Annals of DAAAM for 2010 & Proceedings of the 21st International DAAAM Symposium, Volume 21, No. 1, ISSN 1726-9679
ISBN 978-3-901509-73-5, Editor B. Katalinic, Published by DAAAM International, Vienna, Austria, EU, 2010
Make Harmony Between Technology and Nature, and Your Mind will Fly Free as a Bird
DAAAM Symposium

THE IMPORTANCE OF A POSTPROCESSOR IN DATA TRANSFER FROM APT FORMAT (CAD) IN ISO FORMAT USED BY HEIDENHAIN ITNC 530

GHINEA, M[ihalache]; DOBRESCU, T[iberiu]; NICOLESCU, A[drian] F[lorin] & ENCIU, G[eorge]

Abstract: *Researches in computer aided programming were initiated at MIT, followed by APT processor appearance in order to avoid the incompatibilities between CNC equipments and NC programming software. Since then a lot of versions were developed, improving CNC machining. According with APT philosophy, computer aided programming approach of a part machined on CNC machine tool is realized in two separate phases: processing and post-processing. First of all, geometrical definition of all tool path elements is done, followed by logical linking of these elements, and in the second phase all obtained data in previous phase will be converted in a suitable format for CNC equipment.*
Key words: *APT, postprocessor, CAD/CAM, NC programming*

1. INTRODUCTION

1.1 The postprocessor

The postprocessor is an essential element in mechanical manufacturing, when the machine-tool is equipped with a CNC. The postprocessor represents the link between 3D-CAM software and a CNC machine-tools. Over the world there are a lot of CNC types with personalized features, which are designed somewhat independent of CAM designers. A professional postprocessor can guarantee a manufacturing costs dropping by removing the 3D-CNC conversion errors that can produce fatal collisions and manufacturing errors.

A postprocessor is a computer program that consist in a lot of subroutines, and that transform the output data from processor into a useful data for CNC machine-tools. It can translate the specific language of the CAM software into ISO language for numerical control (for this case, from APT format obtained by CATIA software into ISO file used by Haidenhain iTNC 530 numerical control) (Ross, 1978). APT or Automatically Programmed Tool is a high-level computer programming language used to generate instructions for numerically controlled machine tools.

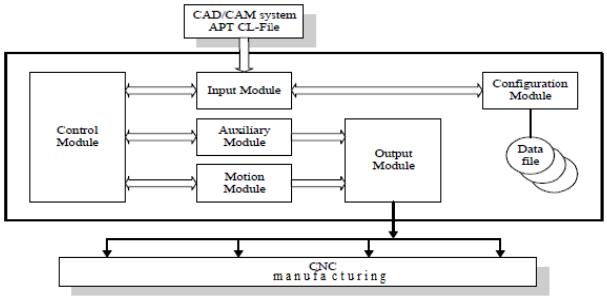

Fig.1. The postprocessor structure

The reason of this extra task is that data files obtained by CAM software commonly cannot be fully assimilated by the numerical control. The final result is a ISO data file used by the almost all known CNCs.

A such of postprocessor (figure 1) contains five major functional modules (Ryu, 2002): control function, input function, auxiliary function, translation function and output fc.

1.2 The postprocessor importance

Therefore the automation of the translation is a complex process due to the a large number of operations, of mathematical transformation etc.

In order to realize this translation, in fact to obtain that specific code recognized by the numerical control, the postprocessor must process a lot of operations, the most important ones of these being presented below:

(1) Reading ATP file;
(2) Switching to the coordinate system of the machine-tool;
(3) Switching from incremental coordinate at absolute coordinate;
(4) Checking of the machine-tool restrains;
(5) Developing feed and rotation speeds;
(6) Developing movement commands ordered by machine-tool/controller;
(7) Allowing linear interpolation, circular interpolation etc;
(8) Storing output data;
(9) Allowing output data printing.

