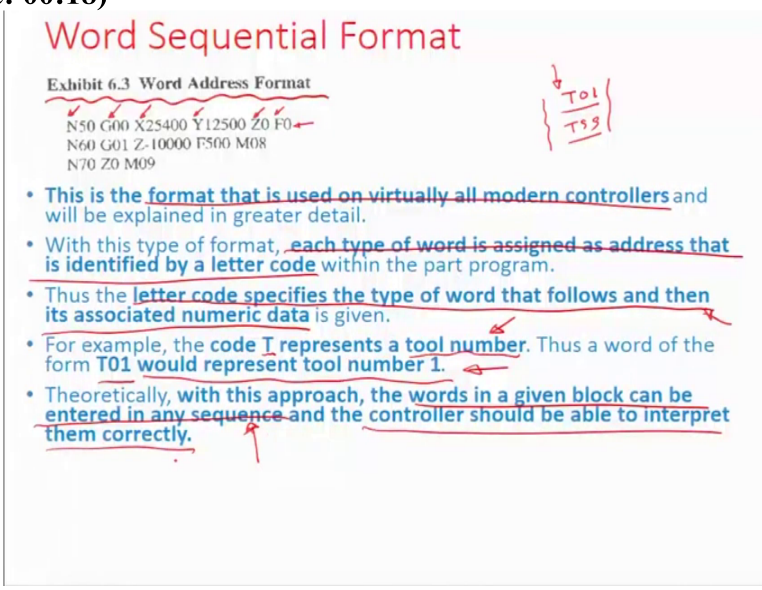
Design Practice - 2
Prof. Shantanu Bhattacharya
Department of Mechanical Engineering
Indian Institute of Technology-Kanpur

Lecture - 28
(CNC Tooling/ Rapid Tooling/ Rapid Prototyping)

Hello and welcome to this design practice to module 28 I will be talking today about the next in the sequence format which is very useful and it is most latest one of the CNC programming aspects which is about the word sequential format.

(Refer Slide Time: 00:18)



As the name means or you know shows up in the format in exhibit 6.3 there are certain words which are used now so that there is no indexing as such or a column header which would be needed to study the column the word itself gets to be a part of the code. For example N indicates the line number G indicates the proprietary code X YZ indicate their axis positions and the word F indicates feed so on so forth.

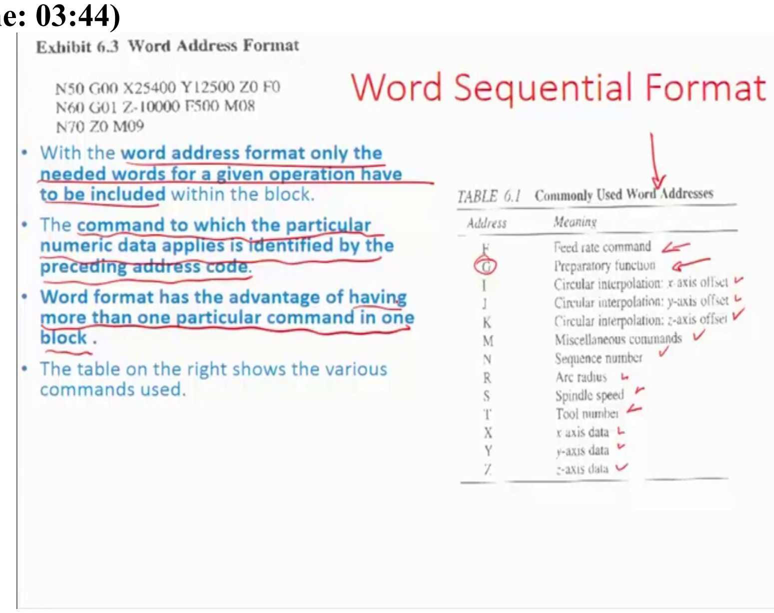
So, many words have been now put in instead of just numerals and the advantage that such a representation has is that every time we do not refer as a programmer to the column header or you know particularly for long programs this is a big issue that the column headers have to be referred again and again and it may not be on the same page okay. So, this format changes the overall level of the inconvenience of usage related to the CNC programming.

