

DEVELOPMENT OF A POSTPROCESSOR FOR TURNING CENTER AND MULTI-TASK MACHINE WITH MULTI-CHANNEL CNC SYSTEMS

Tsvetan Kaldashev¹

Abstract: This article discusses the possibility of developing a postprocessor for turning center with 2 turrets and a multi-channel CNC system with NC program format characteristic of machine Mazak Integrex I series. Typical for these CNCs is the usage of two support programs - one for each turret. The CAD / CAM system PTC Creo is used to solve the problem, where tool transitions are developed for machining the workpieces. The postprocessor is software that translates the CL Data file i.e. turns it into a NC program with preparatory, technological, and supplementary commands to control the machine. G-POST, which is integrated into the PTC Creo CAD / CAM system, is used to develop the post-processor. To solve the problem, a specialized programming language FIL (Factory Interface Language) is used which, with its features and capabilities to work with files in ASCII code, achieves the ultimate goal - NC program in format characteristics suitable for machines Mazak Integrex I series.

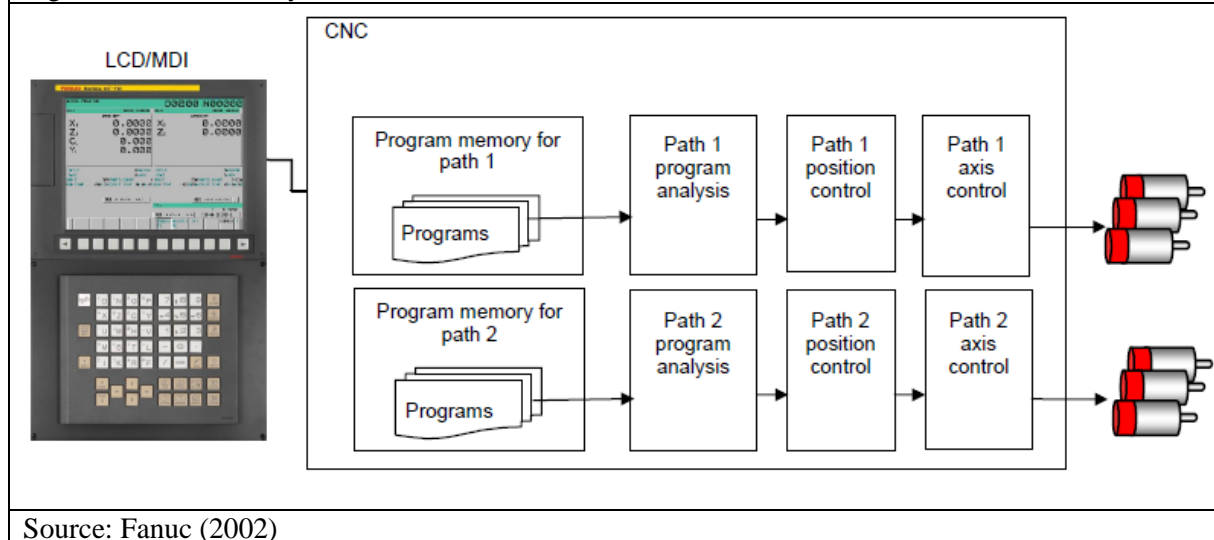
UDC Classification: 004.42, **DOI:** <https://doi.org/10.12955/pns.v1.117>

Keywords: multi-channel, CNC, postprocessor, CAD/CAM

Introduction

For conditions of medium to large-scale production, mill/turn machines with two independent turrets, multi-task machine with upper and lower turret in combination with sub spindle (Grigoriev, 2016) are used. The turrets - lower and upper are controlled separately by two-coordinate systems and these machines have four or more independently controlled axes. This class of machines uses 2-channel CNC systems. They are characterized by the fact that they control the two turrets simultaneously or independently of each other when machining a workpiece. In the programming of such machines, two control NC programs, independent of each other, are compiled for each turret, which are stored in the program memory for the respective path. (Fanuc, 2002) (Fig. 1).

