

Peter Marcinko¹, Dávid Koval'uk²

¹Assistant Professor, Assistant Professor of the Department of production systems and robotics
Technical University of Košice (Košice, Slovakia)

E-mail: peter.marcinko@tuke.sk, ORCID: <https://orcid.org/0000-0003-2853-5755>

ResearcherID: [DFK-9265-2022](https://orcid.org/DFK-9265-2022), Scopus Author ID: [56736964700](https://orcid.org/56736964700)

²PhD student, Phd student of the Department of production systems and robotics
Technical University of Košice (Košice, Slovakia)

E-mail: david.kovaluk@student.tuke.sk

THE USE OF SMARTCOMPONENTS IN THE DESIGN OF COMPLEX ROBOTIC WORKPLACES

This article deals with conceptual designs for welding operations in the ABB simulation environment RobotStudio program. This software is used for offline programming of robots from this robot manufacturer. It contains many useful functions for the creation of robotic workplaces, and one of these functions is the so-called SmartComponents.

The article contains a description and creation of individual objects, mechanisms and devices used in the concept of the workplace, as well as a detailed description of the construction of the workplace. Furthermore, this part contains the generation of signals and the creation of various operations for designing and subsequently simulating the movement of robots performing a technological operation, using the arc welding method.

Keywords: SmartComponents; welding simulation; RobotStudio; Design; Arc welding.

Fig. :6. Table: 1. References: 11.

Relevance of the research. Industrial robotics, specifically industrial robots, have a large presence in various technological sectors in today's world. Although the average person encounter a robot every day, these devices affect several aspects of our lives. They are widely represented, for example, in the automotive industry, where they are most often used for technological operations such as welding, painting or assembly of individual parts. Their use in industry significantly shortens production time and eliminates the strenuous and dangerous work that people would have to do. At the time of the COVID-19 pandemic, some companies began to realize the benefits of robotics and began to gradually implement industrial robots in their production operations [1].

Nowadays, there are several methods of programming industrial robots. One of them is the offline programming method. This method consists in creating the desired robot program in a virtual software environment. For this, simulation software is used, such as RobotStudio along with SmartComponents plugins.

Problem statement. The subject of robotics covers many different areas. Robots alone are rarely useful. They are used together with other devices, peripherals, and production machines. They are generally integrated into a system designed to perform a task or an operation. Robots are very powerful elements of today's industry. They can accurately perform many different tasks and operations and do not require the usual safety and comfort features that a human worker would need. They also enable a certain activity to be carried out continuously and without the break that a human worker would need. Robots are computer-controlled machines that are programmed to move, manipulate objects, and perform work while interacting with their surroundings. To approach this problem, a robotic station was created on which various SmartComponents were tested for different types of mechanisms, e.g., positioner or linear motion of the robot [2].

Analysis of recent research and publications. Offline programming means that the operator builds a virtual scene of the robot's working environment in the programming software and completes the program design of the robot according to production requirements. Offline programming usually allows simulation, but also program testing, e.g., collision detection. Although the offline programming method is usually adopted by traditional industrial robots, this method can be used to program collaborative robots if the online programming method cannot meet the requirements of the task. The advantage of this programming method is that the programming can be done while the robot is still in production on a previous job, so the robot's production time is not lost due to delays in programming the robot to a new task.

Other advantages are efficient programming of program logic and calculation with the state of debugging facilities, locations are built according to models and this may mean that the programmer will have to fine-tune the programs online or use additional sensors, program verification using simulation and visualization, costs independent of production, production can continue during programming, tools to support the process, for example selection of welding parameters [3;1].

ABB RobotStudio is an engineering tool for configuring and programming ABB robots, both real robots in operation and virtual robots on a PC. To achieve real offline programming, RobotStudio uses the ABB VirtualRobot technology, or the so-called virtual controller together with add-ons for importing parts from CAD programs and the library itself, which contains ABB mechanisms and robots.

Due to the large availability of information online and the large amount of work devoted to the basic description of the environment and familiarization with the ABB RobotStudio software, we will not describe the basic environment and functions of this software, but in the following chapters we will describe the SmartComponents with which we can simulate the various required functions and properties.

