficiency of the work done by the robot is based on how well the program was written. Unnecessary movements should be avoided to make the pro-

duction efficient and safe. It is best to keep a program simple and short in order to minimize errors.

An independent program can be called inside a program. In this case, the called independent program is now called a sub program while the program that calls the sub program is now called main program. The main program is written where it controls the flow of the program depending on the use of the machine.

In this thesis, we are to machine 6 work pieces as shown in Figure 25. They are to be fed to the PUMA 6-HS NC machine using the FANUC ARC MATE 100i Robot Arm. In order for us to do this and make the process automated, we made one main program for the robot arm and one main program for the NC machine. Both of these programs run together simultaneously.

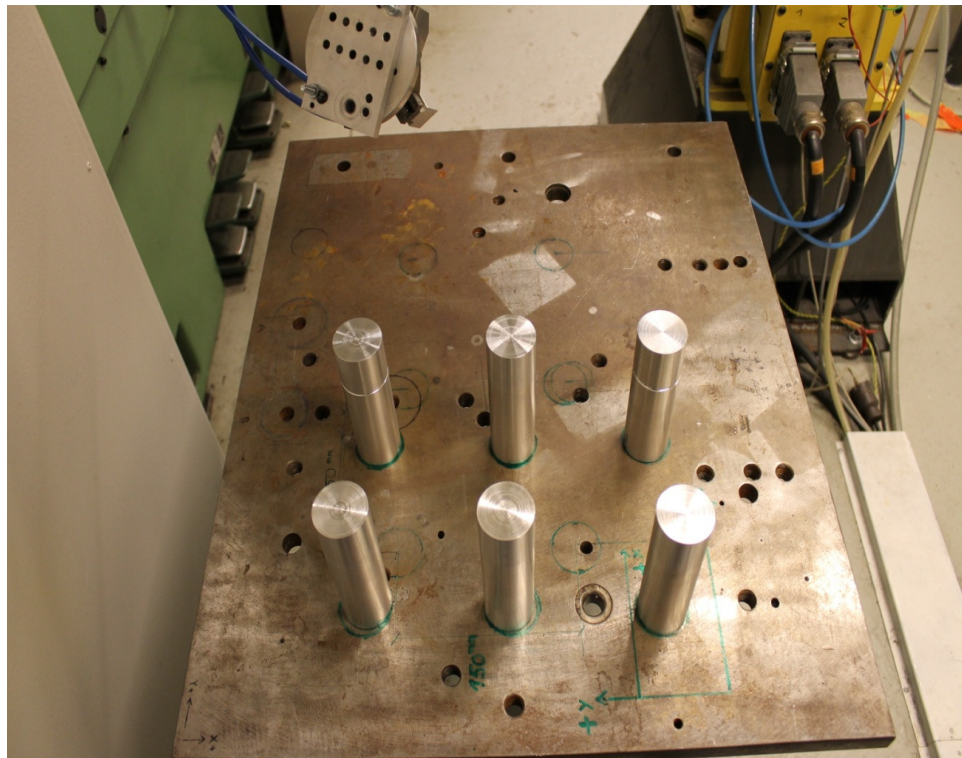


Figure 25 6 Aluminum round bars to be machined

The programs used to manufacture 6 work pieces using the FANUC ARC MATE 100i Robot Arm and the PUMA 6-HS NC lathe machine are discussed below:

7.1 FANUC ARC MATE 100i Robot Arm

The robot arm's main program is consists of eight sub programs and commands line that enables the robot to perform its tasks. Our programs are structured in a way that the robot arm will take one work piece from the table and put it in to the NC machine using 100% of its working speed. After the NC machine has machined the work piece, the robot arm will

take the work piece back to its original position on the table. These tasks will be done over and over again until the robot arm have placed the sixth work piece back on the table. Figure 26 shows the positions of the points during the operation. Refer to appendix 1 for the locations and orientations of the absolute points.