So, this is a format that is virtually used by all modern controllers each type of word is assigned as address it is identified by a letter code within this program. And the letter code specifies the type of word that follows and then it is associated with the numeric data which is again based on the controller capability for example the code T here is not mentioned in this particular exhibit but there is a called T which would represent what is the tool number.

So, typically the words are being chosen in a manner so that it can index very well the purpose that is associated with a set of numerals where this is a part of so. A word in the form T01 for example would represent tool number 1 okay. So, automatically it brings home the capability that 100 tools can be accumulated with the controller capacity which can start from T01 to T99 it automatically means that but the fact that T as a subscript is there in that particular column does not leads to a situation where the column meters are avoided and you can have data often for anybody else to enter ok easily. So, theoretically with this approach the words given in a block can be entered in any sequence ok.

Now because there are you know certain chains which are pulled up in the algorithm that is developed underneath the controller programming that is developed underneath where whatever be the particular number is really based on the word and so the word is a chain to a section which would lead to eventually the implementation of controller commands to the machinery and so the controller should be able to interpret them correctly and sequence may vary accordingly.

(Refer Slide Time: 03:44)

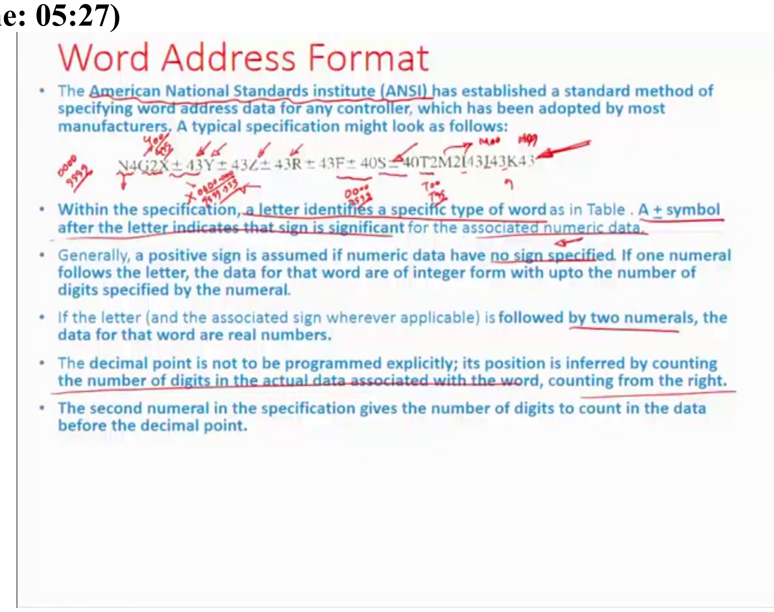


So, these are some of the commonly used words in such word address formats you have F for feed rate command G's preparatory function there are a set of G codes which exist which take care of a lot of preparatory steps associated to the machine motion axis. There is again circular interpolation command along the xyz direction is ijk the miscellaneous commands there are sequence numbers.

You can define in arc radius you could also fix the spindle speed also try to do tool selection and access data along xyz axis so on so forth so. With this word address format only the need for words for a given operation has been the additional included part of what used to happen earlier. So, obviously the command to which the particular numeric data apply is identified by the proceeding address score in terms of the word so it brings a sequence chain you know which would relate irrespective of wherever the word is placed in the sequence.

The particular part of the program of the subroutine of the program which would be able to input the correct data what advantage the; what format has a certain advantage in terms of having more than one particular command in one block. So, obviously you can use many commands.

(Refer Slide Time: 05:27)



And when we look at how internationally standardization has been developed for the word address format it is typically available at the American National Standards Institute ANSI which is defined this controller capability code which comes with all manufacturers with or you know with machines from a variety of different manufacturers to the user. So, the code mentions about the capability of a controller in terms of numbers it can handle N4 typically looks at principle line identification number varying between 0000 to 9999 that is about 10000 different numbers again G2 again indicates 100 different numbers varying between G00 G 99 xyz are having a controller capability of 0000. 000.

I think I had mentioned this why this 4 and 3 are important three places to the right of the decimal four places to the left of the decimal two all the way 9999.999 that is how much how different you know X graduations can be involved. Similarly the same goes true for y and z and even r radius. So, you have capability of the controller to handle numbers up to this extent it is about 10000 different numbers actually much more than 10000.