The postprocessor can be responsible for other functions not just for neutral files translation what are typically for machine-tools. In addition a postprocessor can restore axes displacements, speed limitations, effective work times, data about tools, information's and decisions what can improve resources planning and can lead to the better manufacturing solutions.

The more sophisticated postprocessors can check the program before that CNC starts running. There are a lot of simple rules which postprocessor follows, and if these rules are not respected a visual or acustic warning appears (e.g. when the cutting tool was not chosen at the beginning of the program). After a simple checking of the program, next step is the correction stage. There are a lot of situations when a postprocessor can detect and correct an error. During cutting process can appear a lot of errors like: work cycles as active stages during the tool changing (in fact these must be temporarily canceled), the choice of wrong or inexistent axes (the postprocessor must select those axes that support speed), indication of unavailable cooling liquid (postprocessor must indicate the right available cooling liquid) etc.

The professional postprocessors offer in real time an overview of the CNC machine-tool activity, based on the forecasting a future events during the process. Thus it can be taken the best decision at the right moment, the CNC programmer using these information in order to improve the cutting optimization without any further interventions.

Also, the postprocessors can work in restricted conditions or even when some errors of CAM software or CNC appear. Generally, is much easier to change the postprocessor than the CAM software or to check the CNC controller.

1.3 The postprocessor structure

In figure 1 there is a postprocessor example, having in its basic structure the most important functions: control, auxiliary, input, output and translation. In figure 2 it is presented the data route from APT format into ISO format, accepted by CNC machine.

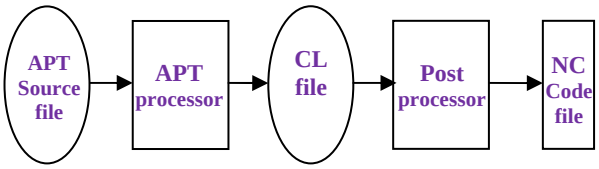

Fig.2. Data transfer from APT format into ISO format

APT source file is written by the user helped by the CAM software. *APT processor* verifies if there are errors in APT source file, errors related by the geometry and movement. *CL file* is a neutral file and contains data about tool position. *Postprocessor* converts the neutral file into ISO file.

2. APT PROGRAM INTO ISO PROGRAM VIA POSTPROCESSOR

It is important to explain the simplicity of the ISO language. ISO language has two types of commands: G commands and M commands. G commands refer to the directions which must be followed (feed, rotational movement, shift displacement on the left or on the right, function by the milling tool diameter). M commands refer to the auxiliary actions regarding the manufacturing process (main spindle start/stop, cutting tool changing, ending program etc.)

ISO commands can be grouped in two: instant commands and modal commands. An instant command is available just in the moment of its reading and a modal command is available until that command will be canceled.

The program sequence shown in figure 3 is extracted from an APT file, designed in the CAM module of the CATIA software. The program sequence shown in figure 3 is extracted from an APT file, designed in the CAM module of the CATIA software.