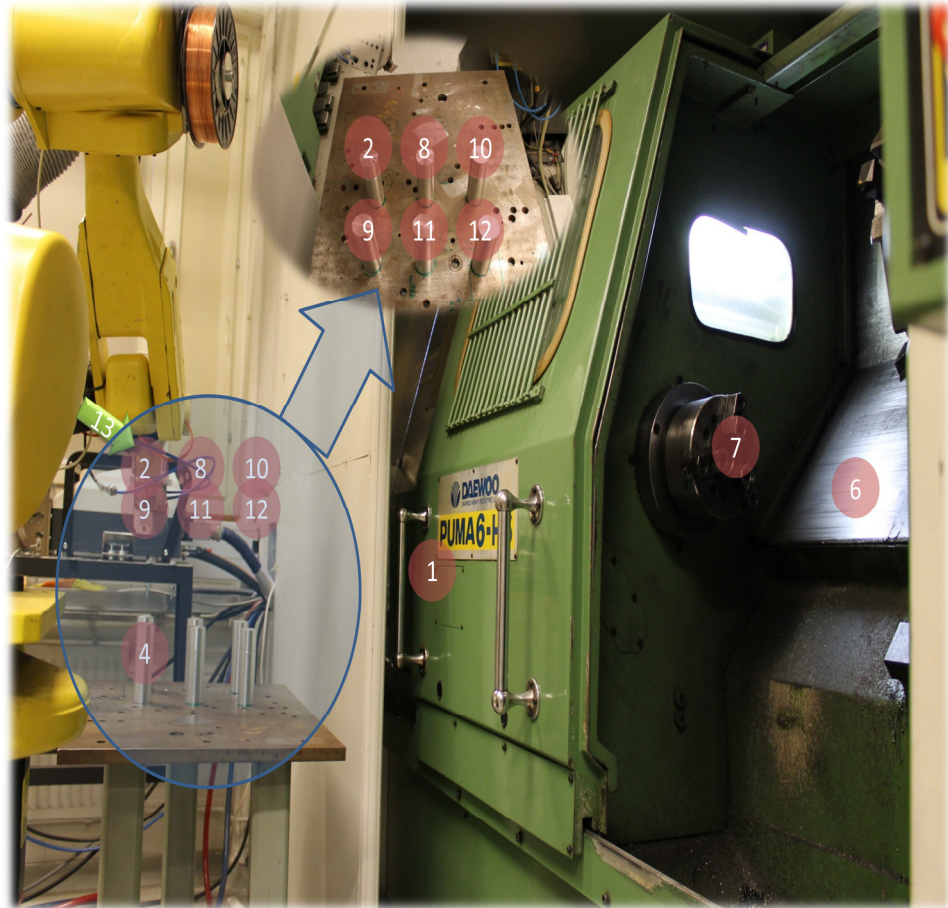


Figure 26 Location of the points during the demonstration

Sub Programs

OPEN

Name of the program

1. RO[1] = OFF

It switches Robot Output 1 off.

2. Wait 1 sec

The robot arm waits for 1 second before reading the next command line. This gives the robot enough time to switch RO[1] off even on higher speed setting.

3. RO[2] = ON

It switches Robot Output 2 on.

4. Wait 1 sec

The robot arm waits for 1 second before reading the next command line. This gives the robot enough time to switch RO[2] on even on higher speed setting.

END

CLOSE

Name of the program

1. RO[2] = OFF

It switches Robot Output 2 off.

2. Wait 1 sec

The robot arm waits for 1 second before reading the next command line. This gives the robot enough time to open the gripper even on a higher speed setting.

3. RO[1] = ON

It switches Robot Output 1 on.

4. Wait 1 sec

The robot arm waits for 1 second before reading the next command line. This gives the robot enough time to grab the piece properly even on higher speed setting.

END

P1

Name of the program

1. L P[2] 250mm/s CNT 50

It moves the robot arm linear to point 2 with a speed of 250mm/s and terminates continuously by 50.

END

P2

Name of the program

1. L P[8] 250mm/s CNT 50

It moves the robot arm linear to point 8 with a speed of 250mm/s and terminates continuously by 50.

END

P3

Name of the program

1. L P[10] 250mm/s CNT 50

It moves the robot arm linear to point 10 with a speed of 250mm/s and terminates continuously by 50

END

P4

Name of the program

1. L P[9] 250mm/s CNT 50

It moves the robot arm linear to point 9 with a speed of 250mm/s and terminates continuously by 50

END

P5

Name of the program

1. L P[11] 250mm/s CNT 50

It moves the robot arm linear to point 11 with a speed of 250mm/s and terminates continuously by 50

END

P6

Name of the program

1. L P[12] 250mm/s CNT 50

It moves the robot arm linear to point 12 with a speed of 250mm/s and terminates continuously by 50

END

Main Program

FAST

Name of the program

1. DO5 = OFF

This command switches DO[5] OFF to receiving signals from the NC machine.

2. J P[2] 10% CNT 50

It moves the robot arm with joint movements to point 2 with 10% of the robot's maximum speed and terminates continuously by 50. Figure 27 shows the robot arm as it executes this command.

