

Manufacturing Systems Technology
Prof. Shantanu Bhattacharya
Department of Mechanical Engineering and
Department of Industrial and Production Engineering
Indian Institute of Technology, Kanpur

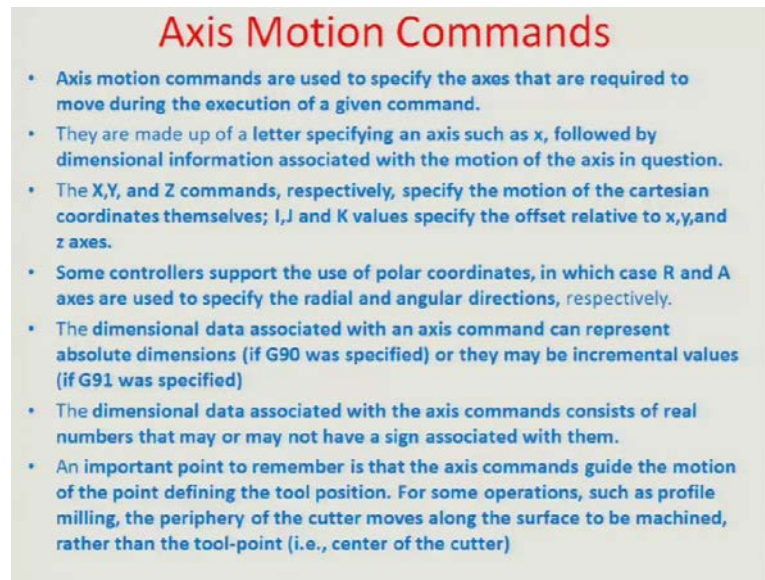
Module- 07

Lecture- 40

Hello and welcome to the Manufacturing Systems Technology module 40. Brief recap of what we did in the last lecture, we talked about the various canned cycles and there are certain G codes or preparatory codes which are available for representing the canned cycles including the basic operations like drilling, counter boring, so and so forth. We also illustrated some of the basic G codes needed for linear interpolation and rapid point to point positioning system, which otherwise is executed using you know, full speed of both the motors or both the axis.

So, today we will be seeing some other commands which are useful for example, list axis motion commands, and miscellaneous commands, and some things related to feed and speed and so discuss in great details about all these different commands. And then probably towards the end of this module, may be in the next module we will also do one program full program NC part program of a certain situation specific situation which has been provided.

(Refer Slide Time: 02:21)



Axis Motion Commands

- Axis motion commands are used to specify the axes that are required to move during the execution of a given command.
- They are made up of a letter specifying an axis such as x, followed by dimensional information associated with the motion of the axis in question.
- The X,Y, and Z commands, respectively, specify the motion of the cartesian coordinates themselves; I,J and K values specify the offset relative to x,y, and z axes.
- Some controllers support the use of polar coordinates, in which case R and A axes are used to specify the radial and angular directions, respectively.
- The dimensional data associated with an axis command can represent absolute dimensions (if G90 was specified) or they may be incremental values (if G91 was specified)
- The dimensional data associated with the axis commands consists of real numbers that may or may not have a sign associated with them.
- An important point to remember is that the axis commands guide the motion of the point defining the tool position. For some operations, such as profile milling, the periphery of the cutter moves along the surface to be machined, rather than the tool-point (i.e., center of the cutter)

So, let us look into Axis Motion Commands. So obviously, these are the set of commands which are used to specify the axis that are required to move during the execution of a given commands. So, there are two different possibilities here: one is the absolute positioning, and another is the incremental; and I think we had talked about this in quite detail manner earlier. And so the absolute in the incremental positioning is totally based on whether you basically reading with reference to the origin or you are going from one point that you have already reached to next available point in the array of motion. So, during the execution of the given command, it is basically the stepper motors which operate in the axis start to get traverse.

As you have seen earlier in almost all the n c programs such commands are given though the coordinate which is involved; for example, let say we say capital X 4 3 meaning there by that X is corresponding to a axis motion command and it majorly represents a certain X coordinate if you talking about absolute positioning with reference to the origin. So, in the next line if there is a certain other X which is there, it will automatically give you a relationship between how to move between the previous X and the next X. So, this is represented in more than two lines, the motion is represented in more than two lines such as system.

