

# Developing lathes postprocessor generating cycles of processing grooves

Tsvetan Kaldashev  
 Technical university of Sofia  
 Sofia, Bulgaria  
 email: kaldashev.cvetan@abv.bg

**Abstract**—This article explores the possibility of developing a post-processor generating circles for radial and facsimile grooves in PTC Creo environments using a generalized postprocessor G-POST and a specialized language FIL (Factory Interface Language). The proposed development approach can also be applied to post-processor generating cycles for rough and fine turning (G71, G72 and G70).

**Keywords**— CNC, postprocessor, canned cycle

## INTRODUCTION

After the development of instrumental transitions to process the workpieces using CAM systems, it is necessary to generate a CN program for a particular machine and control system. For this purpose, it is necessary to perform post-processing, i. e. to transform the CL Data file [4] into the NC program. Here is the CAD/CAM system PTC Creo and the generalized postprocessor G-POST. It is based on an existing postprocessor that creates another new postprocessor [3] for the particular machine and the control system. G-Post offers the opportunity to create post-processors for different types of machines: lathes, milling machines, turning centers, wire EDMs, etc. G-POST uses the information from CL Data File, created by PTC Creo, which is the only input for it. In cases where the created postprocessor does not have the necessary functions or cycles to manage the particular machine using the FIL (Factory Interface Language), they can be created and described with the tools and commands in the language.

As is known in CNC lathe machines are used repetitive cycles. They shorten the production time and the length of the NC program, and that the process can be managed directly by the CNC without the need to regenerate the NC program. The G74 cycle is used to process wide front groove with a narrower tool and holes whose axes coincide with the axis of rotation of the workpiece. Its scheme of operation is shown in Fig. 1. Cycle G74 has the following format [1]:

G74 R(e);  
 G74 X(U) Z(W) P( $\Delta i$ ) Q( $\Delta k$ ) R( $\Delta d$ ) F(f),

where e - retraction amount; X - X-coordinate of point B; U - distance between point A and B (incremental); Z - Z-coordinate of point C; W-distance between point A and pint C;  $\Delta i$  - distance in the X-axis direction;  $\Delta k$  - distance in the Z-axis direction;  $\Delta d$  - relief amount of the tool at the

cutting bottom; f – federate.

If the values in addresses X (U) and P are equal to 0, then this cycle is used for drilling.

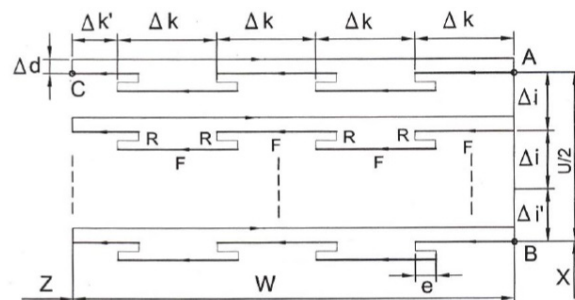


Fig. 1. Working scheme of cycle G74

Similar to Cycle G74 is Cycle G75, which is used to handle broad radial grooves with a narrower tool. The schema and cycle format are similar to the G74.

## I. DEVELOPMENT OF THE POSTPROCESSOR GENERATING CYCLES G74 AND G75

When developing postprocessors generating cycles G74 and G75 the following stages are [5]:

1. Create a Manufacturing User Define Feature (MUDF) that contains all of the technological information such as distance in the X-axis direction ( $\Delta i$ ), distance in the Z-axis direction ( $\Delta k$ ), federate (F), creating a tool path;
2. Describe the cycle in G-POST using the FIL language.

Manufacturing User Define Feature has the task of retrieving and writing the values necessary for defining cycles G74 and G75. To load the relevant parameters in the Tool Motions window, the cycle definition is written as a CL command:

For Cycle G74  
 CYCLE /FACE, RETRACT, & PULLOUT\_  
 DIST : FID\_224, X AXIS, & PECK\_  
 DEPTH:FID\_224,ZAXIS,&STEP\_OVER,FEED,&CUT\_  
 FEED:FID\_224

For Cycle G75  
 CYCLE/RADIUS, RETRACT, & PULLOUT\_  
 DIST : FID\_216, X AXIS, & PECK\_  
 DEPTH:FID\_216,ZAXIS,&STEP\_OVER,FEED,&CUT\_  
 FEED:FID\_216

The meaning of the parameters in the CL command for the two cycles are as follows:

PULLOUT\_DIST - retraction of the tool from the surface

Print ISSN 1691-5402

Online ISSN 2256-070X

<http://dx.doi.org/10.17770/etr2019vol3.4186>

© 2018 Tsvetan Kaldashev

Published by Rezekne Academy of Technologies.

This is an open access article under the Creative Commons Attribution 4.0 International License

being processed;

PECK\_DEPTH - depth of workflow cutout for chip breaking;

These parameters are standard for the PTC Creo system and are used in the development of instrument transitions. The transverse displacement of the tool both on the X axis and on the Z axis depends on the width of the plate and the nose radius. Different widths are used for different grooves sizes, meaning that the transverse displacement will change. For this purpose, this displacement is calculated from the tool width and the nose radius by dependence (1):

$$step\_over=0.6*(tool\_width-2*nose\_radius) \quad (1)$$

This dependency is recorded in the Relations window where the tool\_width and nose\_radius parameter values are internal parameters for the PTC Creo system and it automatically retrieves them from the selected tool which is doing the processing.

