Vol. 06, No. 2 (2024) 593-600, doi: 10.24874/PES06.02.016

Proceedings on Engineering Sciences

www.pesjournal.net

PROGRAMMING METHODS AND PROGRAM VERIFICATION FOR 3-AXIS RECONFIGURABLE HYBRID KINEMATICS MACHINE

Sasa Zivanovic[1]
Goran Vasilic
Zoran Dimic
Nikola Vorkapic
Branko Kokotovic
Nikola Slavkovic

Received 13.07.2023.
Received in revised form 02.10.2023.
Accepted 18.11.2023.
UDC – 004.052.42

Keywords:

Programming, CAD/CAM, STEP-NC, STL, Verification, Virtual machine, Reconfigurable, Hybrid kinematic machine

A B S T R A C T

This paper presents programming methods and program verification for the 3-axis reconfigurable hybrid kinematics machine MOMA V3, which represents an educational desktop milling machine with a horizontal position of the main spindle. The paper considers the different programming and program verification methods. For programming used CAD/CAM system PTC Creo, specialized CAM software CUT3D and new programming method based on STEP-NC. Program verification is based on tool path simulation, material removal simulation and virtual machine simulation. The paper presented the virtual machine in the programming system and the virtual machine integrated with the LinuxCNC control system. Final verification was achieved during the testing on machine by machining the selected workpieces which programmed with different programming methods.

1. INTRODUCTION

This paper considers the 3-axis machine with hybrid kinematics, which was created as a result of upgrading the 2-axis reconfigurable parallel mechanism by adding a serial translatory axis (Vasilic et al., 2021).

The concept of the 3-axis reconfigurable machine with hybrid kinematics, presented in this paper, is based on (Vasilic et al., 2019): (i) a 2-axis reconfigurable parallel mechanism developed as a modular system; (ii) a generalized kinematic model of the applied 2-axis reconfigurable parallel mechanism; (iii) open architecture control system based on LinuxCNC (EMC2); (iv) quick and easy reconfiguration of hardware and control system; (v) the possibilities of using different programming methods; (vi) virtual machine configured in object-oriented programming language Python and implemented in the control system for program verification.

This paper presents part of the research related to the programming methods for considered machine and testing during trial work.

[1] Corresponding author: Sasa Zivanovic
Email: szivanovic@mas.bg.ac.rs

2. RECONFIGURABLE HYBRID KINEMATICS MACHINE MOMA V3

MOMA V3 is established as a modular system on the basis of which the reconfiguration of both the hardware and software parts of the system can be performed.

The parallel mechanism is based on the modular principle, while its reconfigurability is reflected in the fact that the mutual position of modules of the translatory axis can be changed (rotation around the passive rotating axes, for angles α1 and α2), and it is also possible to change the lengths of the legs since there is a family of different legs lengths in three nominal sizes (li= 250, 195 and 180 mm). The configuration of the parallel mechanisms can be easily and quickly changed according to the building program.

As a basis for building a 3-axis reconfigurable machine with hybrid kinematics, selected types of 2-axis reconfigurable parallel mechanisms with an additional serial translatory axis are used, which is shown in Fig.1.

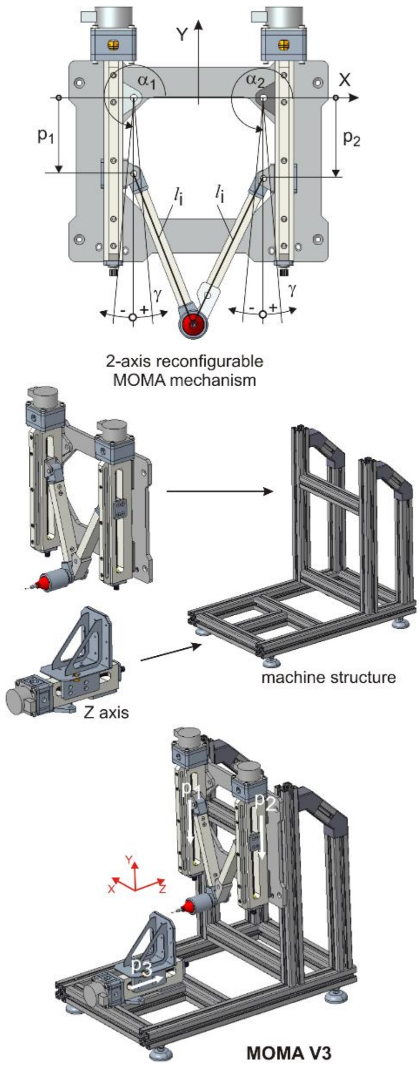

Figure 1. MOMA V3 – 3 axis reconfigurable hybrid kinematics machine

This machine represents a horizontal milling machine with a main spindle that is driven by a reconfigurable parallel mechanism along the X and Y axes, while the workpiece is moved along the Z axis (p3).