Uninvestigated parts of a common problem. Despite a large part of work on the creation of robotic workplaces in offline environments as a whole and individual elements, as well as the interpolation control mode, it is not possible to create complex simulations with technology, for example welding. Then there is a situation where it is necessary to use SmartComponents.

SmartComponents is an object in ABB RobotStudio (with or without a 3D graphical representation) that has behaviour that can be implemented using code or aggregations of other SmartComponents. In the following subsections, we will describe several basic SmartComponents, as we will use and assemble several SmartComponents to simulate some properties in the environment during the design of concepts and the creation of simulations [4].

Research objective. The purpose of the article is to build a robotic workplace in an offline program with industrial robots, a positioner and ABB welding technology that used SmartComponents. To achieve the above, it is necessary to solve the following tasks:

- suitable location of the devices for the entire workplace,
- design logical control for robots and positioner,
- design management,
- testing.

The statement of basic materials. With the help of these SmartComponents, we can detect what kind of object it is and, in combination with other components, perform various functions such as moving the detected object in space or attaching it to another object. The Collision Sensor detects collisions and events in which a collision occurs, or the permitted boundary is exceeded between the first and second object. If one of the objects is not defined, the other will be checked against the entire station. When the signal is active and a collision or near-collision event occurs and the component is active, the Sensor Out signal changes its value to one. Objects that participate in a collision event or near collision event are reported. Line Sensor defines a line using Start, End and Radius. If the active signal is high, the sensor detects objects that cross the line. Intersecting objects are displayed in the Closest Part property, and the point on the intersecting object that is closest to the start point of the line sensors is displayed in the Closest Point property. When an object is detected, the output signal Sensor Out is set. Plane Sensor defines a plane using Origin, Axis1 and Axis2. When an active input signal is set, the sensor detects objects that cross this plane. Intersecting objects are displayed in the Sensed Part property, and the SensorOut output signal is set if an intersection occurs. Closest Object defines a reference object or reference point. If the Execute signal is set, the component finds the Closest Object, Closest Part, and Distance to the reference object, or to the reference point if the reference object is not defined. If a reference object is defined, the search is limited to this object and its parts. When the relevant properties are completed and updated, the Executed signal is set. Fig. 1 shows the setup of the necessary sensors [5].

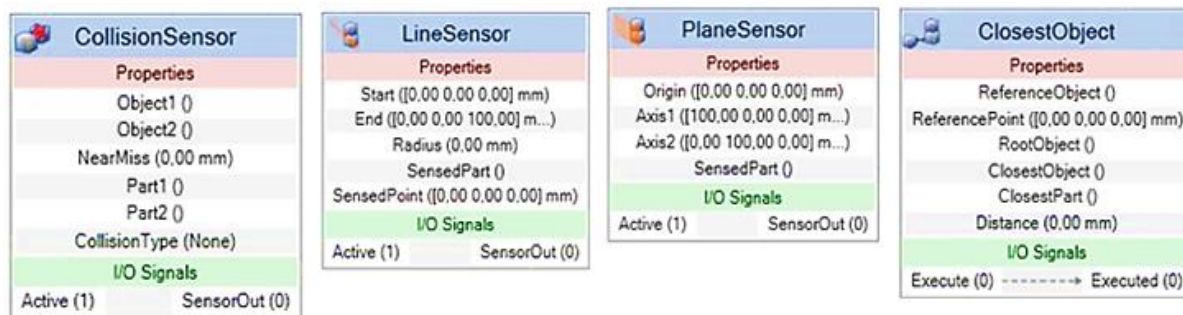


Fig. 1. Sensors

With these components, we can create various actions, e.g., such as attach, detach, or generating a copy of the original object. In some cases, it is advantageous to use these components with sensor components. If the Execute signal is set, the Attacher attaches the defined object in Child to the Parent object. If it is a parent mechanism, the primary Flange object to connect to must also be specified. When the Execute input is set, the child object is connected to the parent object. If the Mount option is enabled, the Child will also be attached to the parent with the specified Offset and Orientation. After completion, the Executed output will be set. Detacher detaches the Child from the object it is attached to if the Execute signal is set. If the Keep position option is enabled, the position will be preserved. Otherwise, the Child is positioned relative to the Parent object. After completion, the Executed signal will be set. The Source property of the Source component indicates the object that should be cloned when the Execute input signal is received. The parent of cloned objects is specified by the Parent property, and the reference to the cloned object is specified by the Copy property. The Executed output signal means that the clone is complete [6].