So, the letter specifying X followed by a dimensional information, is associated with the motion of the axis in question and the XYZ command respectively specify the motion of

the Cartesian coordinates themselves. And I think had illustrated in great details about the values I, J, K, earlier; were I, J, K, really are the representation of the offset that is there are from the central radius. So, that it can give you bases of the beginning position of an arc, you know which the circular interpolation can happen in either in the clock wise or counter clock wise manner. So, the I, J, and K also sort of values which specify the offset relative to the X, Y, and Z axis. So, you know some controllers are really designed for the polar system of coordinates, were you will have a radial and angular direction.

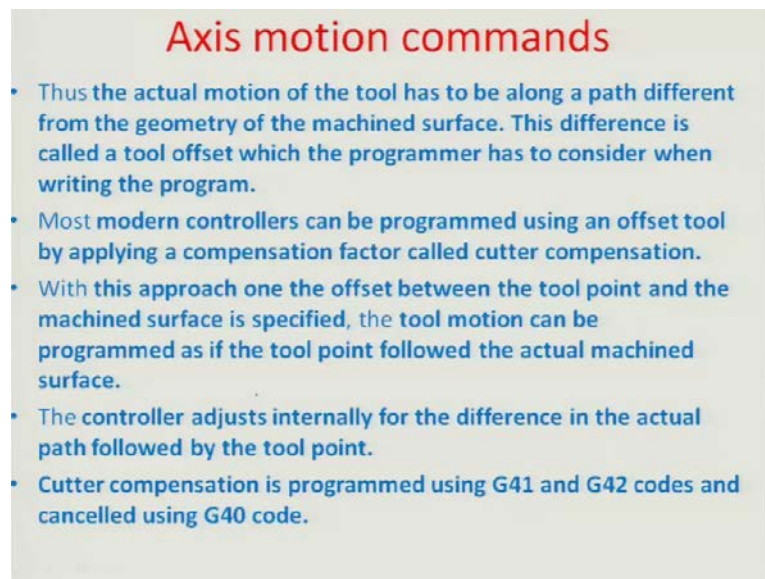
So, for that this subscript that uses R and A, indicating radially outward and inward and then angularly whatever is the direction; obviously, this same methodology has to be followed regarding the positive and the negative directions as help. Has been discussed earlier in case of clockwise or count clockwise rotation concerned with the particular axis of motion. The dimensional data associated with the axis commands of course, comes from as illustrated earlier some of the preparatory functions and this case it is typically G90 or G91 which gives you in sense of absolute of the incremental dimensioning and we have really maintained again and again that these two are completely different systems of part programming. And one is more convenient for the operator to read and just you know as a function of the or the coordinates that the drawing would have try to just mention about those coordinates and let the controller do the remaining job of calculation in between.

So, the dimensional data associated with the axis command contains mostly real numbers. Real numbers meaning you can have up to decimal places; obviously, the decimal places are represented in terms of the two subscript numbers which are associated with that. For examples if I say plus minus X 4 3, it means it is a real number which can go from 0000.000 to 9999.999. So, I think we are at length discussed about this when we were talking about the controller code or the controller specification which is given by the machine manufacture.

So, an important point to remember is that the axis commands guide the motion of the point defining the tool position, and for some operations such as let us say profile milling you know the problem happens because, now the periphery of the cutter moves along the surface that has to be machined, rather than the tool point. And the tool point in this cases sort of you know the center of the cutter. So, I would say that in such situations

when there is circular cutter being used and there is difference between the actual tool point which is more closer to the you know the machined work piece and the radial center of tool. We used ah particular preparatory function which is known as the cutter compensation, which is sometimes given is the part of the controller capability. So, if you are able to just map the tooling center in the way that moves and you have already given the data for the compensation, the cutter compensation value automatically it would describe the tool point based on the program.