3. PROGRAMMING METHODS AND PROGRAM VERIFICATION

The paper considers various methods of programming the machine, suitable for education and rapid prototyping by subtractive processes. This approach can be considered a flexible programming method, which depends on the specificity of the machining task. The available standardized CAD/CAM system PTC Creo is used as the usual programming method. A specialized CAM (CUT 3D) is used for programming based on STL, for rapid prototyping. The application of the new programming method, better known as object-oriented programming or programming using the STEP-NC protocol, is also analyzed.

3.1 CAD/CAM

The basic structure of the system for programming using the CAD/CAM system (Zivanovic et al., 2021) is shown in Fig.2.

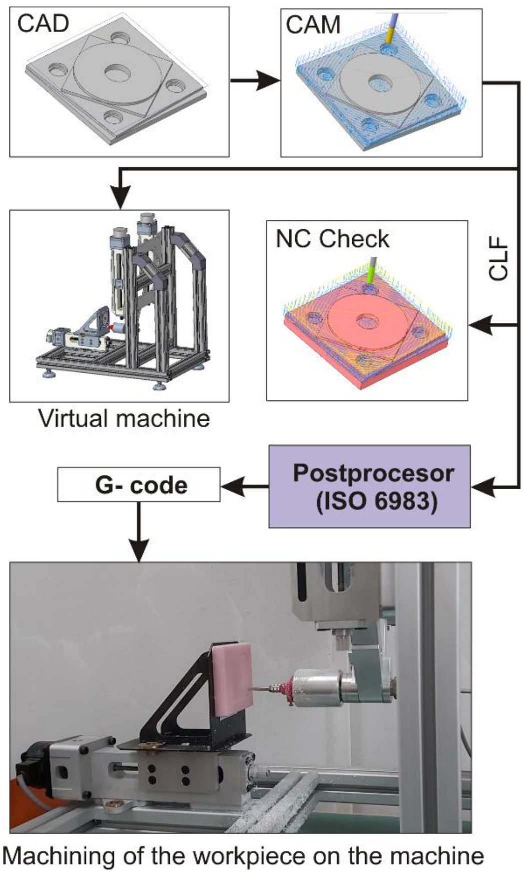

Figure 2. Programming method based on CAD/CAM system (PTC Creo)

Toolpath verification is possible by simulating moving tools along the tool path, material removal simulating, and simulating the working of virtual machine according to the given program in CLF format (Zivanovic et al., 2021, Zivanovic and Puzovic, 2022). Postprocessing is performed for a 3-axis horizontal milling machine, where G code is obtained according to the ISO6983 standard, which in this case is similar in format to programs for Fanuc CNC systems.

The model for the first part planned for machining is a part for testing the accuracy of CNC machine tools according to the (ISO 10791-7, 1998) standard. This part is scaled to fit the dimensions of the considered machine's workspace. The machining of this part is shown in section 4.

The model of this trial part was used in section 3.2 to prepare the program according to the new programming method based on the STEP-NC protocol.

3.2 Programming based on STL files

The considered machine belongs to the category of desktop milling machines that are used for rapid prototyping. In that case, a subtractive process is used, where part fabrication starts with a single block of solid material larger than the final size of the desired object, and the material is removed layer by layer until the desired shape is reached (Zivanovic et al., 2020).

A typical programming method based on STL files is shown in Fig. 3. First of all, need to prepare the model and convert it to STL format.

An example of programming method used specialized software CUT 3D (CUT 3D, 2023) for programming which is based on an STL file.

Programming includes the following stages: (1) loading of STL file and setting orientation and model size; (2) adjustment of the dimensions of the workpiece and the zero point; (3), (4), (5) planning roughing, finishing and cut out a strategy for milling; (6) preview machining – material removal simulation; (7) save toolpaths in G-code using appropriate postprocessor.

After transferring the program to the machine and preparing the machine for work, the fabrication of the prototype using desktop milling follows.

3.3 STEP-NC

This paper also discusses the programming method of the CNC machine tools based on ISO 10303-238 (AP-238) known as STEP-NC (Standard for Product Model Data Exchange for Numerical Control) (Step Tools, 2023), on the example of a 3-axis reconfigurable hybrid kinematic machine.

Today a new standard, STEP-NC, is being used as the basis for the development of the next generation of CNC controllers. The new STEP- compliant NC standard (STEP-NC) is based on the Standard for Exchange of Product data model (STEP) and has been developed to overcome the shortcomings of G-code (Zivanovic and Slavkovic, 2021; Bonnard et al., 2018). The STEP-NC provides object and feature-oriented programming methods and describes the machining operations executed on the workpiece, and not machine-dependent axis motions. It will be running on different machine tools or controllers (Sääski et al., 2005). If the CNC machine tool can interpret only G-code, corresponding translators or converters will be able to postprocess the STEP-NC program into different NC program types.

