# Αριθμητικός Έλεγχος Εργαλειομηχανών

## Ενότητα 13: CNC Lathe

Δημήτρης Μούρτζης, Επίκουρος Καθηγητής
Πολυτεχνική Σχολή
Τμήμα Μηχανολόγων & Αεροναυπηγών Μηχανικών

# COMPUTER NUMERICAL CONTROL OF MACHINE TOOLS

Laboratory for Manufacturing Systems and Automation
Department of Mechanical Engineering and Aeronautics
University of Patras, Greece

**Dr. Dimitris Mourtzis**
**Assistant Professor**

Patras, 2015

# Objectives of section 13

- Describe the **difference** between a ***conventional lathe bed arrangement*** and a ***slant bed arrangement***, listing the advantages of the slant bed for NC
- Explain **axis movement** on a CNC lathe
- Describe the **method of toolholding** used on CNC turning machines
- Explain what a **tool offset** number is
- Describe two methods of **tool selection** used on CNC turning machines
- Describe how **spindle speed** is designated on gear head and variable speed lathes
- Explain how **feedrates** are specified on CNC turning equipment
- Define **TNR**

# Objectives of section 13

- Up to this point, the programming features of CNC mills have been discussed, but **numerical control** is used for **turning equipment** as well

- The **coding** will be a version used with **FANUC lathe controllers**, designed to be generic and so to illustrate the basic programming steps involved

- A numerical control lab in a school ***will have equipment that differs*** in one way or another from that presented here

- Students are advised to **familiarize themselves** with the codes used for the machines they will be using.

# Lathe Bed Design

- Older NC lathes, and those that have been converted to numerical control with retrofit units, look like traditional engine lathes. The ***lathe carriage rests on the ways***

- The ways are in the **same plane** and are **parallel** to the floor

- This arrangement allows the machinist to reach all the controls readily. Since the CNC lathe performs its operations **automatically**, this type of arrangement is not necessary.

- In fact, it is quite awkward since the operator will be busy with other responsibilities while the program is running and will ***not necessarily be there to brush the chips off the ways***.

- In a **conventional lathe bed** arrangement, the chips have nowhere to fall except on the ways. To overcome this problem, many CNC lathes make use of the **slant bed design**

# Lathe Bed Design

- On many NC lathes, the **turret tool post** is mounted on the opposite side of the saddle, compared to a conventional lathe, to take advantage of the **slant bed design**.
- The **slant bed** allows the chips to fall into the **chip pan** rather than on tools or bedways.
- Despite its odd appearance, the slant bed NC lathe ***functions just like a conventional lathe***

## Axis Movement

- The ***basic lathe has only two axes, X and Z***. Since the **Z** axis is always parallel to the spindle, **longitudinal** (carriage) travel is designated **Z**.
- The cross slide movement is **designated X**, since it is the primary axis perpendicular to **Z**. If it were possible to move the carriage up and down, that axis would be **Y**. There is, however, a potential **problem** with this arrangement.
- There appear to be **two Z axes**: ***the carriage movement and the tailstock movement.***

# Lathe Bed Design

FIGURE 1 Slant bed design

# Axis Movement

- To eliminate this problem, the ***tailstock is usually called the W axis*** on lathes with **programmable tailstocks**.

- **Programmable tailstocks**, which are **rear turret assemblies** on CNC equipment, are the third and sometimes fourth axes on more complex equipment.

- In such cases, the axes of the second saddle are usually designated **W** and **U**, with **W being saddle travel and U being cross slide travel**.

- There are some imported lathes on which the X-axis direction is **reversed**. The programmer must **determine** if such a situation exists **before** writing the lathe program

# Toolholders and Tool Changing

- Either a **rigid toolholder** or a **tool turret** is used to ***hold the tools*** on an NC lathe.

- When a **tool change is necessary**, the appropriate ***turret is indexed to the next tool*** needed.
- 
- Simple lathes use six-sided turrets; larger turning machines use eight-, ten-, and twelve-sided turrets.

- With the development of **robotics**, new tool changing and work handling schemes are appearing.