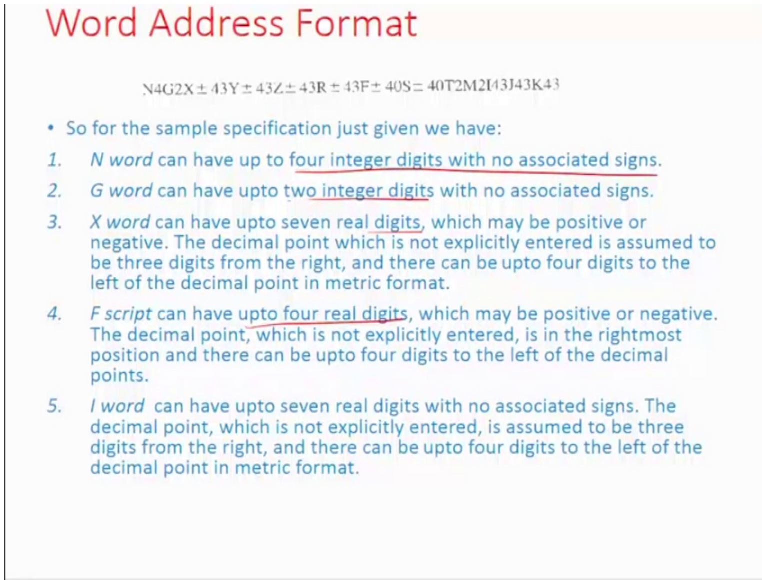
So, 10 to the power of 8 actually numbers that it was 10 to the power of 7 numbers I am sorry when we talk about the controller range you also have options for feed which is between 0000 all the way to 9999 so 10000 feed numbers you have the option of so this is a certain controller that they are defining capability wise option of speed again tool option which can define about hundred tools.

Again the controller way of understanding miscellaneous options from standpoint of 100 different miscellaneous operations M00 to M 99 and then these individual circular interpolation parameters which will again linked up in a similar manner. So, this one line of standard defines how controllers typically are defined within the international market. And within these specifications which are mentioned you know a letter identifies a specific type of word a plus minus symbol after the letter indicates that sign is significant for the Associated numeric data.

A positive sign is assumed if numeric data have no sign specified. So, these are certain basic norms that needs to be followed. So, if one numeral follows the letter the data for that word are of integer form with up to the number of digits specified by the numeral I have already shown you the controller capability. If the letter is followed by 2 numerals the data for that word are real numbers. The decimal points are not programmed it is position is inferred by counting the number of digits in the actual data so share it with the word counting from the right okay.

So, whatever is towards the right of the number indicates how much is towards the right of the decimal and similarly a situation for the left so the second numeral in the specification gives the number of digits to count in the data before the decimal point and so that is how standard Institute has developed this representation of format for every controller.

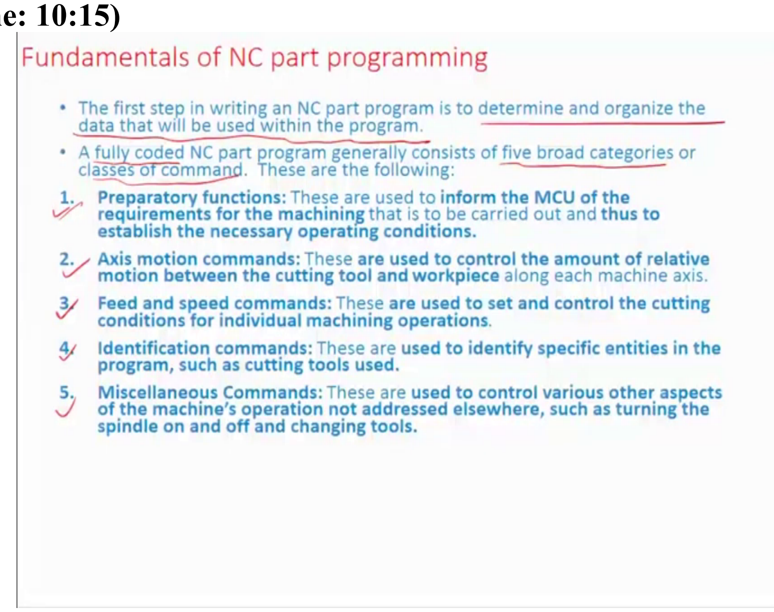
(Refer Slide Time: 09:27)



So, just relooking at it again and what can have up to 4 integer digits no Associated signs G word can have a 2 integer digits X word can have up to 7 real digits which may be positive or negative F script again can have up to 4 real digits which may be positive or negative again decimal point which is not explicitly entered is in the rightmost position. And there can be up to 4 digits to the left of the decimal point.

