

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

Module- 06

Lecture- 36

Welcome to this module 36 on Manufacturing System Technology. So, we were talking great details about the fixed sequential format and we also knew how we you know, there can be a drilling operation which can be stimulated or which can be mentioned in terms of 3 block statements; 3 blocks or 3 statements. Using several words, some of the words are indexing numbers, some of the words are related to whether it is linearly or rapidly able to position, some of the words are related to the xyz dimensions or co-ordinates, some of the words are related to the feed rate of the drill and some of the words are related to the end conditions, for example is the coolant on or coolant of or so and so forth.

So, it was a very simple progression of sequence of commands which was given in terms of numbers. Where now the understanding of the controller already will be in terms of those numbers, and it will also interpret everything in the absolute manner starting from the origin. So, that it can do this job by sending the right amount of current signals to the different drives which are holding; for example, the tool or for example, the work piece. And left alone to it will do some optimization to calculate whether it is the tool which is going to move or the work piece, but your actually programming always as the drill tip or the tool. So, we are following whatever syntax we have actually laid down earlier in this particular method. What are the problems associated with the fixed sequential format and that gives you a basis of the tab sequential format. So, this is how the tab sequential format actually looks into.

(Refer Slide Time: 01:35)

Tab Sequential Format

Exhibit 6.2 Tab Sequential Format

```
0050 TAB 00 TAB -0025400 TAB +0012500 TAB TAB TAB
0060 TAB 01 TAB TAB TAB -0010000 TAB 0500 TAB 08
0070 TAB 00 TAB TAB TAB -0000000 TAB 0000 TAB 09
```

- This format is essentially the same as the fixed sequential format. The difference is that each word within a block is preceded by a Tab character.
- The main improvement over fixed sequential format is that the Tab character for a specified word need not be followed by a numeric data if data are not required in that particular block or they are modal data.
- The Tab character indicates a beginning of a new word but does not specify which type of word; therefore the sequence of the words remain significant.
- Exhibit 6.2 is a repetition of the program represented in Exhibit 6.1 but written in the Tab sequential format.

So, if you look at let us say this particular illustration here. This actually gives you the tab sequential format and you know if I look at one of the problems related to its predecessors that is the fixed sequential format, it is in terms of reputability of the different materials or the different you know numbers. For example, let us look at this particular slide here which talks about the initial basic about the fixed sequential format.

(Refer Slide Time: 02:07)

Fixed Sequential Format

Exhibit 6.1 Fixed Sequential Format

```
0050 00 +0025400 +0012500 +0000000 0000 00
0060 01 +0025400 +0012500 -0010000 0500 08
0070 01 +0025400 +0012500 +0000000 0500 09
```

(7 Commands) ←
Block 1 →
Block 2 →
Block 3 →

- This was the earliest block format used for NC machines and it spawned the name numerical control because in this format only numbers are used.
- With this format each block in the program consists of exactly the same number of words, entered in a specified sequence and each work consists of a fixed no. of data characters. The data characters have positive and negative signs and are interpreted according to their location.
- Characters cannot be added or deleted as this might change the meaning of the Code. Every word must be represented even if the word has zero characters. This frequently results in using unnecessary information and long codes.

So, if I closely look at it, this numbers here for example, the 01 number on block 2 and 01 number on block 3 are repeating. The 0 0 2 5 4 0 0 number is repeating every time. The 0 0 1 2 5 0 0, 1 2 5 0 0 these are also repeating every time. Similarly the number 500 is repeating every time. So therefore, when you are talking about a manual punching

system for a programmer every time you have to take a command without taking it from the last line; you are actually inputting the command yourself in the next line again and you are typing again and again. So, it is definitely something which is not very handy thing it is a probably a time consuming affair and it is not a very productive way of doing the programming. And if instead of this we could actually convert this whole data as a rows and columns.

Let us say there are certain you know classifications and then you are provided. So, this is corresponding to the first row, second row, and third row in this particular manner; and for example, this is corresponding to first second third fourth all the way to seventh column. So, the data here is really given as a column address and a row address. So, you have a one row and one column address here, one row and two column address here, similarly one row and three column address here. One row and four columns similarly three row and four column address here. So, there is some kind of a matrix form in which this data has been laid down. So, it is very convenient for a an operator to, you know to assume if there is a button which allows you to just by a hit of button and two buttons. let say there is a tab button which is very common to all the computers. So, by the allowance of that tab button if I can just fix this particular value to 0 0 2 5 4 0 0, all I need do to do in the second line and the third line is to press the tab two times; so that, this value can be automatically taken from the last column and row address. So, you did not specifically write the 0 0 2 5 4 0 0 here, you can just rather press tab here and tab here so that the values automatically repeated. Similarly tab here and tab here and the values automatically repeated.

