# APPLICATION OF THE STEP-NC STANDARD IN A COMPUTER NUMERICAL CONTROLLED MACHINING DEVICE

**Nazareno de O. Pacheco, nazareno.pacheco@gmail.com**[1]
**Eduardo Harbs, duka_guitta@hotmail.com**[2]
**Roberto S. U. Rosso Jr., rosso@joinville.udesc.br**[2]
**Marcelo da S. Hounsell, marcelo@joinville.udesc.br**[2]
**João Carlos E. Ferreira, jcarlos@emc.ufsc.br**[1]

[1]Universidade Federal Santa Catarina, Departamento de Engenharia Mecânica, GRIMA/GRUCON, Caixa Postal 476, Florianópolis, SC, 88040-900, Brazil
[2]Universidade do Estado de Santa Catarina, Centro de Ciências Tecnológicas, Lab. de Pesquisa em Automação de Sistemas, Joinville, SC, 89223-100, Brazil

***Abstract.*** *This paper shows the use of the ISO 14649 standard (also known as STEP-NC), which is the new standard for interfacing CAM/CNC manufacturing information systems. Currently the exchange of machining information in most manufacturing companies in the world is still carried out using ISO 6983, known as G-code, a standard developed for NC machines in the beginning of the 1960s This traditional standard language links the part design (CAD) with the requirements for machining operations generated by a CAM software. The ISO 6983 standard limits the portability of programs for three reasons: (a) the language focuses on programming the trajectory of the cutting tool tip relative to the axes of the machine instead of the machining processes concerning the part; ( b) the standard defines the syntax of the lines of the program, but in many cases it leads to semantic ambiguity; (c) CNC system vendors usually supplement the language with usually expensive extensions that are not considered in the limited scope of the ISO 6983 standard. In order to contribute to show the functionality of the STEP-NC standard in the control and direct drive of CNC machine tools, a software, hardware and a simple mechanical structure of a CNC milling/drilling machine were designed and implemented, and are described in this work. The software and hardware architecture is open, and adheres to the STEP-NC standard. The developed system uses ISO 14649 files for programming the machine without the use of legacy languages. This allowed validating the use of the standard to drive a machine, showing its flexibility and efficacy for information exchange and machine-tool control.*

***Keywords****: Computer Numerical Control; CAD/CAM; STEP-NC Standard*

## 1. INTRODUCTION

As the first numerically controlled (NC) machines, the current Computerized Numerical Control (CNC) machines continue to use the ISO 6983 programming standard (ISO 6983), known as G-code, which is based on the description of tool movements. Nowadays the exchange of programs between different CNC machines is virtually impossible, unless they have drivers of the same manufacturer and with a similar configuration. For decades these problems have hampered the integration of manufacturing.

However, in the mid-1990s, a new standard (ISO 14649) informally known as STEP-NC has been developed by manufacturers, users and academic institutions around the world seeking to provide an information model for a new generation of intelligent CNCs (Rosso and Newman, 2003). One feature of the STEP-NC standard is that its data model is based on object-oriented features, trying to help eliminate the disadvantages inherent to the ISO 6983 standard.

Considering the above scenario, this paper aims to show that the use of ISO 14649 is feasible for the direct control of machine tools, i.e. without the use of languages such as ISO 6983.

## 2. ISO 10303 (STEP) AND ISO 14649 (STEP-NC) STANDARDS

CAD (Computer Aided Design), CAPP (Computer Aided Process Planning), CAM (Computer Aided Manufacturing), and CAE (Computer Aided Engineering) systems are widely used in product development, and information exchange between these systems is complicated by several factors. To overcome these difficulties and provide a neutral system for registering product information, ISO (International Standards Organization) has developed a set of 10303 standards, known as STEP (Standard for Product Data Exchange Model). And later the ISO 14649 was developed, known as STEP-NC (STEP - Numerical Control), which seeks to build a neutral pattern of data to be used among CNC machines (Newman *et al.*, 2003)

### 2.1 ISO 10303 (STEP) Standard

In design and manufacturing, many systems are used to manage technical product information. Because each system has its own data format, the same information must be updated often in multiple systems, resulting in data redundancy and errors in interfacing. In order to integrate CAD, CAPP, CAE and CAM, the ISO 10303 standard was developed, seeking to standardize products data models (ISO 10303). This standard provides support to product development, so that information remains consistent and integrated throughout the product lifecycle, as shown in Figure 1.

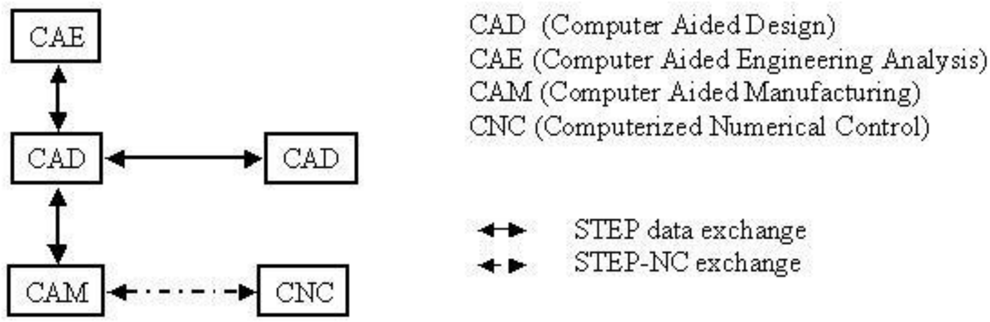


Figure 1. Types of systems that use the STEP/STEP-NC standards (STEP Tools Inc., 2008)

According to Loffredo (2008), the ISO 10303 standard allows the exchange of data through a neutral format between the programs that are used for developing engineering products, as illustrated in Figure 2.