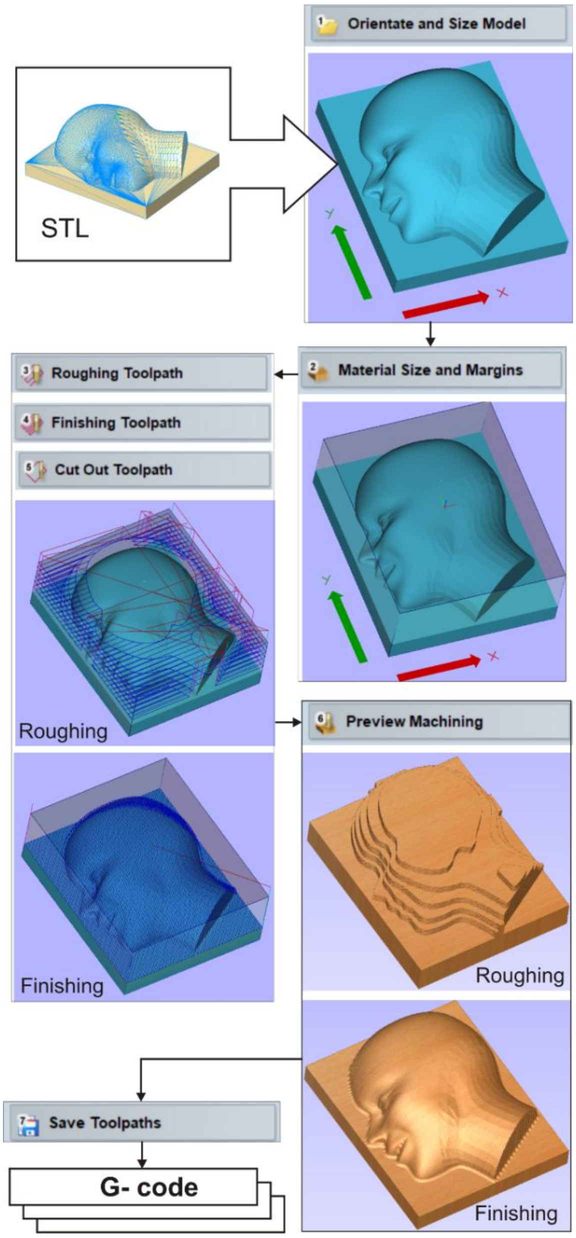

Figure 3. Programming method based on STL files (Cut3D)

At the moment, for most users, this new method of programming based on STEP-NC cannot be completely used with all the benefits provided by STEP-NC, because the resources for its development are not available to everyone.

This section shows two possible scenarios (Sc1, Sc2) code (Zivanovic and Slavkovic, 2021) for the indirect programming method based on STEP-NC which is verified on the considered machine, Fig.4.

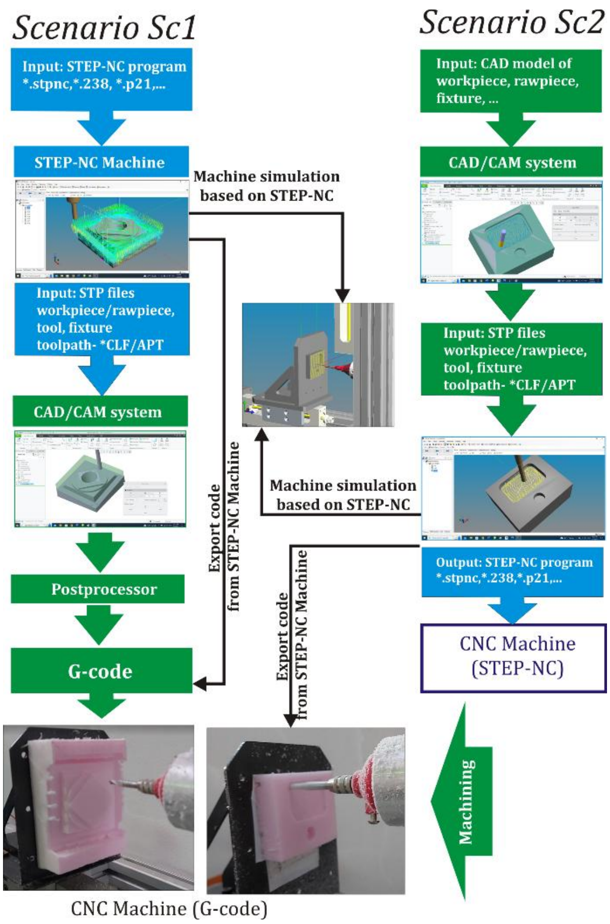

Figure 4. Scenarios for programming based on STEP-NC

These scenarios are (Zivanovic and Slavkovic, 2021):
Scenario 1 - using the native STEP-NC program, post-processing or converting the STEP-NC program into G-code and executing on the machine tool which is only able to interpret G-code files.