So, this is a change in technology which happened and the programming and the controllers who are started to develop in that manner which led to the development of this tab sequential format. Here for example you can see, that you know once you have given a value between two tab options you did not repeat this anymore. So, between the first and second tab automatically this value will be taken, between third and the sorry the second and the third tab automatically there is other value will be taken. And you just stop writing the values in a repetitive manner between the different blocks or different statements. So, definitely this was a an improvement in terms of productively being able to write a program and it was really a good improvement on the fix sequential format as you can see. So, format is essentially the same, there meaning by the sequence of the index number, the rapid positioning number, the x y z co ordinates, you know the other

details related to the starting of the feed or stopping of the feed or for example, the when the coolant on or coolant off. That sequence will not change of the comments, but only thing that you are changing here. Is utilization of the tab allows that you are repeating the values bit automatically or taking the values automatically from the last line when it involves repetition of the value. So, you are saving on the programmers' time this way.

So, tab character indicates a beginning of a new word, but does not specify which type of the word. Therefore, the sequence of the word remains significant from between the different rows. So, that is the tab sequential format. Finally, another opinion that was expressed by many people is that you know, when you talking about these numbers really for a person who is looking into a 360000 lines of programs. It is very in convenient at the end of the day to keep a tally on what follows what you know. Unless you have every time you are comparing the sequence and trying to see what sequence is corresponding to or what is the really the title of this column or the title of this particular column or this column you not able to compare every time and every time you have to really count you know from the left to right to identify which column which row and try to then see what is that function for.

For example, 0 0 is for the rapid positioning which is corresponding to row 1 and column 2. So, this row 1 and column 2 would be same throughout, but supposing there was something corresponding to row 1 and column let say 20. Something like that or may be row 1 and column 50. So, it is very difficult to identify that in a list of such sequences, if you do not have a title for every page as a header title for the particular column. So, therefore, it was you know argue to upon that why the controller should not be able to understand certain Basic English words. And obviously, the controller capability changed the lot during the, I would say the late 70's or in the beginning of even 80's and all. And so there was a substantial amount of technological improvement in the way that controllers are designed or specified and so. It emerged into what is call the current day format which is also known as the Word Sequential Format, which is the probably the newest format that has evolved in the whole NC programming system.

(Refer Slide Time: 07:47)

Word Sequential Format

Exhibit 6.3 Word Address Format

• This is the format that is used on virtually all modern controllers and will be explained in greater detail.

• With this type of format, each type of word is assigned as address that is identified by a letter code within the part program.

• Thus the letter code specifies the type of word that follows and then its associated numeric data is given.

• For example, the code T represents a tool number. Thus a word of the form T01 would represent tool number 1.

• Theoretically, with this approach, the words in a given block can be entered in any sequence and the controller should be able to interpret them correctly.

So, here you have a word as you can see, associated with a certain syntax number which has been given. For example, there is a word N associated indicating that this is probably index number. So, N50 means the 50th line of the program, N60 means the 60th line of the program, N70 means the 70th line of the program so and so for. Similarly you have a set of commands which have evolved which are also known as the G commands, which would really process the controller to do a certain statement. For example, in this block 1 and block 2 you have two different G commands; G 0 0 and G 0 1. So, this corresponds to your rapid positioning, earlier that I had actually illustrated to you what exactly rapid positioning is. And this corresponds to you linear positioning or linear feed, so linear positioning.

So, these two G commands which are also known as preparatory commands. Allows the controller now and presets the controller to carry out a certain operation in a certain manner. So, that is how this is done; obviously, the XY and Z numbers associated with them indicate the read outs in the you know, in terms of the number of co ordinates. Here of course, you can see that X25 4 0 0 has been written meaning there by this corresponds to 25.4r mm. So, this control is probably capable of reading between 99.999 to 00.000, about close to about 100000 values you know in between. And similarly the way that you have defined the Y and Z also remains more or less the same. You have also associated the word feed F from the word feed with respect to this number of 500 Millie meter per minute, which was actually the feed weight of the drilling tip that was being programmed in this way, and then so this is called the feed command.