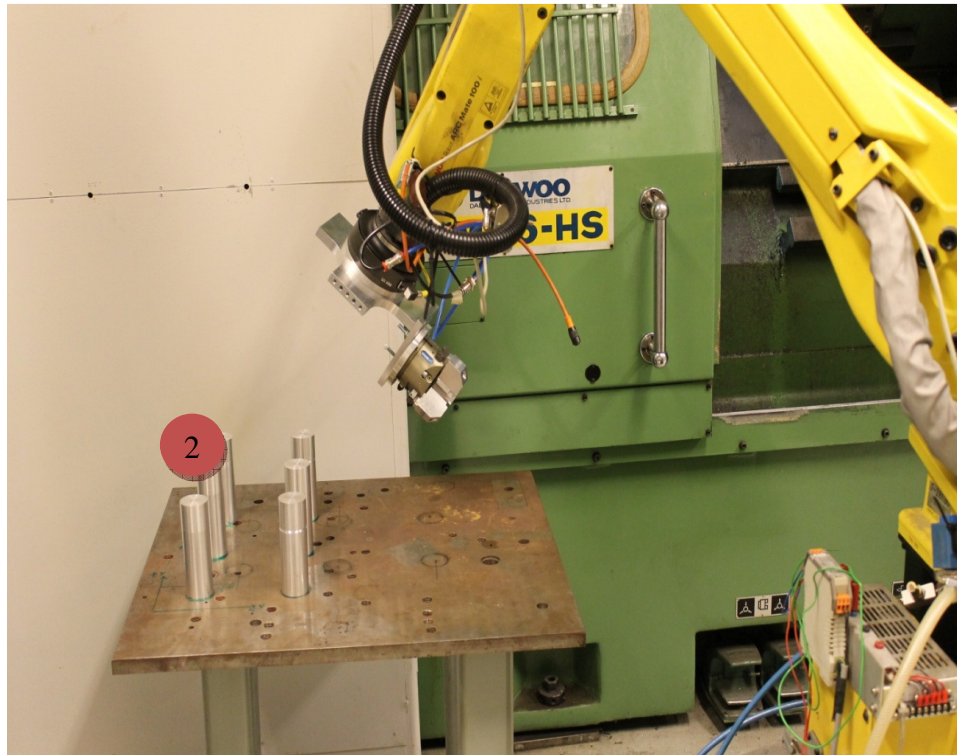


Figure 27 Robot arm as it approaches point 2

3. R[1] = 0

It sets register part number 1 to 0.

4. R[2] = 0

It sets register part number 2 to 0.

5. R[3] = 0

- It sets register part number 3 to 0.*
6. DO5 = ON
It switches DO[5] ON to send a signal to the NC machine. The NC machine starts reading the program.
 7. WAIT 3 sec
The robot arm waits for 3 seconds before reading the next line. This gives enough time for the robot arm to send the signal to the NC machine before switching DO[5] OFF again.
 8. DO5 = OFF
This command switches DO[5] OFF to receiving signals from the NC machine.
 9. WAIT DI5 = ON
The robot arm temporarily stops the program at this line until DI[5] is switched ON by the signal coming from the NC machine (M50).
 10. CALL OPEN
It calls the program OPEN to open the gripper.
 11. LBL 1
Label 1 indicates the start of the loop.
 12. R[1] = R[1] + 1
It is a simple equation for the robot to add a value of 1 to Register 1. This simple equation loops the program.
 13. R[3] = R[3] + 1
It is a simple equation for the robot to add a value of 1 to Register 3.
 14. J P[4] 2% FINE INC
It moves the robot arm with joint movements to point 4 with 2% of the robot's maximum speed and terminates fine as it reaches its destination. It also moves incrementally with respect to its previous location. Figure 28 shows the robot arm as it executes the command. Refer to appendix 2 for the location of Incremental points

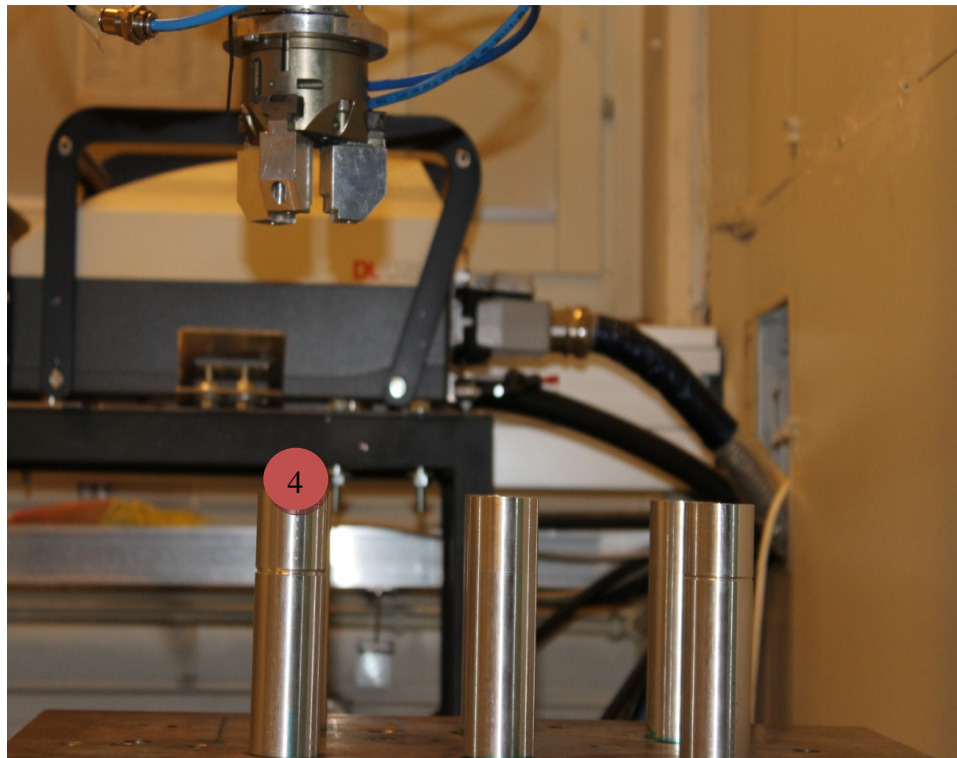


Figure 28 The robot arm as it approaches point 4

15. CALL CLOSE

It calls the program CLOSE to close the gripper and properly holds the work piece.

16. J P[13] 10mm/s CNT 50 INC

It moves the robot arm with joint movements to point 13 with 10% of the robot's maximum speed and terminates continuously by 50. It also moves incrementally with respect to its previous location. The robot arm lifts the work piece to be machined from the table.

17. L P[1] 250mm/s CNT 50

It moves the robot arm linear to point 1 with a speed of 250mm/s and terminates continuously by 50.