(Refer Slide Time: 07:21)



Axis motion commands

- Thus the actual motion of the tool has to be along a path different from the geometry of the machined surface. This difference is called a tool offset which the programmer has to consider when writing the program.
- Most modern controllers can be programmed using an offset tool by applying a compensation factor called cutter compensation.
- With this approach one the offset between the tool point and the machined surface is specified, the tool motion can be programmed as if the tool point followed the actual machined surface.
- The controller adjusts internally for the difference in the actual path followed by the tool point.
- Cutter compensation is programmed using G41 and G42 codes and cancelled using G40 code.

So, the actual motion; obviously, would then be different, quite different then the motion of the tool center. And that is how you can actually program mostly cutter compensation. So, it is programmed using the code G41, G42 and then; obviously, it is the sort of a modal command which once programmed would remain in the system. Every time it would get executed unless you cancel it. So, the cancellation command for that cutter compensation is the code G40. So, these are the three preparatory codes or G codes which are used indicate the cutter compensation part in the program, part program.

(Refer Slide Time: 08:03)

Feed and speed commands

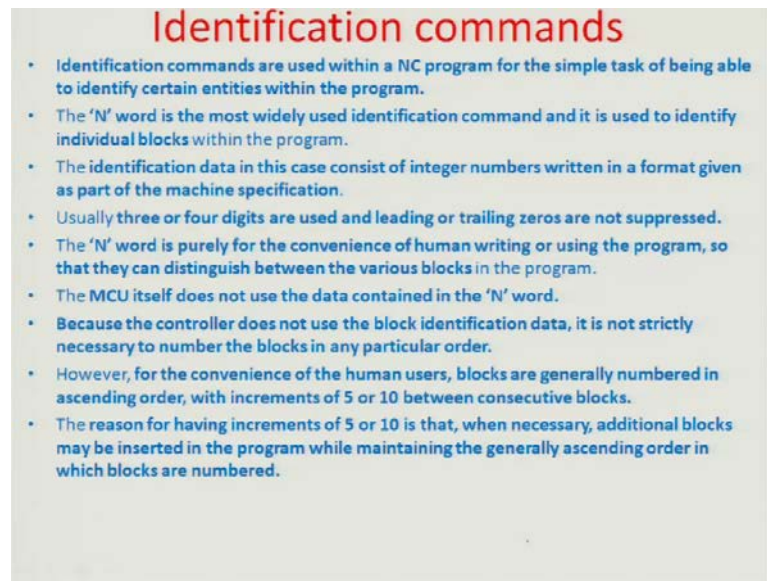
- Feed and speed commands are used to specify the feed rate and speed to use during the machining operation.
- The feed command is specified by the address script 'F' followed by the numerical value of the feed rate required; the speed command is specified by the address word 'S' followed by the required speed.
- The feed rate and the speed used during the machining are of crucial importance in determining how long it will take to make a part.
- The units for cutting conditions can be specified in a variety of ways in the NC program. The feed rate may be specified directly in units/ min. or units/rev., where the units may be in inches or millimeters.
- The preparatory functions G92-G98 are used to designate how the cutting conditions are to be specified.

So, that is all about axis motion commands; obviously, now you are concerned with how the tool moves with respect to the work piece. Once you have given from where to where it should move, it should also be able to describe what rate should be move and so therefore, feed and speed commands come into picture. As the name probably suggest you know feed and speed commands are mostly used to specify the feed rate in the speed used during the machining operation. And the feed command is specify by the address script F, you can see mostly in all the controllers specification the letter F with they with plus minus again and with also. And with numerical value like let say you can define as F plus minus 40. So, again it means that what are the number of digits up to which you can actually programmed. So it can go from 0000 to 9999, different feed values that is what the capability of the controller would them be.

So, similarly when speed command is specified in a similar manner, only thing is the letter S is used in instead of F. And the feed rate in the speed used during machining there of critical importance and that is towards how much time it will take from machining the particular part. So, I think we already know about these terms feed rates or speed, normally feed rate is specified in terms of units per minute or units per revolution; where the units may be described in different systems like the FPS system inches or the metric system millimeters so on and do forth. So, feed rate; obviously, can be either in terms of per unit time or per unit revolution of the work piece. It is very handy as per unit revolution when we talking about a turning center for example. And just need to

mention that the preparatory functions G92 to G98 are generally used to designate how the cutting conditions are to be specified. Whether it is a inch per or unit per revolution or unit per minute rate at which the feed has to be specified so and so forth. So, that is about feed and speed commands. We also have in detail mention the identification commands which are really the address for a given block this start of a given block.

