Journal of Achievements in Materials and Manufacturing Engineering
VOLUME 34
ISSUE 2
June
2009

The visualization of discrete sequential systems

J. Świder*, M. Hetmańczyk
Institute of Engineering Processes Automation and Integrated Manufacturing Systems, Silesian University of Technology, ul. Konarskiego 18a, 44-100 Gliwice, Poland
* Corresponding author: E-mail address: jerzy.swider@polsl.pl

Received 15.02.2009; published in revised form 01.06.2009

Manufacturing and processing

ABSTRACT

Purpose: The paper is focused on the assumptions of the method of discretizing the states of sequential systems for an exemplary robotized metallurgical semi- finished products treatment department. The control of digital equipment is described on the bases of the patterns of control bites sequence configuration. The need for the development of a discretization method stems from difficulties encountered in representing the state of a process of an analogue nature. The crucial barrier is the method of treating the input signal to the visualization system.
Design/methodology/approach: The proposed method is based on determining the variables that control the course of visualization in a manner enabling the treatment of all signals as boolean variables. Such approach requires a proper formulation of the input information flux to secure an explicit identification of the states of the served process.
Findings: On the grounds of the assumptions a system of visualization is created to attempt the service of a robotized treatment centre. The main emphasis is put on devising an unambiguous way of identifying the states of an industrial robot and its cooperating sensors.
Research limitations/implications: The result is the presentation of the combination of control elements in a bite vector used for defining the state of failure or for detecting operational irregularities.
Originality/value: The originality of the paper is the approach towards the problem of processes visualization which may be categorized to a group of discrete processes with elements of analogue control data processing.
Keywords: Automation engineering processes; Robotics; Visualization; Discrete systems; Sequential operation

Reference to this paper should be given in the following way:
J. Świder, M. Hetmańczyk, The visualization of discrete sequential systems, Journal of Achievements in Materials and Manufacturing Engineering 34/2 (2009) 196-203.

1. Introduction

Modern systems of industrial processes automation mostly consist of five basic components: controllers (PLCs, regulators, industrial computers, etc), actuators (electrical, pneumatic, hydraulic and their combinations), control and measuring equipment (sensors and detectors), industrial networks and visualization systems.

The control and supervision [1] over processes by means of SCADA application (Supervisory Control And Data Acquisition) has become a very popular form of programmed automation in industry. SCADA applications are effective tools in centralized and distributed computer control (DCC) systems.

Professional visualization environment is an extended tool for controlling and presenting industrial processes, capable of serving not only simple applications but also complex processes that control a big number of data (sewage treatment plants, chemical processes, etc).