Scenario 2 - enables the generation of STEP-NC programs based on information from the available CAD/CAM system and licensed software STEP-NC Machine, to exchange technology and machining capabilities on machines that can directly interpret STEP-NC programs, but also on machines that can only execute G code using the first scenario.

Both scenarios [8] were tested on the 3-axis reconfigurable hybrid kinematic machine, by translating the original STEP-NC program to G-code for Fanuc CNC systems, while for scenario 2, an indirect programming method was used to generate the STEP-NC program, which is finally also converted to G code for machining process according to scenario 1.

4. VIRTUAL MACHINE IN CONTROL SYSTEM

The virtual machine has a significant role when developing a control system, primarily for machines that have hybrid kinematics, where it is necessary to integrate solutions of inverse and direct kinematics (Vasilic et al., 2017). In this case, the virtual machine was integrated with the LinuxCNC control system. The basic concept of a configuring virtual machine in the control system is detailed shown in (Vasilic et al., 2019).

Configuring a virtual machine is realized under the LinuxCNC software environment and relies on OpenGL and several interface classes written in Python programming language. Python is an interpreted programming language, suitable for scripting tasks, such as the development and configuration of virtual environments (Vasilic et al., 2019).

Based on the developed 2-axis reconfigurable virtual machine (Vasilic et al., 2019), an upgrade of the virtual machine was realized by adding a serial translatory axis and a new supporting structure of the machine.

New models of machine components were imported in ASCII STL format and are connected according to the kinematic model. The file for kinematics was also corrected by implementing an additional serial translatory axis and adjusting the directions of the coordinate axes according to the standard for machines with a horizontal main spindle.

When the virtual machine is configured and integrated within the control system during the execution of the machining program, the virtual machine components are moving identically as on a real machine, which was confirmed by machining experiments in section 5.

The illustration of the operation of the virtual machine is shown in Fig. 5, on the example for which the program was prepared by manual programming, and the machining itself on the machine is shown in Fig. 6.

The presented simulation is shown for the initial variant of the machine, which is active after switching on with parameters $\alpha_1 = \alpha_2 = 270^o$, and the lengths of the legs are 250 mm.

Virtual machine can easily reconfigure using the G code command because in virtual model implemented additional passive virtual axes for reconfiguration. The first two virtual driven axes are connected to the orientation of driven axes, are labelled as A and B, and take values of angles γ_i (γ_1 = A; γ_2 = B). The third virtual driven axis is labelled as C and is associated with leg length (l=C), Fig.6.

The result of such an approach to configuring the control is a reconfigurable virtual machine in Python 3D environment, integrated with graphic interface Axis. Initial configuration, has parameters length of leg l1 = l2 = 250 mm and driven axes parallel to Y axis, (A=0, B=0), Fig. 6a. Changing the machine configuration is done using standard G-code functions using - manual data input (MDI). Examples of configurations and adequate G-code commands reconfiguring four configurations are shown in Fig. 6.