(Refer Slide Time: 10:52)



Identification commands

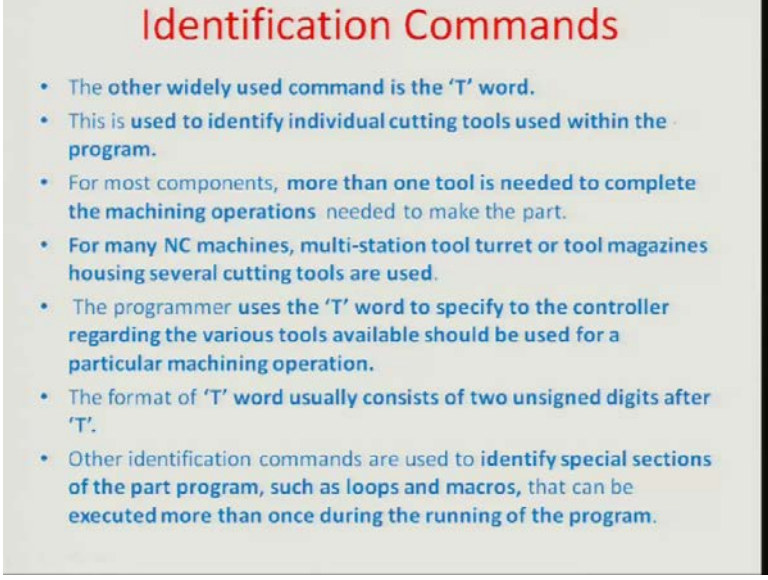
- Identification commands are used within a NC program for the simple task of being able to identify certain entities within the program.
- The 'N' word is the most widely used identification command and it is used to identify individual blocks within the program.
- The identification data in this case consist of integer numbers written in a format given as part of the machine specification.
- Usually three or four digits are used and leading or trailing zeros are not suppressed.
- The 'N' word is purely for the convenience of human writing or using the program, so that they can distinguish between the various blocks in the program.
- The MCU itself does not use the data contained in the 'N' word.
- Because the controller does not use the block identification data, it is not strictly necessary to number the blocks in any particular order.
- However, for the convenience of the human users, blocks are generally numbered in ascending order, with increments of 5 or 10 between consecutive blocks.
- The reason for having increments of 5 or 10 is that, when necessary, additional blocks may be inserted in the program while maintaining the generally ascending order in which blocks are numbered.

So, we generally address this with number N, and the N words mostly the most common identification identifier for an individual block. And although the MCU itself does not use the data contained in the M, but it is actually uses indexing tool mostly, so that you can see what comes in sequence first and comes in sequence later. So, normally I think I have illustrated this earlier, but again I want to retreat that for convenience of the human user it is a good practice that you live gaps in between when you talking about the particularly representing the N values.

For example, you can have one command block to have N05 and following block to have N10, so you are skipping 5 steps in between. And this is normally done in all the programs because if the controller has a capability of going from let us say for example, thousands steps. You always try to cover as much possible the in terms of the N numbers as so that later on any modification in the code etcetera that may be under taking can be easily implemented. So, if you have gaps between the first line and the second line in terms of the identifier numbers and those gaps can be filled on later on if you want to

modify the code in between somewhere suitably, without really that we have into write whole code again. So, this is users convenience point of view that is various N numbers are illustrated on the different command lines and that is how identifications have been done.

(Refer Slide Time: 12:21)



Identification Commands

- The other widely used command is the 'T' word.
- This is used to identify individual cutting tools used within the program.
- For most components, more than one tool is needed to complete the machining operations needed to make the part.
- For many NC machines, multi-station tool turret or tool magazines housing several cutting tools are used.
- The programmer uses the 'T' word to specify to the controller regarding the various tools available should be used for a particular machining operation.
- The format of 'T' word usually consists of two unsigned digits after 'T'.
- Other identification commands are used to identify special sections of the part program, such as loops and macros, that can be executed more than once during the running of the program.

