1. *Aleksandar KOŠARAC*, 2. *Saša PRODANOVIĆ*, 3. *Milan ZELJKOVIĆ*

POSSIBILITIES FOR INTERACTIVE CONTROL OF MACHINE TOOLS IN THE VIRTUAL REALITY ENVIRONMENT

1-2. University of East Sarajevo, Faculty of Mechanical Engineering, BOSNIA AND HERZEGOVINA
3. University of Novi Sad, Faculty of Technical Sciences, SERBIA

ABSTRACT: The virtual product development is a modern approach in the product development process. Apart from the visualization itself, the virtual product development encompasses the whole range of testing in the virtual reality (VR) environment. Different testing performed in the virtual reality environment before prototype production can increase productivity and decrease development time. As an illustration of possibilities of performing different kind of testing in VR environment, this paper shows simulation in the VR environment of the process controlled by programmable logical controller (PLC) and MATLAB/SIMULINK. As an interface between VR and MATLAB/SIMULINK environment VR Toolbox library is used, and between MATLAB/SIMULINK and PLC controller that is OPC standard.
KEYWORDS: Virtual Reality, PLC controller, OPC standard, MATLAB/SIMULINK

INTRODUCTION

Simultaneously with development of information technology, advanced approach to product development is also being improved. All of this is in order to shorten the product development process and minimize the number of physical prototypes. One of these approaches is the development of virtual products which, except for visualization, provides a wide range of testing of the product in the virtual reality environment [2].

The main idea of this paper is the simulation of process control that are realized on the machine tools in the 3D virtual environment. In this case, the virtual model of the machine tool (partially automated pillar drilling machine) is connected with a real PLC (programmable logic controller) using OPC server. This model does not include dynamic simulation. State of the system is changed by the appearance of discrete events. The result of the simulation are movements, which tools perform in 3D virtual environment depending on values of input variables. In this case, the input variable are logical (Boolean) variable representing the state of sensors, buttons, switches that are activated by user in the VR environment.

The simulation can be used in a demonstration of the machine tool functioning and for testing the control algorithm on interactive realistic models of machine tools in the VR environment, where the process is controlled by the real programmable logic controller.

DEFINING THE PROBLEM

The example that is discussed in this paper is pillar drill, shown in Figure 1. We need to create a virtual model of a scene that includes drilling machines, sensors, buttons, switches and other elements of the information lap, so that users can manage the main,

auxiliary and additional movements. Also, it is useful to provide possible solution for, partly automatized, drilling proces on this machine, which means defining the required number of inputs, actuator, and a program written for the PLC controller. It is necessary to develop a model in Matlab / Simulink environment, that manage with virtual models, in accordance with the defined initial conditions, together with PLC controller.
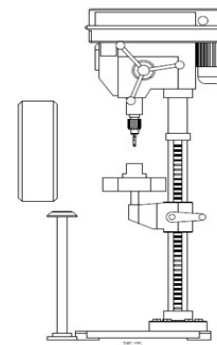

Figure 1. Pillar drill

A brief description of the drilling cycle on the considered machine tools is given below [8]. Workpiece is placed in clamping device, when the sensor LSPP (Limit Switch Part in Place) is activated. After the operator presses two buttons at the same time, the tool begins the main movement (rotation of drills) and simultaneously support motion (linear movement of drill, i.e. drilling machines spindle along the z axis downward). Drive of support movement is automated by kinematic pair of gear - rack. Rack is connected to the double-acting cylinder, that is controlled by the electrically activating monostable directional control valve 5/2. When the tool reaches its lower finite position, which corresponds to the completion of drilling, the sensor,

i.e. limit switch is being activated. LSDD (Limit Switch Drill Down). At that point, we have stop power electromagnet that activates monostable piston direction control valve 5/2, so that the spring returns direction control valve in its start position. Then begins the return movement, i.e. tool moves upward. After tool reach its upper limit position, appropriate limit switch is activated and entire system is being turned off. If the workpiece isn't in the clamping device, i.e. appropriate sensor doesn't detect its presence, process can't start. In order to increase operator safety, security doors close automatically after workpiece was placed and before movement of the tool. Safety door opens when drilling process is completed. Pressing the stop button, rotation of the tool is stopped and linear moving upward is started. After that the door is being open.