- To teach the basics of CNC programming, this text **will focus on nonrobotic tool change**.

# Toolholders and Tool Changing

- The **toolholders** used on NC turning machines are of very **rigid design**

- The tools used for turning are of the **carbide insert type**, made to much more exacting **tolerances** than conventional lathe insert tooling

- A tool change command in a turning program either ***changes the turret position*** or ***causes an automatic tool change***, depending on the type of machine used

# Toolholders and Tool Changing

## Automatic Tool Change

- In a CNC turning program for a machine with a **rigid toolholder**, **M06** is used to initiate an ***automatic tool change***. The **T** address is used (as it is in milling programs) to specify the desired tool. The T address also **calls up the tool offsets**. The format for automatic tool change is:

**M06Tn1 n2**

  - Where **M06** initiates the tool change,
  - **T** is the tool address,
  - **n1** is the tool number, and
  - **n2** is the tool offset number.

- Turret Position **T** is used in a similar manner with ***turret tool selection***. The format is:

**Tn1 n2**

  - Where the **first number** is the turret position and
  - the **second** is the tool offset number.

# Toolholders and Tool Changing

- Since one tool may be used in **several positions**, a turret position is used rather than a tool number. ***The turret position corresponds to the turret station number***.

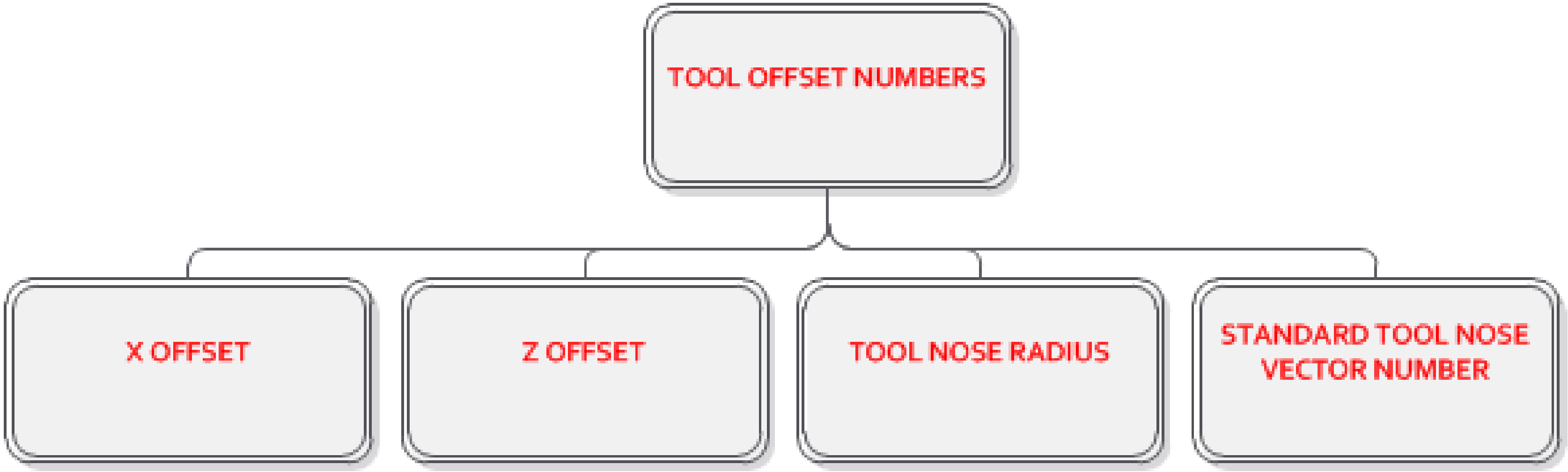
- **T01** will index the tool in **station one** into position.

- Some NC lathes can utilize more than one tool on a single station. It is possible, therefore, for **T0101** to refer to one tool and **T0111** to refer to another. This is referred to as ***piggybacking a tool station.***

- One other point should be kept in mind when changing tools**: the carriage (or tailstock) does not necessarily move to a tool change location**. It is often necessary, therefore, first to move the carriage or tailstock turret **out of the way** before making a tool change.