Figure 1: CNC memory

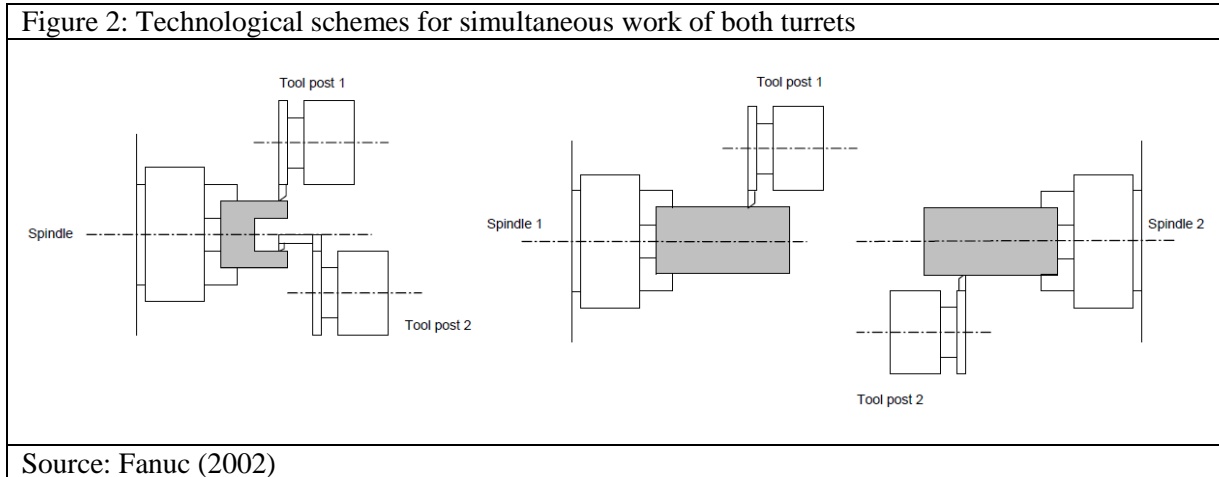


Source: Fanuc (2002)

In order to manage the two turrets while processing each workpiece simultaneously, the processing program for each turret is stored in the program memory. In automatic operation, this function selects turret program 1 and turret program 2 from the program memory for the corresponding turret. When the paths are activated, the programs selected for the respective tool paths are executed simultaneously and independently. In some cases, it is impossible to work with the two turrets at the same time and therefore M-commands are used. They are used to make one turret wait for the other during machining. When the M wait code is specified in the row for one turret during automatic operation, the other turret waits for the same M code to be set before the next line of the NC control program to continue. For Fanuc systems, commands M100 to M103 are used, for Mazak - M950 to M997 (Mazatrol, 2019). In Fig. 2 (Fanuc, 2002) shows typical technological schemes for simultaneous work of the two turrets.

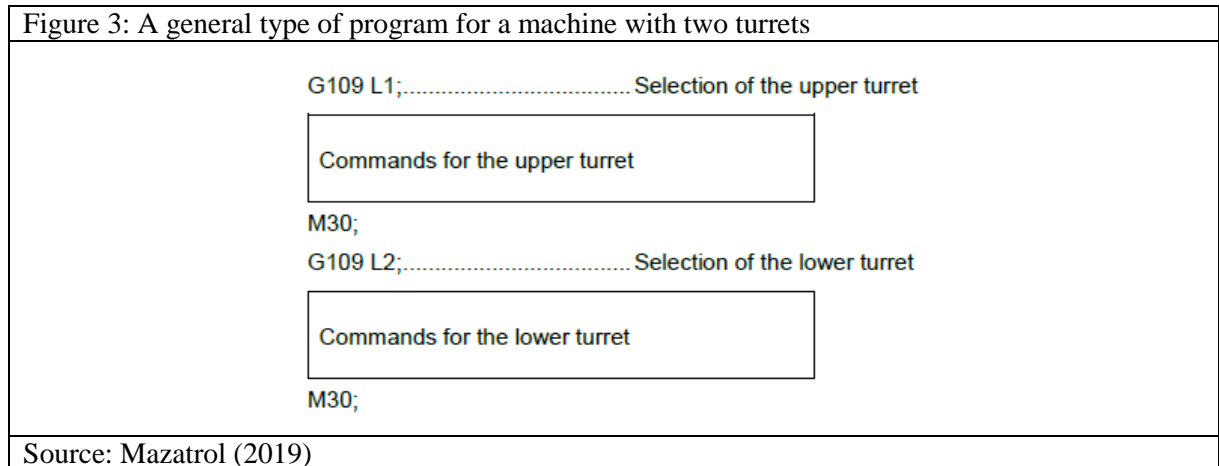
¹ Faculty of Industrial Technology, Technical University of Sofia, Bulgaria, kaldashev.cvetan@abv.bg

