ISBN 978-9940-611-04-0

COAST
INTERNATIONAL CONFERENCE ON ADVANCES IN SCIENCE AND TECHNOLOGY

I INTERNATIONAL CONFERENCE ON ADVANCES IN SCIENCE AND TECHNOLOGY

PROCEEDINGS COAST 2022

FACULTY OF MANAGEMENT HERCEG NOVI

HERCEG NOVI, MONTENEGRO

26-29 MAY 2022

ORGANIZER

Faculty
OF MANAGEMENT
HERCEG NOVI

FAKULTET ZA MENADŽMENT HERCEG NOVI

I INTERNATIONAL CONFERENCE ON
ADVANCES IN SCIENCE AND TECHNOLOGY

PROCEEDINGS COAST 2022

HERCEG NOVI, 26-29 MAY 2022

CO-ORGANIZERS

I INTERNATIONAL CONFERENCE ON ADVANCES IN SCIENCE AND TECHNOLOGY
COAST 2022

Novosibirsk State Technical University NETI

YAROSLAV-THE-WISE NOVGOROD STATE UNIVERSITY

Nikola Tesla UNIVERZITET UNION

ST. PETERSBURG STATE UNIVERSITY OF FILM AND TELEVISION

УДМУРТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ UDMURT STATE UNIVERSITY

University of Maribor
Faculty of Tourism

ФБЕ FBE

ИБЕ IBE

fpep BAR

Academy for human resource development Vimal

PMC PROJECT MANAGEMENT COLLEGE

VUB VISOKA ŠKOLA ZA USLUŽNI BIZNIS

UNIVERZITET PIM BANJA LUKA
UNIVERZITET ZA POSLOVNI INŽENJERING I MENADŽMENT

University under the IPA EurAsEC

PROCEEDINGS

I INTERNATIONAL CONFERENCE ON ADVANCES IN SCIENCE AND TECHNOLOGY
COAST 2022

Book title:
Proceedings COAST 2022

Organizer and publisher:
Faculty of Management Herceg Novi

For publisher:
Irena Petrušić, PhD, Dean

Design and Computer processing:
Sanja Samardžić, MSc, Dragana Savić, Spec. Sci

CIP - Каталогизација у публикацији
Национална библиотека Црне Горе, Цетиње

INTERNATIONAL conference on advances in science and technology (I ; 2022 ; Herceg Novi)
Proceedings/International conference on on advances in science and technology, Herceg Novi, May, 26-29, 2022 = Zbornik radova / Međunarodna konferencija o savremenim dostignućima u nauci i tehnologiji, Herceg Novi, 26-29 maj 2022. godine : Fakultet za menadžment, 2022 (Herceg Novi). - 1011 стр. : илустр.

Радови на срп. и енгл. језику. - Текст ћир. и лат. - Напомене и библиографске референце уз текст. - Библиографија уз сваки рад. - Сажеци на енгл. и срп. језику уз радове.

ISBN **978-9940-611-04-0**
COBISS.CG-ID **23232772**

PROCEEDINGS

I INTERNATIONAL CONFERENCE ON ADVANCES IN SCIENCE AND TECHNOLOGY
COAST 2022

PROCEEDINGS

I INTERNATIONAL CONFERENCE ON ADVANCES IN SCIENCE AND TECHNOLOGY
COAST 2022

PROCEEDINGS

I INTERNATIONAL CONFERENCE ON ADVANCES IN SCIENCE AND TECHNOLOGY
COAST 2022

NOTE:
The authors have full responsibility for the originality and content of their own papers.

Contents/Sadržaj

COMMISSIONING OF ABB COLLABORATIVE ROBOT

Vojislav Vujičić[1], Milan Marjanović[1], Ivan Milićević[1]
[1]University of Kragujevac, Faculty of Technical Sciences Čačak, (Serbia)

Corresponding author e-mail address: vojislav.vujicic@ftn.kg.ac.rs (V. Vujicic)

ABSTRACT:
The aim of this paper is to show the basic steps that are necessary to put new laboratory gear into operation. The laboratory setting is described. ABB RobotStudio was used to create a virtual laboratory by adding an ABB IRB 14000 robot to the work area, a virtual controller, and a simple program for manipulating the workpiece from one side to another with a change of arms. Simulation of the program was done. The commissioning of a robot on a real platform in the laboratory has been done.