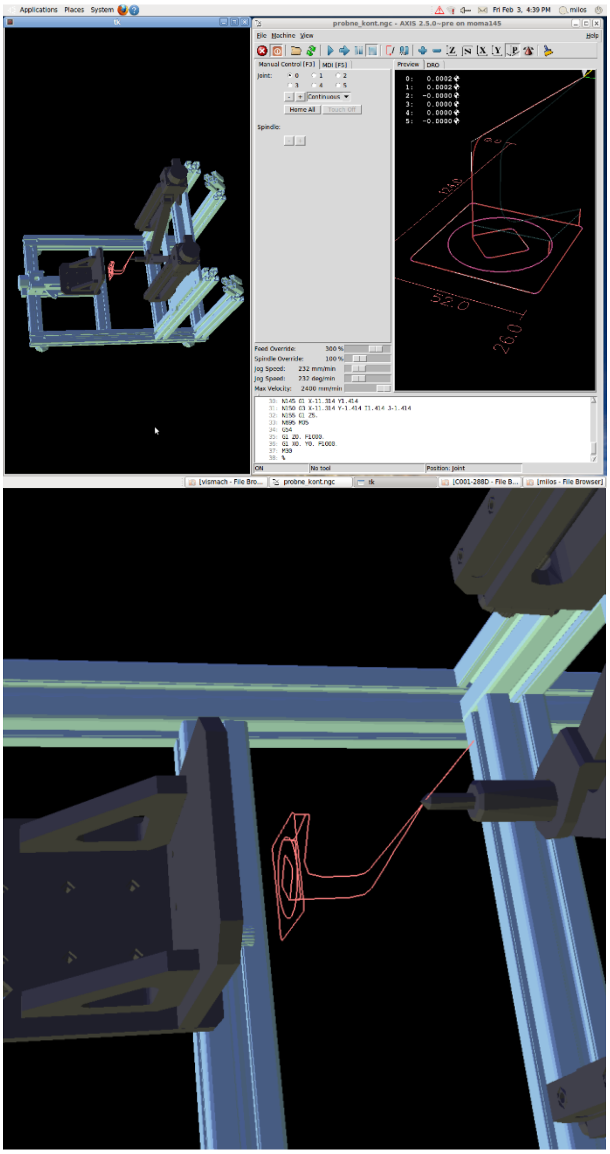

Figure 5. Virtual machine integrated with control system LinuxCNC

The virtual machine configured runs like a real machine and can be applied for verification of the program, before machining on a real machine. Also, this kind of virtual machine can be used for testing developed control, and analysis of machine operation before to its building.

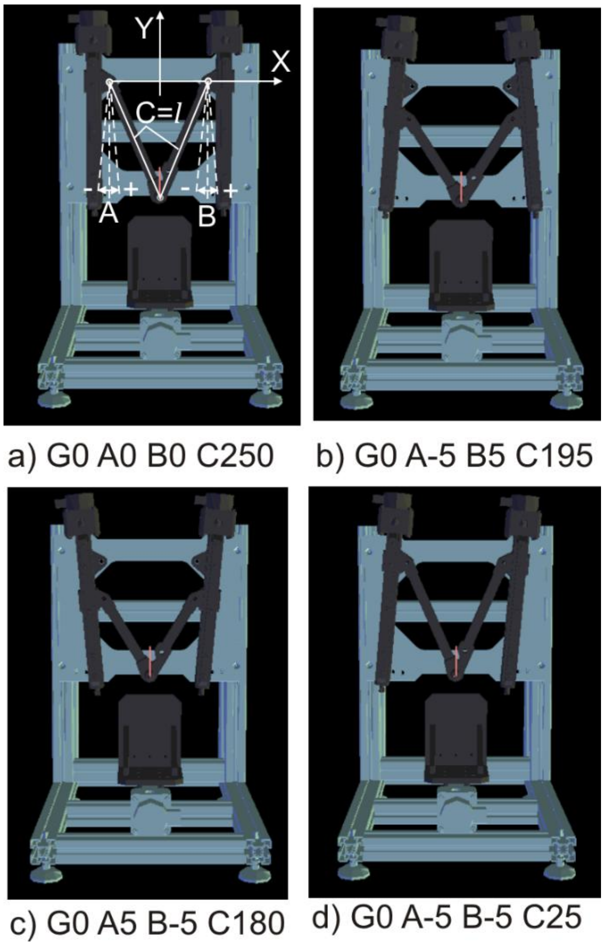

Figure 6. Reconfigurable virtual machine with additional passive virtual axes

5. MACHINING TESTS ON MACHINE TOOL MOMA V3

The verification of the prototype of the MOMA V3 machine was achieved during the trial run by testing different programming methods and machining the selected workpieces.

The trial work includes a complete check of the preparation of the machine for work, such as: checking the kinematics file in the control system according to the installed configuration of the machine, compiling the control, moving the machine to the reference position, setup of the workpiece within the boundaries of the workspace, determining the zero point, loading the program (G-code), testing program on the virtual machine (in the control system) and finally processing the part on the machine.

During the trial run, several groups of different parts were machined. Machined parts can be classified into parts of 2D and 2.5D geometry and 3D parts with sculptural surfaces.

In addition to the discussed programming methods, manual programming was also used for machining 2D circular and square contours to quickly check the accuracy and shape of the machined contours, Fig.7. For all samples, the trial machining was carried out in the soft material using the flat endmill tool with a diameter of 4 mm.

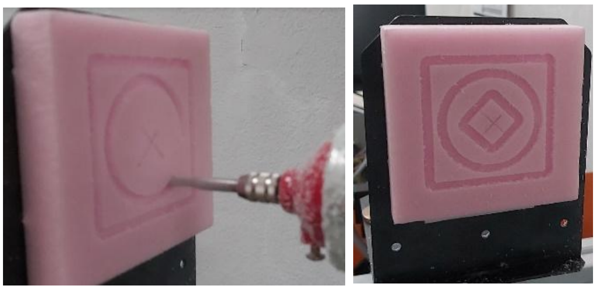

Figure 7. Machining test 1, Manual programming – 2D test

The following is an example from Fig.8, which refers to a scaled and adapted workpiece for testing the working accuracy of CNC machine tools according to the (ISO 10791-7, 1998) standard, which was machined and shown in Fig. 6b.

