A flexible programming and verification methodology for reconfigurable CNC woodworking machine

Article *in* BioResources · October 2024

DOI: 10.15376/biores.19.4.9708-9726

6 authors, including:

Zoran Dimic
Lola Institut

Nikola Slavković
University of Belgrade

Marija Djurkovic
University of Belgrade

Jelena Vidakovic
Lola Institut

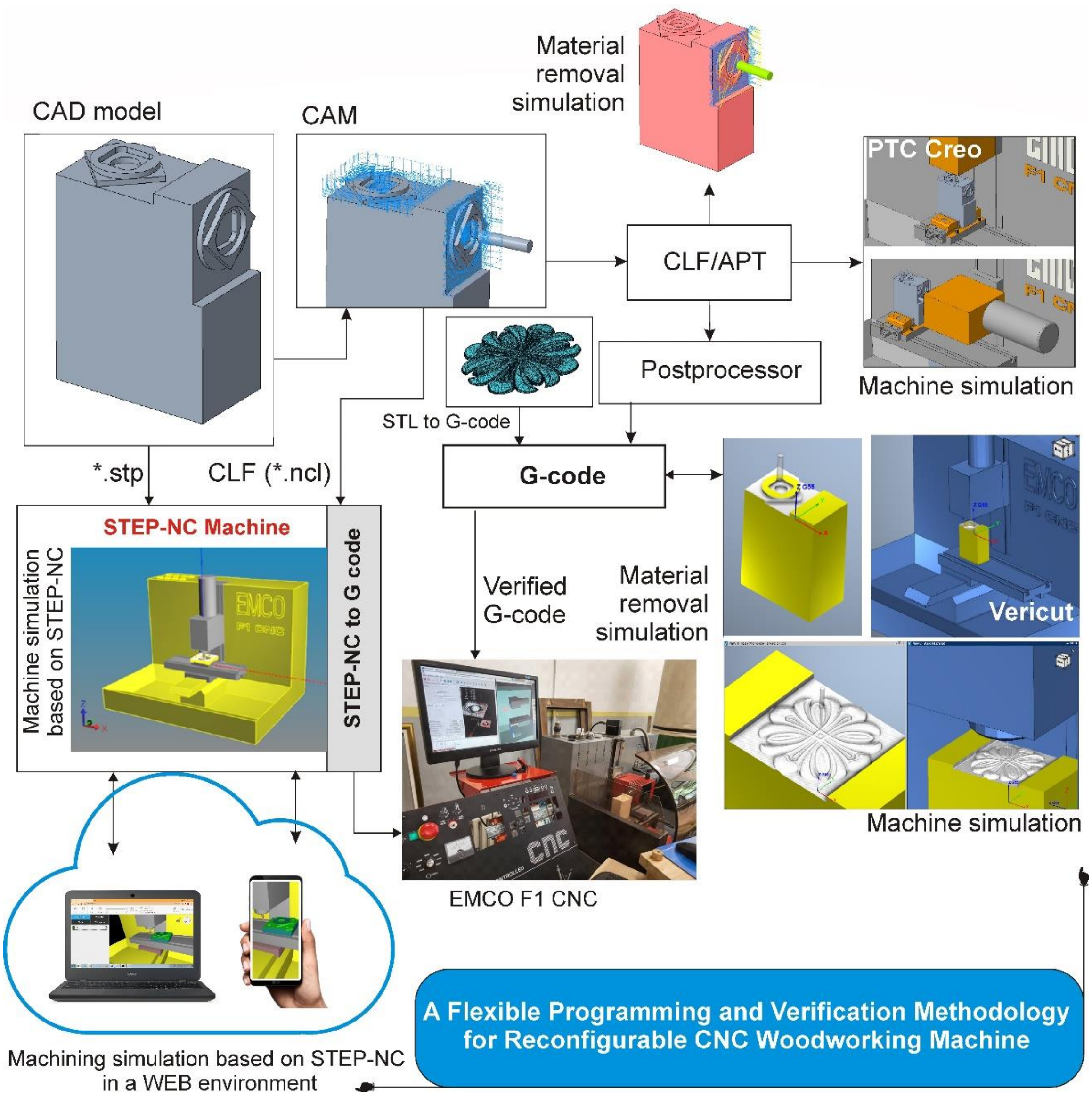
A Flexible Programming and Verification Methodology for Reconfigurable CNC Woodworking Machine

Sasa Zivanovic,[a] Zoran Dimic,[b,*] Mladen Furtula,[c] Nikola Slavkovic,[a] Marija Djurkovic,[c] and Jelena Vidakovic [b]

* Corresponding author: zoran.dimic@li.rs

DOI: 10.15376/biores.19.4.9708-9726

GRAPHICAL ABSTRACT



A Flexible Programming and Verification Methodology for Reconfigurable CNC Woodworking Machine

Sasa Zivanovic,[a] Zoran Dimic,[b,]* Mladen Furtula,[c] Nikola Slavkovic,[a] Marija Djurkovic,[c] and Jelena Vidakovic [b]

An advanced flexible programming methodology for CNC woodworking machines was developed. As the research starting base, a three-axis CNC woodworking machine was used. The developed methodology is proposed for programming, simulation, postprocessing, and machining by woodworking machine. This flexible programming method integrates the standard programming based on CAD, CAD/CAM systems, and STEP-NC protocol through different output files, enabling data interoperability during the realization of the machining tasks. The control system for the machine is configured based on the open-architecture software LinuxCNC to verify the flexible programming method and the results obtained. Programming verification was realized by simulation on a configured virtual machine in different programming environments and finally on a virtual machine integrated with the control system. The results obtained from the study were evaluated comparatively.

DOI: 10.15376/biores.19.4.9708-9726

Keywords: Programming methodology; CAD/CAM; STEP-NC; Reconfigurable; Woodworking machining; Virtual machine

Contact information: a: University of Belgrade, Faculty of Mechanical Engineering Kraljice Marije 16, Belgrade 11120 Serbia; b: LOLA Institute, Kneza Višeslava 70a, Belgrade 11030 Serbia; c: Faculty of Forestry, University of Belgrade, Kneza Višeslava 1, Belgrade11030 Serbia;
** Corresponding author: zoran.dimic@li.rs*

INTRODUCTION

Industries that currently use woodworking CNC machines need solutions that enable them to program, control, and supervise their machines more efficiently. Many solutions introduce different programming environments in order to facilitate the use of wood CNC applications. These solutions are the results of applications that exchange the programming data between control units and several programming software. One of these solutions proposed in this paper uses the CAD, CAD/CAM, and STEP-NC digital data chain for programming. During the programming of CNC machines using different software packages, seamless interaction with virtual machines for verification programs is needed before machining. Program verification is crucial in multi-axis machining and when machines can change structures, such as reconfigurable machine tools.

The CNC woodworking systems present a significant challenge in the era of Industry 4.0 (Lasi *et al.* 2014), which has an initiative that aims to digitalize industrial manufacturing *via* the exploitation of digital technologies. Transformation aims to enable and enhance the production system's programming flexibility, individualization, customization abilities, and reconfigurability. Changes in manufacturing requirements and market demands call for increased flexibility, adaptability, and reconfiguring of machine tools, which include both the physical hardware reconfiguration, as well as their logical reconfiguration manifested in the control system (Pritschow *et al.* 2009).In order to meet such challenges, the present manufacturing system requires a significant transformation in the information handling between CAD/CAM and CNC (Kamran *et al.* 2019).

The application of digital technology (DT), such as computer-aided design (CAD), computer numerical control (CNC), digital communication, digital marketing, *etc*., has been on the increase within the manufacturing industries. It has been widely suggested that the onset of the COVID-19 pandemic accelerated the adoption of DT in manufacturing industries worldwide. Wood products companies that adopted digital technologies boosted their production using CAD and CNC digital technologies. At the same time, those with limited DT adoption were more severely impacted during the pandemic (Ratnasingam *et al.* 2021).

In the field of woodworking, there is a wide technology and a variety of machining operations for milling. With CNC woodworking, it is possible to conceive complex designs that are sometimes impossible to make by conventional cutting methods while minimizing costs, production time, and material (Barata *et al.* 2016; Deus *et al.* 2018). When adopted and using CAD/CAM/CNC philosophy in the domain of woodworking, 3D wood parts that consist of both simple and complex surfaces can be easily created and machined (Krimpenis and Chrysikos 2019).

This study proposes a flexible programming methodology, which represents a procedure for using different digital environments for programming and verifying woodworking programs. The methodology of configuring virtual machines to verify generated programs before execution on a real machine is also presented.

The 3-axis CNC woodworking machine EMCO F1 CNC was used as the testbed for the developed methodology. A new control system based on LinuxCNC was developed for the machine during the previous research (Dimic *et al.* 2024). In recent years, open-source software applications based on the operating system Linux have replaced previous proprietary software in many fields (Wings *et al.* 2015).

LinuxCNC provides a generic control solution for a plethora of CNC machines. It is, therefore, expected that, in some cases, not all its features are utilized because available data are restricted to the physical characteristics and capabilities of the controlled CNC machine (Pantelidakis and Mykoniatis 2024).