Keywords: *ABB, IRB 14000, collaborative robotics, RobotStudio, virtual laboratory, commissioning*

---

1. INTRODUCTION

Robots are quickly becoming a prominent part of the manufacturing industry. Robots are used in a number of ways within different industries, but they can make an especially large impact when it comes to industrial automation.
Traditional robots have been physically boxed off from humans, executing tasks that are isolated from employees, to protect their safety. Collaborative robots, on the other hand, are in such a way to limit the amount of speed and force they can apply, making them safe for humans to be around and work with them together. [1-4]
Collaborative robot systems offer the potential to improve the way we work. They have grown rapidly because of the strong promotion of Industry 4.0 and the launch of more affordable robot technologies. Only a decade ago, they were treated with skepticism, yet today they are the fastest expanding area of industrial robotics. Collaborative robots are quickly becoming a prominent part of the manufacturing industry.
These robots often need human assistance when they're moving materials or loading them into other devices These collaborative robots are designed to work and assist humans, instead of taking their jobs away. They are programmed to perform repeatable tasks where people are needed but not their expertise. So, they are used for simple, repetitive tasks that are dangerous or difficult for humans to perform, such as assembly line operations, machine tending, arc welding, handling of hazardous materials, lab testing and other manufacturing tasks.
The main benefits of collaborative robots are: reducing injuries, decreasing manual material handling workers, reducing production time, decreasing maintenance costs for robotics equipment, increasing production capacity, etc.

2. LABORATORY SETUP

The laboratory for this study is in the Science and Technology Park in Čačak and was funded by the Ministry of Education, Science, and Technological Development of the Republic of Serbia. In this laboratory, two robotic cells were installed. One is the ABB IRB 120 multipurpose industrial robot with six degrees of freedom, described in detail in [5]. Another one is the ABB IRB 14000 YuMi collaborative robot. Fig 1.

Fig 1. ABB IRB 14000 collaborative robot [6]

This is a dual-arm robot made for collaboration with a human on the same task. It has seven degrees of freedom arms, which give flexibility and agility in production with position repeatability of 0.02mm. The robot has a reach of 559mm and a payload capacity of 500g per hand. The IRB 14000 has a smart gripper as an end-effector for part handling and assembling. The gripper has servo fingers as the default module and a vacuum suction tool or vision toll optionally. In this case, one hand has servo and vacuum modules. The second has a servo, vacuum, and vision (Cognex AE3 camera) module [6,7].

Fig 2. Right and left robot Smart Gripper [6]

Both robots are placed in a 1400x2400 work area with a Plexiglas safety zone. The working area has aluminum profiles for mounting additional equipment such as sensors, part holders, part racks, etc. In front of robots, there is a rail for sensors and a terminal box. The robot control unit and terminal box are connected to the control desk equipped with a PLC controller and an HMI panel. With this equipment, robots, work area, safety zone, control desk, and sensors, a lot of industrial applications can be tested in the laboratory. A laboratory setup is presented in Fig 3.

Fig 3. Laboratory setup

2. MAKING VIRTUAL ROBOT ENVIRONMENT

The best way to program ABB robots is to use the ABB RobotStudio software platform. The software also has a virtual controller for robot programming. It is also good for the simulation of robots and other moving mechanisms. This software allows robot programming and program changes without disturbing or stopping production. RobotStudio has been used to make virtual laboratories, and program robots offline. This platform has four main parts: modeling, simulation, controller, and RAPID [8,9].

Modeling allows creating virtual environments by making or importing 3D models of mechanical parts and machines in a production line. The virtual environment is useful to avoid mechanical collisions between robots and other parts of a production line. In this case, the virtual laboratory. Making a virtual laboratory starts by creating a new *Empty Station*. Following that, a working environment was established. The next step is importing a mechanical 3D model of the work area with a safety zone from the 3D modeling software. This step has been thoroughly described in [5]. After adding the work area, a robot must be inserted. ABB IRB 14000 has been added to the *ABB Library*. The position of the robot is adjusted (position (x, y, z): 1604,900, 853, orientation (x, y, z): 0,0, -90). The next step is to insert a smart gripper on the left and right arms. The left arm has servo fingers and a vacuum, while the right has servo fingers, a vacuum, and vision modules. Those grippers must be attached to each hand. After inserting the gripper, it is necessary to add a 3D model of the workpiece to complete the mechanical model of the virtual laboratory. A complete mechanical virtual model is presented in Fig 4.