The I word can have up to 7 real digits with no associated signs the decimal point which is not explicitly entered is assumed to be three digits from the right. There can be up to four digits to the left of the decimal point in the matrix format. I think I have clearly explained to you all this in the last slide on the ANSI format and that is how the word address format goes.

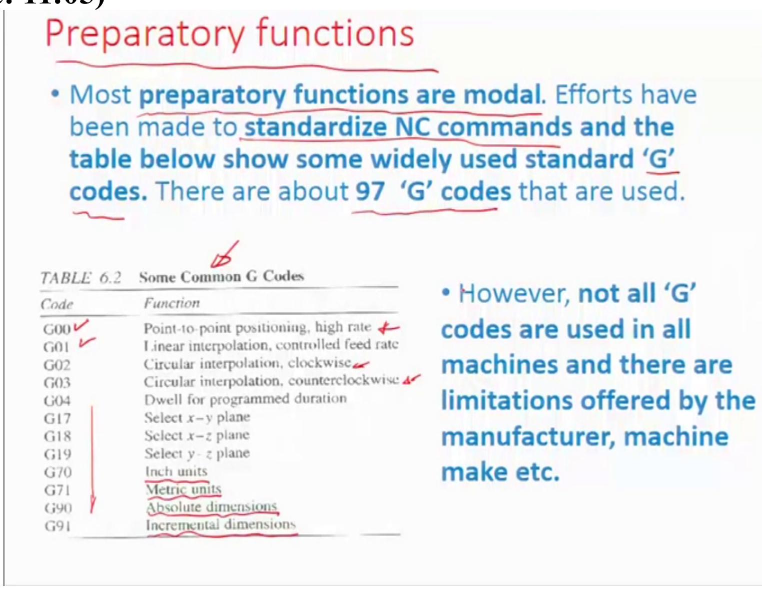
(Refer Slide Time: 10:15)



So, let us now step by step look at how to write such programs on a real sense. So, the first step in writing of course of a part program is to be able to determine and organize the data that will be used within the program for a certain part. Generally a fully coded program would consist of 5 broad categories or classes of commands there would be some preparatory function there would be some exists motion commands.

Some feed speed commands, identification commands and miscellaneous commands these are the must-haves in the different blocks to be able to justify. You know the program make a correct logical sequence.

(Refer Slide Time: 11:05)



So, when we look at each of these types we will need some little discussion related to the different types of commands. So, let us talk about preparatory commands first, so what I did not explain earlier and I can probably take it up now is that there are two different classes of commands one called the modal command another called an on modal command. Depending on whether the commands keep existing if not entered again okay or if not changed so there is a counter you have fixed some numbers.

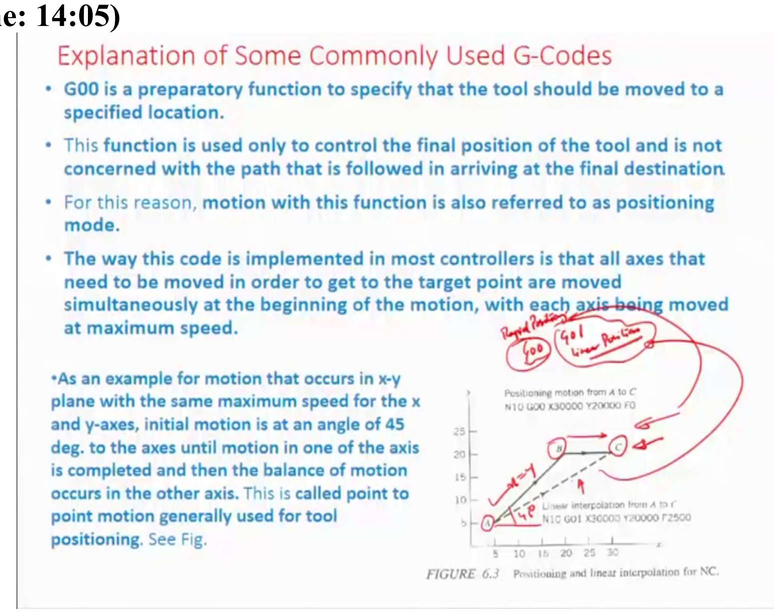
And the number keeps on being taken by the code every time up to an extent you do not change it. So, these are models so in time they are progressing there of course non model where once it has been executed the command would stop and the next step would typically have to be entered back. Let us look at a very basic machining operation like dwell time. So, once for particular step there is a dwell time inserted you would not definitely wish the dwell time to continue until changed.