LinuxCNC is an open architecture control system that runs on most standard PCs loaded with a Linux operating system and supports G code programming and control automation equipment. In the traditional field, it is suitable for controlling milling machines, lathes, 3D printers, laser cutting machines, robots, parallel kinematic machines, and other equipment (Li *et al.* 2015; Kyrychenko 2013).

LinuxCNC has an adjustable modular structure for various hardware and control interfaces and low-level machine electronics such as sensors and motor drives. The modular structure and open source code of LinuxCNC enable the development of kinematics functions for any machine, which makes LinuxCNC suitable for machines with specific kinematic solutions (LinuxCNC 2024), such as, in this case, the kinematics for the reconfigurable machine with horizontal and vertical positions of the main spindle (Dimic *et al.* 2024).

Based on the LinuxCNC software system, the control unit for the woodworking CNC machine EMCO F1 CNC, with reconfiguration options (Dimic *et al.* 2024), was developed, followed by a flexible programming methodology presented in this paper.

EXPERIMENTAL

Description of the Reconfigurable Woodworking Machine EMCO F1 CNC

EMCO F1 CNC is an educational 3-axis milling machine (Fig. 1), which has the possibility of reconfiguration. Different orientations of the milling head reflect reconfigurability. The milling head can be swiveled manually by 90 degrees into a horizontal or vertical position, which makes possible two different configurations in a single machine, as shown in Fig. 2. There is integrated reconfigurability.

However, the original version of the machine did not have the appropriate reconfigurable control system. As initially designed, the machine could work as a vertical or horizontal milling machine, starting a corresponding control system configuration and all axes initialization (EMCO1). In previous research (Dimic *et al.* 2024) for woodworking CNC machine EMCO F1, the authors realized an open architecture control system based on LinuxCNC software. The newly developed control system realized the entire reconfiguration process without reinitializing the machine, using the appropriate workpiece coordinate systems for the horizontal and vertical configuration of the machine, enabling programming in G-code. The programming method is convenient because it is based on the standard for G code (ISO 6983-1 2009). However, in this research study, a flexible programming method was considered. The version includes and integrates the standard programming based on CAD, CAD/CAM systems, and STEP-NC protocol through different output files, enabling data interoperability during the realization of the machining tasks.

The present study considers a flexible programming method using several interchange file formats (STEP, STL) that can be easily exchanged between different CAD, CAD/CAM, or specialized CAM programming software. This way, operators and programmers do not have to change their programming procedures. They can efficiently translate the results of a programmed toolpath, defined in various formats, from available software and postprocessors into the appropriate G-code, according to the ISO 6983 standard.

Fig. 1. EMCO F1 CNC woodworking machine tool

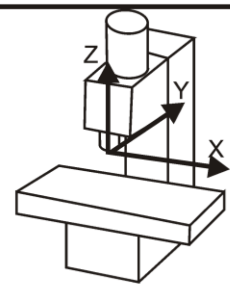
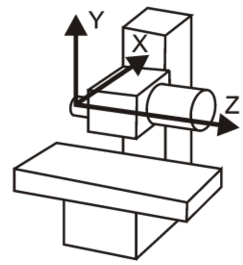
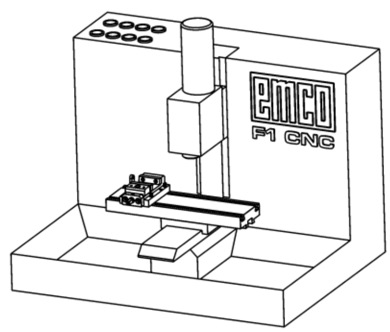
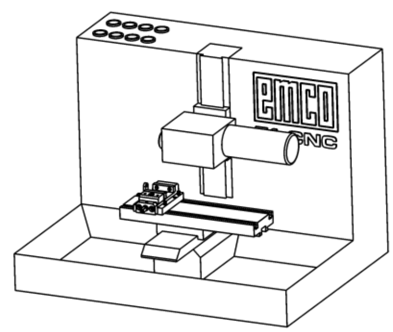
Type	Educational 3-axis vertical / horizontal milling machine	
Configuration	Vertical	Horizontal
Basic configuration machine with coordinate system		
X / Y / Z travel	200 mm / 100 mm / 200 mm	
Kinematic structure	X'Y'OZ	Z'X'OY
Simplified CAD virtual models of machine for simulations		

Fig. 2. Basic EMCO F1 CNC machine configurations, coordinate systems and kinematic structures

Methodology for Flexible Programming Woodworking CNC Machine

The proposed methodology for flexible programming of woodworking CNC machines is a systematic approach for programming and program verification on virtual machines for milling wood parts used in the wood industry. The methodology is described using the IDEF0 diagram (Kim and Jang 2002). IDEF0 is a function modeling methodology for describing manufacturing functions, which offers a functional modeling language for analysis and development, and in this case, integration of programming systems and verification processes on virtual machines (Piccoli and Pigni 2018).

The IDEF0 is used to design a model function as a structured representation of the functions, activities, or processes within the flexible programming method for CNC machining wooden materials based on different programming environments. The top-level child diagram A0 describes a basic flow of activities, as illustrated in Fig. 3.

Analysis of IDEF0 diagrams from Fig. 3 shows the basic flow of activities: A1: Software systems - toolpath generation; A2: Machine simulation in CAD/CAM system; A3: STEP-NC machining simulation; A4: WEB interface for STEP-NC simulation; A5: Postprocessing to G-code; A6: Program verification on Virtual machine in control system; A7: Realization of machining task.

Input data in activity A1 is a CAD model or model in STL format, where, based on this input, different software systems generate tool paths. Output data from this activity are Cutter Location file (CLF) or STEP-NC program according to standard ISO 10303-238. These were prepared in activity A1 in CAD/CAM environment, and STEP-NC Machine represents an input for activities A2 and A3. CNC woodworking machine simulation is available in activity A2, where the toolpath – CLF can be verified in the CAD/CAM system. Also, this activity can be verified by post-processed G-code in the Vericut environment.

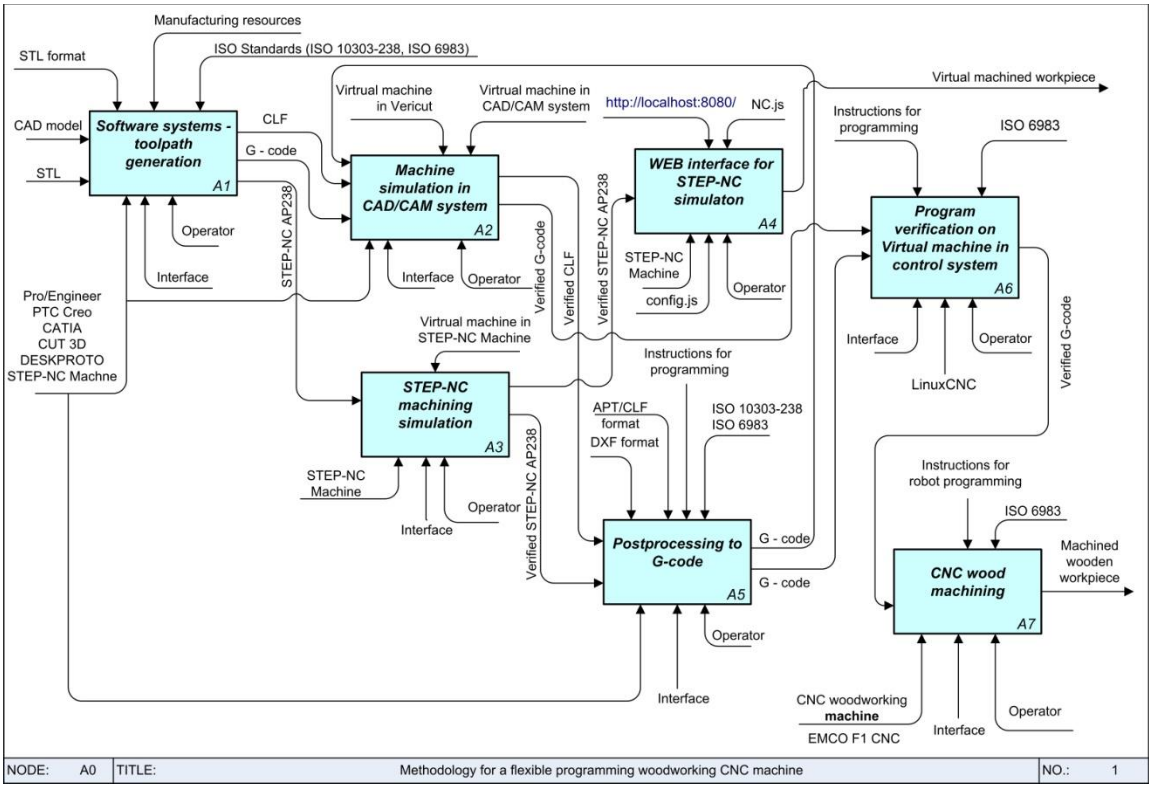


Fig. 3. Methodology for flexible programming woodworking CNC machine

STEP-NC program generated in activity A1 is used as input for activity A3, containing machining simulation on a configured virtual CNC woodworking machine, which can execute programs in the STEP-NC Machine and on a virtual machine in a web environment (Activity A4).