18. L P[6] 75mm/s FINE

It moves the robot arm linear to point 6 with a speed of 75mm/s and terminates fine as it reaches its destination. The robot arm moves slowly as it enters the NC machine to prevent any kind of unexpected movements.

19. L P[7] 10mm/s FINE

It moves the robot arm linear to point 7 with a speed of 10mm/s and terminates fine as it reaches its destination. The robot arm moves in a much slower speed to carefully place the work piece inside the chuck of the NC machine.

20. DO5 = ON

It switches DO[5] ON to send a signal to the NC machine. The NC machine starts reading the program to clamp the work piece inserted by the robot arm in to the chuck.

21. WAIT 3 sec

The robot arm waits for 3 seconds before reading the next line. This gives enough time for the robot arm to send the signal to the NC machine before switching DO[5] OFF again.

22. DO5 = OFF

This command switches DO[5] OFF to receiving signals from the NC machine.

23. WAIT DI5 = ON

The robot arm temporarily stops the program at this line until DI[5] is switched ON by the signal coming from the NC machine (M50).

24. CALL OPEN

After the NC machine holds the work piece by its chuck, the robot arm calls the program OPEN to open the gripper and releases the work piece.

25. L P[6] 75mm/s FINE

It moves the robot arm linear to point 6 with a speed of 75mm/s and terminates fine as it reaches its destination. It carefully moves the robot to point 6 to avoid crashing to the tool of the NC machine.

26. L P[1] 250mm/s CNT 50

It moves the robot arm linear to point 1 with a speed of 250mm/s and terminates continuously by 50. The robot arm moves to a safe position, point 1, so the robot arm is not inside of the NC machine when the machining of the work piece begins.

27. DO5 = ON

It switches DO[5] ON to send a signal to the NC machine. The NC machine starts reading the program to machine the work piece.

28. WAIT 3 sec

The robot arm waits for 3 seconds before reading the next line. This gives enough time for the robot arm to send the signal to the NC machine before switching DO[5] OFF again.

29. WAIT DI5 = ON

The robot arm temporarily stops the program at this line until DI[5] is switched ON by the signal coming from the NC machine (M50).

30. DO5 = OFF

This command switches DO[5] OFF to receiving signals from the NC machine.

31. L P[6] 75mm/s FINE

It moves the robot arm linear to point 6 with a speed of 75mm/s and terminates fine as it reaches its destination. The robot arm moves slowly as it enters the NC machine to prevent any kind of unexpected movements.

32. L P[7] 10mm/s FINE

It moves the robot arm linear to point 7 with a speed of 10mm/s and terminates fine as it reaches its destination. The

robot arm moves in a much slower speed to carefully place the work piece inside the chuck of the NC machine.

33. CALL CLOSE

It calls the program CLOSE to close the gripper and properly holds the work piece.

34. DO5 = ON

It switches DO[5] ON to send a signal to the NC machine. The NC machine starts reading the program to unclamp the work piece from the chuck.

35. WAIT 3 sec

The robot arm waits for 3 seconds before reading the next line. This gives enough time for the robot arm to send the signal to the NC machine before switching DO[5] OFF again.

36. WAIT DI5 = ON

The robot arm temporarily stops the program at this line until DI[5] is switched ON by the signal coming from the NC machine (M50).

37. DO5 = OFF

This command switches DO[5] OFF to receiving signals from the NC machine.

38. L P[6] 75mm/s FINE

It moves the robot arm linear to point 6 with a speed of 75mm/s and terminates fine as it reaches its destination. The robot arm carefully moves the work piece away from the chuck of the NC machine to point 6.

39. L P[1] 250mm/s CNT 50

It moves the robot arm linear to point 1 with a speed of 250mm/s and terminates continuously by 50.

40. IF R[3] = 1 CALL P1

This equation instructs the robot arm to call the program P1 when R[3] (Register 3) is equal to 1. If R[3] does not meet the condition, the robot arm will skip this command line and proceed to the next line.

41. IF R[3] = 2 CALL P2

This equation instructs the robot arm to call the program P2 when R[3] (Register 3) is equal to 2. If R[3] does not meet the condition, the robot arm will skip this command line and proceed to the next line.

42. IF R[3] = 3 CALL P3

This equation instructs the robot arm to call the program P3 when R[3] (Register 3) is equal to 3. If R[3] does not meet the condition, the robot arm will skip this command line and proceed to the next line.

43. IF R[3] = 4 CALL P4

This equation instructs the robot arm to call the program P4 when R[3] (Register 3) is equal to 4. If R[3] does not meet the condition, the robot arm will skip this command line and proceed to the next line.

44. IF R[3] = 5 CALL P5

This equation instructs the robot arm to call the program P5 when R[3] (Register 3) is equal to 5. If R[3] does not meet the condition, the robot arm will skip this command line and proceed to the next line.

45. IF R[3] = 6 CALL P6

This equation instructs the robot arm to call the program P6 when R[3] (Register 3) is equal to 6. If R[3] does not meet the condition, the robot arm will skip this command line and proceed to the next line.

46. J P[4] 2% FINE INC

It moves the robot arm with joint movements to point 4 with 2% of the program's running speed and terminates fine as it reaches its destination. It also moves incrementally with respect to its previous location. The robot arm carefully places the work piece back to its original location.