There are several other identified identification commands like let us say this word T here, which represent the individual cutting tool with the number. So obviously, if there is a tool magazine or turret which contains many tools of tool tips. There is always tendency of tool selection to happen within the, so the first logical step for the machine into be executed in a proper manner should the tool selection in that particular case. So, here for example, T01 would identify the first tool similarly if I am saying that the tool can go up to 20, T 20. So, it means that you can go from all the way from 00 to 99, so about 100 tools that you can actually be able to designate. So, T 20 would be then identifying the capability of that the controller can handle in terms of the number of tools. So, in the same manner you have described the tool selection process, and the other identification commands are used to identify special sections of the part program such as loops and macros. That can be executed more than once during the running of the program.

So, we then come to the last available type of command which is the miscellaneous command, is normally used to control variety of machine function that are not covered

by the other commands done so for like preparatory commands, axis commands, feed speed commands, the identification commands so and so forth.

(Refer Slide Time: 14:17)

Miscellaneous Commands

- Miscellaneous commands are used to control a variety of machine functions that are not covered by the other commands.
- The address word 'M' followed by two unsigned digits is used to specify miscellaneous commands.
- Examples of functions controlled by miscellaneous commands are turning the spindle on and off, turning coolant on and off, initiating a tool change, clamping and unclamping the work-piece interrupting and restarting program execution, stopping the program and rewinding the program.

So, here as the name suggest the first letter M of the miscellaneous is basically used for executing this commands. I will just share with you list of such of commands which are available.

(Refer Slide Time: 14:18)

Miscellaneous Commands

- Generally, miscellaneous commands take effect after execution of the other commands in the block in which they are programmed.
- It is usually permissible to program more than one miscellaneous command in a given block provided they do not have conflicting effects.
- Many of the 'M' codes have been assigned standardized functions. Some 'M' codes are given in the table :

Code	Function
M00	Program stop
M01	Optional stop
M02	End of program
M03	Spindle on CW
M04	Spindle on CCW
M05	Spindle off
M06	Tool change
M07	Mist coolant on
M08	Flood coolant on
M09	Coolant off
M30	End of program—rewind

For example, some commonly used M codes could be how to stop a program M00, M01 is optional stop; 02 is the end of the program. You can also let the spindle turn clockwise

or counter clockwise based on M03 or M04. You can have a spindle off on, you can have a tool change mist coolant on, flood coolant on all these different options given by the various miscellaneous commands. So, these are really the operations which are not been covered so far, in terms of the other functions that have been described earlier. And the generally the miscellaneous commands take effect after execution of the other commands in the block. So, they are last in the row and that is how they are also programmed. If you look at the controller specification also the miscellaneous really comes towards the last of the particular command line.

So usually permissible to produce more than one miscellaneous command in a given block as long as they do not have any conflicting interest for example, you cannot say coolant off and on together. So, that does not create any sound bases as long as there are no conflicting natures of these commands you can have multiple miscellaneous command in every command block. So, that is about it. So, that is sort of bringing us to the end of this particular module when we talking about the various programs.

The only other small thing which we need to mention are the sometimes special characters, like percentage sign etcetera which are done to you know just indicate the start of the program or the stop of the program. So, generally the percentage sign common to all controller is used for indicating the first line of the program. And so these special characters are just over and above what ever has been discussed so far in terms of the G commands the F S. The axis motion commands the identification commands in the miscellaneous commands. So, they are also serving some role the special characters in terms of startup program and stop up program.

Particularly when you are talking about ending the program you sometimes write end of block you know the letters EOB for example, illustrating that the program has a stop on that particular portion. So, this gives you holistic overview of all the modalities associated with the NC program, part program. Now we will do actually a part program in the next module and try to identify in a very nice and intuitive manner how you know the machine path can be planned based on the program.

Thank you.