УДК 681.5

THE IMPACT OF INDUSTRIAL COMMUNICATION PROTOCOLS ON THE DEVELOPMENT OF ROBOTIC CELLS

Marchenko I.S., Sajjad Ahmed M.R.
Belarusian National Technical University
Minsk, Republic of Belarus.

In automated systems, such as a robotic cell, effective communication between various devices is crucial for ensuring smooth operation and synchronization. The devices involved require a carefully sized communication infrastructure to handle the flow of data between them. Sizing the necessary communication involves evaluating factors like data volume, required response times, and the number of devices to be connected.

The communication between devices relies on modern industrial Ethernet protocols, which offer high-speed, reliable, and scalable data transfer capabilities. The protocols used include ProfiNet, Ethernet/IP**,** Modbus TCP/IP and many others, all of the, operate over standard Ethernet networks and are widely adopted in industrial automation. Ethernet/IP is designed for real-time control and data exchange, where precise control of movements and status updates is required. Similarly, Modbus TCP/IP is utilized for communication, enabling the exchange of machining commands, feedback on operational status, and completion signals. These Ethernet-based protocols ensure seamless integration and synchronization of the robotic cell, allowing for efficient and coordinated automation. Those protocols became standard feature for many industrial automation devices, that is why the impact of using them is not that clear in comparison with non-industrial protocols.

During the completion of the diploma project on the topic "CNC-milling machine robotic cell" 6-axis industrial robot HIWIN RA605 has been used. In modern versions of the same robot controller provides Modbus TCP as standard and optional ProfiNet. Using the Modbus and ProfiNet protocols on newer models of the robot controller, it’s possible not to create an executing program for its side at all, but only send them by packages from the PLC.

But in existing robot controller communication is established either via Ethernet TCP/IP or via RS-232. Described case relies solely on TCP/IP for communication, that requires extensive programming to manage interactions between the robot and PLC. Start by designing a clear communication protocol that specifies message structure, data inclusion, and error handling. Utilize socket programming to establish connections through server and client sockets for data exchange. Implement data serialization methods (like JSON or XML) to facilitate sending complex data structures over TCP/IP.

The CNC Milling machine which is used in the same project controlled with Mach3 breakout board, which is enthusiast solution that doesn’t seem to be completely industrial. But even its software provides Modbus RTU and TCP communication protocols, which simplifies integration with external equipment and enhances its programming abilities. Mach3 also provides system variables that are extremely useful for feedback to the PLC when the machining operation is complete.

There is also the technical possibility to integrate devices into a robotic cell by linking their discrete inputs and outputs, which may seem simpler at first glance. However, this will require excessive use of wires, even more redundant programming and more time-consuming commissioning. The maintainability of such solutions is also extremely low.

With the available capabilities, the following interaction algorithm was chosen: the PLC sends commands to the robot to pick up or place parts, and the robot acknowledges these commands. Status updates, such as whether the part was successfully picked up or if an error occurred, are sent back to the PLC for further actions. The PLC is also responsible for coordinating the operation of the CNC machine. It communicates with it using Modbus TCP. The PLC sends signals to the CNC machine to begin machining once the part is placed on the machine's table. Feedback from the CNC machine, such as completion signals, is sent back to the PLC, which then directs the robot to remove the finished part and place it back onto the conveyor belt.

While the robot and CNC machine do not communicate directly, the PLC ensures their synchronization. The PLC sends commands to the robot to pick up, place, and remove parts at the right times, ensuring the workflow is smooth and error-free. By managing the sequence of operations, the PLC prevents delays and overlap, improving efficiency. It monitors various sensors on the conveyor belt, robot, and CNC machine to track the status of parts and machines. This solution also scalable and could be improved by adding HMI panel or connection to some SCADA.

The robot's movements are programmed via HRSS (Hiwin Robot System Software), which provides precise control over the robot's pick-and-place operations. The robot places parts on the CNC machine's table and waits for confirmation that the part is positioned correctly before the CNC begins machining. Once machining is complete, the robot retrieves the part and returns it to the conveyor.

The integration of the HIWIN RA-605 robot, CNC Router 4030, and PLC programming enhances the manufacturing process by automating part handling and machining. This reduces the need for manual intervention, lowers the risk of errors, and increases overall productivity. The precise coordination between the robot, CNC machine, and PLC ensures that the system operates efficiently, with each component working in harmony to meet production requirements.

1. Kief, H. B., Roschiwal, H. A., Schwarz, K. The CNC Handbook: Digital Manufacturing and Automation from CNC to Industry 4.0. – New York, Industrial Press, 2022. — 720 p.
2. Kandray, D. Programmable Automation Technologies. – New York, Industrial Press, 2022. — 528 pages.