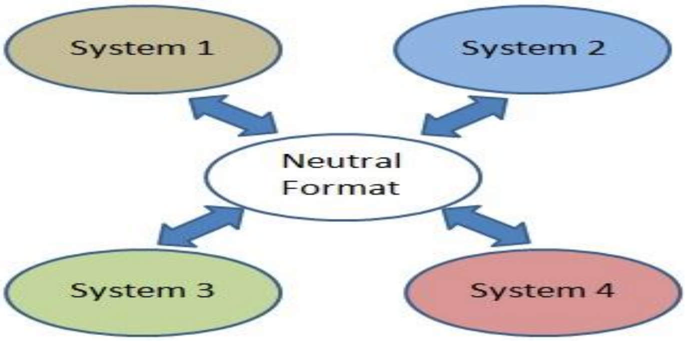


Figure 2. Communication between systems that use a neutral file (Adapted from Zeid, 1991).

## 2.2 ISO 14649 (STEP-NC) Standard

Even with the use of the STEP standard, manufacturers and users are still faced with problems in integrating CAD, CAPP, CAM and CNC. It was the mid-1990s that a new data interface for CNC machine tools based on ISO 10303 was developed, which is referred to as ISO 14649 or STEP-NC (Rosso and Newman, 2003). This standard seeks to provide an object-oriented data model to a new generation of intelligent CNCs. Its innovation, compared to the standard currently used in NC programming, ISO 6983, is the ability to represent part geometry, using the concept of features, providing a more detailed data model with information about geometry, tools to be used, and operations to execute. In addition, STEP-NC enables bi-directional flow between CAD↔CAPP↔CAM↔CNC, allowing changes in product design at any stage in its manufacturing cycle, causing no loss of information. This makes it much easier and faster to update or modify products, as shown in Figure 3, which presents a comparison between ISO 6983 and ISO 14649, where the data flow on the left-hand side is unidirectional, not allowing the flexibility of the manufacturing, whereas on the right-hand side the data flow is bi-directional, allowing process flexibility.

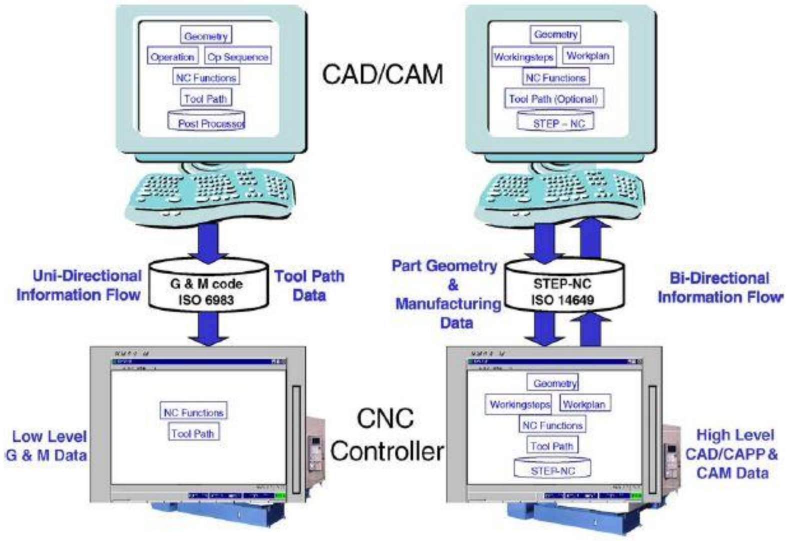


Figure 3. Comparison between the ISO 6983 and ISO 14649 standards (Allen *et al.*, 2003)

Xu and Newman (2006) point out the following benefits from the use of the STEP-NC standard:

- STEP-NC provides a complete and structured data model, linking the geometric and technological information, thus no information is lost along the different stages of the product development process;
- Its data elements are adequate to describe the NC task;
- The data model can be extended to facilitate its utilization (through conformance classes) seeking to combine with the specific abilities of CAM, shop floor control or NC.
- The machining time can be reduced because the intelligent optimization can be built in the STEP-NC controller;
- The post-processing mechanism will be eliminated, since the interface will not need information from a specific machine;
- The machine tools will be safer and more adaptable, since STEP-NC is independent of machine manufacturers;
- In the case of modifications on the shop floor, a feedback signal is sent to the design department, and this would be the case of a bi-directional flow of information between the CAD, CAM and CNC systems;
- The XML file contains the information necessary for the Standard manufacture of the product, which allows the distributed manufacture via the Internet.

The controllers that adhere to STEP-NC are generally classified in three types, as shown in Figure 4 (Suh *et al*., 2003):

- Type 1: These controllers use STEP-NC for conventional control via post-processor;
- Type 2: These controllers have a STEP-NC interpreter as part of the CNC; this type of controller generates the motion corresponding precisely to the STEP-NC file, without any intelligence to establish the machining strategy.
- Type 3: These controllers can execute in an intelligent and autonomous manner the machining task based on the information in the STEP-NC file.