For the new programming method, both scenarios were tested for the application of the STEP-NC protocol. For scenario 1, the original STEP-NC program for machining the test part NAS979 was machined, Fig.9a. The possibility of translating the STEP-NC program into G-code and additional scaling due to the adaptation of the dimensions of the test part to the workspace of the machine was used.

An example of scenario 2, where the STEP-NC program was generated for three operations (working steps - WS): WS_1 milling of a flat surface, WS_2 milling of a cylindrical pocket, and WS_3 milling of a rectangular pocket, Fig.9b. Before the generation of the STEP-NC program, the models of the workpiece, rawpiece, and tool in STEP format, as well as the tool path (CLF) in the CAD/CAM system PTC Creo, were previously prepared. These data were integrated into the STEP-NC Machine software, after which the machining simulation was started.

A simulation of machining a part on a MOMA V3 virtual machine loaded from the machine library as a newly added machine, previously configured in the STEP-NC Machine environment , is shown in Fig.9b. After the simulation on the virtual machine in the STEP-NC Machine environment, the program was translated to G code using the export option to generate the Fanuc G-code that was loaded into the machine control, after which the machining of the part was realized.

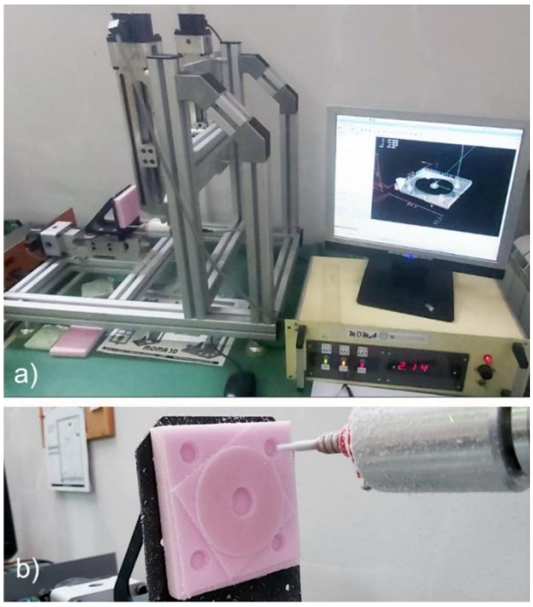

Figure 8. Machining test 2, CAD/CAM programming method – 2.5D test

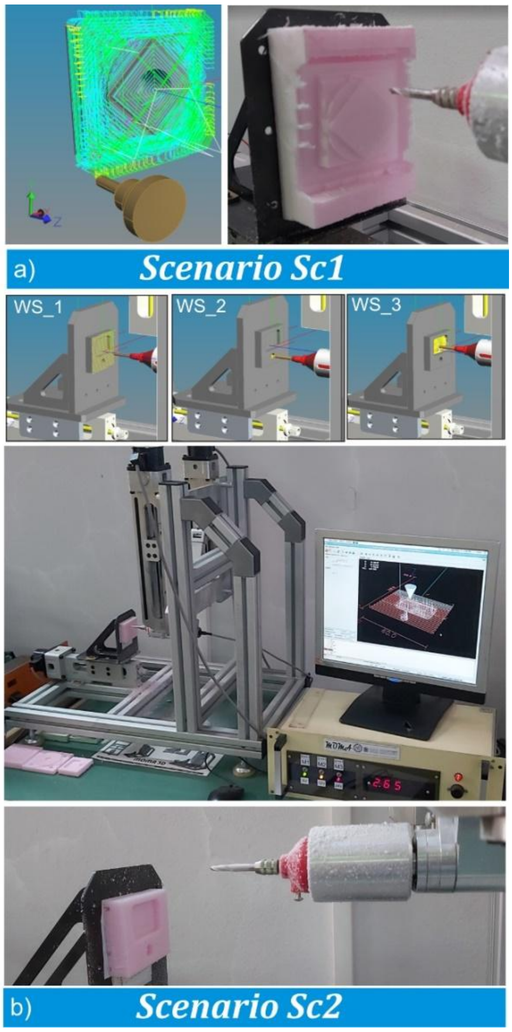

Figure 9. Machining test 3, STEP-NC

An example of machining parts of sculptural forms, as 3D geometry, is shown in the example of machining the profile of the human head (Fig.3), where the model is in STL format, which corresponds to the programming of machines for rapid prototyping. An example of this machining is shown in Fig.10. In this case, pre-machining is first carried out by removing the material by layers (Fig.10a,b,c), after which the final machining follows, to remove the characteristic stepped surface and obtain a smooth surface of the workpiece (Fig.10d,e,f).