Fig. 2. Actions

Description of some 3D models. It will mainly be about 3D models, which are necessary for the design and implementation of the simulation for the workplace concept. Each model was created in SolidWorks 2018. Other used models, mechanisms and various elements were used from the RobotStudio library. In the concept, we will work with the same construction. The structure is made of two pieces of profile steel TR OBD 70×40×5 – 2100, four pieces of profile TR OBD 70×40×5 – 860 and one piece of profile TR OBD 70×40×5 – 460. The individual profiles of the structure will be connected to each other by means of an integral joint – weld. It is a corner weld along the entire length, the height of the weld is 4 mm. In the concept, we will simulate arc welding using industrial robots and one positioner. Due to the need to fix the welded object, it was necessary to design a positioning fixture [7].

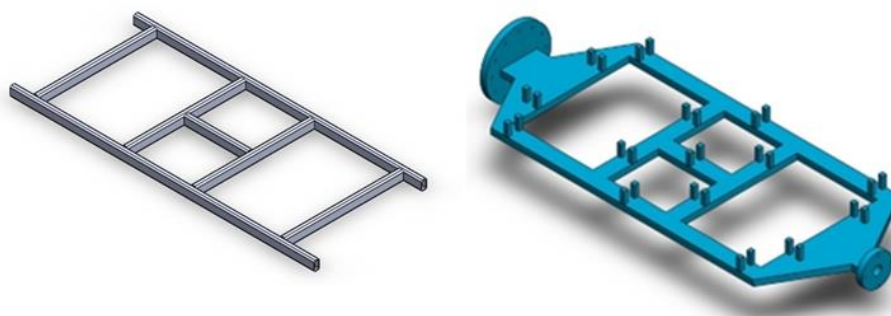


Fig. 3. Holder for steel construct

Due to the large size of the workplace, thanks to the use of components, we will present the layout of the workplace using a Table where the coordinates of the individual mechanisms will be. Each component used in the program has its own local point of origin, which can be used to determine the position in space from the reference point of the entire station [0, 0, 0]. This point is located near the positioner. In Tab. 1, individual rotations of mechanisms and devices around the axis are also included. The Y axes of both robots are not fixed, as both robots move using linear units. The entry point "A" is the point of the "Source" component that creates the palette. The output point "F" is the point on the path of the IRB 6660-100/3.3 robot, where the structure will be released from the gripper [8].

Table – Location of the devices

	X	Y	Z	R_x	R_y	R_z
Entry point "A"	4740	13477	415	0°	0°	0°
Exit point "F"	6220	-5410	209	0°	0°	0°
IRBT 6004	3125	2090,5	13	0°	0°	-90°
IRB 6660	3600	-	397,5	0°	0°	-90°
IRBT 2005	-1842	2074,5	0	0°	0°	-90°
IRB 2600	-1550	-	460	0°	0°	-90°
IRBP K-600/1200	1070	-3000	0	0°	0°	180°

Each performed task at the workplace is triggered after the previous task is completed. The whole process is automated, and we do not need to interfere with the program during the simulation. The first task is performed between points A, B. At point A, a pallet with a holder and profiles is created, which is then moved to point B. Between points B and D, the second task is performed, which starts after detecting the presence of the pallet at point B. At point B the robot gradually removes individual profiles and at point D places them on the fixture. The departure of the empty pallet from the workplace is carried out between points B and C after the completion of the second task. After saving the profiles on the fixture at point D and if the fifth task of the workplace (simulation of the journey - welding) is completed or if there are no other fixtures on the second fixture profiles, the fourth task is performed, changing the positions of the positioner. If there are profiles at point E on the fixture and the fifth task has not yet been performed after the position change of the positioner has been performed, then the fifth task (path simulation - welding) will begin. After finishing the fifth task and at the same time, if there are profiles on both fixtures, the sixth task begins, which is performed between points D and F [9].

After creating the required actions and processes that take place at the workplace, it was necessary to create the correct signal connections, the so-called design. This is what the station logic is for, which can be found in the Simulations tab, the Configure section. In the logic of the station, we connected all the necessary taps and input signals with which we simulate the entire process.