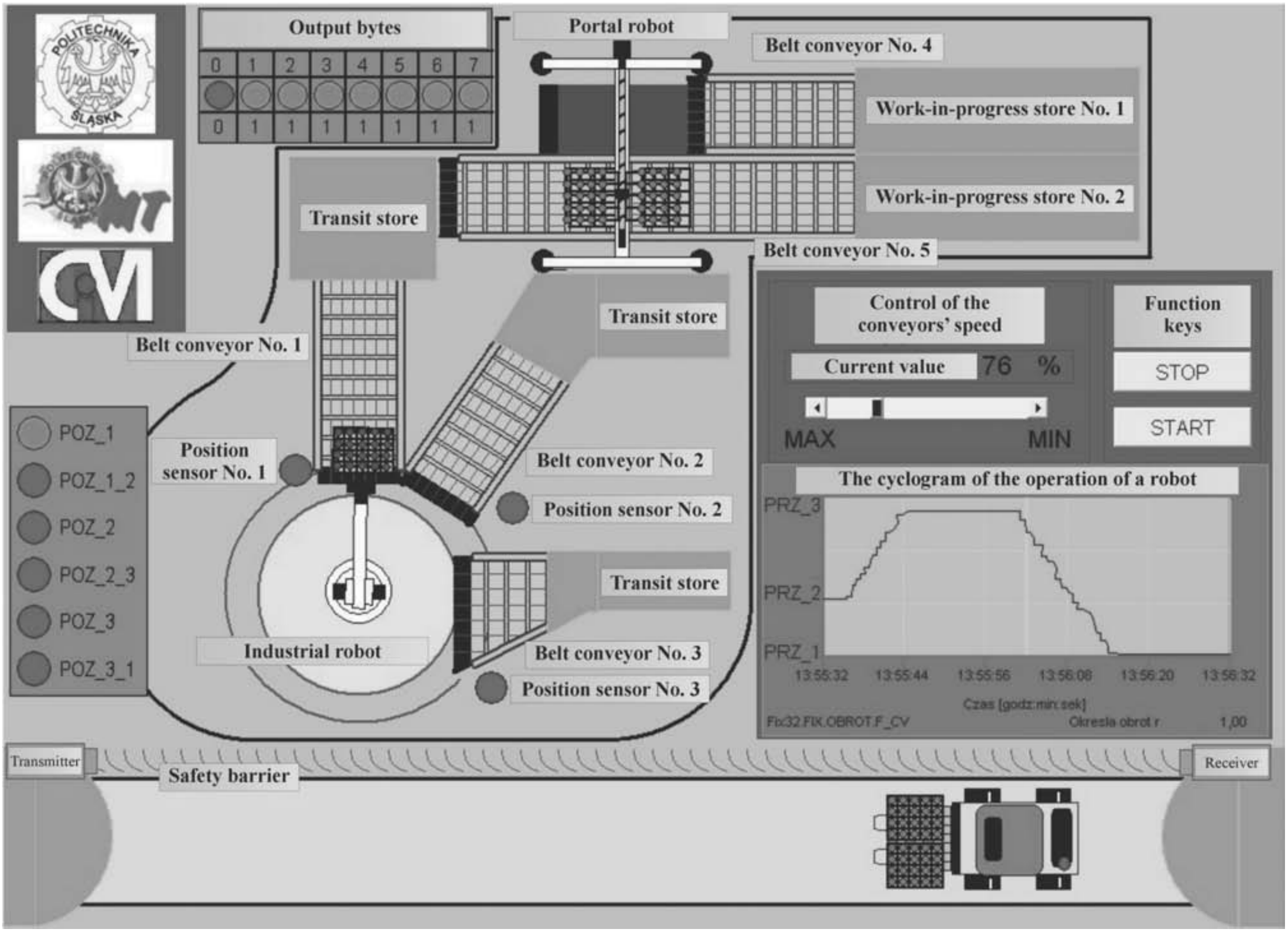

Fig. 1. Visualization window of a robotized centre

All available software has a certain implemented language (Visual Basic for Application, Cicode, CitectVBA) rendering almost unlimited possibilities of programming.

Industrial applications that use automatically controlled equipment [2] are mainly operated by means of customized controllers and properly selected modules and communication standards [3].

A common form of visualization is the programming of sequential systems, where the initiation of successive events occurs as a result of the completion of previous process stage.

Sequential systems include: robotized treatment centers, assembly lines, conveyors connected into a joint transporting sequence. Such systems, for safety reasons, require an extended monitoring procedure.

2. Analysis of an industrial system

2.1. Robotized centre - an operational algorithm

A sequential model of a robotized reloading centre equipped with additional conveyors is shown in Fig. 1.

The principle of the system operation is based on algorithms making up a network of procedures (Fig. 2). The treated elements and empty containers are fed onto the conveyors from the input stores marked as 1 and 2. The objects grouped into several batches by means of conveyor 4, are directed to the reorientation stand, where they wait to be picked by a gripping device of a portal robot. The containers fed from the second input store are coded and assigned to a given batch of objects. Next, the set (container and elements) is forwarded to store 1, where it is conveyed to the work stand of a stationary robot by means of belt conveyor 1.

The robot collects the successive sets and next conveys particular elements to their destination, the empty containers go to store 2 conveyor, whereas the treated elements to store 3 conveyor. In the next stage, semi-finished products are moved from the stores to machine tools (not marked in the Figure), where, after treatment, they are loaded to previously assigned containers.

The algorithm is executed cyclically and its form explicitly defines the operational mode of the centre.