The input in the A5 activity is the verified tool paths in CLF and STEP-NC format, where the postprocessing into the G-code is executed. For the CNC woodworking machine considered, it is necessary to have a converter to convert STEP-NC programs into an appropriate program format. For this purpose, available convertors from the STEP-NC Machine are used because the developed control system (LinuxCNC) uses G-code with syntax very similar to Fanuc CNC systems.

After obtaining the G-code, final program verification on the Virtual machine integrated with the LinuxCNC-based control system is realized in activity A6. CNC wood machining is realized in activity A7, where the result is the machined wooden workpiece. A digital twin is a virtual machine that can simultaneously execute programs in real-time with a real machine. In this case, the digital twin integrated with the control system shown in activity A6 allows offline program verification, virtual machining, and monitoring during real machining.

Configuring Virtual Machine

Virtual machining simulation is very important in modern manufacturing and is performed before real machining. Machine simulation is important because it (i) enables program verification and prevents collisions during machining, (ii) shows machine tool operators what to expect from new programs, (iii) increases machining safety, (iv) enhances visualization of the machining process, and (v) provides training and education without using production time or risking a crash of the real machine.

The usual generalized procedure for configuring virtual machines is illustrated in Fig. 4 by the stages to be carried out until the definition of the virtual machine is ready for simulation of the work according to the given program. These stages are:

- Description of the kinematic structure;
- Configuring CAD model of the virtual machine, *i.e.*, modeling the basic components of the machine tool. If the machine tool model already exists, load the finished virtual machine model in STEP or STL formats;
- Defining the kinematic joints and the mobility of the machine axes involves analyzing the machine tool model and integrating the corresponding kinematic connections, enabling the mobility of the machine components. Here, it is necessary and correct to define the directions and limitations of the translatory and rotary axes of the machine tool. At the end of this phase, analyze the mobility of virtual machine axes within the defined limits;
- Virtual machine machining simulation according to the given program, depending on the applied programming environment and programming method. A given program can be a tool path (CLF), G code, or STEP-NC program. These machining simulations allow the loading of the complete virtual machine for program verification purposes.

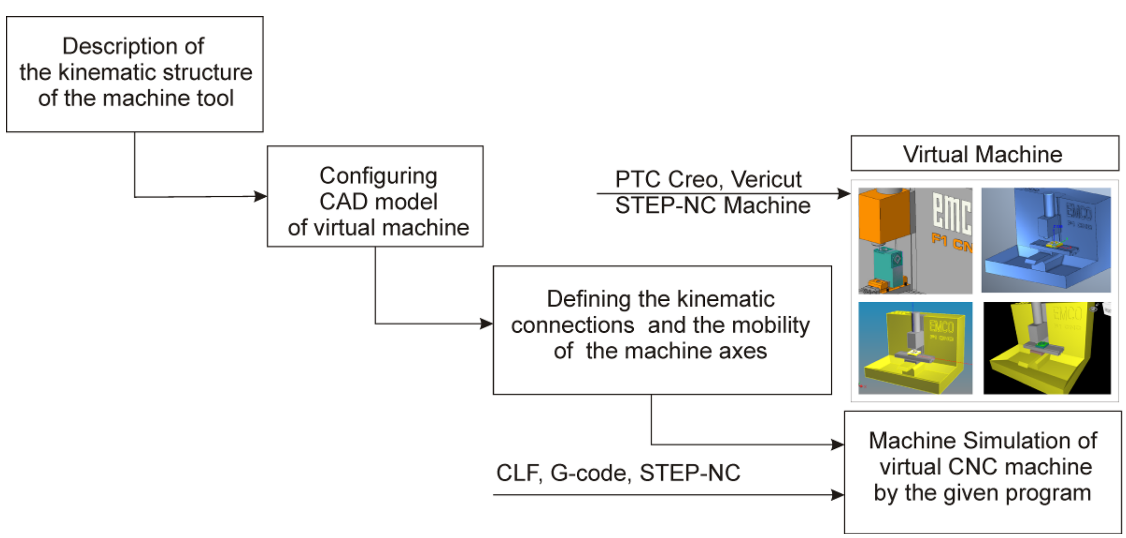


Fig. 4. Basic procedure for configuring virtual machine

The virtual machine in Vericut

The machining simulation software Vericut was used to simulate the G-code-based program. The application of this software offers the possibility of simulation toolpath and material removal simulation but also the possibility of including a configured virtual machine (Zivanovic *et al.* 2023a).

The kinematic structure of the woodworking CNC machine X'Y'OZ describes the arrangement and type of connections between the base and the moving components of the machine. It is necessary to define exact connections between moving components, *i.e.*, BASE, TOOL, and STOCK. The fixed part of the machine is the base, with the tooling branch (Z axis), and the components of the workpiece branch (X' and Y ' axes) should be added, as shown in Fig. 5.

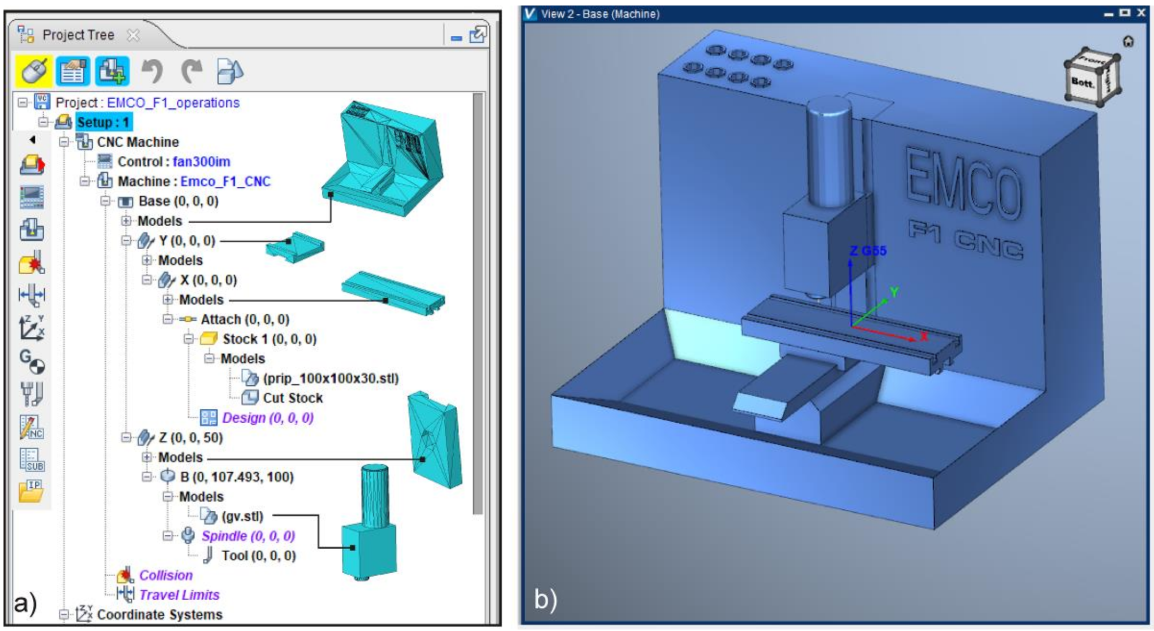


Fig. 5. CNC wood machining simulation based on G-code in Vericut environment

The virtual machine configuration in Vericut starts from the base as the first fixed component, to which the translatory axis Y' is added and axis X' is added as a part of the workpiece branch. These axes represent the translatory worktable on which the attached workpiece (Stock) is placed and based. On the other hand, the fixed base is connected to the remaining axis Z. The vertical translational axis Z is added to the base. The main spindle (Spindle) and the tool (Tool) are connected to the Z axis, which completes the kinematic structure of the machine.

After configuring the virtual machine, it should be saved and added to the database of existing virtual machine tools in Vericut. For the realization of the processing simulation project, it is necessary to define (Vorkapic *et al.* 2020; Zivanovic *et al.* 2023a): (i) workpiece (Stock), (ii) working object, coordinate system (Program zero point), (iii) zero-point position adjustment on the virtual machine), (iv) tools used in the process, and (v) NC programs. An example of one performed process simulation is shown in the next sections.

A virtual machine in STEP-NC Machine

Because a control unit cannot interpret the STEP-NC program, this section demonstrates how to configure and use a virtual machine with a capacity for woodworking machining simulations when the machine works using the program based on the STEP-NC. The own configured virtual machine can interpret the STEP-NC program in the software STEP-NC Machine. It is necessary to follow the next procedure to prepare a woodworking virtual machine for the STEP-NC environment (Zivanovic *et al.* 2018): (i)

Configure a CAD model of the woodworking CNC machine tool; (ii) Prepare model of the machine tool in the STEP format by the protocol AP203, AP214 or AP224; (iii) Load model of the machine in STEP format in software ST Viewer; (iv) Prepare XML file which includes characteristics of a machine tool based on STEP file (Fig.6a); (v) Include both files for the machine (STEP file and XML file) in the folder machine in software STEP-NC Machine; and (vi) Start STEP-NC Machine, and in the Machine tool drop-down menu it will appear the new configured machine tool ready for Virtual machining, based on STEP-NC program.