- It may also be necessary to program a dwell **(G04)** to **halt the program**, giving the tool time to index to ***position safely***

# Toolholders and Tool Changing



# Toolholders and Tool Changing

## Tool Nose Radius And Standard Tool Nose Vector Numbers

- The **tool nose radius** and **tool nose vector** numbers are optional. They are entered if using **cutter comp**
- Cutter diameter **compensation** is called ***tool nose radius compensation (TNR comp)*** on turning machines
- The **tool radius** tells the MCU the **amount of compensation** that is to be used. With NC machining centers this value was entered in a **comp register**
- **TNR comp** is utilized just as **cutter comp** was in section 10. It can be used to **program** the part line or **fine tune** the tool path to **compensate** for tool wear

# Toolholders and Tool Changing

- The **major difference** is that lathe tools are not completely **circular** as is a milling cutter

- To aid in proper **compensation** of the tool path and correctly identify alarm conditions, a **tool nose vector number** is entered in the register.

- **Tool nose vector numbers** tell the MCU the **orientation** of the tool nose.

- These directions are referred to as ***vectors***

- Each vector has a number associated with it that is **used to describe the tool orientation** to the MCU

# Toolholders and Tool Changing

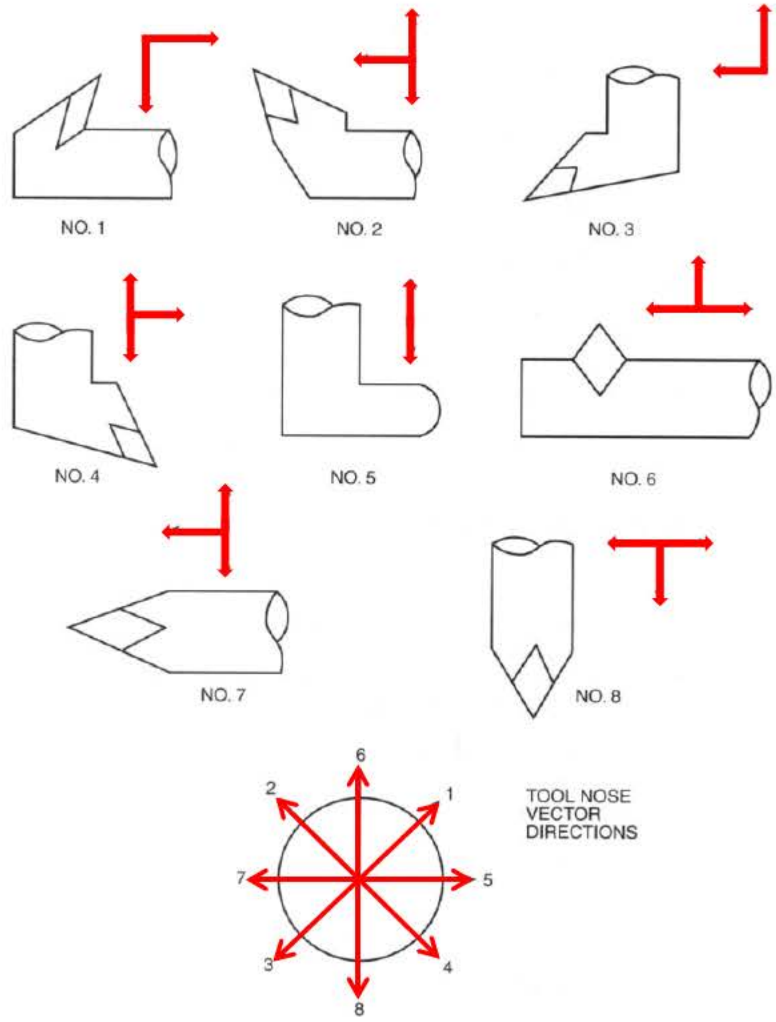



**FIGURE 2 Tool nose vectors**

- The **various directions** in which a tool may be **oriented**.

# Toolholders and Tool Changing

## Tool Edge versus Centerline Programming

- The **tool nose** may be programmed in one of two ways when **TNR comp** is not active:

  - by the **tool edge** or
  - by the **tool nose radius centerline**.

