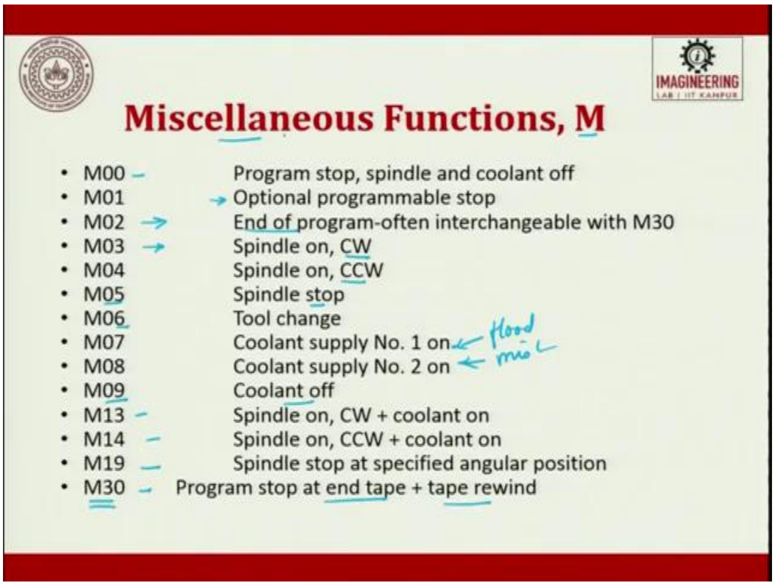
Computer Integrated Manufacturing
Professor. J. Ramkumar
Department of Mechanical Engineering and Design Program
Indian Institute of Technology, Kanpur
Lecture 21
CNC Part Programming (Part 3 of 4)

(Refer Slide Time: 00:19)

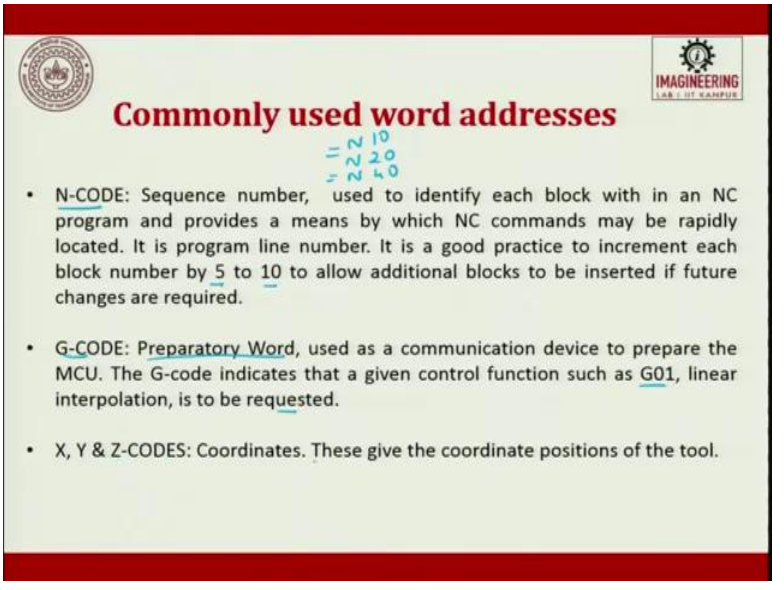


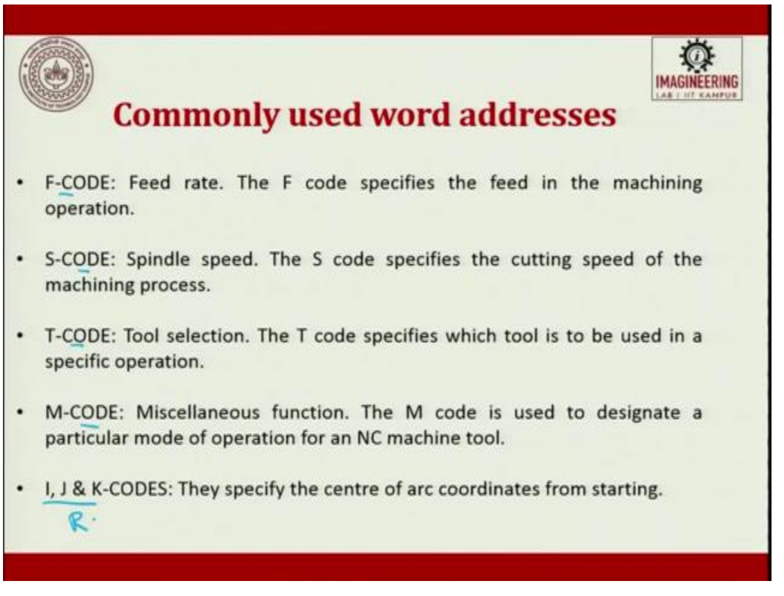
The next important topic in CNC is going to be, miscellaneous code functions. So miscellaneous codes are, these are all important from the point of your execution. G00 command are preparatory code, they will be used for simulation, dry simulation and see on the system whether the profile, whatever is cutting is to your requirement. But when we start introducing the miscellaneous codes, these codes are going to help me while manufacturing the product.

So M00 is program stop, or spindle and coolant off, we use M00 command. M01 is optional stop or a dual cycle, M02 is end of the program, M03 is spindle move clockwise direction, M04 in counter clockwise direction. If we want to stop the spindle, we give M05. Then for a tool change we give M06, coolant supply on.

So, it can be, this can be flood type, this can be mist type. So we say, M07, M08 for coolant supply and for offing it is M09. And, M13 and M14 are spindle on clockwise plus coolant on, spindle on, counter clockwise plus coolant on will be M14. So, M19 is spindle stop at a specified angular position, we use M19. And M30, is program stop at end tape plus tape rewinding is M30. So, we can use these commands for executing the part in a CNC machine.