After recording the cycles as a CL command, a MUDF is created where, with its help, the instrumental transitions are created more quickly, enhances the productivity of work with CAM system, thus not passing through all stages of creating a technological operation (fig. 2).

In order to create tool transitions for grooves processing, only the workpiece, the machined geometry, the machine, the start and the end points (for automatic tool change) must be defined in advance. These several steps are sufficient to insert a MUDF that automatically creates the instrument transition and loads the cutting mode parameters. In this case, MUDF is used as a template that carries all the technological information, but the user can set (change) different cutting tools and modes than those in the template.

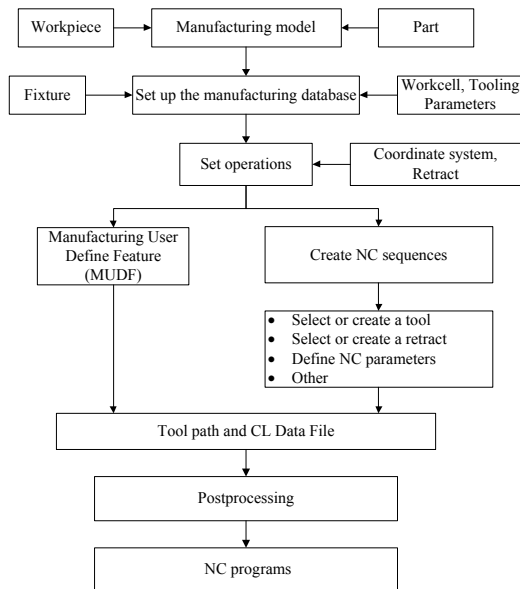


Fig. 2. Block diagram for creating a lathe operation using MUDF

For postprocessor development, G-POST and FIL are used to describe the cycles pre-recorded as a CL command. To process CL records CYCLE/FACE ..., CYCLE/RADIUS.... the following FIL record is used:

```

CIMFIL/ON,CYCLE
CTYPE=POSTF(7,4)
CASE/CTYPE
WHEN/ICODEF(FACE)
G74=1
RET=POSTF(7,6)
XDEPTH=POSTF(7,8)
ZSTEP=POSTF(7,10)
FED=POSTF(7,12)
XX=POSTF(26,5,5,1)
WHEN/ICODEF(RADIUS)
G75=1
RET=POSTF(7,6)
XDEPTH=POSTF(7,8)
ZSTEP=POSTF(7,10)
FED=POSTF(7,12)
XX=POSTF(26,5,5,1)
WHEN/OTHERS
XX=POSTF(26,5,5,0)
ENDCAS
XX=POSTF(13)
CIMFIL/OFF
    
```

This FIL record is used to take the values of the CL Data file for the respective cycle (G74 or G75) and set them to RET (retract distance), XDEPTH (peck\_depth), ZSTEP (step\_over), FED (cut\_feed). After assigning the variables values with command XX = POSTF (26,5,5,1) goes directly to a GOTO motion record. This command works like IF - THEN - ELSE or CASE - WHEN operators to find a specific subtype [2]. The GOTO movement record has saved the G74 and G75 output format. The motion track has the following structure:

```

CIMFIL/ON,GOTO
XX=POSTF(13)
IF(G74.EQ.1) THEN
G74=0
CURX=POSTF(7,6)*2
CURZ=POSTF(7,8)
XX=POSTF(14)
XX=POSTF(14)
LASTZ=POSTF(7,8)
LASTX=POSTF(7,6)*2
XX=POSTF(25,1)
DO/ENDO2,ILOOP=1,10
ILOOP=1 $$ infinite loop
DMY=POSTF(14) $$ Get next CL (TAPERD)
CLC=POSTF(7,2) $$ $FCLASS, 2d
parameter of record
SCL=POSTF(7,3) $$ $FSUBCL, 3rd
parameter of record
IF(CLC.EQ.2000.AND.SCL.
EQ.(ICODEF(CYCLE))) THEN
MOD=POSTF(7,5)
ILOOP=21 $$ Exit the loop
ELSE
XX=POSTF(13) $$ Process
ENDIF
ENDO2) CONTIN $$ End of DO loop
XX=POSTF(25,0)
IF(MOD.EQ.ICODEF(OFF)) THEN
    
```

```

POSTN/OUT,G,74,R,RET
POSTN/T,G,74,X, LASTX,Z, LASTZ,P,XDEPTH
,Q,ZSTEP,F,FED
ENDIF
ENDIF
IF (G75.EQ.1) THEN
G75=0
CURX=POSTF(7,6)*2
CURZ=POSTF(7,8)
.....
.....
POSTN/OUT,G,75,R,RET
POSTN/T,G,75,X, LASTX,Z, LASTZ,P,XDEPTH
,Q,ZSTEP,F,FED
ENDIF
CIMFIL/OFF
    
```

In the GOTO record, an initial check is made whether a G74 cycle command has been received and if the IF (G74.EQ.1) THEN condition is met, the CURX, CURZ, LASTZ, and LASTX variables are assigned respectively for start and end coordinates of the X and Z axis path. With the DO / Loop function, the remaining CL records are processed until a CYCLE record is found, then the MOD variable is assigned the cycle state value. An IF statement verifies whether the end of the cycle is reached and whether the condition is filled with a record of the type POSTN/OUT,G,74,R,RET

```

P O S T N / O U T , G , 7 4 , X , L A S T X ,
Z,LASTZ,P,XDEPTH,Q,ZSTEP,F,FED
    
```

a cycle G74 is generated. Similarly, a command G75 is generated.