47. CALL OPEN

The robot arm calls the program OPEN to open the gripper and releases the work piece.

48. J P[13] 10% CNT 50 INC

It moves the robot arm with joint movements to point 13 with 10% of the robot's maximum speed and terminates continuously by 50. It also moves incrementally with respect to its previous location.

49. IF R[1] < 3 JMP LBL 1

This condition instructs the robot arm to jump to label 1 if R[1] is less than 3. This condition allows the robot arm to repeat its tasks starting from label 1 so it can place new work piece in to the NC machine.

50. R[1] = 0

It resets the value of R[1] to 0

51. R[2] = R[2] + 1

It is a simple equation for the robot to add a value of 1 to Register 2. This simple equation loops the program.

52. J P[9] 10% CNT 50

It moves the robot arm with joint movements to point 9 with 10% of the robot's maximum speed and terminates continuously by 50.

53. IF R[2] < 2 JMP LBL 1

This condition instructs the robot arm to jump to label 1 if R[2] is less than 2. After placing 3 work pieces in the NC machine, this condition allows the robot to repeat its tasks again starting from label 1 to place the other 3 work pieces in the NC machine.

54. J P[2] 50% CNT 50

It moves the robot arm with joint movements to point 2 with 10% of the robot's maximum speed and terminates continuously by 50. After performing all the tasks, the robot arm moves back to its home position point 2 where the user could safely switch off the robot arm.

END

This line indicates that the robot arm's program has ended.

7.2 Programming the required part for NC lathe machine

In this practical work the part has been programmed manually as the requirement for the piece was to keep it as simple as possible. The robotic arm was integrated with a Puma CNC lathe which is essentially a turning machine. So, a cylindrical piece was machined and the diameter had to be compromised according to the gripper available. There were three different process were involved: facing, roughing and finishing. Two different tools were selected to perform these operations which were tool no 3 and tool no 7 from the turret. Tool no 3 was chosen to perform the facing and the roughing and tool no 7 was for the finishing cycle. This Puma lathe has a hydraulic chuck which consists of three jaws. The diameter of the jaw had to be changed according to the diameter of the part. The diameter of the part was kept 40 mm and the diameter of the machined part was kept 35mm as the range of the pneumatic gripper was small. The turret (See figure 31) was placed as far as possible from the spindle in the beginning of the operation to ensure that the robotic arm has enough space to load the NC. The zero point of the program was set on the right side of the piece touching it by the tool in a minimum portion. And this coordinate was $(x,z) = (0.0, -134.717)$.

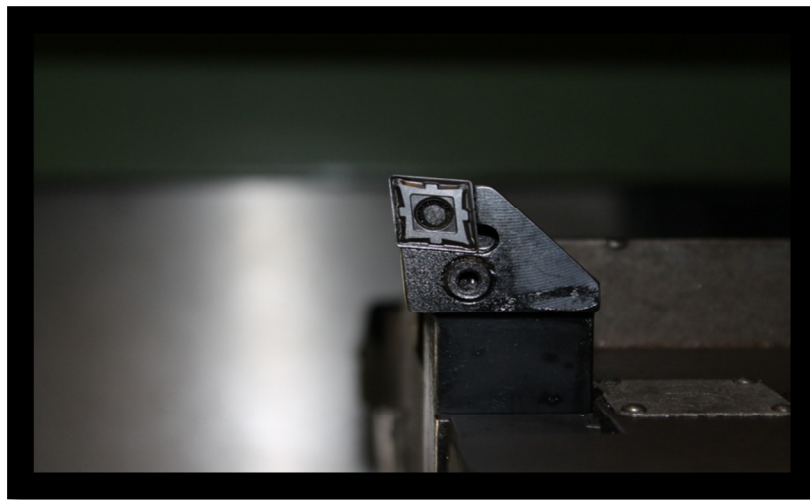


Figure 29 Tool 3

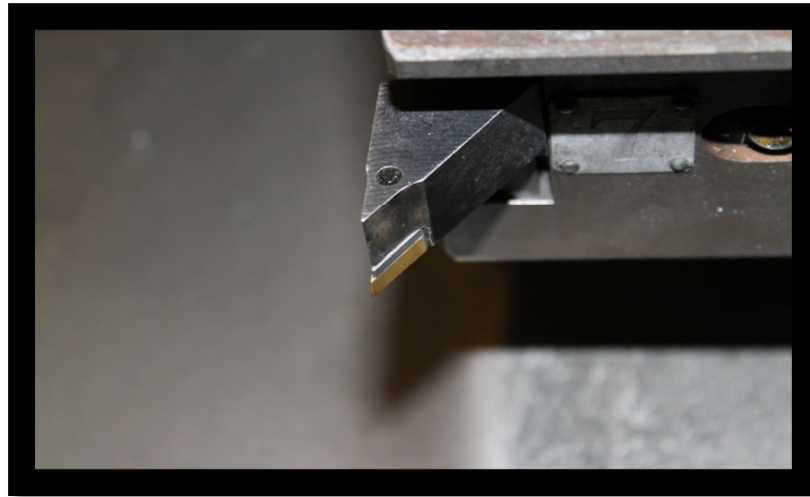


Figure 30 Tool 7



Figure 31 Turret in the Puma lathe

In the programming part a main program and sub-program was created for the operation. The main program consists of calling the subprogram 6 times as the practical work requires demonstrations of 6 cycles. The sub-program consist all the necessary machining processes. The main program is labeled as O9998 and the subprogram is labeled as O9997. This CNC machine does not have an automatic door yet so a program stop command (M00) has been used so the operator can manually open and close the door during the operation. Now, it's a semi-automatic production system but after the installation of a new automatic door, this system can be considered as a fully automated production system. The addresses used in these programs are in following table:

Table1 Addresses for the program letter in NC part programming

Code	Description
O	Program label
N	Line label
G	Preparatory function
X, Y, Z	Dimension word
F	Feed
S	Spindle speed
T	Tool function
M	Miscellaneous function
p	Program designation
S	Second miscellaneous function

Sub program

%

O9997

Name of the program

N10 M68

It closes the chuck.

N20 M50

It sends a signal to the robot arm that the NC machine is temporarily on hold so the robot arm may resume running again. NC machine will not continue reading the next line until the robot arm switches DO[5] on that sends a signal to the NC machine to resume running again.

N30 M00

It temporarily stops the program so the operator may close the door of the NC machine. The NC machine will be on hold until the “start cycle” button is pressed to resume the program.

N40 G54

This command line instructs the NC machine to use the work coordinate system 1 that sets X coordinate to 0.000 and Z coordinate to -134.717.

N50 G00 X200. Z100.

It rapidly moves the tool to the tool changing position which is at coordinates X 200 and Z 100 mm.

N60 T0303

It activates the tool placed on 03 for rough cutting.

N70 G50 S3000

It sets the maximum spindle speed to 3000mm/s.

N80 G96 S400 M3

It sets the cutting speed to 400mm/s and switches the spindle on that spins clockwise.

N90 G00 X47. Z-2.

It rapidly moves the tool to X47 and Z-2 coordinates.

N100 G01 X-2. F0.1

It commands a feed linear movement to X-2 and sets the feed rate to 0.1. Instead of stopping the cutting tool to X0, stop the cutting tool to X-2 to ensure that the end of the work piece was properly faced. This command line allows the NC machine to face the piece.

N110 G00 X40. Z2.

It rapidly moves the tool to X40 and Z2 coordinates.

N120 G71 P130 Q150 U0.2 W0.2 D2000 F0.3 S400

Rough cutting cycle. The cutting cycle starts from line N130 to N150 with 0.2mm tolerance with respect to X and Z axes. The depth of cut is 2000 μ m with a feed rate of 0.3 having a cutting speed of 400mm/s.

N130 G00 X35. Z2.

It rapidly moves the tool to X35 and Z2 coordinates.

N140 G01 Z-70.

It commands a feed linear movement to Z-70.

N150 X42

It commands a feed linear movement to X42. Instead of stopping the cutting tool to X40, stop the cutting tool to X42 to ensure that the tip of the tool will not touch the work piece when it is going back to its tool changing position. Placing the cutting completely out of the work piece's diameter will avoid damaging the work piece.

N160 G00 X200. Z100. T0300

It rapidly moves the tool to the tool changing position and cancels the tool placed in 03.

N170 T0707

It activates the tool placed on 07 for finishing.

N180 G42 X40. Z2.

It sets the tool to traverse to the right of the contour looking in the feed direction.

N190 G70 P130 Q150 F0.1

Finishing cutting cycle. The NC machine reads lines N130 to N150 to perform a finishing cutting cycle with a feed rate of 0.1.

N200 G40

It cancels the cutter compensation G42.

N210 G00 X200. Z100. T0700 M5

It rapidly moves the tool to the tool changing position and cancels the tool placed in 07. It also stops the spindle from spinning.

N220 M00

It temporarily stops the program so the operator may open the door of the NC machine. The NC machine will be on hold until the "start cycle" button is pressed to resume the program.

N230 M50

It sends a signal to the robot arm that the NC machine is temporarily on hold so the robot arm may resume running again. NC machine will not continue reading the next line

until the robot arm switches DO[5] ON that sends a signal to the NC machine to resume running again.

N240 M69

It opens the chuck so the robot arm may now take the work piece.

N250 M99

It sends a signal to the NC machine that the sub program is has ended.

%

Main Program

%

O9998

Name of the program

N10 M69

It opens the chuck so the robot arm can insert the work piece. The NC machine's chuck must be closed first in order for the machine to run a program.

N20 G54

This command line instructs the NC machine to use the work coordinate system 1 that sets X coordinate to 0.000 and Z coordinate to -134.717.

N30 G00 X200. Z100.

It rapidly moves the tool to the tool changing position which is at coordinates X 200 and Z 100 mm. The NC machines tool must be placed in these coordinates to make room for the robot arm. It is wise to check first the clearance for the robot arm inside the NC machine.

N40 M00

It temporarily stops the program so the operator may open the door of the NC machine; assuming the NC machine's door is closed. The NC machine will be on hold until the "start cycle" button is pressed to resume the program.

N50 M50

It sends a signal to the robot arm that the NC machine is temporarily on hold so the robot arm may resume running again. NC machine will not continue reading the next line until the robot arm switches DO[5] ON that sends a signal to the NC machine to resume running again. The NC machine sends a signal to the robot arm that it is ready for the work piece to be inserted in the chuck.