The usual XML file structure for the description structure of machine tools and robots for machining has the following parts: (i) Machine name (EMCO_F1 V-H); (ii) Control algorithm (BCGantry); (iii) Name of the STEP file of the machine (emco_f1_asm.stp); (iv) Description of the machine base structure, which is stationary; (v) Description of tool side structure (<chain target="tool">); (vi) Definition of placement location for tool; (vii) Description of workpiece side structure (<chain target="workpiece" >); (viii) Definition of placement location for workpiece; (ix) Description of NC axes and their feeds, constraints, mutual position, as well as the sequence of rotary axes), Fig.6a.
According to this methodology, any other woodworking CNC virtual machine tool can be configured and loaded in the STEP-NC Machine environment, and then STEP-NC programs can be executed.

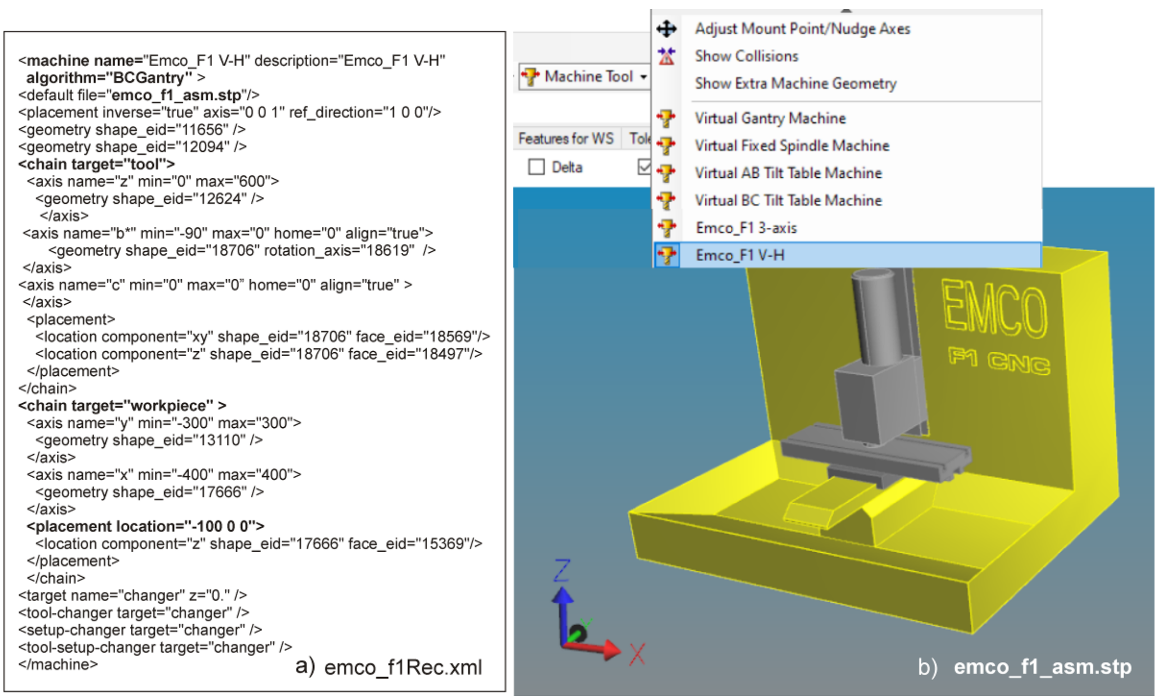


Fig. 6. Example of XML (a) and STP (b) representation of CNC woodworking machine EMCO F1 CNC for machine simulation in STEP-NC Machine environment

Programming and Program Verification on a Virtual Woodworking Machine

In this section, applied programming methods are given within the methodology for flexible programming woodworking CNC machines and effective methods for program verification before machining.

Programming and simulation in CAD/CAM environment

As one of the basic programming environments, the CAD/CAM system PTC Creo was used. The basic structure of the program system used for programming is shown in Fig. 7. The part related to programming is very conventional, using a postprocessor to convert the CL file into G-code. The programmer starts from the CAD model in a common way, in this case, in the CAD/CAM system PTC Creo that generates toolpath (CLF/APT). The generated tool path is tested through material removal and machining simulation on the virtual machine. Using the configured postprocessor, CLF obtains the program in G-code. CLF or G-code can used as an input for machine simulation in PTC Creo or Vericut environments. After successfully verifying the program, the machine can start machining the part.

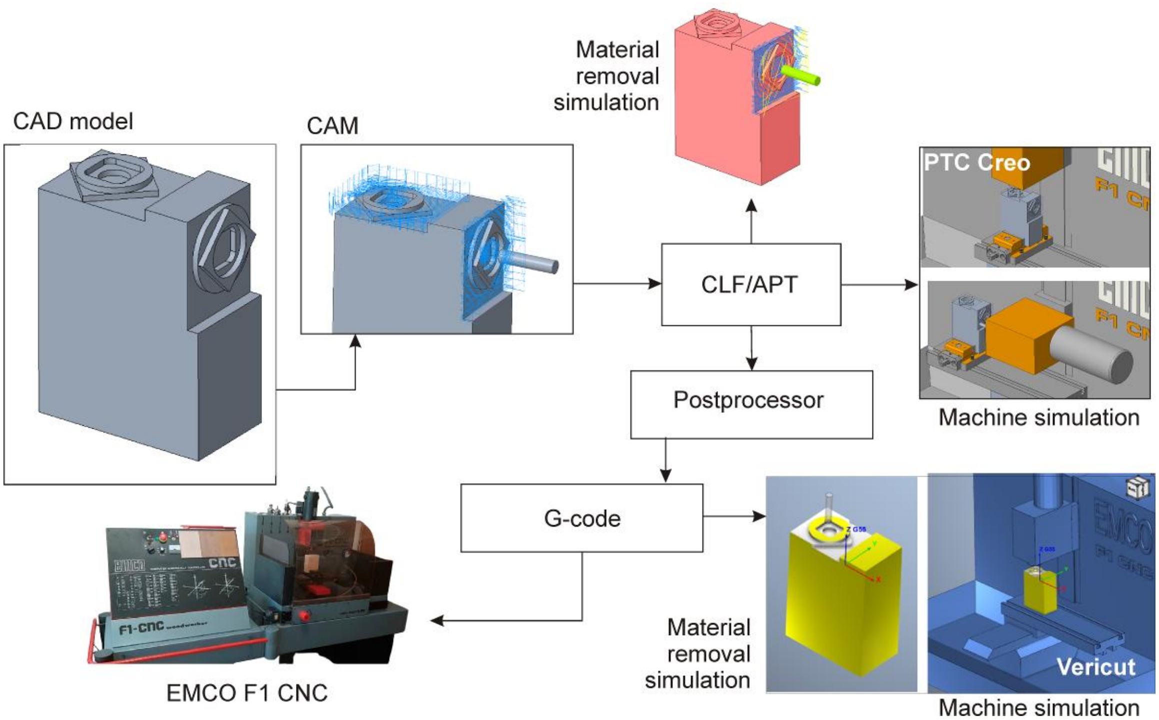


Fig. 7. Programming and simulation in CAD/CAM and Vericut environment

Programming and simulation based on STL

The typical process for rapid prototyping subtractive technology is also known as desktop milling. This procedure is very often used for CNC woodworking machining. This programming method is based on a model in STL format.

During programming, the following stages are passed: (1) CAD modeling of the prototype; (2) Model conversion to STL; (3) Loading of STL model as input in specialized CAM system; (4) Setup of the model in working area; (5) Planning roughing and finishing strategy for milling; (6) Generating adequate G-code; (7) G-code verification with material removal simulation; (8) Fabrication of prototype using CNC woodworking machine.

There are many specialized software packages for machining based on STL files. Some of them are CUT3D, Deskproto, Mesh CAM, *etc*. These software packages are easy to use and fast roughing and finishing tool path generation. An example of the application programming method based on the STL model using specialized CAM software is shown in Fig. 8. Figure 4 shows an example of the generated tool paths for roughing and finishing.

In the same software, program verification was performed using material removal simulation. After that, the postprocessing to obtain the G code follows. A material removal simulation in another program, such as the CIMCO editor, verified the post-processed G code.

Based on this example, machining had been performed. The results of that verification were presented in the subsection Machining test.