After the postprocessor has been developed, a control program for the workpiece shown in Fig. 3. In Fig. 4 shows the program where it was generated for a 3mm wide tool.

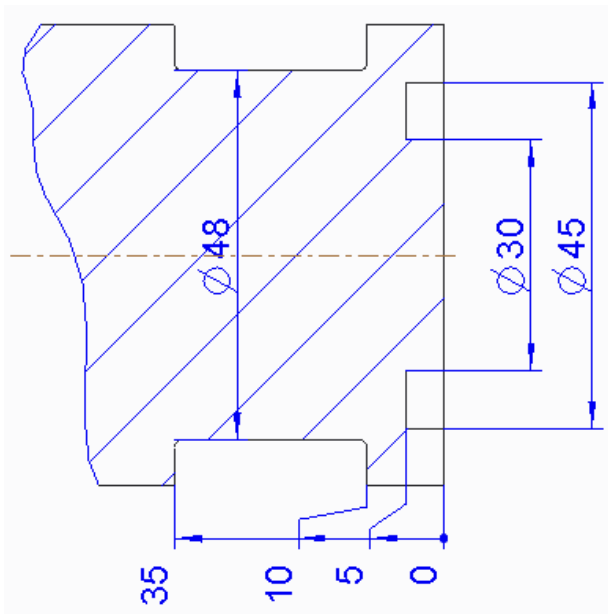


Fig. 3. Part

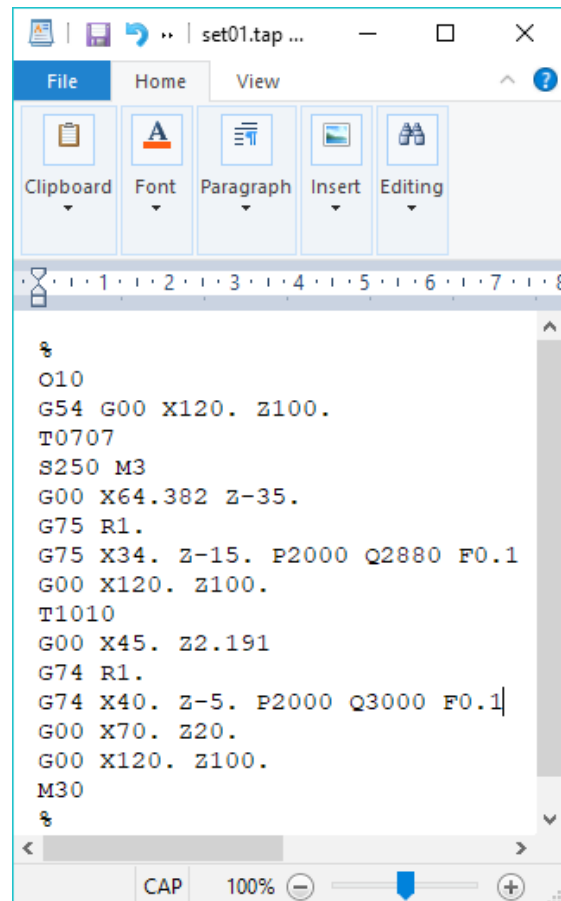


Fig. 4. Generated NC program

## II. CONCLUSION

1. Developed and tested experimentally postprocessor generating cycles for processing radial and front grooves, which has its practical advantages;
2. This approach to the development cycles for lathes can be applied to generate control programs using cycles G70 and G71, G76.

## III. REFERENCES

- [1] Doosan Puma series, Programming manual, MXTTPGE68;
- [2] FIL Help V6.6;
- [3] HJ. Patthi Bin Hussain, Zhu Yuesheng, Murthy Rallapalli and J.Mahmud, Development of Post Processor for Dual-Spindle Turning Center, Applied Mechanics and Materials, 2014, Vols. 541-542, pp 804-807, ISSN 1662-7482;
- [4] Čuboňová, N., Postprocessing of cl data in CAD/CAM system edgecam using the constructor of postprocessors, Manufacturing Technology, Vol. 13, Iss. 2, 2013, pp 158-164, ISSN 1213-2489;
- [5] Hadjiiski, P. I., Kaldashev, T. P., Creation of leather postprocessor generate repeated cycle in an environment of Pro/Engineer, International scientific conference-FIT 2012, pp. 245-249, 2012, Bulgaria, ISBN 978-954-438-994-9.