(Refer Slide Time: 01:57)



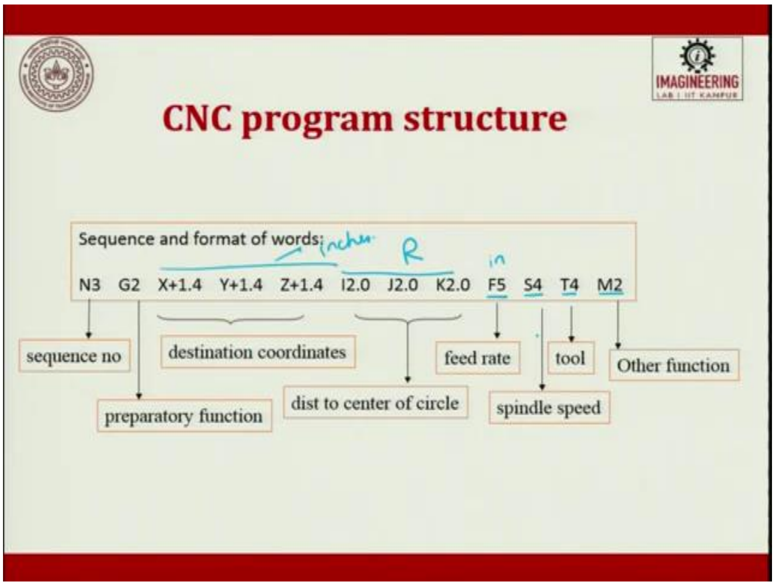


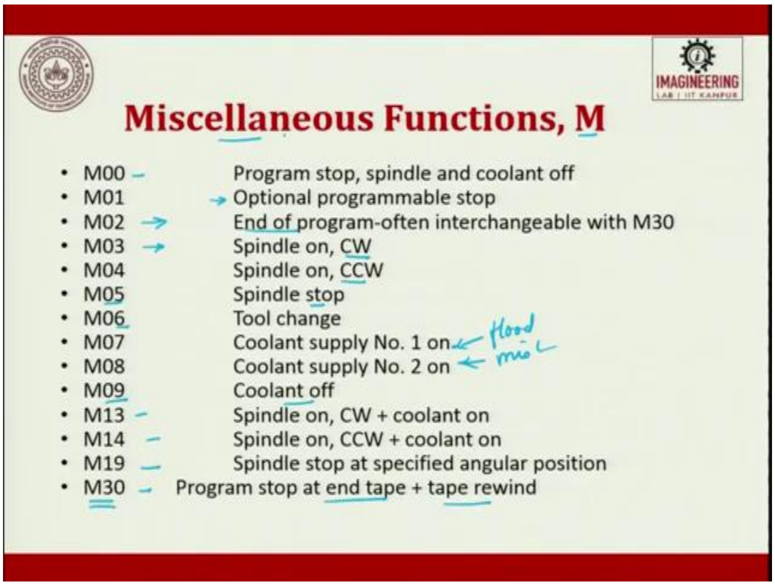
So commonly used to word addresses are, N code. So, N code is the sequence number used to identify each block with an NC program and provides a means by which NC command may be rapidly located. For example, N10, N20, N40. So, N is nothing but the line number. It is program line number. It is a good practice to increment each block number by 5 or 10.

So that, in case if you want to do some additions it can be easily inserted in between those lines. G codes, G codes are preparatory words used to as a communication device to prepare the MCU. G codes indicate that a given control function such as G01, linear interpolation is to be requested. G codes are preparatory words. X, Y, Z codes are coordinates.

Then we have F for feed rate, S for spindle speed, T for tool number, M for miscellaneous code. I, J, K code for the, they specify the centre of R coordinates from starting. So, this is what I said I, J, K or R command can be used. The R command can be used for a radius, if you call it on a canned cycle, R command is used to for the reference plane.

(Refer Slide Time: 03:28)

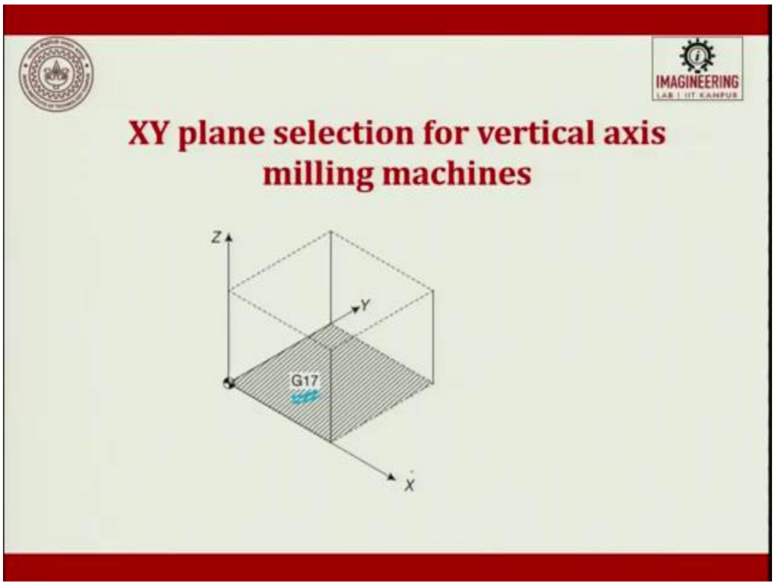




So, this is a structure of a sequential block. So, you will have a sequential number, then you will have a preparatory function, then you will have, here it is all written in inches, these are all in inches. So, it is X plus 1.4, Y plus 1.4, Z plus 1.4. So, these are the destination coordinates. I, J, K are the distance to centre of circle, x axis, y axis, k axis offset. Or if you do not want, you can write it in R command, which talks about the radius. Then we have feed rate again in inches, spindle speed, and then we have tool number, and then miscellaneous function M02.