The utilization of data structures adherent to STEP-NC enables the use of integrated information models, and not subject to interpretation errors. This approach is especially suitable for integrating CAD, CAPP, CAM and CNC through product and manufacturing data models. The use of STEP compliant data from part design to manufacture should significantly reduce the time for product launch. It should also reduce costs due to reduced paper circulation, less programming errors, better use of computing and manufacturing resources. Over time, more software companies will adopt this technology, seeking to improve data quality, and more efficient information exchange compared with current procedures. Another key factor in the use of standards is the reduction in the time to create and develop parts, thus reducing significantly the manufacturing costs (Rosso and Newman, 2003; Suh et.al, 2003; Nassehi *et al.*, 2006).

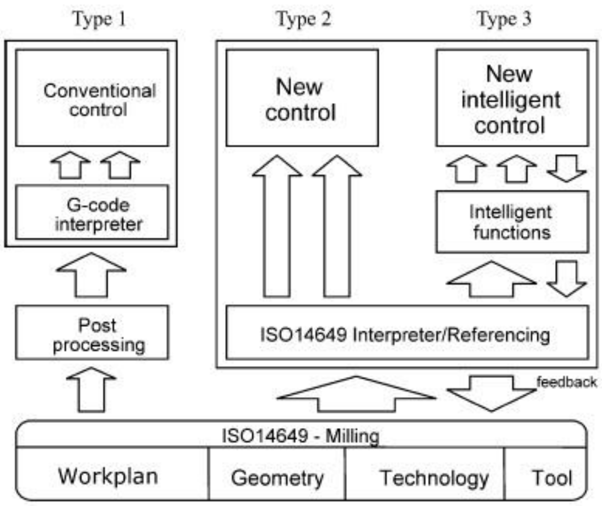


Figure 4. Types of STEP-NC controllers (SUH *et al.*, 2003)

The ISO 14649 standard is an ARM (Application Reference Model), while ISO 10303-238 is an application protocol (AP) that implements the AIM (Application Interpreted Model) in a STEP context, allowing better integration with other application protocols of ISO 10303. The contents of the AP-238 are more detailed and complex, resulting in greater difficulties in implementing the ISO 14649 (Feeney *et al.*, 2003). For this reason, in this work we chose to use ISO 14649.

Figure 5 shows an excerpt from a STEP-NC program (in the part 21 format of the STEP standard) for testing the prototype. It can be noticed from the file excerpt that it corresponds to a part with thirteen holes, and the necessary information for machining each hole is included, among which is the hole diameter, tool material, tool type, setup, etc.

```
ISO-14649-21;
HEADER;FILE_DESCRIPTION(('Nazareno Pacheco, Roberto Rosso,
Eduardo Harbs ISO 14649-11 FURO1',
                 'PROGRAM WITH A ROUND_DRILL'),'1');
FILE_NAME('FURO1.STP');
FILE_SCHEMA(('MACHINING_SCHEMA','MILLING_SCHEMA'));
ENDSEC;
DATA;
#1= PROJECT('EXECUTE FURO1',#2,(#4),$,$,$);
#2= WORKPLAN('MAIN WORKPLAN',(#11,#12,#13,#150,#151,#152,#153,
#154,#155,#156,#157,#158,#159),$,#8,$);
#4= WORKPIECE('SIMPLE WORKPIECE',#6,0.10,$,$,$,($,$,$,$));
#6= MATERIAL('ST-50','STEEL',(#7));
#7= PROPERTY_PARAMETER('E=200000N/M2');
#8= SETUP('SETUP1',#71,#62,(#9));
#9= WORKPIECE_SETUP(#4,#74,$,$,());
#11= MACHINING_WORKINGSTEP('WS DRILL HOLE1',#62,#17,#20,$);
#12= MACHINING_WORKINGSTEP('WS DRILL HOLE2',#62,#18,#20,$);
#13= MACHINING_WORKINGSTEP('WS DRILL HOLE3',#62,#19,#20,$);
.................................................................
#17= ROUND_HOLE('HOLE 1',#4,(#20,$),#81,#64,#58,$,#26);
#18= ROUND_HOLE('HOLE 2',#4,(#20,$),#82,#64,#58,$,#26);
#19= ROUND_HOLE('HOLE 3',#4,(#20,$),#83,#64,#58,$,#26);
.................................................................
#20= DRILLING($,$,'DRILL HOLE1',10.000,$,#44,#45,#41,$,$,$,$,
$,#46);
#26= THROUGH_BOTTOM_CONDITION();
#31= TWIST_DRILL(#32,2,.RIGHT.,.F.,0.840);
#32= MILLING_TOOL_DIMENSION(1.000,$,$,$,$,$,$);
#41= MILLING_MACHINE_FUNCTIONS(.T.,$,$,.F.,$,(),.T.,$,$,());
#44= MILLING_CUTTING_TOOL('SPIRAL_DRILL_D1MM',#31,(#
126),90.000,$,$);
ENDSEC;

END-ISO-14649-21;
```

Figure 5. Example of a STEP-NC program

## 3. OPEN ARCHITECTURE CNC ADHERENT TO THE STEP-NC STANDARD

In order to show the functionality of the ISO 14649 standard in its direct application to a CNC device, the prototype shown in Figure 6 was built. The mechanical structure consists of parts obtained from two printers, a scanner and a doorway for fastening one of the axes.