PLC CONTROLLER OMRON

Here we use programmable logic controller manufacturers OMRON, Series CJ1, which belongs to the class of microcontrollers [5]. This type of PLC are used in many industries, on the CNC machine tools for the control of individual modules, machines for different purposes, the packaging, the operations with various materials, in automated plants, etc. Product Group CJ series is intended for the use from simple sequential control to the powerful and fast models that can manage the system with a large number of input / output units. Complete modularity in composing this PLC provides the ability to apply a wide range of central processor units (CPU), which are fully compatible and can be quickly and easily combined with any other module of PLC. Central processor unit CPU11, which is used in this device, is designed for smaller systems, such as automatic machines, control devices, etc. The controller has a digital input unit with 16 points of connection, and a digital relay output unit with 16 points of connection.

Figure 2. Ladder Diagram

Figure 3. Wiring the input and output modules in case of physical model

Standard interface RS 232 is used for its connection with computer. Programming of PLC is performed using a graphical programming language called Ladder Diagram (LD) CX Programmer, figure 2, as a common platform for programming CS/CJ/C/CV serie OMRON controllers [4]. Figure 3 shows the wiring of I / O modules for the example under consideration.

APPLICATION OF OPC STANDARD FOR THE DATA EXCHANGE BETWEEN MATLAB SOFTWARE AND PLC

In order to exchange data between programmable logic controllers and virtual reality environments a suitable interface is required. As a solution, it is possible to apply the OPC standard, (abbreviation OLE for Process Control), which means Microsoft technology OLE Object Linking and Embedding [1], [7]. In this case, we use the CX OPC server that provides the ability to connect and exchange data between OMRON's PLC series CS/CJ/C/CV and CP, and applications operating in the Microsoft environment and have the support for the OPC standard. CX-Server OPC supports all Omron communication protocols. It should be noted that it is possible to use other software, such as MATRIKON OPC, which supports communications with OMRON PLCs. On the other hand, Matlab / Simulink has library of OPC Toolbox, which enables access to the OPC server in real time. In other words, utilization of this library enables reading of data in real time, that OPC server takes from controller or sending of data from Matlab to the PLC. Figure 4 shows the way of data exchange between the PLC and Matlab.
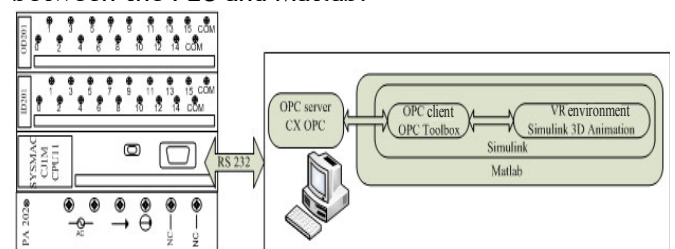

Figure 4. Data exchange between the PLC & Matlab[3]

CREATING MODELS OF DRILLS IN CATIA SOFTWARE AND VIRTUAL SCENE USING VRML LANGUAGE

Having in mind the complexity of virtual scenes, CATIA software for general purposes had been used to create 3D models, figure 5, recognizing that the same could be done by using any other CAD software system, which allows recording files in .wrl format.
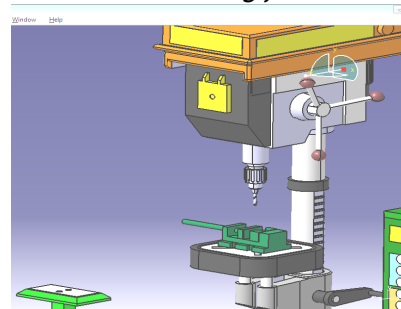

Figure 5. Model of pillar drill in the CATIA software

In developing the model it is necessary to plan in advance all movements that are performed by individual components or sub-assemblies in the VR environment, in order to properly set up a parent-child relationship, as well as take into account the orientation of the coordinate system in Matlab.