The programming of this class of machines is done using CAM systems. After the development of tools transitions for processing of surfaces of the workpiece, the CAM system creates CL Data file that records the coordinates of the trajectory of the tool to the machine coordinate system. To use the coordinates of the CL Data file processing detail it is necessary to perform post-processing i.e. translate the CL Data file into an NC program for a specific machine and CNC. The postprocessor is connected to the CAM system and can only work with it. The purpose of this article is to develop a postprocessor generating NC program in Mazak Integrex I series machine format, where programming can be done with the PTC Creo CAM system.



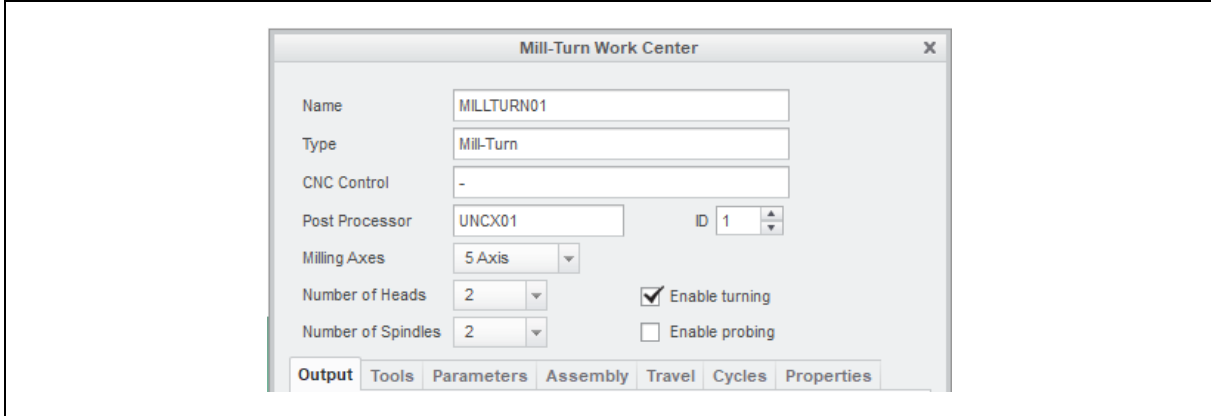
Structure of a postprocessor generating NC programs for machines with two turrets

In the development of tool paths for mill/turn and multi-task machines using CAM systems it is necessary, after generating the NC programs, for the postprocessor to create two programs - one for each turret. Fig. 3 shows a general type of program for a machine with two turrets (Mazatrol, 2019). To develop the postprocessor, a generalized postprocessor (Astinov,1987) G-POST and specialized FIL language (Factory Interface Language) is used to describe more specific features of the machine. With the FIL language, the following tasks can be accomplished: modify the generated NC program; add and change words in the postprocessor; read or write data from external ACSII code files, and connect to other applications (www.austinncc.com). It makes it possible to generate NC programs for CNC machines, where the machining of the workpiece surfaces can be performed without chip removal, for example, ball-burnishing process (Slavov, 2018; Slavov, 2017).