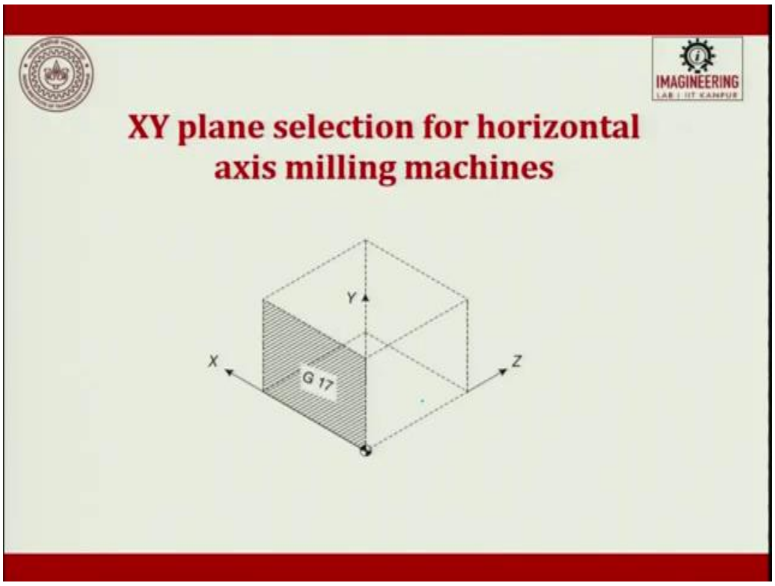
What does a miscellaneous function M02? End of program often interchangeable with M30. So, you can give M02 or M30. And in this block, you should understand, M when we try to write the M code, it will be almost at the last. So, you will just to follow the sequence, in a fixed block. So, what will happen is, you will have to fix all the X, Y, Z, all the data has to be filled in.

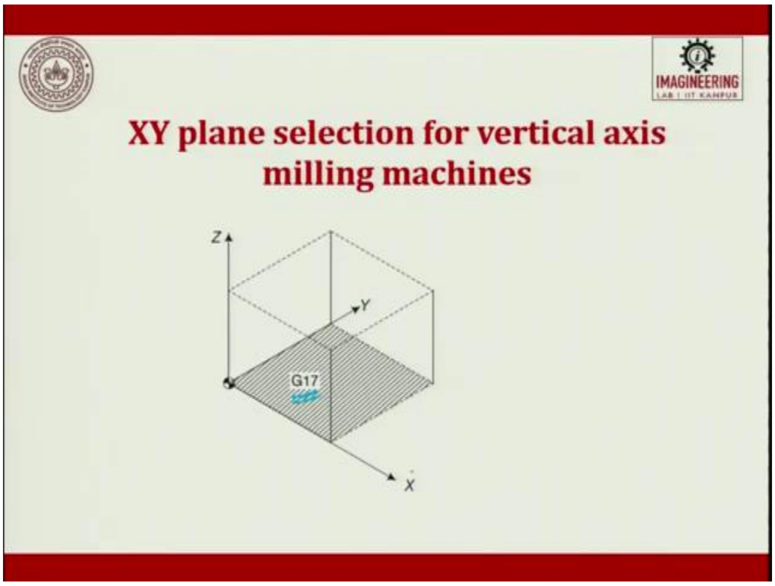
(Refer Slide Time: 04:42)



So, this is what is the XY plane. So now we are going to see, the G16, 17, 18 planes, whatever we have run. So, XY plane selection for vertical axis milling machine. So, this is the G17 code which is used on XY plane.

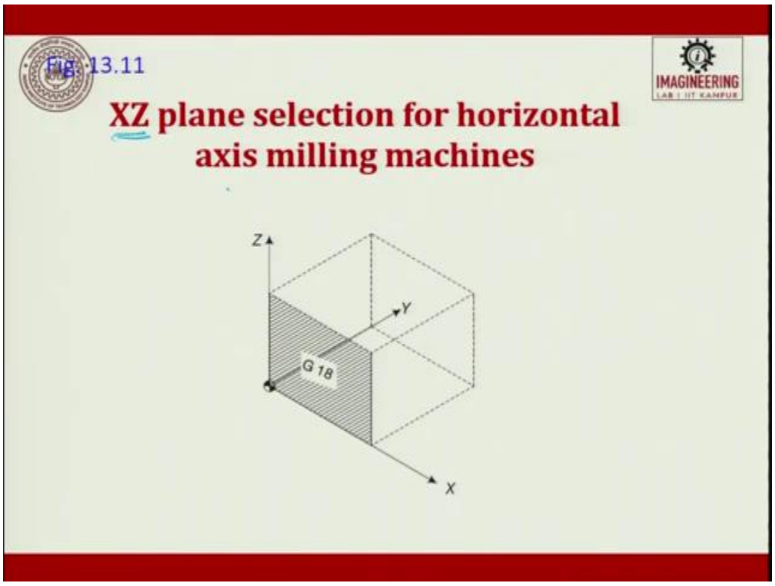
(Refer Slide Time: 05:02)





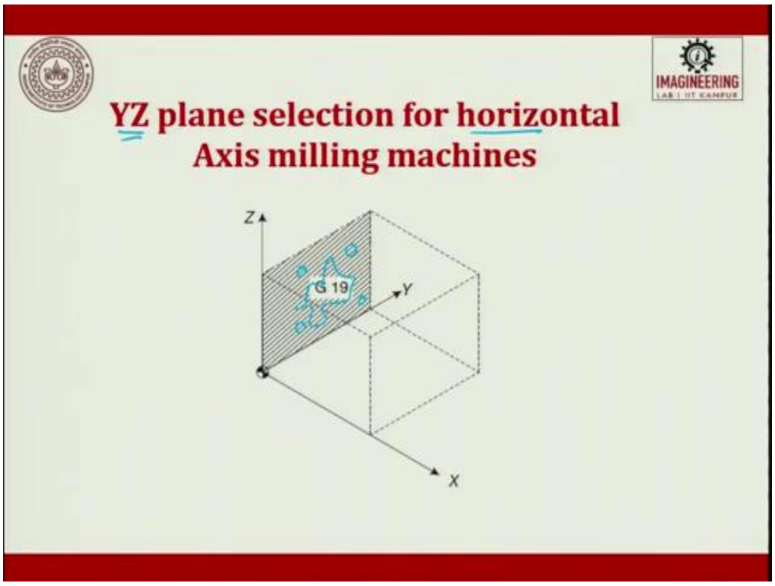
So, this is the XY plane selection, on a horizontal axis milling machine. Please see the difference. This is XY plane for vertical axis milling machines, bottom XY plane. When I do the same thing for a horizontal axis, XY plane I, this comes like this. So, this is X and Y, G17.