To create a virtual scene, VRML (Virtual Reality Modeling Language) is used , which is suitable for easy conversion with other 3D formats [6]. VRML is an
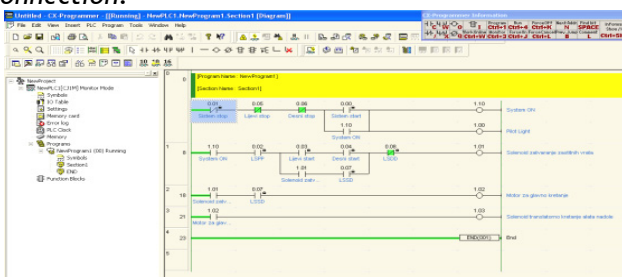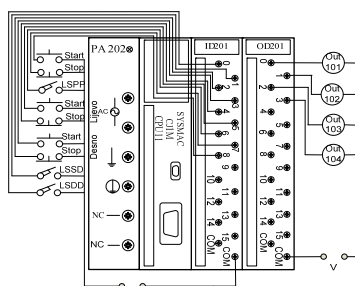
ISO standard for interactive, three-dimensional web graphics, or create virtual worlds on the Web. Virtual scene does not imply only objects that are visible in the scene, there are still lighting, background, camera position, the command to define the paths between nodes, the script nodes, which are easy to define the behavior of nodes, and many more, all to enable different interaction for user, as well as object control in the VR environment. Previous considerations [2], [3] have explained the application of different nodes to define geometric shapes, description of properties, as well as the nodes for grouping and their fields.

Without repeating of the foregoing, the paper contains a brief explanation of the specific mechanisms for communicating between nodes, important for this paper.

Since VRML has a specific purpose - to describe three-dimensional scenes and objects, it requires many features of generalized programming languages. The language itself is not object-oriented, but using Prototype mechanisms it can be derived object-oriented model, i.e. encapsulation and "reusability". This means, using prototype (PROTO node), allowing the definition of new nodes by the user. Using the prototype enables defining the objects in the scene that are "repetitive", such as sensors of object's presence, security door, buttons and switches for handling machinery, etc. Some nodes generate events in relation to the changing environment or user actions. These nodes are called sensors (Anchor, Collision, CylinderSensor, PlaneSensor, ProximitySensor, SphereSensor, TimeSensor, TouchSensor, VisibilitySensor) and represented essential elements that provide interaction with user. This approach uses a Touch Sensor that detects the mouse interaction with geometry. Event management is carried out using the ROUTE command, i.e. by writing a special script. Once the user completes an event (a mouse click in this case) to a geometric shape, ROUTE command defines paths between nodes, which generate events and those who receive them. At that time, the given geometric shape, sensor and ROUTE command are in the same node for grouping.

VRML files are displayed in a VRML browsers that can be a standalone application or VRML plugin html browser. The most popular VRML browsers are: Cortona VRML Client, Blaxxun Contact, Cosmo Player and others. For the purposes of this paper, the virtual scene is displayed in the application called Simulink 3D Animation Viewer, which is part of the library of Simulink 3D Animation.

MATLAB SIMULINK MODEL

Movement of kinematics modules as the basic kinematic entity, is being controlled from Simulink environment, figure 6. In that way, the VR Sink and VR Source blocks of libraries Simulink 3D Animation are used for the connection between Simulink and virtual scene (Figure 8). The blocks: OPC Configuration, OPC Read and OPC Write from the OPC Toolbox library are used for connecting with PLC.