Before developing a tool path, it is necessary to set the machine, the number of controlled axes, the number of turrets, and the number of spindles (Fig. 4). When developing a tool path, it is necessary to indicate to the system which revolving head the processing is made - 1, 2, etc. After the development of the tool path, the coordinates of the nodes of the tool trajectory, the revolving head with which the machining is performed, the spindle (main or opposite) and many other information are written to the so-called CL Data File (cutter locations data file), which is the only one input for the postprocessor. On this basis the NC program for a specified machine and CNC system are generated.

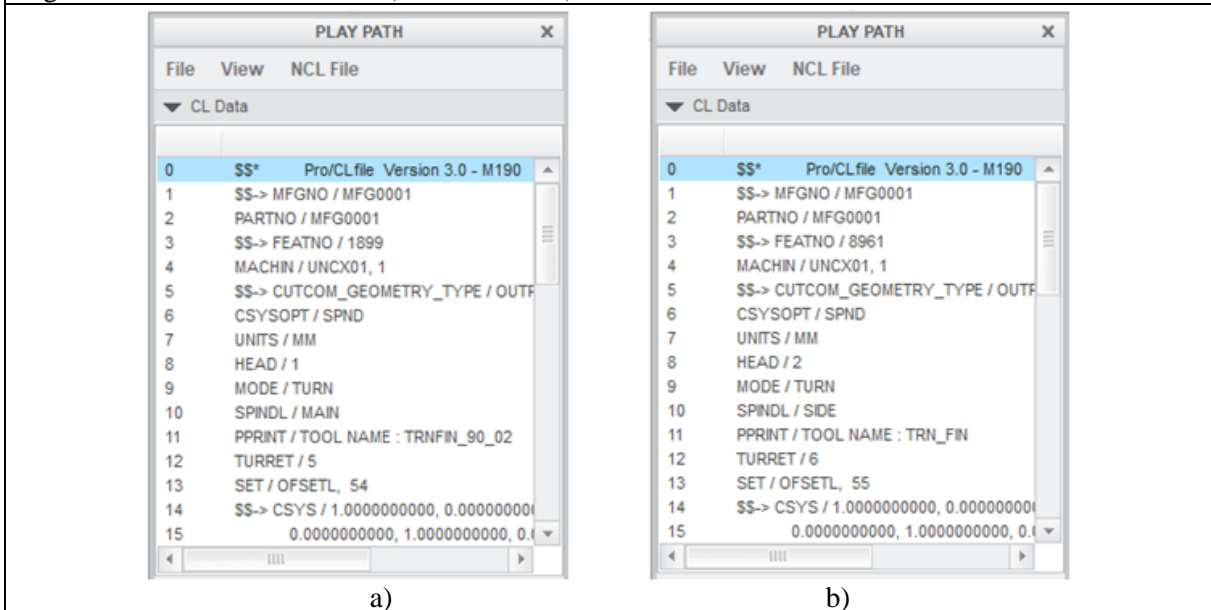
Figure 4: Set machine tool



Source: Author

Fig. 5 shows a portion of the CL Data file where processing is performed with HEAD 1 on the main spindle (Fig. 5a) and with HEAD 2 on the counter spindle (Fig. 5b). All CL commands (HEAD, MODE, SPINDL, TURRET, etc.) are standardized and used by the postprocessor while processing information from the CL Data file. Thus, after generating the NC program, these CL commands are transformed into G-, T-, S-, F- and other commands. For example the CL command TURRET/5 which is associated with calling of the tool with which the processing is performed, and thus after the post-processing, in the NC program, there will be a command T0505 (for lathes machines) or T05 M06 (for milling machines).