(Refer Slide Time: 05:26)



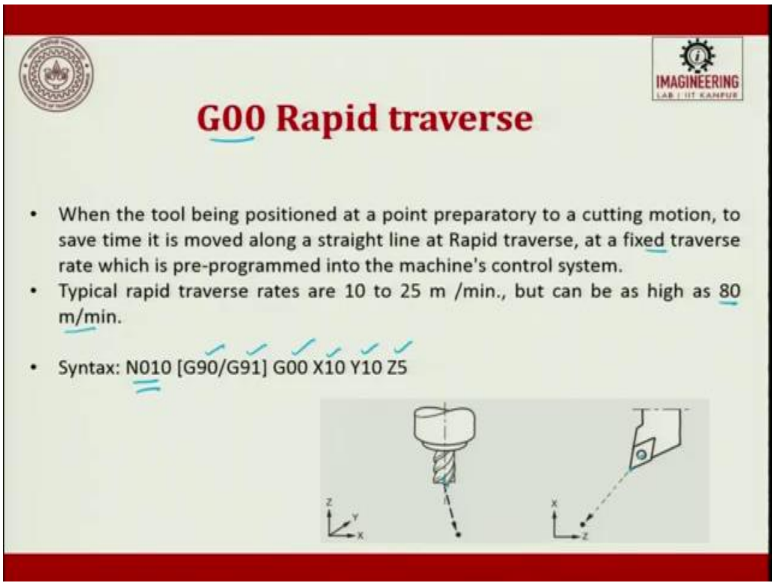
So, the XY plane for horizontal axis X is at plane, horizontal axis milling machine X and is at, so this is G18. X is at plane selection for horizontal axis milling machine.

(Refer Slide Time: 05:44)



So, YZ for horizontal axis milling machine is G19. So, you see for from machine to machine, horizontal to vertical axis when you do, so you can see the planes differ. You have to define these planes, then after defining this plane in this plane, what are all the operations to be done. So, then you can ask that to do.

(Refer Slide Time: 06:07)

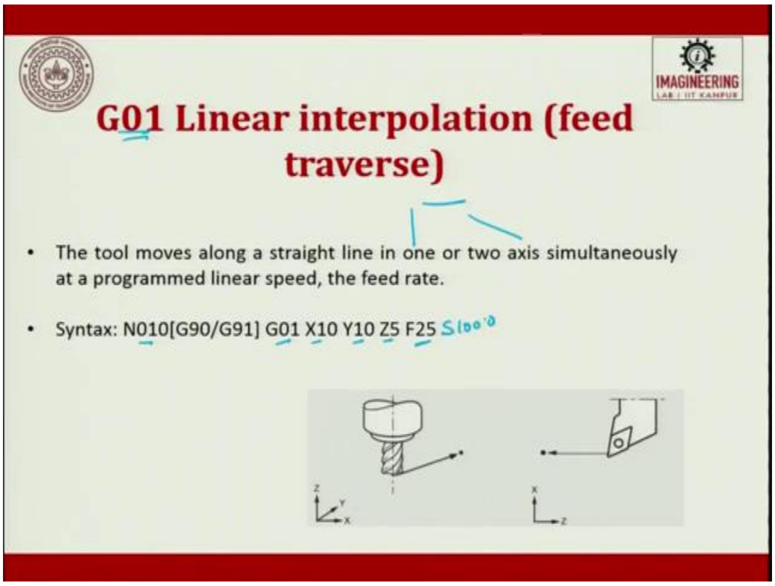


So, now whatever we have studied in rapid traverse, let us have a pictorial representation and understand. When the tool is being positioned at a point preparatory to a cutting motion to save time it is moved along a straight line at rapid traverse, at a fixed traverse rate which is preprogramed into the machine’s control system. Fixed traverse is the highest feed rates, whatever it is. Typical feed rates are 10 to 20 meters per minute, but can be as high as 80 meters per minute.

So, the syntax when we use this G00 command we will say, this is the numbering of the blocks, so N0010 and then you can say whether to follow up absolute scale or incremental scale any one. Then you G00 which is rapid traverse, and then you will define X and Y and Z motion. What is that, so this will be the destination point which G00 has to, the tool will move.