So, you will; obviously, have also a speed in a speed command for defining the RPM, because later on you know drilling was no longer done at a constant you know RPM in the interest of machining optimizations. Because; obviously, it is also a describing process and there has to be a sort of optimum cutting rate, which is a function of the drill speed as well. So, you have different variability's in the drill speed introduced into the NC system, to have optimization of the cutting tip; because cutting tip is very expensive. And you have to create situation where that tip without changing can do multiple turning operations. So, therefore, there was a speed command also associated with the whole the NC programming system. And then; obviously, this whole coolant business coolant start and stop, and these are some of the miscellaneous functions.

For example, you can clamp or declamp a work piece or coolant can be started off or on at certain point which are really the miscellaneous functions associated with the machining, but not really actively involved as a working participant for the machining process. So, these are now classified as a miscellaneous command and so the word M comes out of there and there is an index now related to M1 M2 M3 and so on up to probably M30 or M40; there are many such miscellaneous commands which are available as index in the program.

So, you can see the same 3 blocks have been written in the word sequential format in a manner now, that I can probably go to this row and this column and understand what this element is. So, for example, this corresponds to M09 which means that the coolant should stop. This corresponds to Z09 meaning there by that the Z has gone to the position co ordinate before. What is also important for me to tell you is that, this what address format has also been designed keeping in the tab option available. So, every time you did not write the X and Y, once you have written the X and Y it automatically gets translated and only thing which changes is the Z value. From 0 you have changed to minus 10; that means, the detail has moved into the block at 10 mm and then again you are going back to the Z0; so therefore, it is re translating back to the Z0 option. And this whole thing is what in a very convenient and consolidate manner you can look into an at every spot you can identify what a particular number or a particular command would be using or would be used their for. So, this is what the latest development has in this area, and this is also known as the Word Sequential Format.

Let us now come to little more details about the different preparatory functions the miscellaneous functions; or for example, how do you lay out the XYZ data in such a

controller design.

(Refer Slide Time: 12:31)

Exhibit 6.3 Word Address Format

```
N50 G00 X25400 Y12500 Z0 F0  
N60 G01 Z-10000 F500 M08  
N70 Z0 M09
```

- With the word address format only the needed words for a given operation have to be included within the block.
- The command to which the particular numeric data applies is identified by the preceding address code.
- Word format has the advantage of having more than one particular command in one block something that would be impossible in the other two formats.
- The table on the right shows the various commands used.

Word Sequential Format

TABLE 6.1 Commonly Used Word Addresses

| Address | Meaning |
|---------|---------------------------------------|
| F | Feed rate command |
| G | Preparatory function |
| I | Circular interpolation: x-axis offset |
| J | Circular interpolation: y-axis offset |
| K | Circular interpolation: z-axis offset |
| M | Miscellaneous commands |
| N | Sequence number |
| R | Arc radius |
| S | Spindle speed |
| T | Tool number |
| X | x-axis data |
| Y | y-axis data |
| Z | z-axis data |

And the first thing that comes to our mind is that you know, there are many such commonly used words in the word addresses format, there can be a feed rate command given by F, there can be a preparatory function command and there are many such G commands. Varying between G 012 almost G 99 about 100 also, which have been probably developed for different controllers which have different capabilities. So, it actually prepares as you how already seen the controller to do some carry outs on particular job and then there are commands related to circular interpolation in the x-axis y-axis and the z-axis. I am going to in detail explain about what these circular interpolations mean I J K corresponding to X Y Z.

There are miscellaneous commands like you saw in the case of coolant earlier and; obviously, the sequence number command of the identification number N which is there. And then there can be various other commands like the arc radius, the spindle speed, the tool number, the X, Y, and Z data on the axis which can be repeated. We which need not be repeated once it has been into the block unless there is a change they can be treated as a modal command I will keep on taking those values between the different lines or the block. So, in a way that is how the whole word sequential format has evolved and now you have pro pre much everything mentioned in place about the different words.

So, let us now probably close the topic right now for discussion and in the next session I would like to actually continue this further and talk about a basic international coding

standard also known as the American National Standard Institute NC based code for a controller and it is a one line option which defines the capability of a controller it comes with all the machines. And we will have to start making the NC programming before immediately after you look into this particular one line code which comes with controllers.

So, it gives you an idea of the controller capability. And in one line you can also find out what is the sequence in which you have to lay out these different commands like the N command, the XYZ command, the F command, the miscellaneous command, the speed command all those things. So that, you can have the controller to understand these lines in a certain sequence in a certain manner that you would like to address. So, in the next module we will like to take this up in more detail and do some realistic programming upon some parts. So, that you understand and get a feel of how you as a programmer can begin your journey into this NC programming world.

Thank you so much.