- Tool edge **programming** is adequate for **simple straight line cuts** where the part surfaces intersect each other at right angles.

- **Problems** are encountered, however, when ***angles and especially arcs*** are programmed this way

# Toolholders and Tool Changing

- If the tool edge is programmed, the **I** and **K centerpoints** of the illustrated arc must be **shifted**
- This results in a ***tool path that does not follow the desired arc*** exactly.
- The amount of error that is induced **depends on the size of the cutter** and the **radius of the arc**.
- In any case, tool edge programming **should not be used** when encountering **arcs** and **angles**
- Tool centerline programming is **identical** to the centerline programming done when **milling**

# Spindle Speeds

- **Spindle speed** is specified using an **S address**, just as in milling. On turning machines with a gear head design, the spindle speed is changed by shifting gears in the headstock
- On **gear head machinery**, there are usually **two or more** gear ranges
- An **M** function is used to select the gear range in which the desired speed is located
- **M40** through **M46** generally serve this purpose
- For **gear head** examples in this text:

  - **M40** will be used for low range
  - **M41** for mid range, and
  - **M42** for high range

# Spindle Speeds

- The following chart shows a sample of **speed ranges for gear head machines**. This chart is not for a particular machine but is **representative** of the type of **spindle speed spread** found on a machine

- Some CNC turning machines use a **variable speed drive** with which an **infinite number of speeds** are available between the highest and lowest speeds.

- In these cases, the speed is selected using the **S address** as it is in milling

| LOW RANGE | | | |
|---|---|---|---|
| 10 | 15 | 20 | 25 |
| 30 | 40 | 50 | 65 |
| 75 | 90 | 110 | 125 |

| MEDIUM RANGE | | | |
|---|---|---|---|
| 55 | 70 | 95 | 120 |
| 140 | 155 | 175 | 200 |
| 235 | 260 | 290 | 300 |

| HIGH RANGE | | | |
|---|---|---|---|
| 285 | 335 | 380 | 450 |
| 530 | 660 | 900 | 1200 |
| 1800 | 2100 | 2500 | 3000 |

- **Chart** shows a sample of **speed ranges** for gear head machines

# Feedrates

- With a CNC lathe, assigning **feedrates** is quite simple.

- A **G98** or **G94** code (depending on the controller) tells the MCU that the following feedrate is in inches per minute.

  - For example, **G98 F7** specifies a ***feedrate of 7 inches per minute***. A **G99** or **G95** in a turning program specifies a **feedrate in inches per revolution**.

  - For example, **G99 F.015** specifies a ***feedrate of .015 inch per revolution***

# Machine Origin and Work Coordinate Systems

- An NC lathe generally has a **fixed zero position** assumed by the executive program upon power-up. This position is known as the ***home zero*** or ***machine origin***.

- The physical location of this position **varies** from controller to controller and machine model to machine model.

- It is usually one of two locations: **X0 = centerline of the spindle**, **Z0 = the chuck mounting surface of the spindle**, or **X0 = extreme X + location**, **Z0 = extreme Z + location**.

- It is usually necessary to ***establish a zero point*** on the part different from the machine origin location. This position is called the ***work coordinate system or part zero***.

- There are two methods used to accomplish this.

# Machine Origin and Work Coordinate Systems

- The first method involves the use of an axis preset command—**G50**. The **G50 transfers** the zero point from the **home zero to the coordinates specified** with the command.

- The **format** for a **G50** command is:

G50 Xxx.xxxx  Zzz.zzzz

Where:

**G50** = the axis preset command

**xx.xxxx** = the X-axis distance to the part zero

**zz.zzzz** = the Z-axis distance to the part zero

# Machine Origin and Work Coordinate Systems

- A **G50** command is issued at the **start of each tool**. Since the programmer will not know the axis preset distances in advance, **zeros** should be used or some other prearranged value in the **G50** line.
  - The actual values will be ***determined by the setup person*** and edited in the control when the job is set up.

- The second method uses registers called ***work coordinates.*** These are registers in the MCU that tell the MCU the **distance** from **home zero to the part zero**.