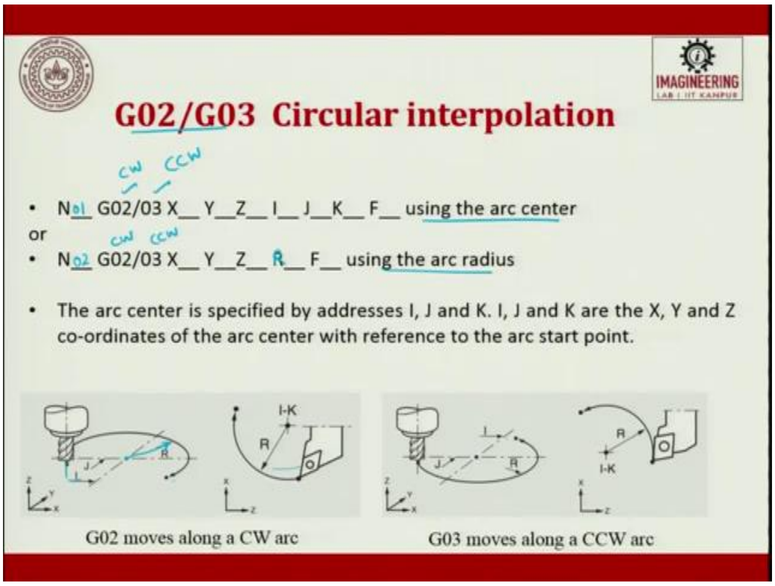
So, when we tried to do on a turning, it is, it can be with respect to this or with respect to the nose radius. You have to define it prior or it can be with respect to the centre of the tip or for the periphery of the tip, depending upon the tool register, whatever you have done accordingly, you change the central point. You are to look for the machine, how does they take the reference points. When suppose if they take it for this, then you have to give these points as offset. If you take it for the centre, then the diameter or the radius has given as offset.

(Refer Slide Time: 07:37)



When you look at G01, the tool moves along a straight line in 1 or 2 axes, that is what I said, horizontal vertical 2 axes are taper. Simultaneously, at a programmed linear speed, the feed rate. So here, we follow the speed and feed rates what are defined. So here, what we have to do is we write the blog N010, G90 or G91 absolute or increment, G01 command, X, Y, Z and we define the feed. So, you can also have the speed, speed 1000. So, at a programmed linear speed and feed rates, the program, the block will execute.

(Refer Slide Time: 08:19)

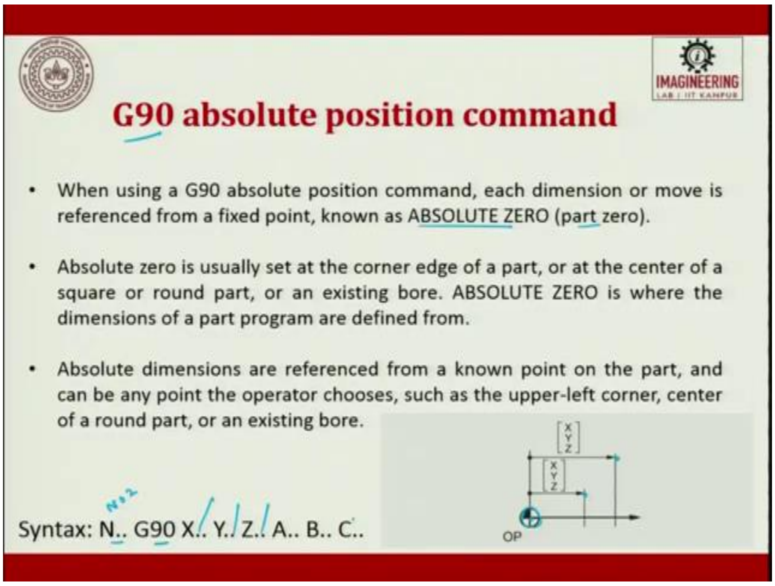


When we talk about circular interpolation, G02 or G03, it is circular interpolation. There are 2 formats. One is number block 01, so then you can choose either G02 or G03 this is circular clockwise. This is circular counter clockwise. And then X, Y, Z. Then you will have an offset in x axis, offset in y axis, offset in z axis, then you will have a feed rate. So, using the arc centre.

So here, the other option can be N02, same code G02 clockwise or counter clockwise X, Y, Z. And then you define a radius R, and a feed. So, using the arc radius, the arc centre is specified by the register I, J and K. I, J and K are the X, Y and Z axis coordinates of the arc centre, with the reference to the arc starting point.

So it is here, it can come up to here. So, it is I, J, then you will have, so it will move from here to here. And this point is defined and then you define that radius R. So, it is I, N, K you have a swing like this. So, it is G03. Counter clockwise you can have I and J will be this, and then you will have a radius from the centre point. So, it is I, N, K for turning, so I N K for turning.

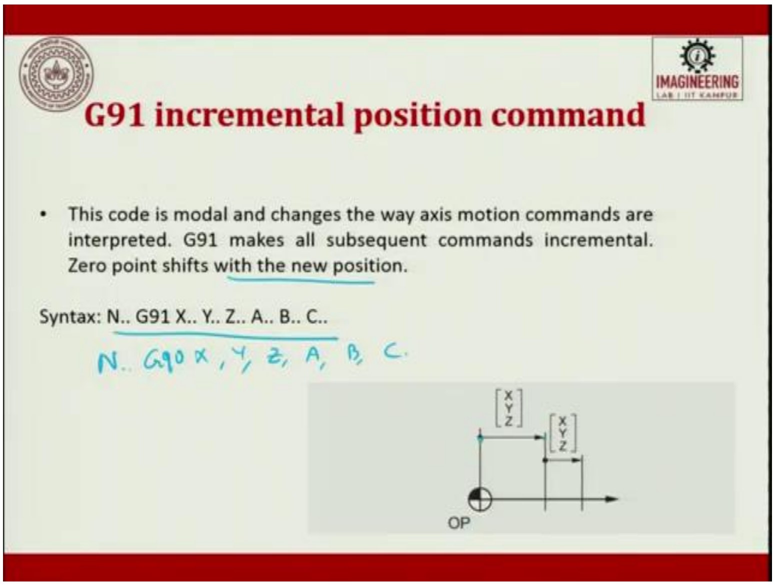
(Refer Slide Time: 09:50)



G90 obsolete position command, each dimension or move is referred from a fixed point known as absolute zero point, of the part zero. The absolute zero is usually set as a corner edge of a part or at the centre of the square or round part, or an existing bore. Absolute zero is where the dimensions of the part programming are defined from. So, this is what is the absolute, this is what does the (0, 0). From this point, I am trying to define the X, Y, Z for this and the X, Y, Z for this.