VR Sink block "enters" the value of its ports into virtual world through the fields defined in the dialog box called Block Parameters, figure 9.

VR Sink block is responsible for the display of virtual scene in the application Simulink 3D Animation Viewer. The application is run after the user double-clicks the left mouse button on the VR Sink block. Inputs of VR Sink block directly change the characteristics of objects in the virtual scene.

Defining characteristics of virtual objects, which are changing, enables selection of the option "Simulation > Block Parameters ..." from the menu into application Simulink 3D Animation Viewer. On the right side of the window is a tree for a given virtual scene that is defined in the VRML file, whose name is shown in the left side of the window. This tree shows the structure of the VRML file. Blocks with names are marked with red arrow and available from the Matlab environment. Blocks without name, but whose children have the names are also marked with a red arrow. Fields of blocks that can be changed, have a square next to the name. Selecting a square enables changing of field values from Simulink (simultaneously, input with corresponding name is generated on the VR Sink s-block). If it is need to make changes in the VRML file, the option "File> Open in Editor" from application Simulink 3D Animation Viewer should be chosen. Then the .wrl file is opened in V-Realm Builder VRML editor, which is distributed with MATLAB, figure 7.
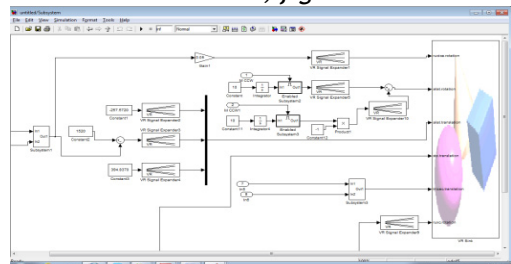

Figure 6. Block diagram – connection between Matlab/SIMULINK and VR environment
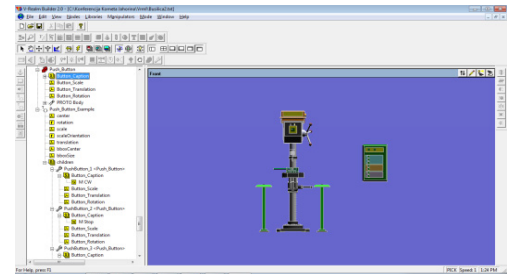

Figure 7. Virtual world shown in V-Realm Builder editor

Figure 8. Virtual scene

Particular changes to VRML files can be manually performed, due to .wrl is text file, but it is necessary to know the syntax of VRML. Application of VRML editor is definitely more comfortable and easier way, but not always sufficient.

It has already been said that the Simulink model does not involve dynamic simulation, and system state is being changes with the occurrence of discrete events.

But simulation is changing in time, which is called a continuous simulation. The reason for this is in the fact that the position of the tool is determined by the output of the block Integrator, whose values is changed continuously in time, figure 10.
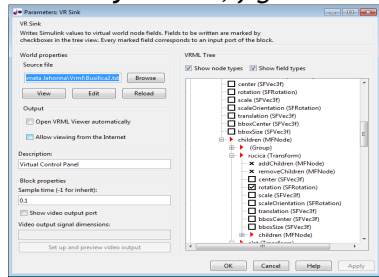

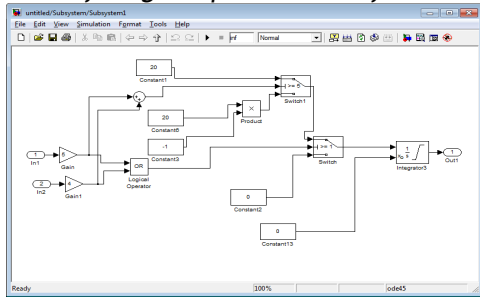
Figure 9. Defining the parameters of VR Sink block