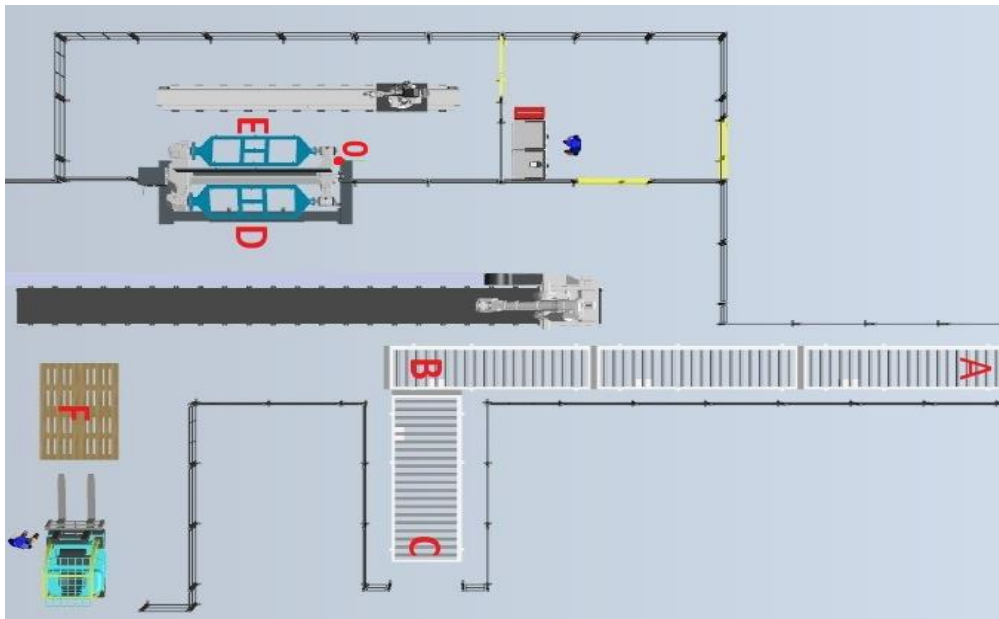


Fig. 4. Workstation

We connected all the signals from both virtual controllers as well as the SmartComponents we created. For some commands, we also used logical gates such as AND, NOT or OR, which ensure the correct functioning of the entire station. All signal connections in the station logic are shown in Fig. 5 [10].