- If a machine has more than one available work coordinate, **multiple zero points** may be used for complex programming.

- Another advantage to multiple work coordinates is the ***ability to have more than one program loaded*** in the MCU, each with its **own work coordinate**.

- This is a decided advantage when running several repeating jobs through a turning center.

# Machine Origin and Work Coordinate Systems

- Each **work coordinate** is called by a **G code**. If a program were to use four work coordinates, they would be selected by the codes **G54**, **G55**, **G56**, and **G57**.
- The first work coordinate (**G54** in this case) is the default work coordinate.
- This work coordinate is **automatically activated** upon power-up.
- If using only the default work coordinate, the **G code** may be omitted.
- The work coordinate values are **entered by the setup person** when the job is prepared.
- The programmer must **instruct** the setup personnel the position on the part of the ***part zero location***.

# Quicksetters

- A fairly recent development has been the use of ***quicksetters*—arms with tool sensors** on them.

- During job setup, the arm is lowered into position, the operator **jogs a tool** to the presetting position, and **touches it off on the sensor**.

- The **quicksetter** automatically ***sets the values of the work coordinate and the tool offset registers***

- ***Quicksetters*** can also measure the size of tools **before** cutting starts, and

- ***Check for tool damage or breakage*** during the machining operation.

# Summary

- CNC turning machines often use a **slant bed arrangement** to protect the machine ways from chips. Although different in appearance, the ***functioning of a slant bed and conventional bed machine is identical***
- There are ***two basic axes, X and Z***, on a CNC lathe. If the lathe has additional axes, they are generally designated **U** and **W**
- **TNR** stands for **tool nose radius** compensation. **TNR** is the **equivalent** in CNC turning to cutter diameter **compensation** in milling
- A **tool turret** or a rigid toolholder is used to **hold the tools** on an NC lathe
- **Tool offset** are entered into the MCU **prior** to running the program to **compensate** for minor setup adjustments
- A standard **tool nose vector number** is used to ***identify the orientation*** of a particular tool when using **TNR**

# Summary

- A **tool change command** in turning programs will either **change the turret position** or cause an automatic tool change, depending on the type of machine used

- The tool change format for turret changing is: **T n1 n2**

  Where **T** is the tool change command, **n1** is the turret position and **n2** is the tool offset

- The format for automatic tool change is : **M06 T n1 n2**

  Where **M06** initiates the tool change, **T** is the tool address, **n1** is the tool number, and **n2** is the tool **offset** number

- **Spindle speeds** are specified directly using the **S address**. On gear head machines, it is necessary ***to specify the gear range*** when selecting a range **outside** the active one.

- **Feedrates** on CNC lathes can be specified either in **inches per minute** (using **G94** or **G98**), or in **inches per revolution** (using **G95** or **G99**).

- To set a part at **X0/Z0** point, it is necessary to **transfer the machine origin** to the workpiece using a G code

# Vocabulary Introduced in this section

- Centerline programming
- Lathe bed
- Quicksetter
- Slant bed
- Tool edge programming
- Tool nose radius
- Tool nose vector number
- Tool offset numbers
- Tool turret
- Turret position

# End of Section

# Funding


- This educational material has been developed in the teaching duties of the respective educator.
- The Project "**Open Academic Courses at the University of Patras**" has funded only the reformation of the educational material.
- The Project is implemented within the context of the Operational Programme "Education and Lifelong Learning" (EdLL) and is co-funded by the European Union (European Social Fund) and national resources.

European Union
European Social Fund
OPERATIONAL PROGRAMME
EDUCATION AND LIFELONG LEARNING
investing in knowledge society
MINISTRY OF EDUCATION, LIFELONG LEARNING AND RELIGIOUS AFFAIRS
MANAGING AUTHORITY
NSRF
2007-2013
programme for development
EUROPEAN SOCIAL FUND
Co- financed by Greece and the European Union

# Reference Note

# License Note

# Notes Preservation

Any reproduction or modification of this material must include:

- the Reference Note
- the License Note
- the Notes Preservation statement
- the Third Parties’ Works Note (if exists)

as well as the accompanying hyperlinks.