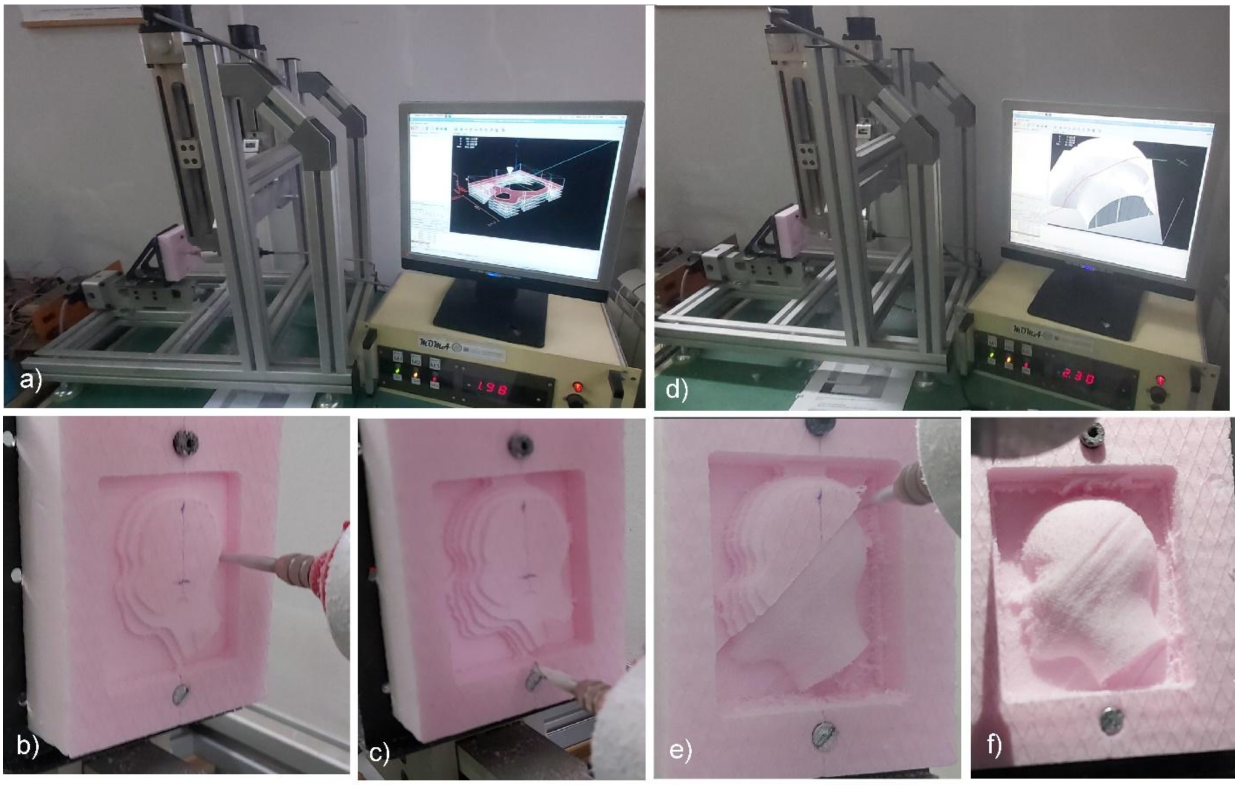

Figure10. Machining test 4, 3D tests, programming method based on STL files

Based on the milling selected parts carried out during the test operation of the machine, it can be concluded that the configuration of the prototype of the new machine with hybrid kinematics, as well as the appropriate developed control system for this machine, are correct, i.e., the virtual machine gives a faithful representation of the toolpath as on the real machine. Based on the performed experiments, it can be observed the suitability of the considered machine for: (i) practicing work on reconfigurable machining systems, (ii) verification of the NC program (G-code) on a virtual machine before its execution on the actual machine, (iii) practicing work in reconfiguring the control system when the machine configuration changed, (iv) education for programming with different programming methods.

6. CONCLUSION

The paper presents programming methods and testing of a horizontal 3-axis reconfigurable machine with hybrid kinematics, which contains a reconfigurable parallel mechanism MOMA that in combination with a serial translational axis, can represent a realistic concept for the construction of industrial machines of this type. The machine presented in this paper is an educational, laboratory-type machine with an open architecture control system (LinuxCNC).