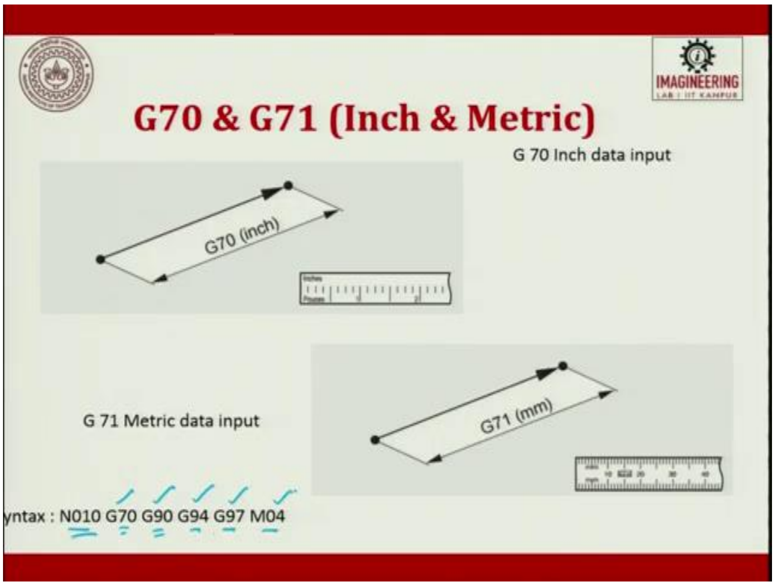
Absolute dimensions are referred from a known point on the part, and can be by any point the operator is chosen, such that the upper left corner, centre of the round part or at the existing bore. So, he can choose anywhere. So, this is the left bottom, left top, extreme corner, whatever he wants he can do. And everything will be with respect to this point, we try to take the dimensions. So here syntax, N we have 02, then G90 for absolute, X, Y, Z, are further coordinates we say A, B, C, are for the rotation about XYZ. So, this is what we designed in the absolute code.

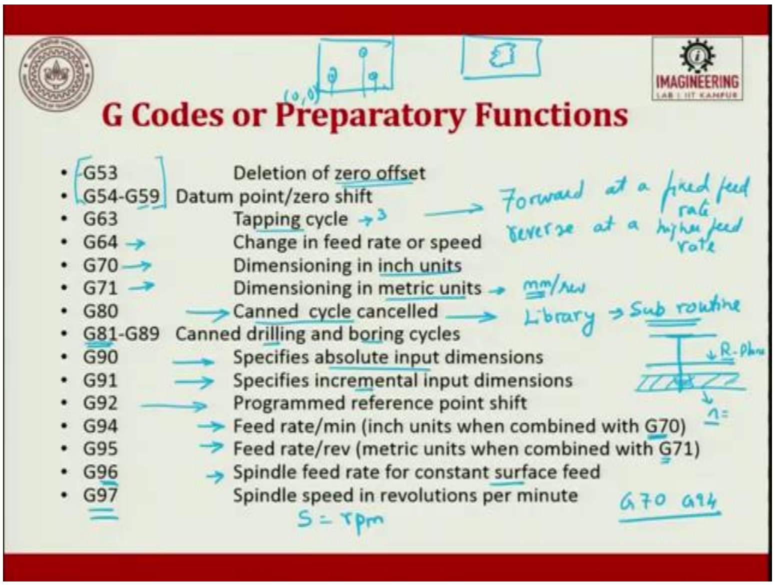
(Refer Slide Time: 11:20)



When we talk about increment, it is modal and changes the way axis motion command are interpreted. G91 makes all subsequent commands incremental. Zero floating shifts with the new position. So, this will be written like this. So, there is no difference, syntax will be almost the same. This is for (X, Y, Z), if you want to define (A, B, C) it will be like this. This is G90, this is G91. So, everything is respect to this point, you go here and then with respect to this point, you go to the next point. So, every point where you stop becomes a floating zero point.

(Refer Slide Time: 12:02)