Figure 5: Part of CL Data File a) for Head 1; b) for Head 2

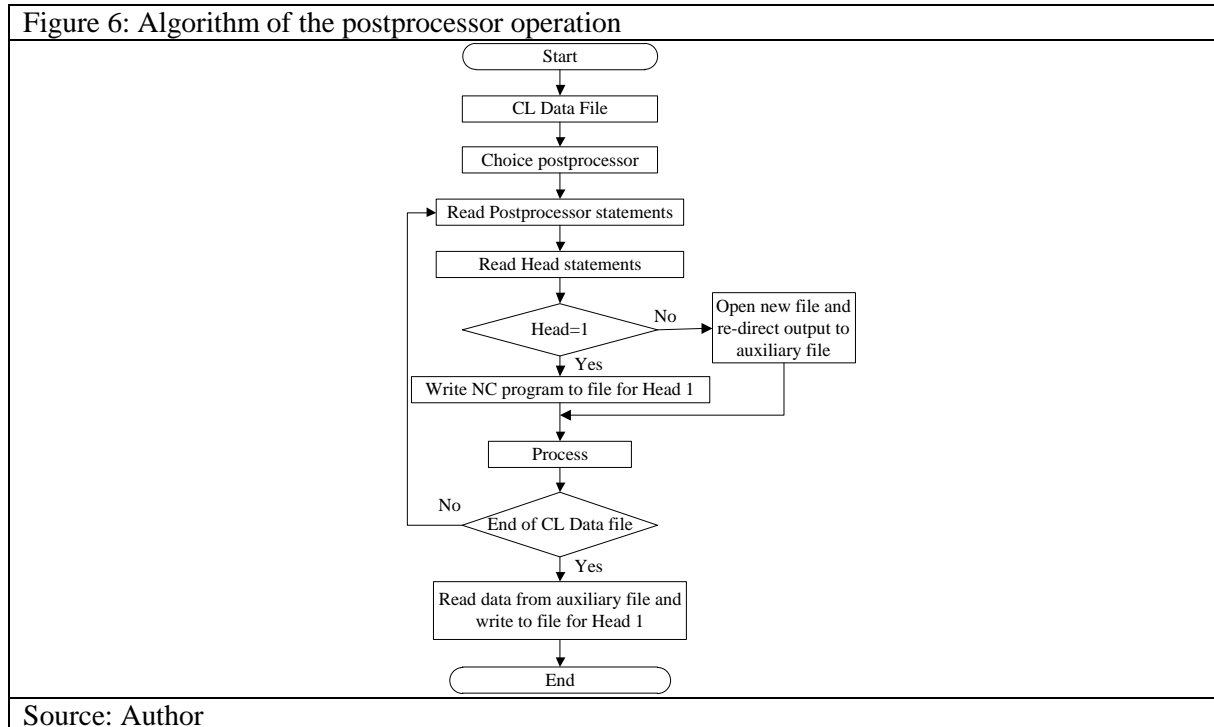


Source: Author

The algorithm used by the postprocessor is shown in Fig. 6. After selecting a postprocessor and reading the commands stored in the CL Data file, a check is made for the Head record. If processing is performed with Head 1, NC program is output to a file automatically created by G-POST. When processing with Head 2, a new auxiliary file opens which records the commands and coordinates for processing it. When the end of the CL Data file is reached, the commands and coordinates written to the auxiliary file (those for Head 2) are read and saved to the file containing the coordinates and commands for processing with Head 1. If the end of CL Data file is not reached, it reads the next CL record. The algorithm is repeated until the end of the CL Data file is reached.