So, once executor it is automatically set back to counter 0 unless you again reprogram it in the other case for example something like a proprietary command like a rapid positioning. Once the rapid positioning aspect is over until you decide that a different form of positioning needs to be in place it should keep continuing so these are modal commands. So, most of the propriety functions are model so they have to be changed if you want them to have different effects efforts have been made to standardize more or less all the NC commands.

And these codes are typically you know which describe the proprietary function are also very well known as G codes there are about 97 different G codes that controllers are used to handle. Some of the major ones are represented in this table 6.2 starting from G00 which is a point-to-point positioning at a high rate we also call it rapid positioning. G01 which is a linear interpolation controlled feed rate.

G02 circular interpolation clockwise or circular the interpolation counterclockwise and then ever idea whether these different codes G04 for example is a dwell for program duration 17 a select xy-plane, 70 was 4 inches units, 71 is metric units. So, these are preparing the axes to start reading the numeral data numeric data associated with the part geometry, 90 is absolute dimensions, 91 is incremental dimensions and so on so forth. So, these are some commonly used preparatory codes in almost all the programs.

(Refer Slide Time: 14:05)



Now if we look at the differences there are very subtle differences between the preparatory functions for example if we look at the case of G00 as opposed to 011 being the rapid positioning another being the linear positioning the utility here is quite different because the rapid positioning comes into picture when there is no tool engagement with the workpiece typically when you are trying to take something back to the origin or big taking back the tool to the point of engagement where machining motion would start.

You can rapidly do that because 2 does not scratch the particular workpiece surface. But once it is engaged and once the machining operation starts there has to be a well-defined feed rate which would be in accordance with the material properties, tool properties, the failure criteria as you know all these different issues associated with such tooling. So, the rapid positioning is about a path which is represented here okay very well derived by the solid line.

Let us say in certain instance certain operation you would like to go from the point A or position A to the point C. So, there are two ways of doing it one is of course a direct translation from A to C through the dotted line and another is going from A to B and then from B to C the question that is asked normally is that which of these paths are more in terms of time and which may be more lesser in terms of time and so it is really about utilizing controller capabilities.

So, when we are talking about this path definition through two different coordinates namely the X and the y coordinate what we mean is that there are two different spindles one in the X direction which has a motor which will rotate and execute different values of X and another in the Y direction which will execute different values of Y. So, supposing the motors were to operate at full speed then obviously the X will always be equal to Y in that situation.

However the Point C as you can see in this figure may not satisfy that criteria where X = Y because C is slightly more towards the right on the X then on Y if they were to move equally with respect to each other. And so therefore a best idea for minimum time arrival of a tool which were at point A to point C is to operate to an extent possible where X could be equal to Y okay where the maximum speed or the full speed of both the motors could be any shear it.