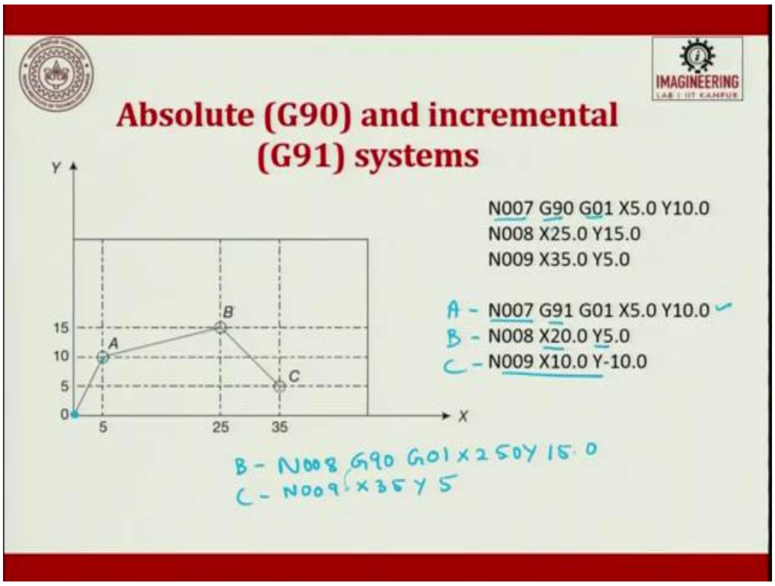


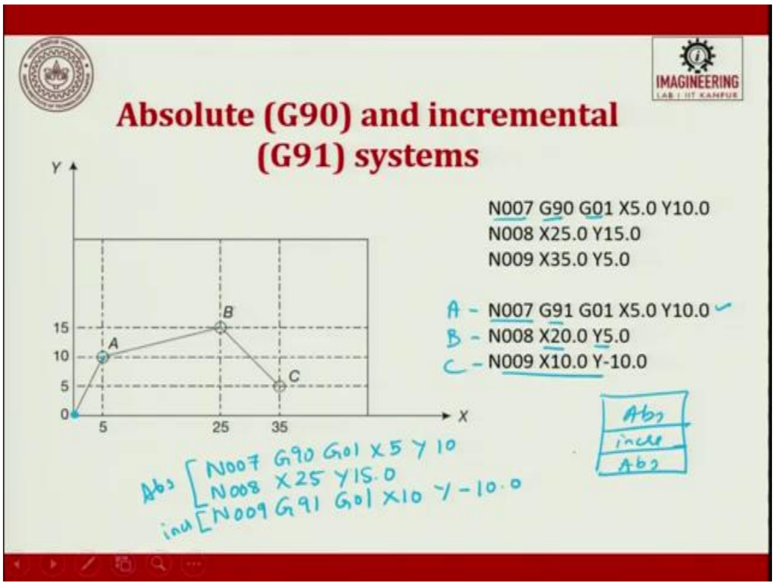


This, 70 and 71 it is an inches. So M, it is in metric. So, this is how syntax is. You can write it as N10, 70 which is in inches. Then you call it as G90, which is in absolute scale. Then 94, 97 and M04. Let us see, what is 94.

It is feed rate in mm per minute, and then 97 is spindle speed in revolutions per minute. So, we have put all those points here. So, it is 1 syntax you can call. G94, G97 and M04 is spindle on. So, all these things are done in 1 line. So, you should understand this syntax. G9, 70 is inches absolute. Then it is, you can talk about inches per minute and then this can be in RPM, and then this can be the spindle on. So, all these things can be defined in 1 block.

(Refer Slide Time: 13:07)





So, this is absolute and increment. I am just comparing. So, N07 is the seventh line of that particular program. You have called G90, which is absolute and then you said G01 moved to this point A, which is X5 and Y10. So, the next point is going to be, if you are, if I am going to move here, it is going to be next to G, N. So, to have a safer role it will be N008, G90 then you will say G01, X25, Y15.

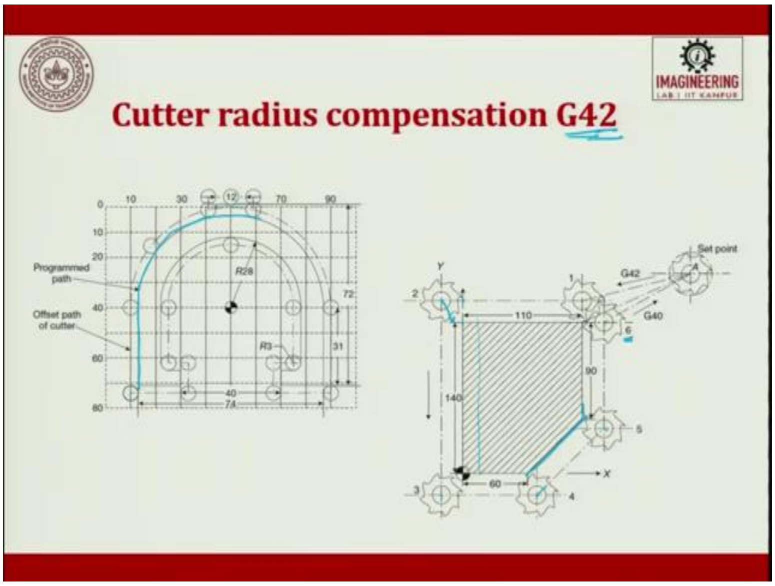
So here, I am just trying to play safe. You need not call at every block, the previous G code, whatever you have used, you need not call. If you call it, you maintain it so that you will know, where you make an error. Otherwise, it is not necessary to call. The next sentence for, this is for B, and this is for N009, I just say X35, Y5 that is all. So, rest all it is assumed from by the modal command, it takes G90, it takes G01.

So, this is what it is. So here what we have done this with respect to absolute, with respect to a standing reference, we have noted down all these points. Now let us do the same thing with increment. When we do with increment, this is so your N07 which is here A. So, here what you do is you write G91, G01, you say 10 and 5 or, so, X is 5, and Y is 10, this is point A. So now, you have already moved 10 and 5, so when I moved to point B it is going to be, I will reduce 5, 25 minus 5 which will be 20, and this will be 15 minus 10, 5. So, incrementally it will be 20 comma 5, so this is your B point.

When you try to talk about the C point, it is going to be 35 minus 25, which is 10. And then Y axis falls from 15 to 5. So, 5 minus 15 which is minus 10. So, this is what is written here. Now, I am sure by looking into this program, you will be able to solve absolute and incremental code usage. In a program you can have several portions, which is absolute and increment. Let us do, 1 trial.

So, I will now try to do a program where N007, G90, G01, X5, and Y10. The next line I am writing it as, N008, X25 and Y15. The third line, I can write it like this, G91, G01. So now, X and Y. So, it has come up to 25, so now from 25 I am going to go to 35. So, I will say, X10 and Y -10. So, I have used in the same program of absolute, I have converted to incremental. So, this is possible, I have just made 1 block as incremental, possible. So, in your program, you can write half of the program in absolute, half you can write it an increment and then once again you can write it in absolute, to meet out your comfort zone.