N60 M98 P9997 M50

This command line calls the program "O9997" and reads it. After reading the program "O9997" that machines the first work piece, the NC machine sends a signal to the robot arm that it is temporarily on hold so the robot arm may resume running again.

N70 M98 P9997 M50

This command line calls the program “O9997” and reads it. After reading the program “O9997” that machines the second work piece, the NC machine sends a signal to the robot arm that it is temporarily on hold so the robot arm may resume running again.

N80 M98 P9997 M50

This command line calls the program “O9997” and reads it. After reading the program “O9997” that machines the third work piece, the NC machine sends a signal to the robot arm that it is temporarily on hold so the robot arm may resume running again.

N90 M98 P9997 M50

This command line calls the program “O9997” and reads it. After reading the program “O9997” that machines the fourth work piece, the NC machine sends a signal to the robot arm that it is temporarily on hold so the robot arm may resume running again.

N100 M98 P9997 M50

This command line calls the program “O9997” and reads it. After reading the program “O9997” that machines the fifth work piece, the NC machine sends a signal to the robot arm that it is temporarily on hold so the robot arm may resume running again.

N110 M98 P9997 M50

This command line calls the program “O9997” and reads it. After reading the program “O9997” that machines the last work piece, the NC machine sends a signal to the robot arm that it is temporarily on hold so the robot arm may resume running again.

N120 M30

This command line instructs the NC machine to end the whole program.

%

7.2.1 General precautions for the lathe

Safety goggles to be wear all the time during operation. It helps to avoid any kind of injuries in the eye. Chips and different small particles could be flying around during any operation. Safety shoes are also suggested. Because the production floors sometimes maybe wet, greasy or the surface could be rough and any slipping accident could cause a life threatening harm. The machine is not to be operated while wearing gloves as it increases the possibility of pressing wrong buttons which could cause a serious breach in the operation. Adequate light should be provided around the machine and the perimeter around the machine should be kept dried, clean and in good order. Usage of compressed air should be avoided to remove dust, chips etc. deposited in the machine, switchboard, NC control unit etc. Nothing should be placed near the machine; otherwise it becomes an obstacle during the operation. The work bench to be set on the machine must have sufficient rigidity and proper steps should be taken so that the

surface of the work bench will be slip free. Tools, work pieces or other items cannot be placed on the machine, especially on the moving parts (Puma lathe instruction manual, 1-1).

Precaution for machine operation

Indexing the turret head should be done carefully. Otherwise, tool tips may sometimes accidentally come into contact with the bed, carriage, cover, tailstock etc. Tools are not being setup in a way that they protrude more than the specified length.

Work piece chucking

Chucking the work piece is a tricky action. The work piece has to be balanced during chucking. Spindle is not to be run if the piece is unbalanced. Using standard chuck jaws is advised. The outside circumference of the jaws must be kept within the outside diameter of the chuck when the jaws are set in the chuck. Chucking cylinder pressure should be set so that the centrifugal force available when the spindle is rotating does not exceed the chucking force. Notes have to be taken (while forming the jaws) of the forming ring grip position and the shape to which jaws are to be formed. While chucking and centering a work piece, special note has to be taken of the weight of the work piece, the size and shape of the center hole and the thrust force applied to the center hole.

Completion of operation

The power supply to the machine should be turned off when operations are over. And the machine has to be cleaned according to the given procedure. Leaving the power supply on could cause a lot of electrical damages including miss functioning of the machine. Rust preventive oil should be applied to the sideways and the individual sections. Extra care has to be taken while using water-soluble coolants.

8 CONCLUSION

After running a few test programs, a final test was made to see the result of the demonstration. The outcome was excellent! The demonstration was done in two parts, each time for 3 pieces. The whole process took approximately thirty minutes.

During the practical part of this thesis, some problems were encountered that almost stopped us from reaching our goal. These problems sounded serious when we first encountered them. But as we proceed, we have found out that these problems can be solved by modifying some parts of the robot arm.

The working space for the robot arm inside the NC machine is very limited. It is best to program the NC machine to put the tool turret as far as possible from the chuck so that it gives more room for the robot arm. The end-effector of the robot arm also plays an important role to maximize the

working area inside the NC machine. The original orientation of the gripper does not give more room for the robot arm to move. To address this problem, the tip of the robot arm was modified by installing an adaptor between the robot arm and the end-effector. The new orientation of the gripper gave us enough space to place the work piece inside the chuck.

Using the original size of the gripper, the maximum dimension of the work piece that can be manufactured must not exceed 45mm. Knowing that 45mm is the maximum opening of the gripper, blanks with 40mm diameter were machined. It is not good to make a 45mm diameter work piece for it will not have any clearance between the work piece and the gripper. Having a clearance of 2.5mm, on both sides of the work piece, allows the robot arm to grip it properly.

Few changes in the cell could improve the outcome of this project dramatically. A gripper with better gripping range could give the designer more independence in designing the work part. Bigger space inside the cell would assure more free movement for the robotic arm and could increase the process speed. Also, this demonstration would be fully automated if an automatic door was installed in the NC lathe and run the whole program by just pushing two buttons (one to start the robot arm and the other one to start the NC machine).