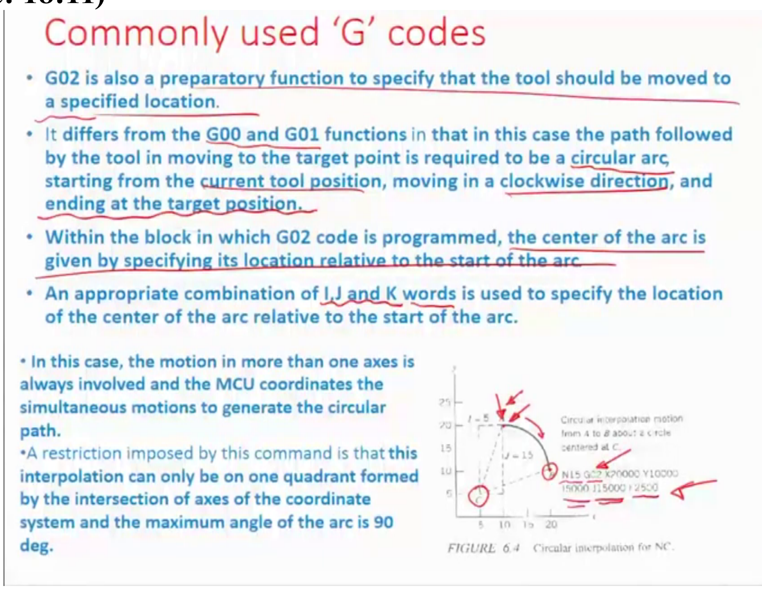
And then once it reaches that point namely B where the angle here is about 45 degrees corresponding to X equal to Y line then you can actually switch off one motor and let only the excess motion be covered in the x-direction again at full speed. So, time wise in a MC XY table it makes perfect sense if I say that the path ABC is more quickly traversed in comparison to AC.

You can see the limitation in AC is that one of the motors will definitely particularly in the x-direction has to be slower so that it reaches between A and C.

So, the question again comes into which n is a better motion and so one from ABC is considered to be rapid positioning G00 and the one which is actually along the dotted line here either a certain fixed rate we can call this to be linear positioning G01. So, these are inserted there the sudden subtle differences between the different G codes which are often on being visited by a programmer okay.

So, typically the rapid positioning is also known as point-to-point motion or point-to-point positioning. And the linear positioning is a rate driven, feed rate driven positioning system.

(Refer Slide Time: 18:11)



Let us look at some other aspects which is about the circular interpolation G02 and 03, so as we all understand G02 and G03 corresponds to whether or the arc of a circle is described in the clockwise sense or the counterclockwise sense. It is a proprietary function to specify that the tool should be moved to a specified location it differs definitely from G00 and 01 functions in that in this case the path followed by the tool is moving from target point to the target point in a circular manner.

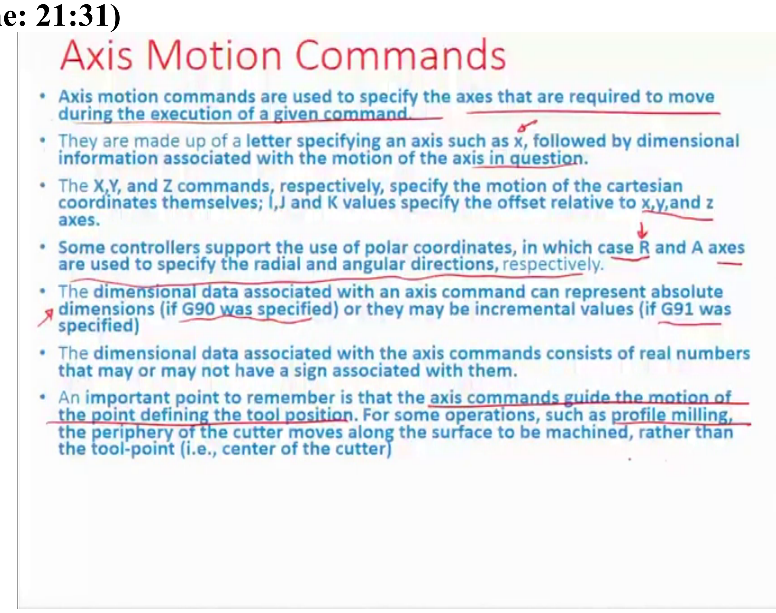
So, there is a circular arc which is being described in order to reach the target okay starting from the current tool position and you know you can move in the clockwise direction if you have G02 if it is G03 it can be the anti-clockwise direction and so you should have an ending target position as well provided an early position where earlier on a position earlier on where it started to execute the arc.

So, within the block in which G02 is programmed for example this right here shows a block N15 where this G02 comes up as a preparatory command. The center of the arc is given by specifying its location relative to the start of the arc to an appropriate combination of words I J and K. In this case for example it has a certain X and Y position which shows where it starts from so these are the starting positions namely X20 and so we have the end positions here at X20 Y10 corresponding to the point B up to which the arc has to be described.