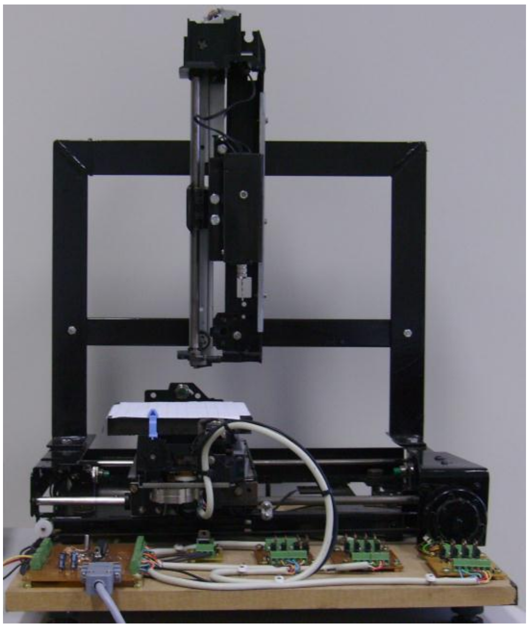

Figure 6. CNC drilling device adherent to STEP-NC

The drilling CNC device has three axes, and the Z axis corresponds to a portable drill that was mounted on one of the printers' structure. The part to be drilled is fixture on a horizontal table that has movements along the X and Y directions. The architecture consists of an open CNC, as shown in Figure 7. The building blocks of this architecture are:

- **Compiler**, which is responsible for interpreting the STEP-NC file;
- **Table**, which contains and organizes the machining information, such as: types of tools, dimensions and positions of the holes;
- **Microcontroller**, which receives the machining information, and forwards the commands to the drives of the machine motors;
- **Serial communication, motors drives, and machine drives**, which are respectively responsible: for the information exchange between the computer and the control board, for driving the motors of the machine axes, and for driving the machine's spindle axis.

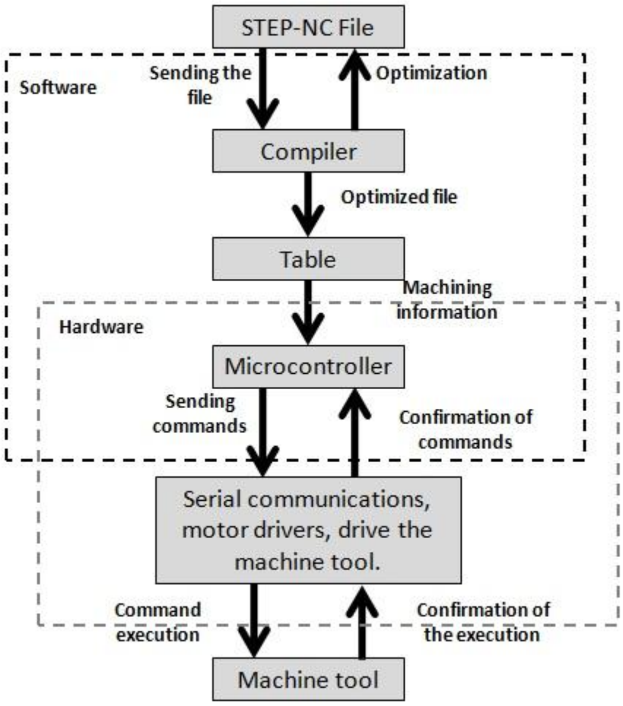


Figure 7. Open architecture developed in this work

According to Aho *et al.* (1995), a compiler is a program that reads a file written in a language called the source language and translates it into an equivalent file in another language, called the target language. In this work a compiler was developed to read a file in ISO 14649 (Part 21) and translate it into a table containing the information necessary for machining the part. This compiler was developed in the Java language.

After opening the STEP-NC file, the compilation is performed. The steps to this process are: lexical analysis, syntactic and semantic analysis, code optimization, and target code generation.

In the first stage the analysis of the code characters and the storage of the tokens are carried out. These tokens are the symbols that refer to the source language. In the second stage the sequence of tokens and file structure are checked. If an error is found, the program displays an error message to the user informing its origin. As this work is restricted to the drilling process, the compiler was built using a minimum number of entities from the standard, reducing the number of symbols for the analysis. Of course in future works, when other processes will be considered, other entities in the standard will be included in the system.

After reviewing the entire file, the recognition of the information necessary for machining is performed. This information includes the X and Y coordinates of each hole, tool diameter, tool name, hole diameter, feed control, spindle speed, and reference plane. Figure 8 shows a flowchart with the stages of the developed compiler.