SOURCES

A comprehensive study of operational condition for turning process. 2010. International journal on emerging technologies. Accessed: 24.11.2011.

<http://researchtrend.net/ijet/MOHAN&%20SINGLA.pdf>

Cartesian/Gantry Robot. Robotmatrix.org. Accessed: 31.10.2011.

<http://www.robotmatrix.org/CartesianGantryRobot.htm>

Chapman, W. 2004. Guide to Machining Operations. Ohio: Hanser Gardner Publication.

End effector. Copyright © 2008. Whatis. TechTarget. Accessed: 01.11.2011.

http://whatis.techtarget.com/definition/0,,sid9_gci521693,00.html

End Effectors. RobotWorx Experts in Automation. 370 W Fairground St, Marion, OH 43302. Tel: 740-383-8383. Accessed: 31.10.2011.

<http://www.robots.com/faq.php?question=end+effectors>

End-effectors. ICC Innovative Conveyor Concepts, Robots. 910 Fountain Parkway Grand Prairie, TX 75050. Tel: (866) 501-6530. Accessed: 31.10.2011.

http://www.iccrobotics.com/robot_end_effectors.html

Fanuc ArcMate 100i Robot. RobotWorx Experts in Automation. 370 W Fairground St, Marion, OH 43302. Tel: 740-383-8383. Accessed 26.10.2011.

<http://www.robots.com/fanuc.php?robot=arcmate+100i>

Fanuc Robot Series. 1996. (R-J2 Controller) ARC TOOL Setup and Operation Manual. Fanuc LTD.

Fanuc Robotics. 1999. Arc Mate 100i. FANUC Robotics North America, Inc. Accessed 27.10.2011.

http://icrrobotsolutions.com/Documents/Fanuc%20Data%20Sheets/Fanuc_ARC_Mate_100i.pdf

Groover, M. 2001. Automation, Production Systems and Computer-integrated manufacturing. London: Prentice hall.

Industrial Robot. Wikipedia the free encyclopedia. Accessed 26.10.2011.

http://en.wikipedia.org/wiki/Industrial_robot

Instruction manual, PUMA 6/6HS/8 CNC LATHE. Changwon, Korea.

Magnet type end effector. Total Tool Solutions INC. Tel: (586) 757-0444. Fax: (586) 757-7208. Accessed: 31.10.2011.

<http://www.totaltoolsolutions.com/products-services.html>

Robot Software. Wikipedia the free encyclopedia. Accessed 28.10.2011.

http://en.wikipedia.org/wiki/Robot_software#Examples_of_programming_languages_for_Industrial_Robots

Robot Types. RobotWorx Experts in Automation. 370 W Fairground St, Marion, OH 43302. Tel: 740-383-8383. Accessed 26.10.2011.

<http://www.robots.com/faq.php?question=robot+types>

SCARA vs. Cartesian Robots. RobotWorx Experts in Automation. 370 W Fairground St, Marion, OH 43302. Tel: 740-383-8383. Accessed 26.10.2011.

<http://www.robots.com/blog.php?tag=44>

Schneider Jr, G. 19.01.2010. Copyright © 2011. Turning tools and operations. American machinist. Penton media, Inc. Accessed: 22.11.2011.

<http://www.americanmachinist.com/304/Issue/Article/False/85333/Issue>

Types of Robots. Copyright © 2009. Loop Technology Limited. Tel: +44(0) 1305 257108. Accessed 25.11.2011.

<http://www.looptechnology.com/robotic-robot-types.asp>

Absolute locations and orientations of the FANUC ARC MATE 100i Robot Arm used to help manufacture six work pieces in the PUMA 6-HS NC Lathe Machine.

Positional Information	Location (mm)		Orientation (deg)	
P[1]	x:	105.046	w:	86.364
	y:	73.926	p:	6.892
	z:	211.128	r:	1.061
P[2]	x:	-1212.791	w:	92.222
	y:	410.914	p:	1.232
	z:	34.005	r:	78.355
P[6]	x:	141.497	w:	-0.618
	y:	1444.191	p:	53.283
	z:	-38.956	r:	-90.899
P[7]	x:	102.857	w:	-0.618
	y:	144.190	p:	53.283
	z:	-38.956	r:	-90.899
P[8]	x:	-1212.790	w:	92.222
	y:	533.029	p:	1.233
	z:	34.105	r:	78.355
P[9]	x:	-1074.939	w:	92.222
	y:	410.917	p:	1.233
	z:	28.413	r:	78.355
P[10]	x:	-1212.790	w:	92.222
	y:	655.144	p:	1.233
	z:	32.730	r:	78.355
P[11]	x:	-1074.939	w:	92.222
	y:	533.032	p:	1.233
	z:	28.435	r:	78.355
P[12]	x:	-1074.938	w:	92.222
	y:	655.146	p:	1.233
	z:	27.893	r:	78.355

Incremental positions of the FANUC ARC MATE 100i Robot Arm used to help manufacture six work pieces in the PUMA 6-HS NC Lathe Machine.

Positional Information	Location (mm)		Orientation (deg)	
P[4]	x:	0.000	w:	0.000
	y:	0.000	p:	0.000
	z:	-139.104	r:	0.000
P[13]	x:	0.000	w:	0.000
	y:	0.000	p:	0.000
	z:	139.104	r:	0.000

2-D drawing for the work part