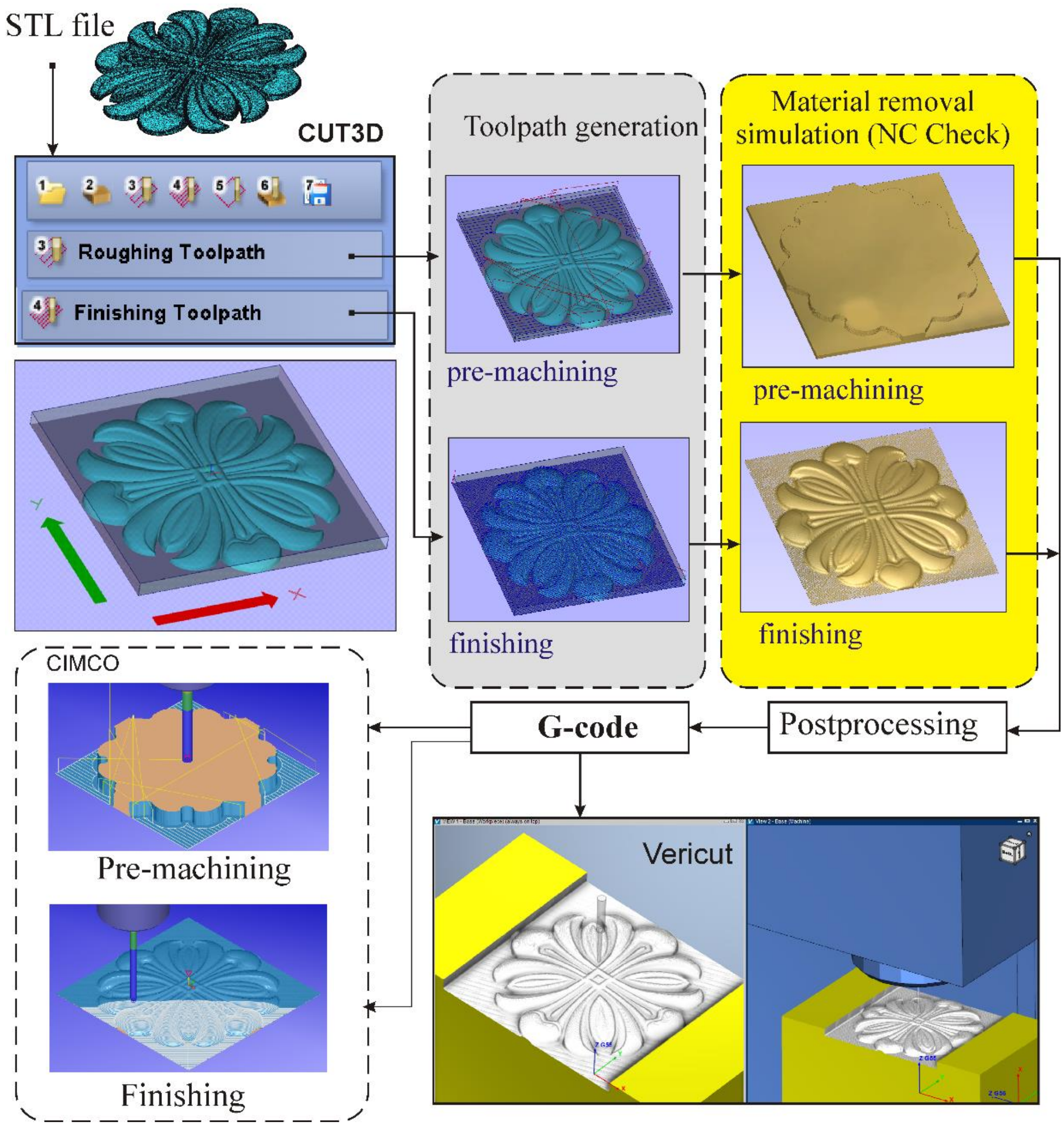


Fig. 8. Rapid prototyping using subtractive technology based on STL files

Programming and simulation in STEP-NC Machine environment

For programming a CNC woodworking machine in a STEP-NC Machine environment according to ISO10303-238, it is necessary to prepare models of the workpiece, raw stock material, and tool in STEP format in the available CAD/CAM environment, as well as toolpath trajectory (Cutter Location File – CLF). Thus, prepared files are input into the software STEP-NC Machine, using option Import (Fig. 9b), where a program in the P21 format is generated, as shown in Fig. 9a.

The flexibility of the programming method in the STEP-NC Machine environment is reflected in the possibilities used and combine STEP files and toolpaths APT/CLF from different CAD/CAM systems (PTC Creo, Pro/Engineer, CATIA, NX Siemens), Fig. 9b, and generate an integrated program in format P21.

Since the direct execution of the program in STEP-NC format requires a control unit with a suitable STEP-NC interpreter, for now, this programming method is applied indirectly by converting to G-code. A significant advantage of STEP-NC is the high level of information in the STEP-NC file, which is suitable for storing and exchanging the complete technology for processing a part, regardless of which machine the machining process will be performed on. The advantage of the STEP-NC Machine software is the existence of Export code options that have adequate postprocessors for the most well-known control units (Fanuc, Siemens, Haas, Okuma, Heidenhain,...), in Fig. 9c, which enable translation into the desired G code for the machine.

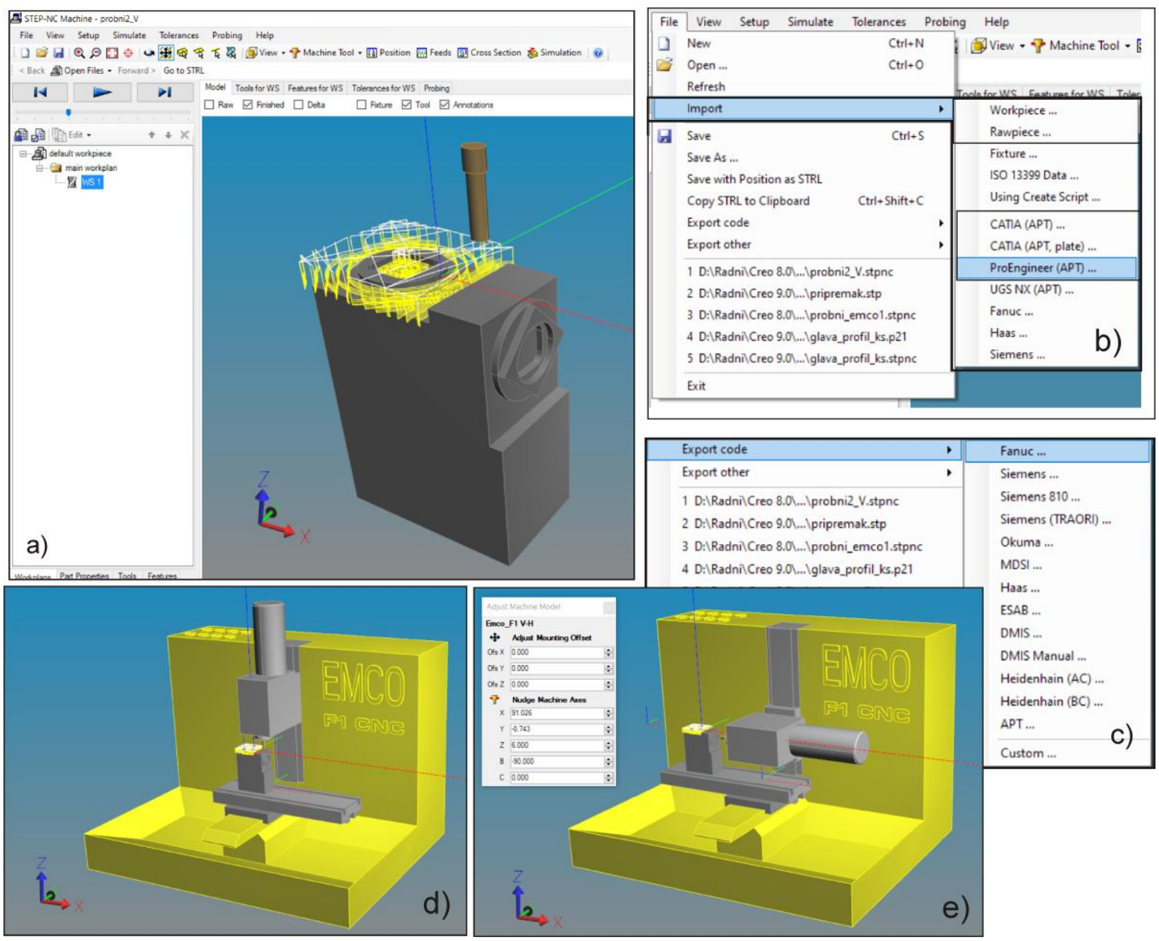


Fig. 9. Programming and simulation based on STEP-NC protocol

The virtual machine in the STEP-NC Machine is configured with a rotary axis on the main spindle, which enables the transition from vertical (Fig. 9d) to horizontal (Fig. 9e) configuration of the machine during the simulation of the STEP-NC program.

Web interface for the virtual machine tools

A configured virtual machine in a STEP-NC Machine environment can, through a Web interface and local server, enable accessibility on a cloud platform for simulation and monitoring. The Web interface based on STEP-NC was used, NC.js. The NC.js package is implemented in JavaScript. The web interface on the local server can use the STEP Tools commercial software STEP-NC Machine (Step Tools 2024), which can display the 3D models for the machining workpiece, tools, and material removal simulation. After virtual machining, it is possible to download the virtual machined part as an STL file for further qualitative analysis.