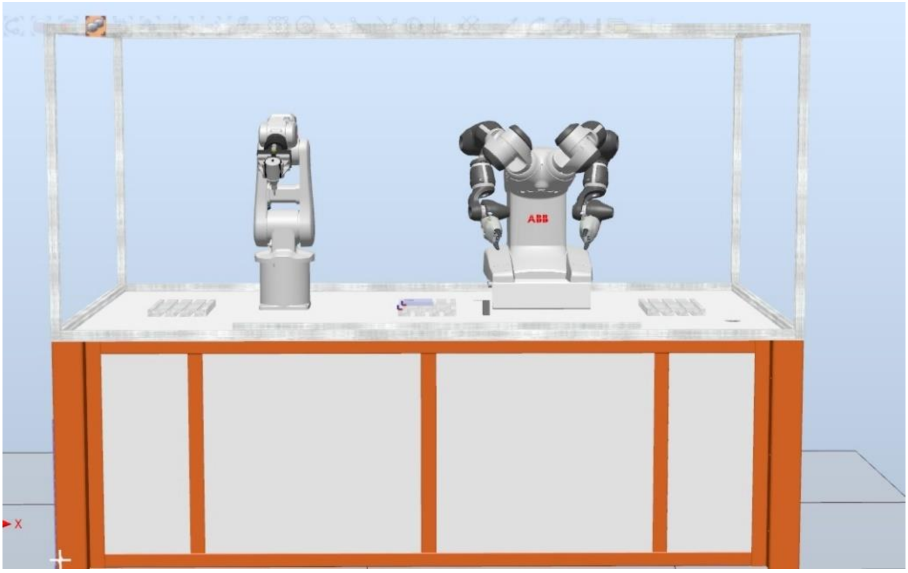

Fig 4. Laboratory virtual station

After adding all the necessary mechanical parts into the virtual model, a virtual controller must be added to the project. Adding a controller is done by the following procedure: in the *Home,* section, select the option *Virtual Controller – New Controller.* Controller with RobotWare version 6.11.01 for IRB 14000 0.5kg 0.5m has been added. This step is crucial for offline robot programming. The program that is going to be made is to lift the workpiece from one side of the work area with one (right) hand, transfer it to another (left) hand, and put it down on the other side of the work area.

Programming ABB robots starts by defining *Targets.* The *target* is the point that the robot must reach. The main target is the home position target. The easiest way to achieve this target is to send grippers to their home position. This should be done with the following instructions: *Gripper-Modify-Move To Pose-Jump Home*. In this way, the gripper is placed in the home position, and this position has to be saved into a target with the *Home-Teach Target* instruction. This procedure has to be done for both hands. Each hand has its own targets. The right arm has to approach the workpiece from the home position to a position for catching trout (above the workpiece) a few targets (Target 10, 20, 30). At Target 30, the Gripper has to be closed to grab a workpiece. Then it goes to the target for exchange - Target 40. Figure 5a shows the right Smart Gripper position in defined targets. From those targets, a path has to be made (Path_R). This path contains all the made targets and *Move (J-joint or L-linear)* instructions to make a move from one to another target. The left arm has targets to approach the exchange point with the right arm (Target 10, 20, 30), the extraction from exchange point (Target 35), Targets (40, 50, 60, and 80) for transferring workpieces to another point on the work area. The last Target 70 is an extraction from Target 80 to avoid a Gripper collision with the workpiece. Figure 5b shows the position of the left Smart Gripper in defined targets. With those targets, a path has been made to (Path_L).

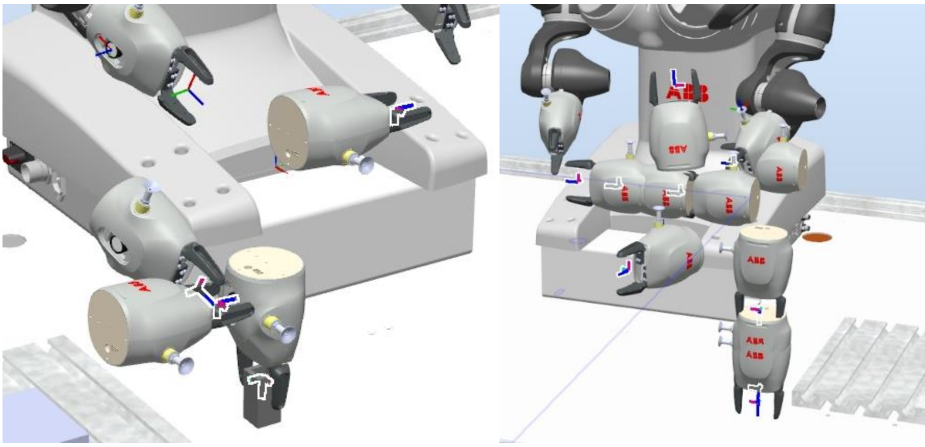

Fig 5a. Right gripper in target positions, **Fig 5b.** Left gripper in target positions