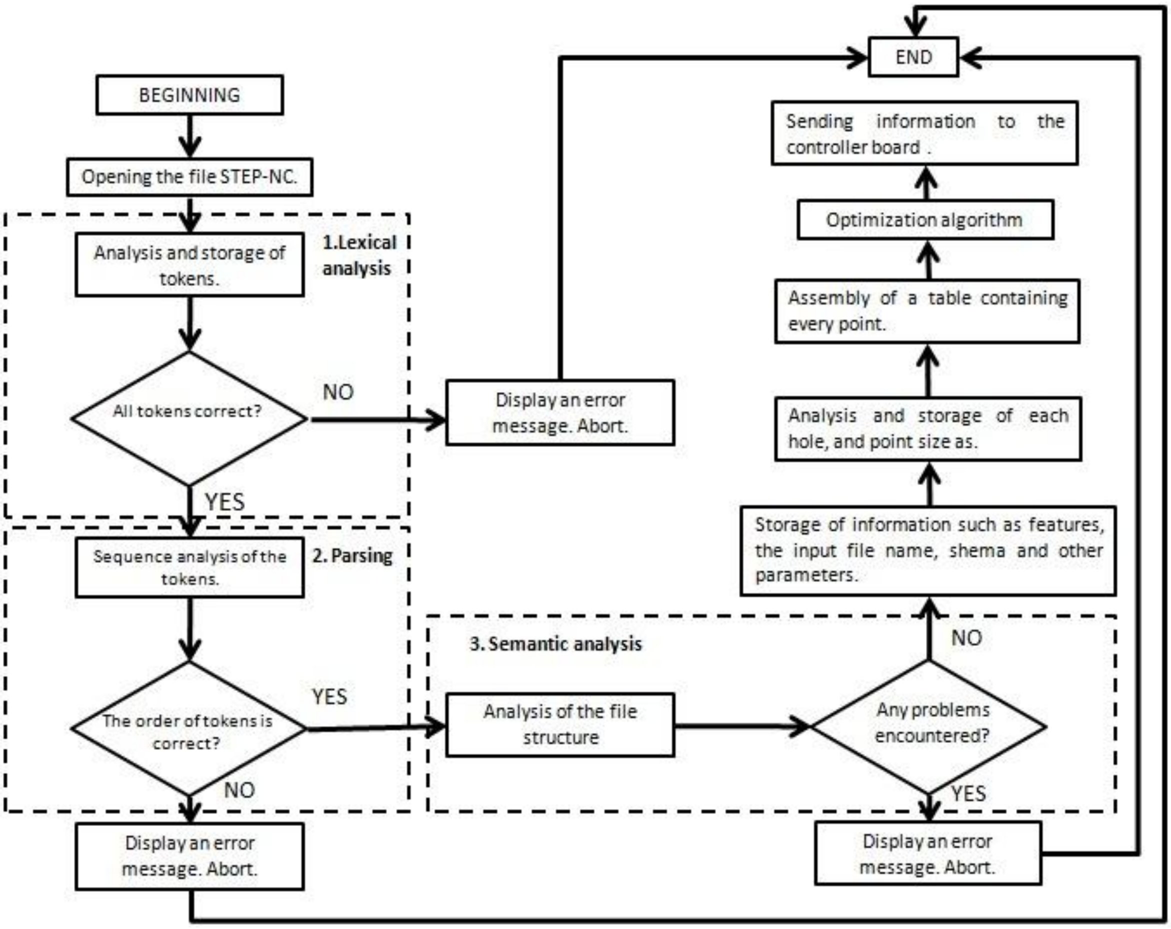


Figure 8. Flowchart containing the steps of the developed compiler

The compiler's function is to read an ISO 14649 file and translate it into a table containing the information necessary for machining the part. Along with the mapping of STEP-NC file, an optimization algorithm was implemented, which seeks to generate the best path to machine the part. This algorithm determines the point that is closest to the machine zero, and the next point is referenced by the former. A message with the modification resulting from the optimization of the work plan is shown on the screen when file is compiled. It is up to the user to accept or not to change the line of the Workplan, which is the element of STEP-NC responsible for the work plan of the part. Figure 9 shows the interface of the compiler with the message from the optimization algorithm. After the user makes his/her choice, a table is generated by the compiler containing the hole positions, hole diameters and some more machining information.