Basic steps for building and running a server can be viewed at Step Tools 2023. The NC.js package is built from the source using the Node Package Manager. The procedure assembles all dependencies, including a pre-built version of the STEPNode plugin, and then it starts the server on a specified STEP-NC file. First, node installation is needed; the NC.js source code from GitHub must be downloaded after installation. At the top-level directory of the NC.js package, the file “config.js” can be edited, and it is possible to specify which processing program is used in STEP-NC format and on which machine the processing will be performed, with the exact paths to these files.

After successful installation of the Node and NC.js packages, options for building and running the server are available. When the server runs with the defined program and machine in NC.js, and the web browser starts at the address http://localhost:8080/, a machining setup is shown, as shown in Fig. 10.

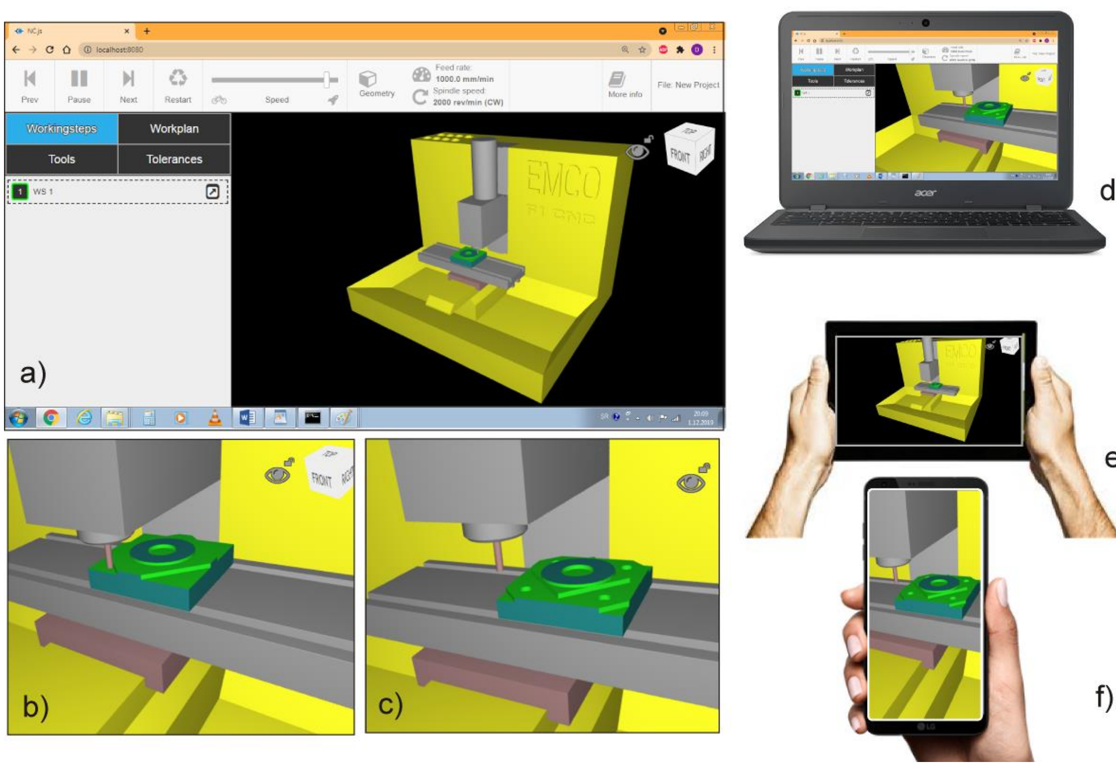


Fig. 10. Example of the CNC wood machining simulation based on STEP-NC in a WEB environment

In this manner, interactive websites can be developed to facilitate the sharing of machining programs and receive feedback on machined parts. An illustration of STEP-

NC's application in the Industry 4.0 era is demonstrated through Collaborative Manufacturing, integrating product design, manufacturing, and inspection processes (Step Tools 2024).

Virtual machine in control system LinuxCNC

The retrofitting of the control system for the presented EMCO F1 CNC machine is based on an available open architecture control software, LinuxCNC (Dimic *et al.* 2024). Years of research and development on the LinuxCNC program have produced a solid, time-proven machine control platform.

Integrating a virtual machine into an open architecture control system based on LinuxCNC allows machining simulation in the OpenGL 3D environment. Configuration of the interactive virtual machine as a digital twin is done *via* Python programming, allowing primitive graphics modeling and integration of the developed models directly into the LinuxCNC control system. Such a virtual machine enables G-code program verification and machining process monitoring (Rakic *et al.* 2021; Dimic *et al.* 2024).

The basic concept of configuring a virtual machine is presented in Rakic *et al.* (2021) and Dimic *et al.* (2024). The first step in the procedure is to prepare ASCII STL files of basic machine modules, such as a stationary base and modules for three coordinate axes (X, Y, and Z). The imported machine parts are oriented and placed in the virtual environment according to the adopted reference frame, resulting in a fully functional virtual machine. OpenGL libraries within Python were used to complete the machine model and connect axes with LinuxCNC control signals. Such a virtual machine enables G-code program verification and machining process monitoring, Fig.11.

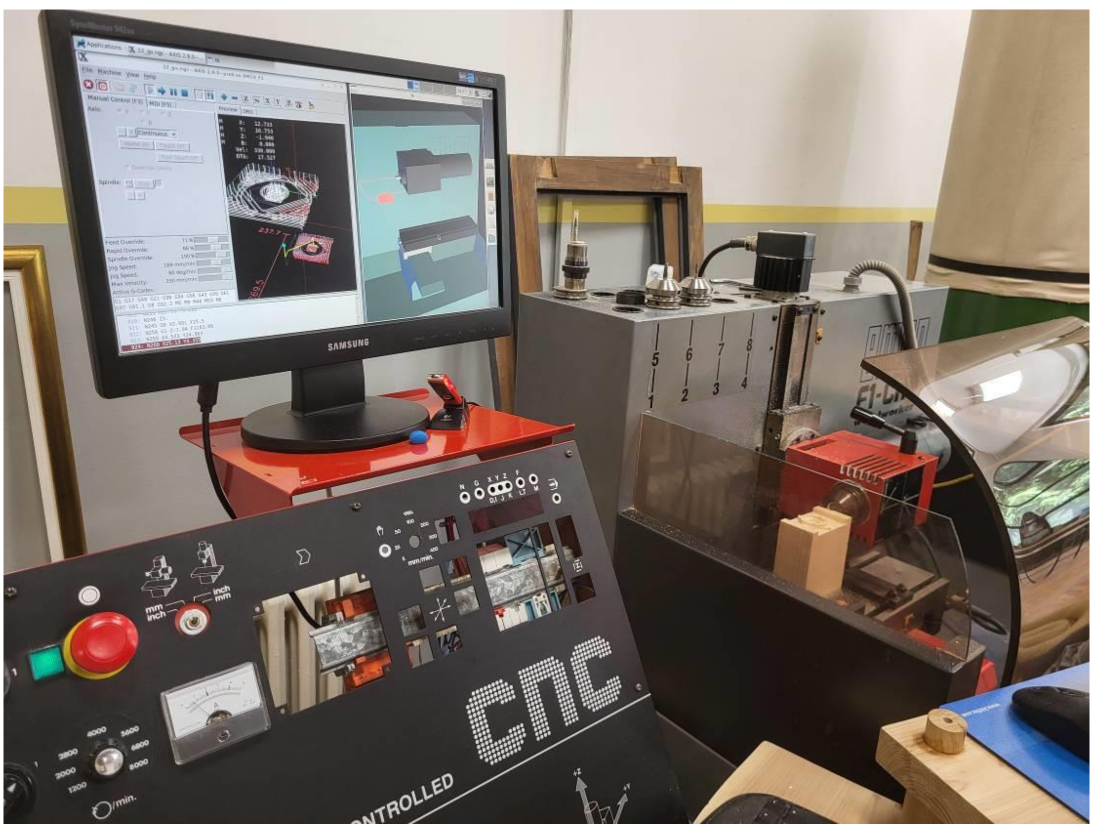


Fig. 11. Example of virtual woodworking CNC machine EMCO F1 integrated with a reconfigurable control system based on LinuxCNC

The virtual machine is integrated with AXIS GUI and placed in a separate window. Virtual machines as digital twins can draw toolpaths, considering each axe's limitations. During the execution of the machining program, the virtual machine components are moving in real-time, fully synchronized with the moving components of the real machine, with no visible latency, as shown in Fig. 11.

The virtual machine can be used to simulate and verify the machining programs directly in the Axis GUI, or it can be used for monitoring machine activity during machining operations since the real and virtual machine movements are identical (Rakic *et al.* 2021).

Here, the importance of the virtual machine is even greater because it can change its configuration while using a unique control system. At the same time, due to the limited workspace for the machine when it works in a horizontal configuration, it is very important to verify the program's feasibility before starting its execution.