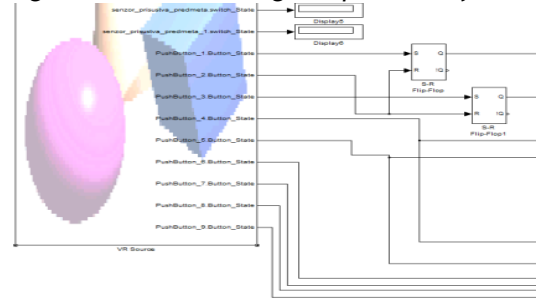Figure 10. Determining the position of tools

Figure 11. VR Source block

VR Source block is very similar to the VR Sink block. Using this block, we can run applications Simulink 3D Animation Viewer, and edit .wrl files. Also, the defining characteristics of virtual objects can be performed as previously described. The difference is that this block reads different values from the virtual world and forwards them to the Simulink model. Figure 11 shows VR Source block, where the outputs contain values of logical variables, that represent the state of sensors, switches and buttons in the virtual scene. Blocks OPC Read and OPC Write communicate with OPC server and OPC server with PLC. Figure 12 displays the address (called tags) input and output terminals on the PLC that are defined on the OPC server.

Figure 12. Addresses of input and output terminals

OPC Write block sets the (forced) state of bits at the input terminals, depending on the initial conditions (sensor state), while the OPC Read block takes the value of the output terminals of the controller, that represent a control signals for kinematic modules of machine tools in a virtual environment.

CONCLUSIONS

The paper presents the possibilities of using connection between Matlab and PLC, where OPC server has utilized as interface for data exchange. In the virtual reality environment, in addition to the visualization, various tests of products can be performed. One of the approaches involves the management with 3D virtual models using real programmable logic controllers. This concept can be used for different purposes like testing of the developed control algorithm, shortening product development time and increasing productivity, as well as for educational purposes.

Realization requires a standard PC platform, which is an advantage because of the low cost.

The disadvantage is quite a complex process of creating such a scene since it is required the knowledge of a CAD tool (which can export geometry into .wrl file format), VRML language, JavaScript i.e. Java programming language, Matlab, and the knowledge of working with PLC, i.e. programming of PLC.

REFERENCES

[1] Guzina, M., Stupar, M., Naumović, M.: Possibilities of communication between MATLAB/Simulink environment and processes controlled by programmable logic controllers using OPC concept, Infoteh-Jahorina Vol. 8, Ref. A-15, p. 66-70, March 2009. (in Serbian).

[2] Košarac, A.: Design automation and visualization of flexible technology structures running, Master thesis, Faculty of Technical Sciences, Novi Sad, 2010. (in Serbian).

[3] Košarac, A., Zeljković, M.: Simulation of process controlled by Programmable Logic Controller PLC in the Virtual Reality Environment, Procedings – CD rom, 11th INTERNATIONAL SCIENTIFIC CONFERENCE MMA 2012 – Advanced Production Technologies, Faculty of Technical Sciences, Novi Sad, 2012, pp. 239-242, ISBN 978-86-7892-419-4

[4] Matić N.: Introduction to industrial PLC controllers, mikroElektronika, Beograd, 2001. (in Serbian).

[5] OMRON, Technical Library, Multilanguage edition, 2005.

[6] Pașc, I., Țarcă, R., Popențiu-Vlădicescu, Florin.: The VRML model and VR simulation for a SCARA robot, Annals of the Oradea University, Fascicle of Management and Technological Engineering, Volume VI (XVI), 2007.

[7] Persin, S., Tovornik, B., Muskinja, N.: OPC-driven Data Exchange between MATLAB and PLC-controlled System, Int. J. Engng Ed. Vol. 19, No. 4, pp. 586-592, 2003 TEMPUS Publications.

[8] Webb, W. J., Reis, A. R.: Programmable Logic Controllers, Principles and Applications, Fifth Edition, Pearson Education, Inc., Upper Saddle River, New Jersey 07458, 2003