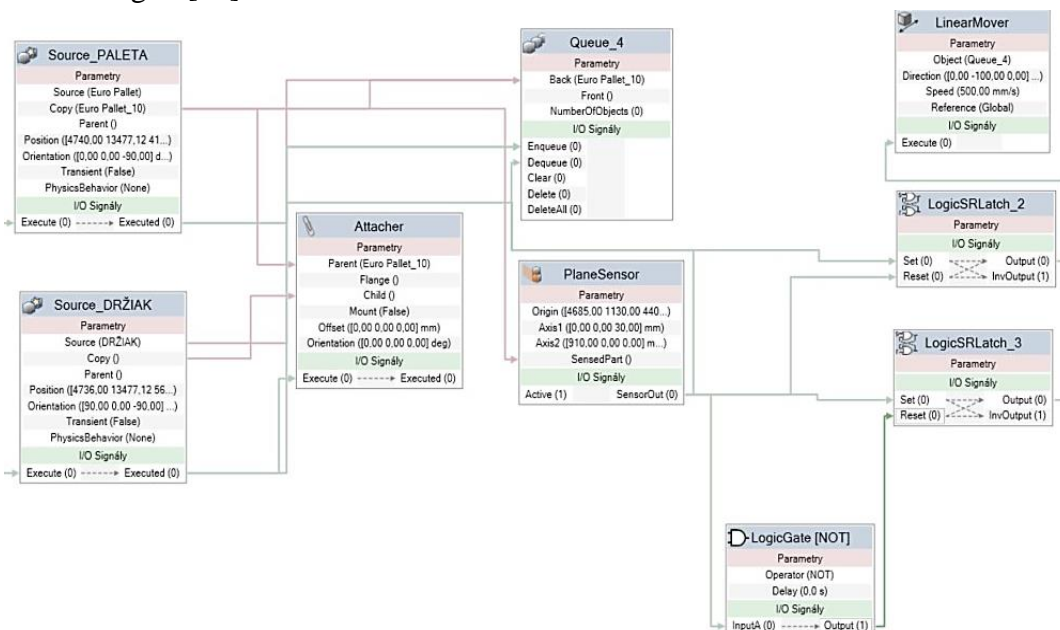


Fig. 5. Design for logical control and management for workstation

The last task performed at the workplace is the removal of the structure from the preparation and its subsequent placement on the prepared pallet. We perform this task using the IRB 6660 robot, with which we also performed the first task at the workplace. The proposed control for the entire testing operation was successful [11].

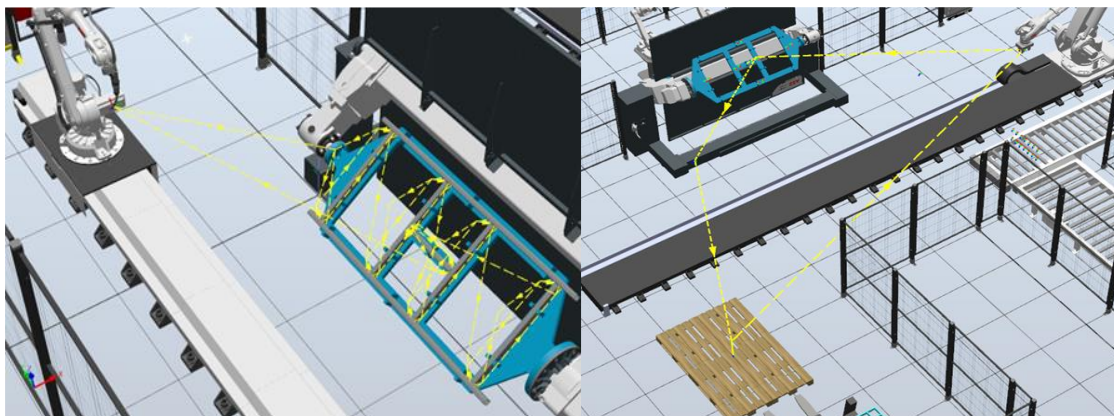


Fig. 6. Testing proposed control with SmartComponents

Conclusions. The goal of this paper was to design an offline program using SmartComponents to control a robotic workstation. This station contained the IRB 6004 robot, which removed steel profiles from the belt conveyor and inserted them into the fixture in the positioner. This positioner then ended by 180 degrees where another operation took place and that was arc welding. At the same time, another frame for welding was unfolded on the other side of the positioner with the help of the first robot. After welding the frame, the positioner rotated, and the robot removed the welded frame to the pallet. This complex operation of storing the frame, welding and subsequent transfer to the pallet would not be possible without the use of SmartComponents.

Acknowledgements. This article was created thanks to the KEGA project support: 004TUKE-4/2021 Development of innovative teaching materials for learning multi-agent robotics.

References

1. Hudák, J.: Konceptný návrh robotizovaného pracoviska pre zváranie konštrukcií v simulačnom protredí RobotStudio. Diplomová práca. Košice, 2022.
2. Marcinko, P., Semančík, K.: Robotized workplace for pick and place operation in simulation program RobotStudio. 2021. In: Technical Sciences and Technologies. Chernihiv (Ukraine): Chernihiv National University of Technology, Roč. 24, č. 2 (2021), s. 92-98, ISSN 2411-5363.
3. Semančík, K., Návrh robotizovaného pracoviska v simulačnom prostredí RobotStudio. Diplomová práca. Košice. 2021.
4. Holubek, R., Delgado, S. D., Košťál, P., Ružarovský, R.: Offline Programming of an ABB Robot Using Imported CAD Models in the RobotStudio Software Environment, 2014-12. Applied Mechanics and Materials, vol. 693, p. 62-67.
5. Semjon, J., Daneshjo, N., Fabian, M.: Off-line programování robotu pomocí simulačního programu RobotStudio. In: IT CAD. Vol. 21, no. 6 (2011), p. 36-38, ISSN 1802-0011.
6. Zaidan, S.: A Work-Piece Based Approach for Programming Cooperating Industrial Robots. Munich: Herbert Utz Verlag HmBH, 2013. ISBN 978-3-8316-4175-8.
7. Silva, F., M.. Targets Creation in ABB RobotStudio 5.15.02. ISEP – Instituto Superior de Engenharia do Porto, Project: ROBIN – Simulation and Offline Programming of Industrial Robots. 2016.
8. Marcinko, P., Semjon, J., Jánoš, R.: Vytváranie robotických buniek pomocou offline prostredia ABB RobotStudio I. – 1. Vyd. – Košice: Technická univerzita v Košiciach – 2021. 162 s. [CD-ROM]. ISBN 978-80-553-3831-6.
9. Šablatura, J., Lipina, J.: ABB RobotStudio – návody, Laboratorní cvičení v oboru II. Ostrava 2012. Vysoká škola báňská – Technická univerzita Ostrava, ISBN 978-80-248-2753-7.
10. Hashem, E.: Assembly Parts Project using RobotStudio. University of Maribor, 2018.
11. Cohal, V.: A welding technology using RobotStudio. MATEC Web Conf., Vol. 112, 2017. 21st Innovative Manufacturing Engineering & Energy International Conference – ImanE&E 2017.