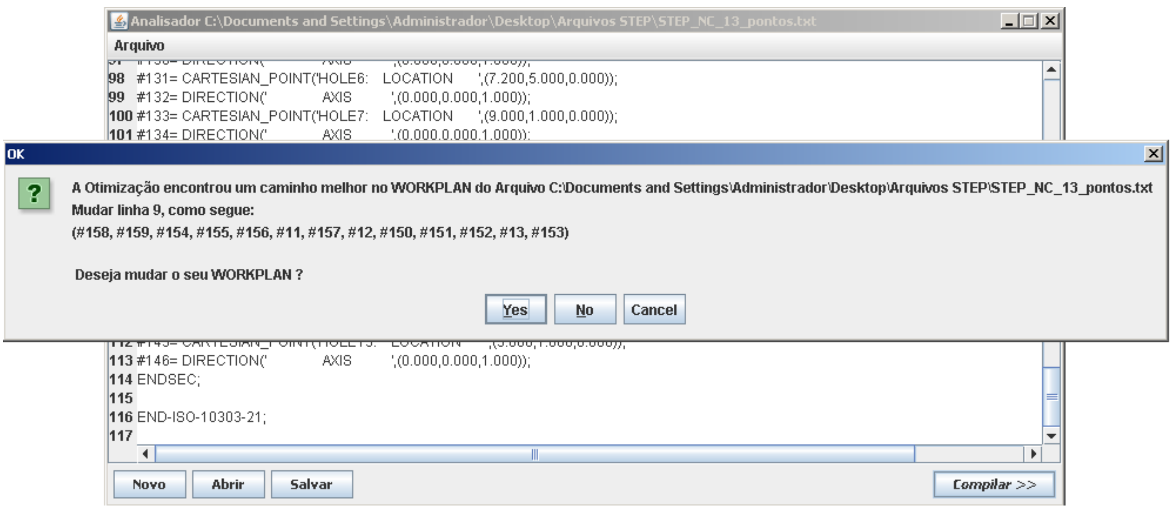


Figure 9. Developed compiler and the window for choosing the alteration of the Workplan seeking to optimize the drilling of the holes

In order to input data and perform machine interaction, a software interface was developed in C++. This interface is responsible for exchanging information with the micro-controller board through the PC's serial port, besides providing a field for machine settings. In this interface the table generated by the compiler is loaded, and then it asks the microcontroller to run the table as shown in Figure 10.

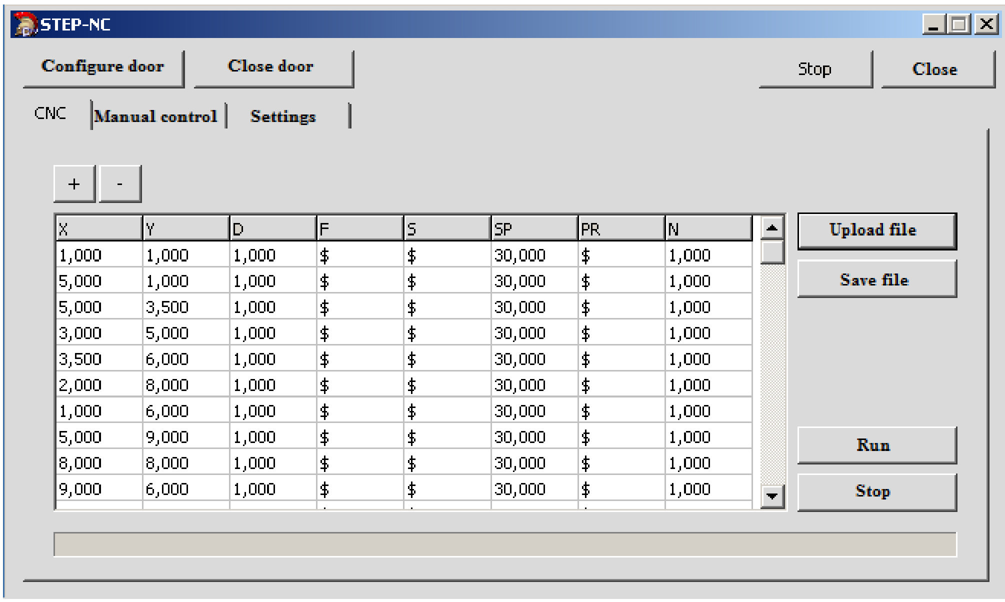


Figure 10. Interface for loading and executing the STEP-NC file

Figure 11 shows the configuration screen of the machine. On this screen the user can configure the working area, the speed of each axis with its respective acceleration ramp, and the precision of the steps in the X, Y and Z axes, and in the approximation of the cutting tool in the Z axis. For example, when a 130x130mm working are is configured, which is the case with the prototype, if the STEP-NC file has any point outside this configured work area, an error message is output. These configurations ensure the adaptability and interoperability of the implemented system.