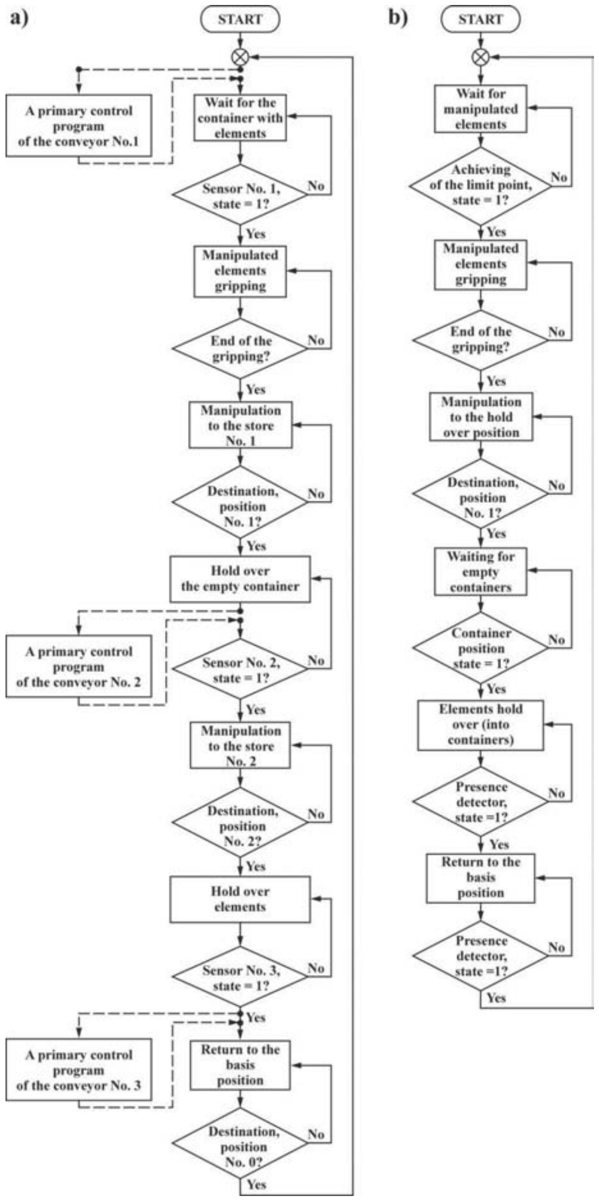

Fig. 2. Operational algorithm for: a) a stationary robot, b) a portal robot

2.2. Identified problems, simplifying assumptions

Despite an apparent simplicity of operation, the presentation of the states of the actuators still poses a big problem, due to the following reasons:

- the system of identifying and orienting the industrial robot is based on encoders, the resolution and response rate of which must be adjusted to the dynamic parameters of the manipulator (maximal acceleration and velocity),
- the characteristics of the process, determined by high efficiency requirements, enforce the analysis of a big number of transmitted data that change in dozens of milliseconds [4],
- centralized control systems are characterized by the distribution of actuators, which obstructs their supervision and the control of the states,
- manipulators, robots and automated internal transport means impose the use of special, safety-enhancing elements (curtains, pressure mats, etc).

Moreover, in view of numerous additional equipment, mutual combinations between the elements must be identified, as well as the impacts of the state changes on the process [5].

Internal conditions (Fig. 2) are stored in the form of actions, recorded inside a schedules editor (Fig. 3).

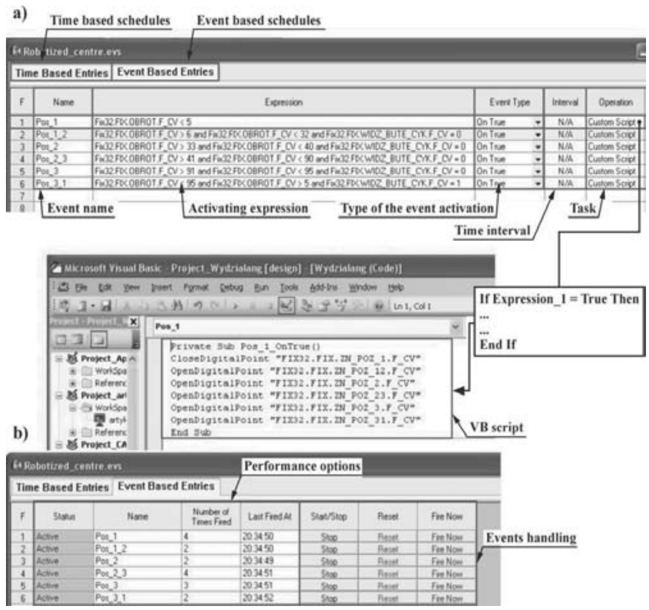

Fig. 3. Windows of defined schedules: a) a configuration view, b) the active state view

The fundamental assumption of the discussed approach to visualization is:

- the simplification of the control algorithm (reduced number of variables in the control code and simplicity of the program, supervising the process enable faster processing of the program data and updating the inputs/outputs),
- the omission of inessential stages (in the case of the robot: the path between intermediate positions is run at constant speed, the key moment is when the robot reaches the discarding or collecting position),
- the separation of the objects that control the detection of failure or hazardous states (for example, break of the safety barrier, operational failure or collision of the robot) into two elements: PLCs (superior equipment supervising the process) and the robot control system (only hazardous states connected with the motion of the robot, collisions, etc),
- the achievement of the condition where all control and supervision signals have a two-state form,
- the shifting of the responsibility for the form and performance of visualization to a PC computer (SCADA node).