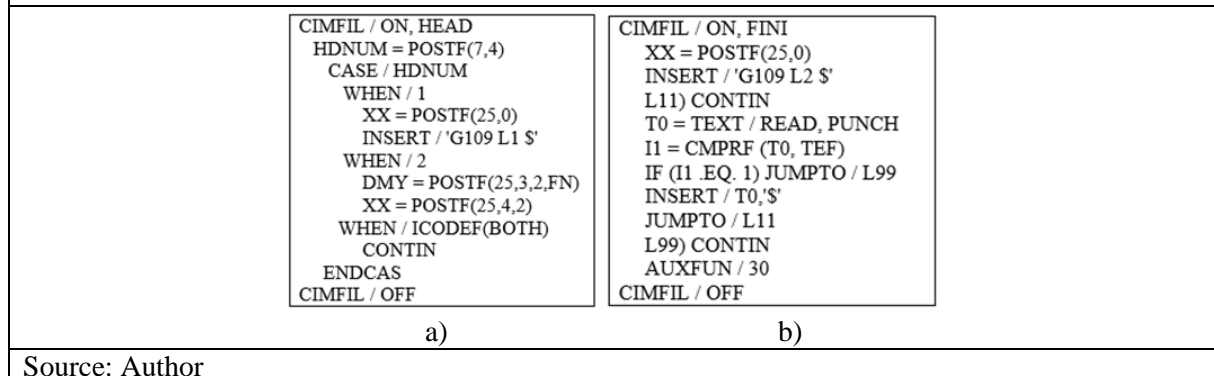
The processing of information from the CL Data file according to the algorithm of Fig. 6 is realized with the FIL language in the environment of generalized postprocessor G-POST. Head records are processed with the following FIL code (Fig. 7a)

Figure 6: Algorithm of the postprocessor operation



Source: Author

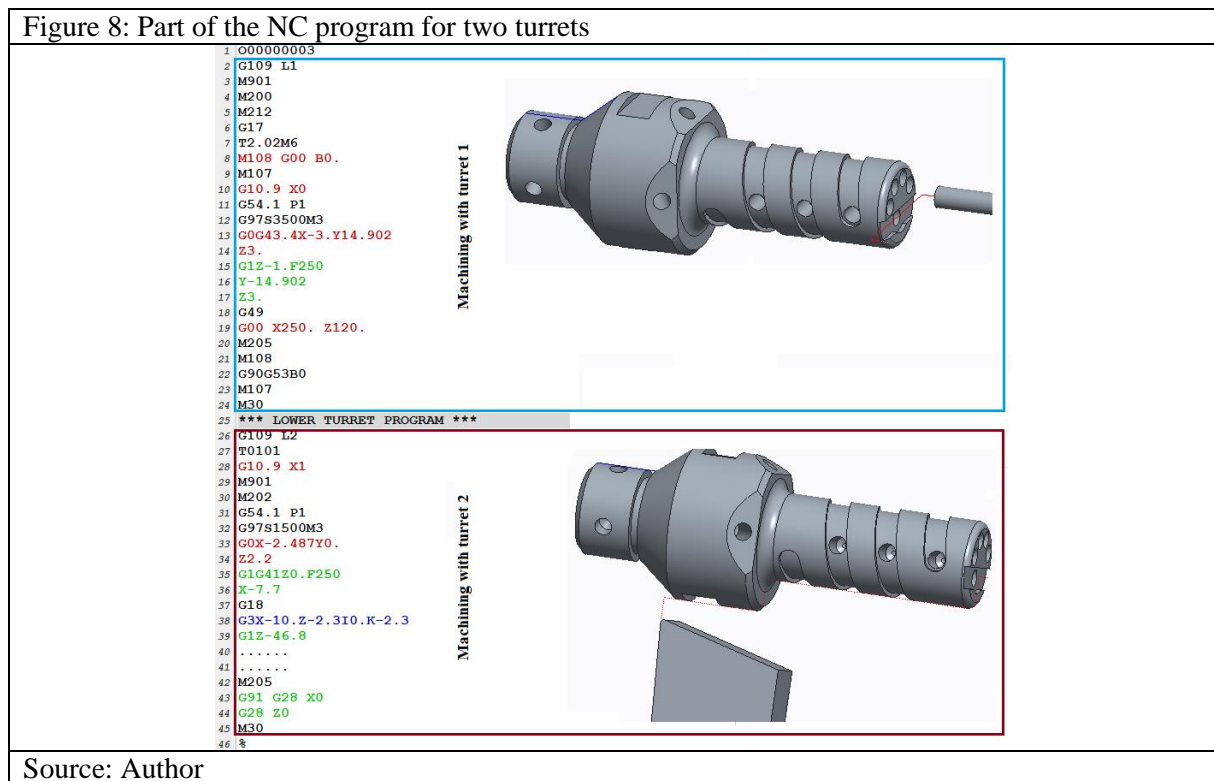
Figure 7: FIL code a) for Head records, b) for reading the supplementary FN file and adding it to the main file.