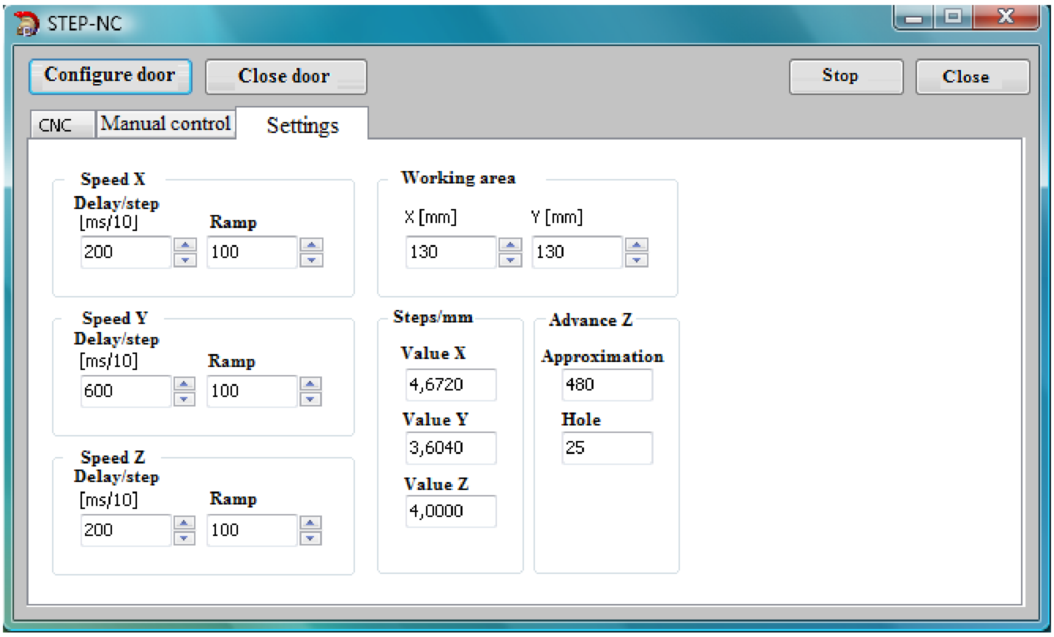


Figure 11. Configuration screen

## 4. TESTS AND RESULTS

For conducting the tests, STEP-NC files (part 21 format) were prepared, corresponding to different configurations of holes in the part. The compiler then performs the analysis of the files. If the compiler finds any inaccuracy, it indicates to the user (through a pop-up window) the row and column where the error is located. If the file is correct, the compiler generates a table with the machining data, such as: position of the holes, diameter of holes, tool types, axis feed, reference plane, and tool rotation. Figure 12 illustrates the drawing of the part that was considered for the tests, and Figure 13 shows a photograph of the part after being machined.

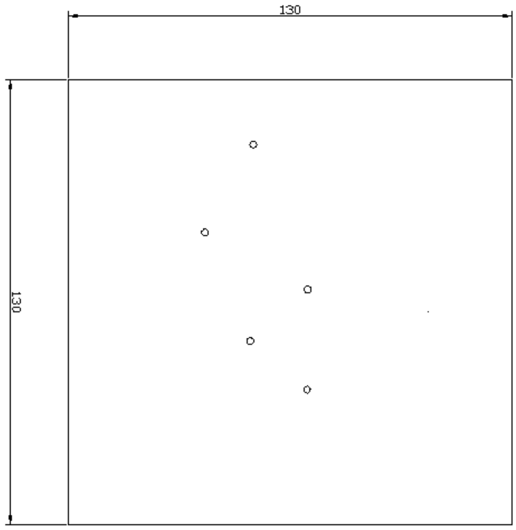


Figure 12. Drawing of the part

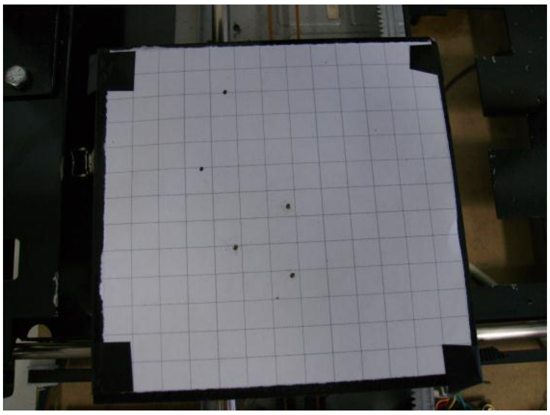

Figure 13. Photograph of the machined part

Tests were performed with different amounts of points in order to analyze the behavior of the optimization algorithm. Table 1 shows the results of tests performed on files that do not use the algorithm and those that use the optimization algorithm, and Figure 14 shows the machining time resulting from the use of the optimizer, compared with the times without using the optimization algorithm.

In Table 1 it can be noticed that with the increase of the number of holes the optimization algorithm resulted in a better performance. This difference in machining time can be noted by observing the operation of the machine, because when it leaves the home position the first time, it takes a while to machine the first hole, so when there are just a few holes the machining time becomes significant. However, in the case of a larger number of holes, the total machining time decreases because the cutting tool vertical travel time is diluted in the amount of existing holes in the part.

Table 1. Comparison between the results obtained with and without the use of an optimization algorithm

| *Item* | *Amount of points* | *No optimization (sec)* | *With optimization (sec)* | *Reduction (%)* |
|---|---|---|---|---|
| 1 | 13 | 112 | 110 | 1.79 |
| 2 | 36 | 367 | 288 | 21.53 |
| 3 | 50 | 501 | 382 | 23.75 |
| 4 | 100 | 972 | 754 | 22.43 |

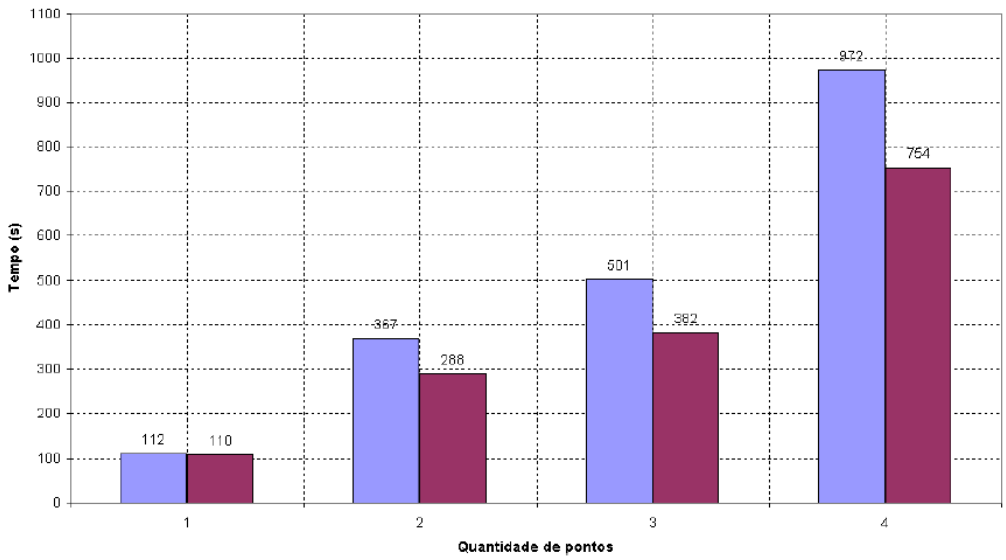


Figure 14. Machining holes without (blue) and with (magenta) the use of an optimization algorithm