After the programming is finished, the simulation could be done to check how the program works. The simulation starts by selecting the option *Play* in *Simulation* part. When the simulation starts, the robot goes from target to target, making moves defined by paths. The first simulation shows that arms are not synchronized, so the program has to be changed. Changes that have to be made are:

a) The right arm has to wait until the left arm takes over the workpiece.
b) Wait until the left arm extracts from the right gripper zone.
c) The right arm can go to a home position, and the left arm can put the workpiece in its location in the work area.
d) Wait until both arms are in the home position.

The program must have instructions for the synchronization of arms. The easiest way to add those instructions is to go to RAPID (code editor) and add the necessary instructions. To make changes in the RAPID program, they have to be transferred from the virtual station to RAPID with the option: *Home – Synchronize – Synchronize to RAPID.* After that, code lines could be added. To make synchronization of arms task lists and identification points for synchronization possible, they have to be added to the arms RAPID code. The code lines are shown below:

```
PERS tasks task_list{2}:=[["T_ROB_L"],["T_ROB_R"]];
VAR syncident sy01;
VAR syncident sy02;
VAR syncident sy03;
```

After that, *WaitSyncTask* instructions have to be added into the paths, in the position where one arm has to wait for another to finish the movement. Modified RAPID programs for the right and left arm are shown side by side below:

```
PROC Path_R()
    Reset R_Grip;
```

```
PROC Path_L()
    Reset L_Grip;
```

```
    MoveJ
Home_R,v300,z100,Servo\WObj:=wobj0;
    WaitSyncTask\InPos,sy01,task_list;
    MoveJ
Target_10,v300,z100,Servo\WObj:=wobj0;
    MoveJ
Target_20,v300,z100,Servo\WObj:=wobj0;
    MoveL
Target_30,v300,fine,Servo\WObj:=wobj0;
    WaitTime 1;
    Set R_Grip;
    MoveJ
Target_40,v300,z100,Servo\WObj:=wobj0;
    WaitSyncTask\InPos,sy02,task_list;
    Reset R_Grip;
    WaitSyncTask\InPos,sy03,task_list;
ENDPROC
```

```
    MoveJ
Home_L,v300,z100,Servo\WObj:=wobj0;
    WaitSyncTask\InPos,sy01,task_list;
    Set Reset_Pos;
    Reset Reset_Pos;
    MoveJ
Target_10,v300,z100,Servo\WObj:=wobj0;
    MoveJ
Target_20,v300,z100,Servo\WObj:=wobj0;
    WaitSyncTask\InPos,sy02,task_list;
    MoveL
Target_30,v300,z100,Servo\WObj:=wobj0;
    WaitTime\InPos,1;
    Set L_Grip;
    MoveL
Target_35,v300,z100,Servo\WObj:=wobj0;
    WaitSyncTask\InPos,sy03,task_list;
    MoveJ
Target_40,v300,z100,Servo\WObj:=wobj0;
    MoveJ
Target_50,v300,z100,Servo\WObj:=wobj0;
    MoveJ
Target_60,v300,z100,Servo\WObj:=wobj0;
    MoveL Target_80, v300, fine, Servo;
    WaitTime\InPos,2;
    Reset L_Grip;
    MoveJ
Target_70,v300,z100,Servo\WObj:=wobj0;
ENDPROC
```

Each RAPID code change necessitates reverse synchronization: *RAPID – Synchronize – Synchronize to STATION*. Then simulation of a changed program could be performed. A simulation of the final version of the program with the right (red) and left (blue) tool center point (TCP) trace is shown in Fig 6.