Отримано 14.10.2022

Петер Марцинко¹, Давид Ковалюк²

¹доцент, доцент кафедри виробничих систем і робототехніки
Кошицький технічний університет (Кошице, Словаччина)

E-mail: peter.marcinko@tuke.sk, ORCID: <https://orcid.org/0000-0003-2853-5755>

ResearcherID: [DFK-9265-2022](https://orcid.org/0000-0003-2853-5755), Scopus Author ID: [56736964700](https://orcid.org/0000-0003-2853-5755)

²аспірант кафедри виробничих систем і робототехніки
Кошицький технічний університет (Кошице, Словаччина)

E-mail: david.kovaluk@student.tuke.sk

ВИКОРИСТАННЯ РОЗУМНИХ КОМПОНЕНТІВ У ПРОЄКТУВАННІ СКЛАДНИХ РОБОТИЗОВАНИХ МІСЦЬ

Метою цієї статті було роз'яснити роботу зі SmartComponents. Ми познайомилися з методами програмування промислових роботів. Ми представили програмне забезпечення моделювання RobotStudio від ABB, яке використовується для автономного програмування роботів. Представлено деякі SmartComponents, які використовуються в цьому програмному забезпеченні моделювання для виконання різноманітних необхідних дій на робочому місці.

На початку роботи увага присвячена презентації створених нами моделей для проєктування, а окрему главу ми присвятили створенню власного функціонального механізму за допомогою програмного забезпечення RobotStudio, яке згодом використовували при проєктуванні робочих місць. Далі представлено окремі пристрої, які використовуються на робочому місці. Їх компонування і подальше створення завдань виконуються на робочому місці за допомогою інтелектуальних компонентів і різноманітних сигналів, що створюються в контролерах роботів. Це необхідно для моделювання моделей у програмі RobotStudio, а також у програмах САПР, особливо SolidWorks і Creo. Для другої концепції ми розглянули функцію Multi Move.

Після складання обох робочих станцій ми повинні декілька разів змодельовати працездатність робочих станцій і перевірити їх функціональні можливості. У подальшому це дозволяє порівняти обидві концепції з точки зору ефективного використання пристроїв, які містяться в розроблених концепціях. Після створення обох конструкцій у програмі RobotStudio можна буде заощадити витрати на придбання обладнання або простої між операціями.

Заміна одного конвеєра, по якому на піддоні рухаються три різні сталеві профілі, та використання трьох різних конвеєрів дозволять скоротити час зварювання, і тим самим досягти більшої кількості виготовлених конструкцій. За попередніми оцінками в першій концепції ефективність робота IRB 6660 буде на 7,2 % нижче, ніж у випадку другої концепції. Це пов'язано з тим, що за першою концепцією було виготовлено за одну годину загалом 13 штук, а за другою їх було 16. Це забезпечило більш ефективне використання робота IRB 6660 у другій концепції.

Ключові слова: SmartComponents; моделювання зварювання; RobotStudio; конструкція; дугове зварювання.

Рис.: 6. Табл.: 1. Бібл.: 11.