And we would like to definitely find out the start position from the last step in the programming of the code where the tool was which is the position A in order to describe this arc we need to define the center about which it has been described and for which you have these I and J commands which correspond to I 5000 and J 15000. And of course a feed rate which is about 2500 millimeters per minute in this particular case.

So, essentially what is happening is that you are taking this input position where the tool is starting from the last command line or the last block and you define what is the end position and you also define what is the center along which a motion has to execute the very nature of G02 makes the interpolation go in a clockwise direction. So, obviously from A the tool would start moving towards B and not revert back. And so that is how you define G02 or subsequently G03 circular interpolation.

(Refer Slide Time: 21:31)



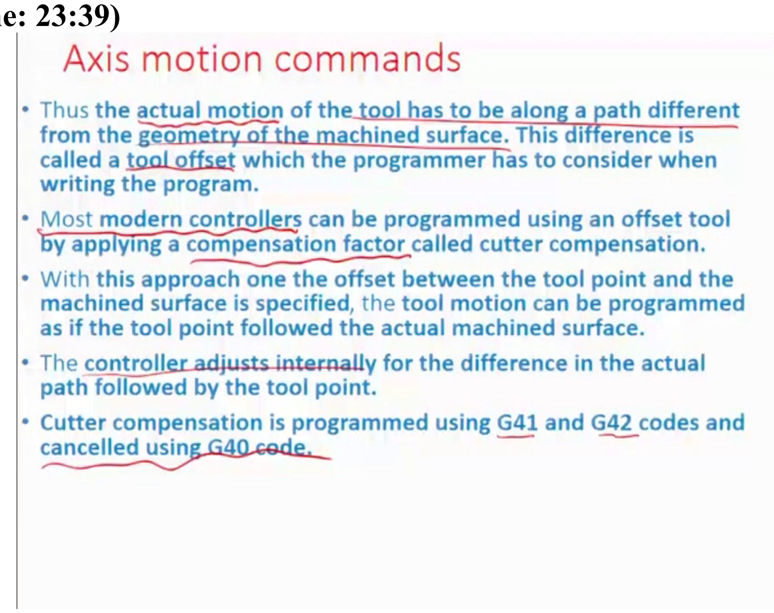
So, there are also axis motion commands which are used to specify axes that are required to move during the execution of a given command they are made up of letters signifying an axis such as small x for example followed by dimensional information associated with the motion of the axis in question. The X Y and Z commands respectively specify the motion of the Cartesian coordinates IJK specify the offsets relative to the XY and Z axis.

Some controllers support the use of polar coordinates also in which case there would be the additional definition of radius R and axes are used to specify the radial and angular direct also respectively. So, if you want to reset the axis of motion in terms of the dimensional data that you input so you can actually specify whether you want to read data on absolute basis or if you would like to read them on an incremental basis.

You could also have some codes namely G70 associated with whether the dimensions that need to be followed of the dimensional system that needs to follow is inches or 471 it is metric units. So, these are some basic preparatory functions needed for setting up or setting stage for the machining to happen. 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 the center of the cutter.

So, one has to be careful of the cutter compensation and other aspects when you program such particular motions.

(Refer Slide Time: 23:39)



So, obviously because of the radius of a circular tool the actual motion of the tool has to be along a different parallel path from the geometry of the machine surface. This is typically known as the tool offset because you are assuming that you are sitting as a programmer at the center of such a circular disc which has edges which are starting to engage with workpiece and perform cutting action etc.

So, most of the modern controllers though do have a compensation factor which you can put in so that you can still assume you yourself to be in contact with the workpiece profile and the compensation factor automatically would generate a parallel path. But such compensations are sometimes programmed with G41 and G42 codes and you can cancel such companies using G40 codes. So, but then one has to understand the basic a sense when you use such codes whether you are in a parallel path or whether you are at the point of engagement of the tool to the workpiece.

Rest of the things the controller's adjust internally for difference in the actual path followed by the tool point accordingly once the G command G 41 and 42 are applied. So, I would like to now finish this particular module maybe in the next module we will do some real programming for a certain part with all these knowledge base that we have developed.

So, that you can become trained in carrying out such small programs so once we do that then we will start various aspects of rapid tooling and prototyping which is actually very important from a standpoint of 3D shape visualization or creation thank you very much for being with me.