(Refer Slide Time: 16:59)



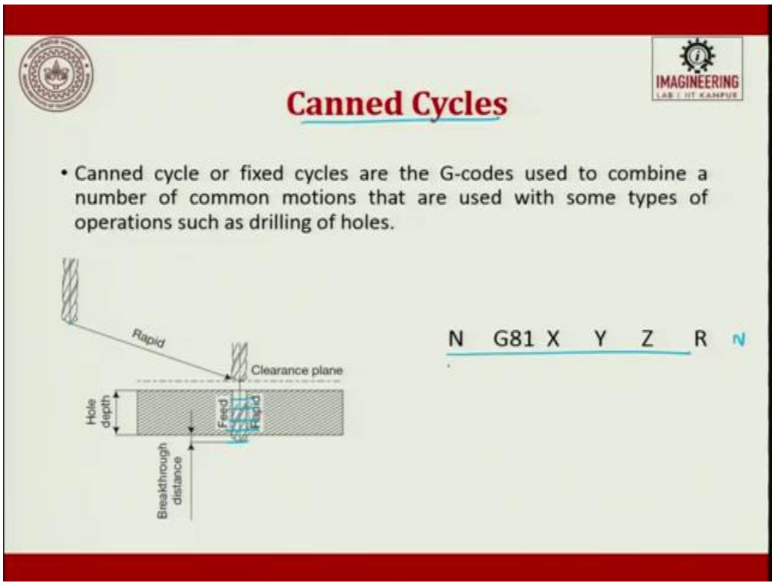
So, the cutter radial compensation, cutter radial compensation G42. So, look at it. So, this is your programmed path. So, this is your reference. So, now you are trying to take, this is your programmed path, this is your programmed path. So now, your cutter is moving to the programmed path. This is your cutter; this is your diameter. So now, what are you trying to do is, you are trying to move because the cutter diameter comes in contact and then you have to generate this profile. So, cutter edged comes and touches the profile.

So now, the cutter edge if it has to touch, the centre has to be offsetted. So now you see, there is an offset path of the cutter which is generated, which is this. This is an offset path with a dash, dash line which is there. So, this is what is created. So, this one is G42 command, so this is offset. So, when you look at it, this is the centre. So, your first point is here, so here is your first point. So, the cutter comes in contact with our piece, it starts cutting. So, from here, the centre is moved here, and this is your offset diameter.

So, and when it comes here, so if you look at it, if it exactly stops here, so then it will try to cut inside the material. So, in order to avoid this, what we do is we offset this by half of the diameter, move to the extreme out. And then from here, so your points should be one here, shifted from, that means the tip must touch here. So, the centre must be shifted.

So, this is your point 2, then comes your point 3, then goes to your point 4, then you see here, you will move along this point 5. So, point 5 also you see here, there is a taper which is there. So, what we do is we try to define up to this point 5 and then the cutter starts moving along this line and goes here. So, this is nothing but cutter radius compensation G42. We said G41, G42. This is cutter compensation G42 code for the outer.

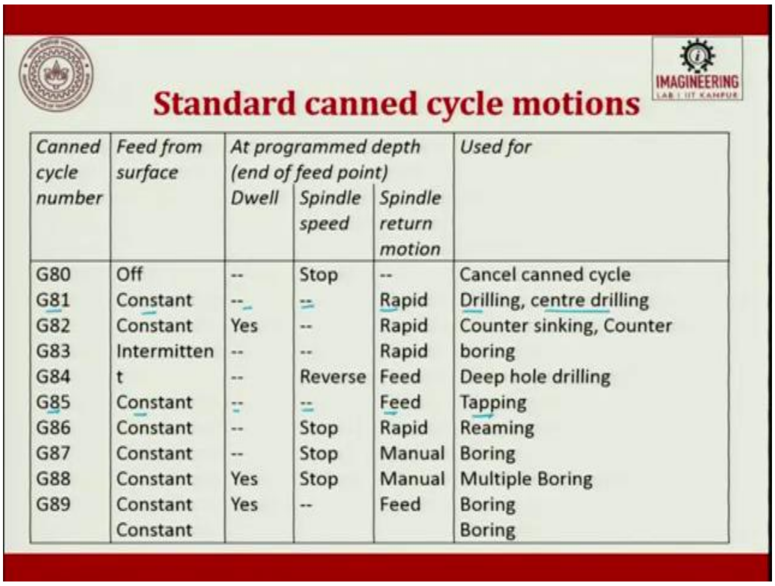
(Refer Slide Time: 19:35)



When we see a Canned cycle, a Canned cycle or a fixed cycle are G codes used to combine a number of common motions, that are used with some type of operations such as drilling of a hole. So, from here rapidly it comes to this plane and this plane is called as a clearance plane. And from here, you can keep going inside and outside. So, number of times it can peck and then do the operation.

So, if you put that in a code form, N0 and then G81. X, Y, Z reference plane, you can define the number of times it has to go. And then, it also tries to tell what will be the projection of this, outside the workpiece also can be defined. So, this is how a Canned cycle works.

(Refer Slide Time: 20:26)



Standard canned cycle motions

Canned cycle number	Feed from surface	At programmed depth (end of feed point)			Used for
		Dwell	Spindle speed	Spindle return motion	
G80	Off	--	Stop	--	Cancel canned cycle
G81	Constant	--	--	Rapid	Drilling, centre drilling
G82	Constant	Yes	--	Rapid	Counter sinking, Counter
G83	Intermitten	--	--	Rapid	boring
G84	t	--	Reverse	Feed	Deep hole drilling
G85	Constant	--	--	Feed	Tapping
G86	Constant	--	Stop	Rapid	Reaming
G87	Constant	--	Stop	Manual	Boring
G88	Constant	Yes	Stop	Manual	Multiple Boring
G89	Constant	Yes	--	Feed	Boring
	Constant				Boring

The various canned cycles, and their numbers are given for your understanding. Let us take only one. So, let us take G81, which is a feed from the surface is going to be constant. So there is no dwell, the spindle speed will be the same, when it is returning back it will have a rapid speed.

All the drilling cycles and centre drilling will use it. Let us look at tapping, G85 the feed from the surface will be constant. The programmed depth, end of the feed point there will not be any action, so that will not be reversing. The feed rate, return motion will be there. This is a tapping cycle. So, this is how canned cycle works. Thank you.