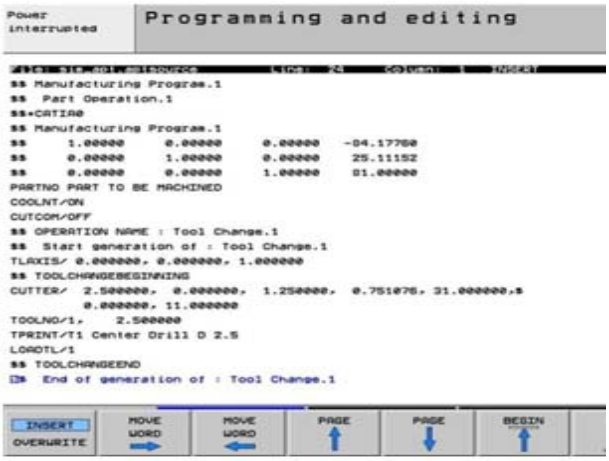

Fig.3. APT program

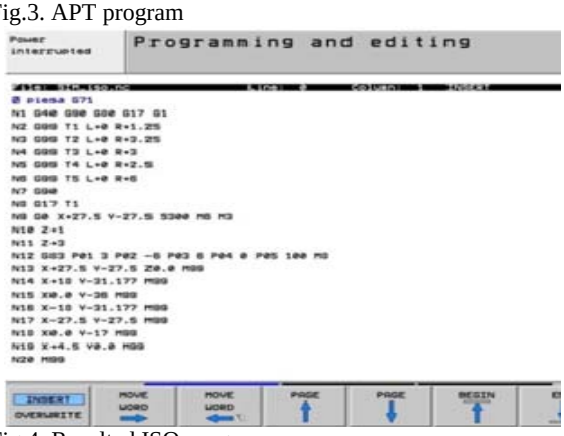

Fig.4. Resulted ISO program

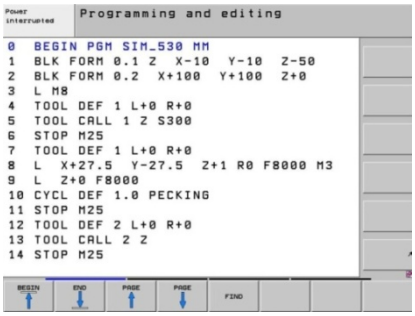

Fig.5. Haidenhain iTNC 530 program

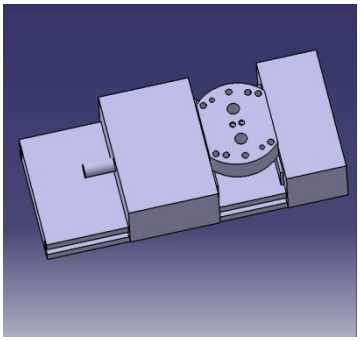
Fig.6. The part virtually designed and processed by milling

In pictures shown above there is the same sequence of the program in different formats (APT – figure 3, ISO, after postprocessing in figure 4, and directly written on Haidenhain NC – figure 5). In figure 6 there is a screen captured from CATIA with the processed part obtained by milling, process controlled by NC program, ISO version.

3. CONCLUSIONS

This paper presents the postprocessor role when cutting manufacturing becomes just a numerical controlled process. The diversity of the machine-tools, of the CNC controllers, of the CAM software and the geometry of the processed parts, ask for a faster NC programming. It is very well known that a CNC machine-tool works with a low level programming language, and this kind of program almost is impossible to be obtained by modern CAM software, which offers a high level programming language – APT format. Thus, a postprocessor, realized in a common programming language, can be the most useful solution to make real the communication between CAM software and CNC machine-tools.

4. REFERENCES

Lungu, I. & Miclosinam C. (2001) Stadiul actual al procedeelor de programare a masinilor unelte cu comanda numerica, http://www.uem.ro/universitatea/Facultatea_de_Inginerie/Catedra_de_DisciplineTehnologice/MICLOSINA%20CALIN/Lucrari%20stiintifice%20%20S.l.%20Ing.%20Calin%20MICLOSINA/Stadiul%20Actual%20Al%20Procedeelor.pdf, *Accessed*: 2009-04-22

Rochem S. (1996). La chaine CFAO, http://marauder77150.free.fr/cfao.htm, *Accessed*: 2010-02-12

Ross, D.T. (1978). Origins of the APT Language for Automatically Programmed Tools, *ACM SIGPLAN Notices*,Vol. 13, No. 8, August 1978, pp.61-99

Ryu, G.S. (2002). Implementation of web – based NC postprocessor builder, *KSIAM IT*, Vol.2, pp 91-98, ISSN:1226-943

Toader, S. (2008). Devenind realitate, http://www. Cad,report,ro/cadrep98.02/04.htm, *Accessed*: 2009-12-02