RESULTS AND DISCUSSION

The machining experiments on the available reconfigurable CNC woodworking machine aimed to validate the flexible programming and verification methodology.

Machining test

Machining experiments were conducted with several examples. The first test was a machined test workpiece from Fig. 10. This test workpiece was programmed using a CAD/CAM system and STEP-NC protocol. The first machining experiment was carried out in the wood material using the flat endmill tool with a diameter of 6 mm. The workpiece was set up on the machine using a vacuum fixture. Figure 12 shows machining with a vertical machine configuration. This model for the first experimental testing was prepared according to the example for testing the working accuracy of CNC machine tools.

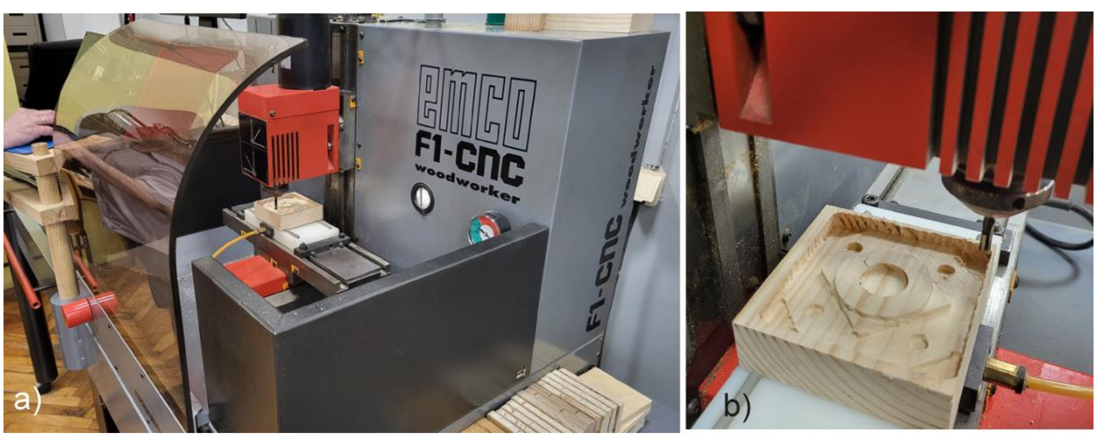


Fig. 12. Machined test workpiece with the vertical configuration of the machine

Figure 13 shows a machined workpiece based on an STL model with the vertical configuration of the machine. Sculptural surfaces, which are very common in woodworking, are machined this way. A vacuum fixture is also used to set up the workpiece on the machine. Machining was carried out using the wood material using the flat ball mill tool with a diameter of 6 mm.

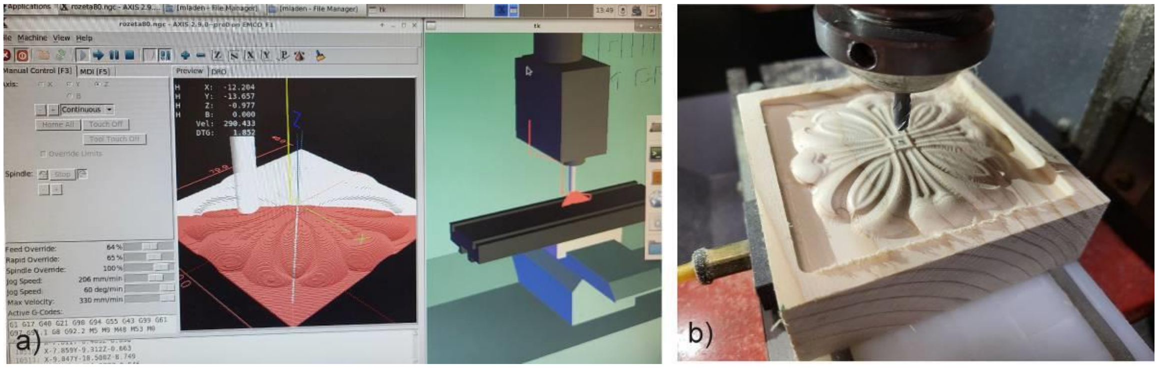


Fig. 13. Machined workpiece based on STL model with the vertical configuration of the machine

The machining test workpieces were verified into two planes during the testing in Figs. 14 and 15. These examples include machining test workpieces by reconfiguring the machine from vertical, Fig. 14(a-b) to horizontal, Fig.14 (c-d). This machined test is a very important virtual machine integrated with the control system LinuxCNC. It is used to check the accuracy of the setup workpiece and the possibility of machining itself without collisions.

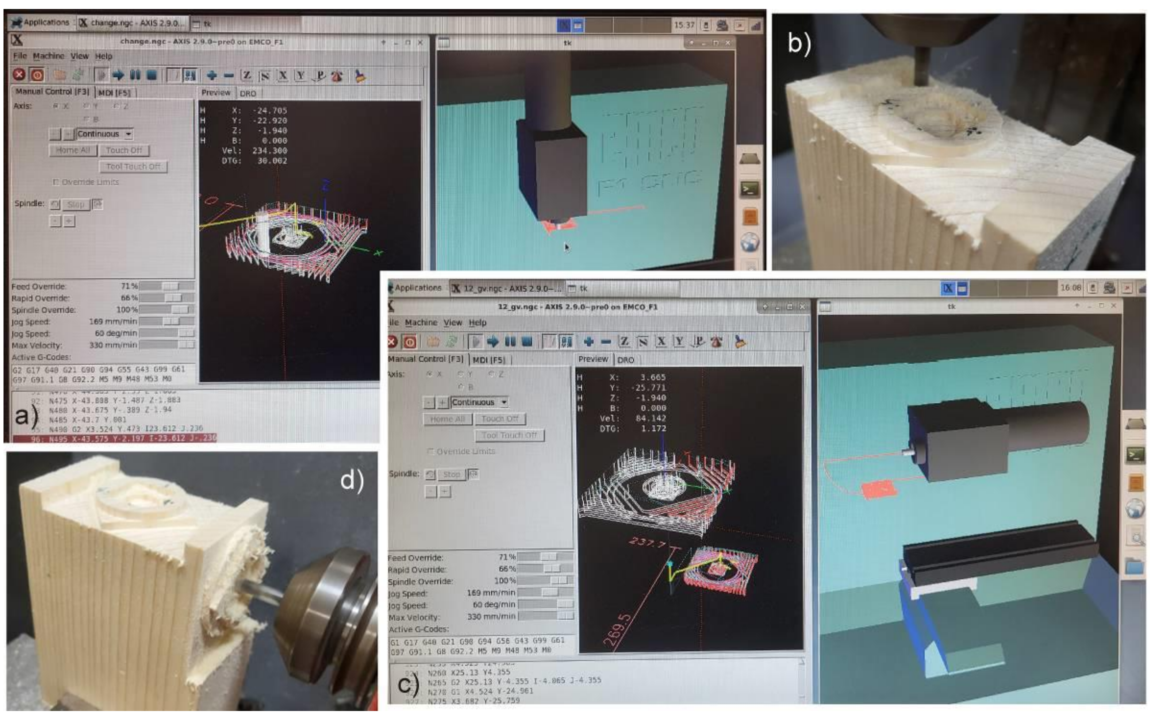


Fig. 14. Example of machining test workpieces with a reconfiguring machine from vertical to horizontal

Examples of finished machined different test workpieces are shown in Fig. 15. The final machined test workpiece with the vertical position of the main spindle is shown in Fig. 15a, then the reconfiguring of the main spindle to the horizontal position (Fig. 15b), and finally the final machined part with the horizontal position of the main spindle (Fig. 15c). A universal mechanical clamp was used to set up the workpiece on the machine. Machining of the wood was done using the flat-end mill tool with a diameter of 6 mm.

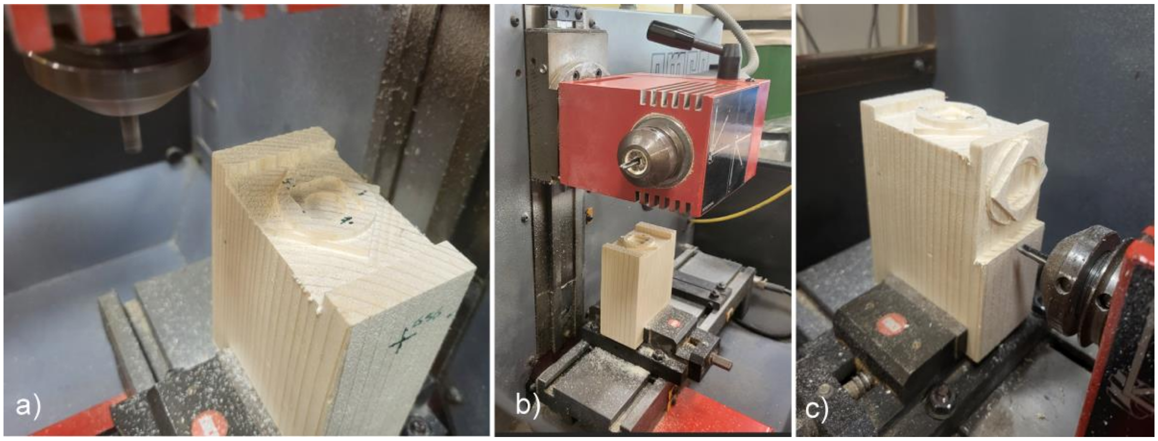


Fig. 15. Machined test workpieces with a reconfiguring machine from vertical to horizontal

The proposed methodology integrates programming and program verification in the programming and control systems, which most commercial machines do not have. Common programming approaches of woodworking CNC machines do not include program verification by machine simulation.