Source: Author

When processing the FIL Head record, a HDNUM variable is assigned the head (turret) number. The CASE operator (FIL, 2017) is used to check the HDNUM variable. It allows branching of the various FIL statements, depending on the value of the variable (string or real type). This is similar to the IF operator, but much more flexible. If HDNUM=1, i.e. currently working with HEAD 1, the command POSTF (25,0) sets the output of the program to be displayed in the current (main) file created by G-POST. If HDNUM=2, i.e. it is currently working with HEAD 2, then a command POSTF (25,3,2, FN) opens a supplementary file with the name assigned to the variable FN. Once this file is opened, with the command XX=POSTF (25,4,2) the second support program outputs to the supplementary file FN. It is also possible for the two heads to work simultaneously. Then the variable HDNUM=BOTH is used. Upon reaching the end of the CL Data file, as already mentioned, the Head 2 driver needs to be read from a supplementary file named FN and written to the main one (the Head 1 file). This is accomplished by the record shown in Fig. 7b. This FIL record uses unconditional transition operator JUMPTO (FIL, 2017). Reading a record from the supplementary file created with the command XX=POSTF(25,n) is performed with the READ, PUNCH command, where the value is set to variable T0. The CMPRF function is used to compare two strings. In this case, a comparison is made as to whether the current value of T0 is equal with TEF, where TEF=TEXT/ERROR \$ EOF'. The TEF value is a service variable indicating the end of the auxiliary file. If they do not match i.e. I1 is not equal to 1, then the current value of T0 is written to the main file. The JUMPTO/L11 entry returns to the beginning, where a new value is assigned again to variable T0. When variable I1 is equal to 1, it

means that the end of the supplementary file is reached, with JUMPTO/L99 continuing to execute in a row labeled L99. Record AUXFUN/30, outputs the M30 command to end the NC program. With the postprocessor developed in this way, a NC program for a machine with two turrets is generated. Fig. 8 shows a NC program for machine with two turrets with the structure shown in Fig. 3.



Conclusions

The proposed approach for generating NC programs for 2 turret machines can be applied to all CAM systems using a generalized postprocessor G-POST and the specialized FIL language. The structure of the NC program of the postprocessor developed in this way is valid for Mazak machines having two turrets. Considering the algorithm and the FIL commands used with minor changes, the structure of the NC program can be changed for other machines with two turrets and multi-channel CNC systems.

Acknowledgements

The author would like to thank the Research and Development Sector at the Technical University of Sofia for the financial support

Reference

- Astinov, I., & Todorov, N., (1987). Basic functions of a generalized postprocessor in a mechanical CAD/CAM system, *IV International Symposium "Automation and scientific instrumentation '87"*, 313-318;
- FANUC Series Oi & Oi Mate Model D (Lathe) – Operators Manual. (2002).
- FIL Version 6.7 Factory Interface Language. Programming Language for G-POST. (2017).
- Mazatrol SmoothX for Integrex I series, Programming manuals, Eia/Iso Program. (2019).
- Grigoriev, S., & Martinov, G. (2016). An ARM-based Multi-channel CNC Solution for Multi-tasking Turning and Milling Machines, *7th HPC 2016 – CIRP Conference on High Performance Cutting*, 525-528. doi: <https://doi.org/10.1016/j.procir.2016.04.036>
- Slavov, S. (2018). An algorithm for generating optimal toolpaths for CNC based ball-burnishing process of planar surfaces. *Advances in Intelligent Systems and Computing*, 365–375. doi: https://doi.org/10.1007/978-3-319-68324-9_40
- Slavov, S., & Iliev, I. (2017). A study on the variability of the pressing force during processing of regular shaped roughness by using new kinematical scheme for ball burnishing process. *MATEC Web of Conferences*. doi: <https://doi.org/10.1051/mateconf/201711202006>