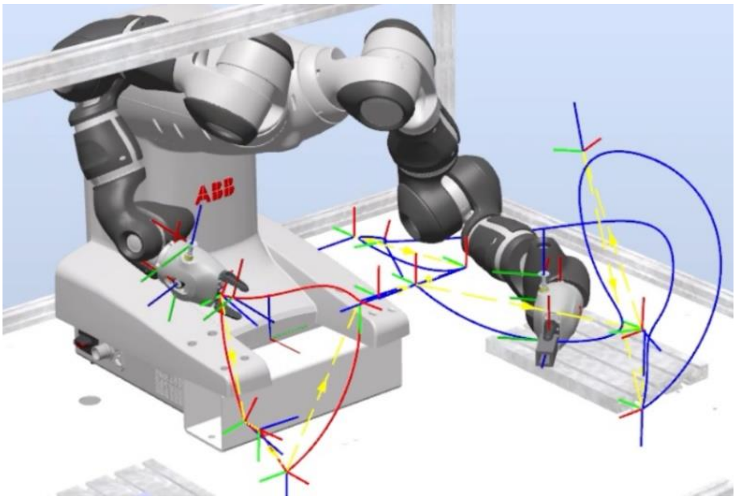

Fig 6. Simulation of robot movement with TCP trace

2. ROBOT COMMISSIONING

If the program simulation is good, the program can be written into the real controller and commissioning can start.
The PC and robot need to be connected via LAN (Service port on the robot). Then a real (online) controller has to be added into RobotStudio with the option: *Controller – Add Controller – Add Controller (available on a network)-OK.* An online controller will be added. The next relation between virtual and online controllers must be made using the option: *Controller – Create Relation.* Making a relation between these two controllers allows transferring the program from virtual controller to online (from PC to robot) and vice versa.
After transferring the program from the computer to the robot, the commissioning can be started. The commissioning is performed with FlexPendant. It is a lightweight remote-control unit equipped with an HMI touch panel, joystick, different types of buttons, an E-stop button, and a „dead man's switch" connected to a robot controller. FlexPendant is shown in Fig 7.
The first thing to do is debug the program. A robot has to be in *Manual* mode. In this mode, a robot is moving at a reduced speed. On an HMI panel, the FlexPendant programs for the right and left arms have to be opened from *Menu – Program Editor – (T_ROB_R and T_ROB_L).* Then the *Debug* option needs to be selected, and *PP to Main* needs to be executed. This will put the program pointer to the first instruction in the main function and has to be done for both arms. The next step is to start the program step-by-step, using the next instruction button. This way, the robot executes one instruction at a time, so target positions can be checked. If some targets have to be repositioned, *Jogging* mode has to be enabled from the menu. Jogging mode is active for one arm (Mechanical unit) at a time. Repositioning of the arm is done using a joystick.

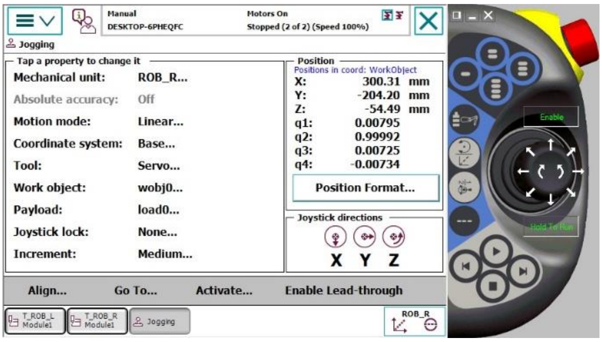

Fig 7. FlexPendant jog mode

If there is a need to change the position of a target, by jogging the robot is brought to the desired position. Then the position is modified using *Modify Position* option. This position has to be confirmed. In this case position of Target 30 of the right arm has been adjusted. This is a target for picking a workpiece. And Target 30 and 70 of the left arm have been modified to pic a workpiece from the right arm and put the workpiece down in the work area.

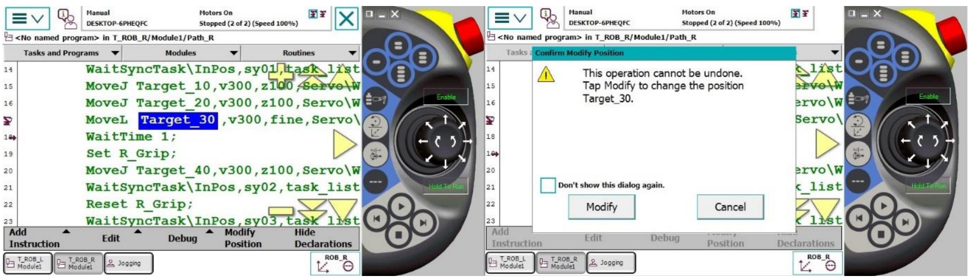

Fig 8. FlexPendant position modification

FlexPendant can also be used for other types of settings and adjustments in the robot controller. When all adjustments have been done, a robot should be switched to *Auto*, production mode. ABB IRB 14000 collaborative robot in motion has been shown in Fig 9.