The considered 3-axis reconfigurable machine with hybrid kinematics MOMA V3 follows current trends in the development of modern machine tools such as: (i) modernization of the mechanical structure in terms of introducing new concepts of machine tools based on parallel and hybrid kinematics; (ii) adaptation to the specific needs of certain branches of industry, by introducing reconfigurable and adaptable machine tools; (iii) application of the developed machine in education and research; (iv) consideration of new methods in CNC programming, based on the STEP-NC protocol; (v) adoption of digitization and virtualization approaches by the globally adopted strategies for improving industrial production under the Industry 4.0 paradigm.

Acknowledgement: The presented research was supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia through the project “Integrated research in macro, micro, and nano mechanical engineering” (contract no. 451-03-47/2023-01/200105 dated 3 February 2023) and by contract 451-03-47/2023-01/ 200066 dated 2023.

References:

Bonnard R., Hascoët J.Y., Mognol P., & Stroud I. (2018). STEP-NC digital thread for additive manufacturing: data model, implementation and validation. *International Journal of Computer Integrated Manufacturing, 31*(11),1141-1160. doi:10.1080/0951192X.2018.1509130

Cut 3D. (2023). Retrived from https://www.vectric.com/products/cut3d

ISO 10791-7 (1998). Test conditions for machining centres — Part 7: Accuracy of a finished test piece

Sääski, J., Salonen, T., Paro, J. (2005). Integration of CAD/CAM and NC with STEP-NC, VTT Information Service, Finland. Retrieved from https://publications.vtt.fi/pdf/workingpapers/2005/W28.pdf.

Step Tools, Inc., (2023). Retrived from http://www.steptools.com/

Vasilic, G., Zivanovic, S., & Kokotovic, B. (2017). Modelling and analysis of 3-axis reconfigurable hybrid kinematics mechanism with translatory actuated joints. Paper presented at the *Proceedings of the 5th International Conference on Advanced Manufacturing Engineering and Technologies, NEWTECH 2017*, Editors: Majstorovic, V., Jakovljevic, Z., Lecture Notes in Mechanical Engineering, pp. 429-441, 5th – 9th June 2017, Belgrade, Serbia, Springer International Publishing AG 2017.

Vasilic, G., Zivanovic, S., Kokotovic, B., & Dimic, Z. (2019). Configuring and analysis of a class of generalized reconfigurable 2-axis parallel kinematic machine. *Journal of Mechanical Science and Technology, 33*(7), 3407-3421. doi: 10.1007/s12206-019-0636-z

Zivanovic S., Popovic M., Vorkapic N., Pjevic M., & Slavkovic N. (2020). An Overview of Rapid Prototyping Technologies using Subtractive, Additive and Formative Processes. *FME Transactions, 48*(1), 246-253. doi:10.5937/fmet2001246Z

Zivanovic, S., Slavkovic, N. (2021). Programming of machine tools and robots for machining using STEP-NC in the era of Industry 4.0, Keynote Lecture. Paper presented at the *Proceedings of the 15th International Conference on Accomplishments in Mechanical and Industrial Engineering DEMI 2021*, pp. 3-26, University of Banjaluka, Faculty of Mechanical Engineering.

Zivanovic, S., Tabakovic, S., Zeljkovic, M., & Dimic, Z. (2021). Modelling and analysis of machine tool with parallel–serial kinematics based on O-X glide mechanism. *Journal of the Brazilian Society of Mechanical Sciences and Engineering, 43*(456). doi: 10.1007/s40430-021-03171-6

Zivanovic, S., Puzovic, R. (2022). Converting of virtual machine from different CAD/CAD environments and its configuring in desired environment, *Tehnika 72*(1), 47-54. doi: 10.5937/tehnika2201047Z

Sasa Zivanovic
University of Belgrade, Faculty of Mechanical Engineering,
Belgrade, Serbia
szivanovic@mas.bg.ac.rs
ORCID 0000-0003-4950-8203

Goran Vasilic
Academy of technical vocational studies, Belgrade, Serbia
gvasilic@atssb.edu.rs
ORCID 0000-0003-0465-192X

Zoran Dimic
LOLA Institute,
Belgrade, Serbia
zoran.dimic@li.rs
ORCID 0000-0002-1496-4981

Nikola Vorkapic
University of Belgrade, Faculty of Mechanical Engineering,
Belgrade, Serbia
nvorkapic@mas.bg.ac.rs
ORCID 0000-0001-5081-4828

Branko Kokotovic
University of Belgrade, Faculty of Mechanical Engineering,
Belgrade, Serbia
bkokotovic@mas.bg.ac.rs
ORCID 0000-0002-0847-3676

Nikola Slavkovic
University of Belgrade, Faculty of Mechanical Engineering,
Belgrade, Serbia
nslavkovic@mas.bg.ac.rs
ORCID 0000-0003-1147-284X