The presented programming methodology includes a new programming method based on the STEP-NC protocol, an alternative to G code. As a new programming method, STEP-NC will enable the application and exchange of complete machining technology in the STEP-NC file. This programming method raises the digitization and visualization of the entire machining process to a higher level, which can also be available in the web environment (Zivanovic *et al.* 2023b).

The presented results fully confirm the methodology for programming and verification of the program, since the programmed parts were successfully machined, including machining with a reconfiguration of the machine, which implies a change in the position of the main spindle, Fig. 15. All programs were first verified in the programming system, and then on the virtual machine in the control system, as a final form of verification.

CONCLUSIONS

1. The paper presented the developed advanced concept of flexible programming and verification methodology for reconfigurable CNC woodworking machines. The methodology allows programs to be prepared for different purposes in the woodworking industry. Classical programming methods were used, but a method based on STL models for machining sculptural surfaces was also used as a new programming method based on the STEP-NC protocol.
2. For all programming methods, verification was enabled using virtual machines in the programming environment and the control system LinuxCNC. The configuring of virtual machines in different environments is also very important because CNC woodworking systems present a significant challenge in the era of Industry 4.0, which aims to digitalize and virtualize CNC machines through digital technologies.

3. The control system based on LinuxCNC is a low-cost system that is especially suitable for retrofitting and advancing older machine tools. Linux is open-source software that can be reorganized, changed, and restructured for the desired purpose. Due to the modularity, flexibility, openness, and availability, the proposed system is convenient for machine tools with specific reconfigurable kinematic structures and unique functionalities. Using LinuxCNC software in CNC machines makes it possible to stay up-to-date in design and production processes. The LinuxCNC control and flexible programming methodology will also reduce production costs.
4. The methodology was successfully verified through machining experiments on a reconfigurable woodworking CNC machine. All programs were verified twice on the virtual machines during the programming test workpieces. First, on the configured virtual machine in the programming environment, and second, on the virtual machine integrated with a LinuxCNC control system.

ACKNOWLEDGMENTS

The paper is a part of the research done within the projects supported by the Ministry of Science and Technological Development of the Republic of Serbia by contract no. 451-03-65/2024-03/200105 dated 5 February 2024, by contract 451-03-66/2024-03/ 200066 for 2024 and by contract 451-03-65/2024-03/200169 for 2024.

REFERENCES CITED

Barata, T. Q. F., Rodrigues, O. V., Matos, B. M., and Pinto, R. S. (2016). "Furniture design using MDF boards applying concepts of sustainability," *Product: Management & Development* 14(1), 68-83. DOI: 10.4322/pmd.2016.008

Deus, P., Alves, M., Vieira, F., and Bilesky, L. (2018). "Analysis of the cutting parameters in front milling for medium density fiberboard," *BioResources* 13(2), 3404-3410. DOI: 10.15376/biores.13.2.3404-3410

Dimic, Z., Zivanovic, S., Pavlovic, D., Furtula, M., Djurkovic, M., Rakic, A., and Kokotovic, B. (2024). "Reconfigurable open architecture control system with integrated digital twin for 3-axis woodworking milling machine," *Wood Material Science & Engineering.* DOI: 10.1080/17480272.2024.2318024

ISO 6983-1. (2009) "Automation systems and integration — Numerical control of machines — Program format and definitions of address words Part 1: Data format for positioning, line motion and contouring control systems," International Organization for Standardization, Geneva, Switzerland.

Li, B. C., Yu, T., and Liu, P. (2015). "Kinematics simulation and control system design of the three DOF parallel mechanism," Joint International Mechanical, Electronic and Information Technology Conference JMIET 2025, pp. 750-755.

Kamran, L., Aini, Z. A. K. and Yusri, Y. (2019). "The importance of STEP-NC in the IR 4.0 manufacturing systems," *International Journal of Advanced Trends in Computer Science and Engineering* 8(4), 1568–1570. DOI: 10.30534/ijatcse/2019/80842019

Kim, S. H., and Jang, K. J. (2002). "Designing performance analysis and IDEF0 for enterprise modelling in BPR," *International Journal of Production Economics* 76, 121-133. DOI: 10.1016/S0925-5273(00)00154-7.

Krimpenis, A.A., and Chrysikos, M. (2019). "3D parametric design and CNC manufacturing of custom solid wood electric guitars using CAD/CAM technology," *Wood Material Science & Engineering* 14(2), 66-80. DOI: 10.1080/17480272.2017.1379035

Kyrychenko, A. (2013). "Open source CNC control for parallel kinematic machine tool," Proceedings of scientific works of the Kirovograd National Technical University. Machinery in agricultural production, industrial engineering, automation," *Kirovograd* 26, 63-66.

Lasi, H., Fettke, P., Kemper, H. G., Feld, T., and Hoffmann, M. (2014). "Industry 4.0," *Business & Information Systems Engineering* 6, 239-242. DOI: 10. 1007/s12599-014-0334-4

LinuxCNC. (2024). "LinuxCNC Documents" (https://www.linuxcnc.org/), accessed 28 july 2024.

Pantelidakis, M., and Mykoniatis, K. (2024). "Extending the digital twin ecosystem: A real-time digital twin of a LinuxCNC-controlled subtractive manufacturing machine," *Journal of Manufacturing Systems* 74, 1057-1066. DOI: 10.1016/j.jmsy.2024.05.012

Piccoli, G., and Pigni, F. (2018). *Information Systems for Managers: With Cases* (4th Ed.), Prospect Press, Burlington, VT, USA.

Pritschow, G., Wurst, K-H., Kircher, C., and Seyfarth, M. (2009). "Control of reconfigurable machine tools," in: *Changeable and Reconfigurable Manufacturing Systems*, H. A. ElMaraghy (ed.), Springer-Verlag, London, pp. 71-100. DOI:10.1007/978-1-84882-067-8

Rakic, A., Zivanovic, S., Dimic, Z., and Knezevic, M. (2021). "Digital twin control of multi-axis wood CNC machining center based on LinuxCNC," *BioResources* 16(1), 1115-1130. DOI: 10.15376/biores.16.1.1115-1130

Ratnasingam, J., Ioras, F., Choon Liat, L., Ayenkaren, J., Yan Yi, L., and Ab Latib, H. (2021). "Digital technology application among Malaysian value-added wood products manufacturers," *BioResources* 16(2), 2876-2890. DOI: 10.15376/biores.16.2. 2876-2890

STEP Tools (2023) "Inc, Build and run server," (https://steptools.github.io/NC.js/build.html), accessed 10 October 2023.

Step Tools (2024). "Digital thread for manufacturing," (http://www.steptools.com/), accessed 12 april 2024.

Vorkapic, N., Zivanovic, S., Slavkovic, N., Dimic, Z., and Kokotovic, B. (2020). "Configuring of 3-axis vertical CNC machine for rapid prototyping with two translatory and one rotary axes," *Advanced Technologies and Materials* 45(1),17-25. DOI: 10.24867/ATM-2020-1-003

Wings, E., Müller, M., and Rochler, M. (2015). "Integration of real-time Ethernet in LinuxCNC," *The International Journal of Advanced Manufacturing Technology* 78(9-12), 1837-1846. DOI: 10.1007/s00170-015-6786-y

Zivanovic, S., Slavkovic, N., and Milutinovic, D. (2018). "An approach for applying STEP-NC in robot machining," *Robotics and Computer–Integrated Manufacturing* 49, 361-373. DOI: 10.1016/j.rcim.2017.08.009

Zivanovic, S., Vorkapic N., and Dimic, Z. (2019). "Configuring the programming and control systems for a mini 3-axis CNC machine tool on the Raspberry Pi platform," *Tehnika* 68(6), 823-831, DOI: 10.5937/tehnika1906823Z

Zivanovic, S., Slavkovic, N., Tabakovic, S., and Z. Dimic. (2023a). "Programming and program verification for robots that perform machining tasks," *Tehnika* 73(2), 167-176. DOI: 10.5937/tehnika2302167Z.

Zivanovic, S., Dimic, Z., Rakic, A., Slavkovic, N., Kokotovic, B., and Manasijevic, S. (2023b). "Programming methodology for multi-axis CNC woodworking machining center for advanced manufacturing based on STEP-NC," *Wood Material Science & Engineering* 18(2), 630-639. DOI: 10.1080/17480272.2022.2057816

Article submitted: June 25, 2024; Peer review completed: July 17, 2024; Revised version received: July 31, 2024; Accepted: September 10, 2024; Published: October 29, 2024.
DOI: 10.15376/biores.19.4.9708-9726