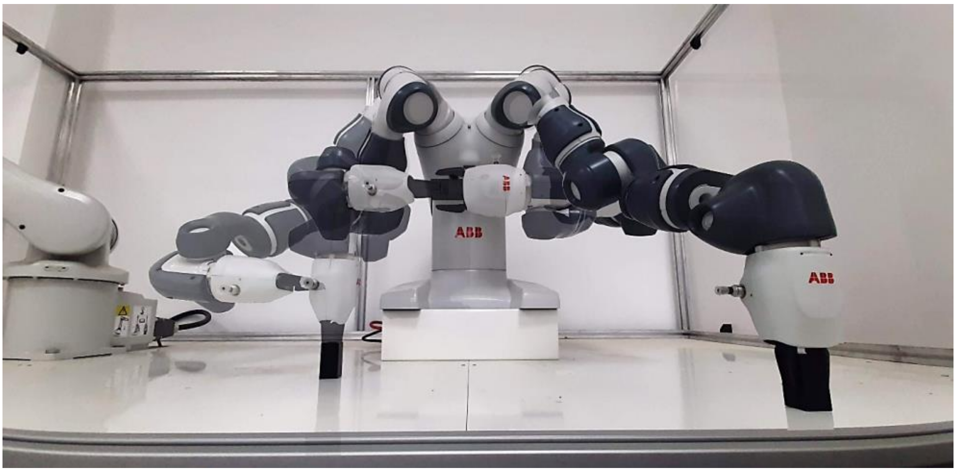

Fig 9. ABB IRB 14000 in motion

3. CONCLUSION

The opening of a new laboratory equipped with two ABB robots within the Science and Technology Park in Čačak created new opportunities for the development of robotics at the Faculty of Technical Science in Čačak. A virtual laboratory with ABB IRB 120 and ABB IRB 14000 has been formed using RobotStudio software. This allows programming a robot from home using a virtual environment.

A sample program has been made in a virtual laboratory, checked, and adjusted through simulation. The program was put into the real controller. It was debugged, and some targets were adjusted. After that, the ABB IRB 14000 commissioning has been done.

In this way, it went through the whole procedure of commissioning an industrial robot.

The following research will be focused on collaboration on difficult tasks, collaboration with humans, and usage of vison system on Smart Gripper.

4. ACKNOWLEDGMENT

This study was supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia, and these results are parts of the Grant No. 451-03-68/2022-14/200132 with University of Kragujevac - Faculty of Technical Sciences in Cacak.

5. LITERATURE

[1] Collaborative Robots [Online]. Available: https://new.abb.com/products/robotics/collaborative-robots [Accessed 11.3.2022.]

[2] (2015) KUKA LBR iiwa. [Online]. Available: https://www.kuka.com/en-us/products/robotics-systems/industrial-robots/lbr-iiwa [Accessed 11.3.2022.]

[3] (2015). UR3 Robots. [Online]. Available: http://www.universal-robots.com/products/ur3-robot/ [Accessed 11.3.2022.]
[4] (2015). Rethink Robotics. [Online]. Available: http://www.rethinkrobotics.com/ [Accessed 11.3.2022.]
[5] Vujičić, V., Milićević, I., Marjanović, M., Dučić, N., Jovičić, A., Dragićević, S.: Offline Robot Programming Using ABB RobotStudio, X International Scientific Conference Heavy Machinery HM 2021, Proceedings HM 2021, ISBN: 978-86-81412-09-1, pp. C79-C83, University of Kragujevac, Faculty of Mechanical and Civil Engineering Kraljevo, 23-25th June 2021, Vrnjačka Banja, Serbia.
[6] Product specification IRB 14000 [Online]. Available: https://search.abb.com/library/Download.aspx?DocumentID=3HAC052982-001&LanguageCode=en&DocumentPartId=&Action=Launch [Accessed 11.3.2022.]
[7] Podobnik. J., Šlapah, S., Rejc, J., Kamnik, R., (2018). Osnove robotike: Laboratorijski praktikum, Založba FE, Universa v Ljubljani, Ljubljana
[8] Adit, M., (2020). HandBook Guidance on the programming of ABB YuMi IRB 14000. Otto-von-Guericke-University Magdeburg 10.13140/RG.2.2.12746.18881.
[9] RobotStudio [Online] Available: https://new.abb.com/products/robotics/robotstudio [Accessed 11.3.2022.]

COAST 2022

FACULTY OF MANAGEMENT

www.confcoast.com
Zemunska 143, Meljine 85 348

HERCEG NOVI

2022