## 5. CONCLUSIONS

The importance of portability of machining information either through programs or data interfaces has been growing steadily in manufacturing. The interoperability between computer systems to support manufacturing is essential to achieve the goal of integrating computer-aided manufacturing. In this work the ISO 10303 (STEP) and ISO 14649 (STEP-NC) standards were applied in the direct control of a CNC drilling device. This direct control is partly because it is not necessary to use a post-processor to translate the file from STEP-NC into G-code (ISO 6983). It was shown that the use of these standards has a significant advantage compared with G-code, which is the absence of dialects, enabling process interoperability. It was also shown that a CNC machine can be driven without the use of intermediate languages.

In the development of this work publications on recent developments related to STEP-NC applications were sought, and studies that use similar approaches were published by Calabrese (2007) and Suh *et al.* (2003). However, those implementations used a post-processor to translate the STEP-NC file into G code (corresponding to the A type shown in Figure 4). This type of implementation has the limitation resulting from the non-parity between ISO 14649 and ISO 6983. Since the prototype developed in this work has a STEP-NC interpreter as part of the CNC, it is therefore classified as type B according to Suh *et al.* (2003) (see Figure 4). That is, the proposed prototype does not use legacy systems, with a solution that contributes to the integration of manufacturing.

Despite the limitations of the built equipment, it was used adequate to show the application of an open architecture CNC adherent to STEP-NC, since the exchange of manufacturing information between the computer and the device was effectively carried out.

For future efforts it is aimed to apply the proposed approach to other features such as pockets, shoulders, slots, and chamfers. We also intend to extend the system over the TCP/IP in order to provide remote machining via the Internet.

## 6. REFERENCES

Aho, A.V., Sethi, R., Ullman, J.D., 1995, “Compilers: Principles, Techniques and Tools”, Ed. Rio de Janeiro: Guanabara Koogan (in Portuguese).

Allen, R.D, Newman, S.T., Harding, J.A., Rosso Jr. R.S.U., 2003, "The design of a STEP-NC compliant agent based CAD/CAM system", Proceedings of the 13th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM 2003), Tampa, FL, USA, pp. 530–540.

Calabrese, F. and Celentano, G., 2007, "Design and Realization of a STEP-NC Compliant CNC Embedded Controller", Emerging Technologies Factory Automation, pp. 1010-1017.

Feeney, A.B, Kramer, T, Proctor, F., Hardwick, M., Loffredo, D., 2003, "STEP-NC Implementation – ARM or AIM?", White Paper. ISO T24 STEP-Manufacturing Meeting, San Diego, USA.

ISO 6983, 1982, "Numeric control of machines-Program format and definition of address words-Part 1: Data format for positioning, line motion and contouring control systems".

ISO 10303-1, 1994, "Industrial Automation Systems and Integration Product Data Representation and Exchange – Overview and Fundamental Principles", International Standard, ISO 10303-1. ISO TC184/SC4. [S.1.].

ISO 14649-1. 2003, "Data Model for Computerized Numerical Controllers: art 1 - Overview and fundamental principles". [S.l.].

Loffredo, D., 2008, "Fundamentals of STEP Implementation", 1st March 2008, <http://www.steptools.com/library/fundimpl.pdf>.

Nassehi, A., Newman, S.T., Allen, R.D., 2006, "STEP-NC compliant process planning as an enabler for adaptive global manufacturing", Robotics and Computer-Integrated Manufacturing. pp. 456-467.

Newman, S.T., Allen, R.D., Rosso Jr., R.S.U., 2003, "CAD/CAM solutions for STEP-compliant CNC manufacture", Computer Integrated Manufacturing, pp. 590-597.

Rosso Jr., R.S.U., Newman, S.T., 2003, "Estrutura de Dados para Sistemas CAD/CAM aderente à STEP", Livro de Actas do VI Congresso Ibero-Americano de Engenharia Mecânica - Universidade de Coimbra, pp. 1019-1024.

STEP Tools Inc., 2008, "What is STEP?", 2nd March 2008, <http://www.steptools.com/library/standard/step_1.html>

Suh, S., Lee, B.E., Chung, D.H., Cheon, S.U., 2003, "Architecture and implementation of a shop-floor programming system for STEP-compliant CNC", Elsevier, pp. 1069-1083.

Xu, X.W. and Newman, S.T., 2006, "Making CNC machine tools more open, interoperable and intelligent – a review of the technologies", Computers in Industry, vol. 57, no. 2, pp. 141-152.

Zeid, I., 1991, "CAD/CAM Theory and practice", McGraw-Hill: New York.

## 7. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.