Table 1.
Content of the control block

No.	Instructions (line number)	The input or output pattern	Actions
1.	(0) WAITAND 0 XXXXXXXXXXOOOOOC 1	000001XXXXXXXXXX	The program searches through the contents of the block verse by verse (from 0 to 13), in accordance with the refresh time imposed by the user. The last verse include the return command (a jump to the first verse), which defines the program loop. If values of the WAITAND instruction are reached, the DC block sends the corresponding pattern to the output.
2.	(1) OUTPUT CCCCCCCO	01111111	
3.	(2) WAITAND 0 XXXXXXXXXXOOOOCO 3	000010XXXXXXXXXX	
4.	(3) OUTPUT COCCOOCC	11001101	
5.	(4) WAITAND 0 XXXXXXXXXXOOOCOO 5	000100XXXXXXXXXX	
6.	(5) OUTPUT OCOOCCOO	00110010	
7.	(6) WAITAND 0 XXXXXXXXXXOOCOOO 7	001000XXXXXXXXXX	
8.	(7) OUTPUT CCOOOOCC	11000011	
9.	(8) WAITAND 0 XXXXXXXXXXOCOOOO 9	010000XXXXXXXXXX	
10.	(9) OUTPUT OOCCCCOO	00111100	
11.	(10) WAITAND 0 XXXXXXXXXXOCOOOO 9	010000XXXXXXXXXX	
12.	(11) OUTPUT OOCCOCOO	00101100	
13.	(12) GOTO 0	-	

2.3. Replacement of the analogue representation of the states with a control bites sequence

The assumptions of the method require a two-state representation of the value of the signals, as it is the simplest form of identifying the states of objects [6,7].

From the point of view of operational principles, the entire process is classified to the group of discrete processes with determined repeatability (a definite cyclogram of the operation of the robot).

The position of the robot may be recorded in the form of a register consisting of 8 or 16 bites [8]. The data bite may represent the number of impulses (the initial value of the encoder) designating the position of the manipulator (only the vertical axis is considered). The acceptance of the representation of the position of the manipulator on the grounds of the servo-drive encoder signals is correct, however, it is difficult to implement because of fast changes of the sensor initial values.

In the discretization [9] of the state of the operation of the manipulator reference points are assumed in the form of the positions taken at particular operational stages. Thus, there are three basic positions (in which the service operations are performed: collection or discarding of elements) and three intermediate positions (taken in the course of the motion between the destinations).

The identification of the destination positions is a result of detecting the absence or presence of the elements that are to be collected or discarded. The points defined in such manner combine all actuators, whereas the task of supervising and correct positioning of the robot is taken over by its control system [10].

On the basis of bytes the achievement of a particular position may be controlled by several actuators, for example: stopping the conveyor, permitting the access of the robot, opening/closing the gripping device, adjusting the feeder, valve, etc. by means of Device Control blocks run on the patterns of contacts defined by the user (Fig. 4, Table 1)

The principle of processing the input pattern into the output one is relatively simple and was presented on the discussed example. The controller monitors an arrays of discrete devices, in which 32 logical states were identified (0 or 1) represented by 16 bites.

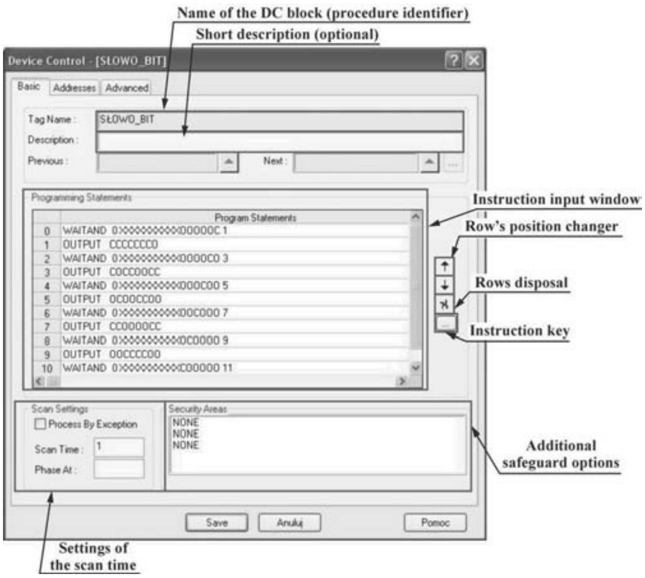

Fig. 4. The view of the control block window of iFIX equipment

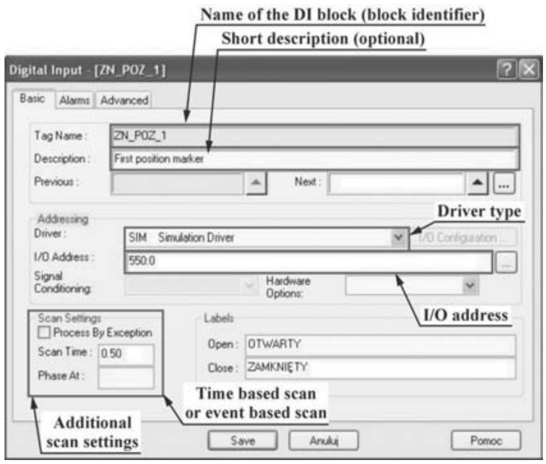

Fig. 5. The view of the digital input window (a first position marker) of iFIX equipment

No. of the input address	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Input pattern	C	C	C	O	X	O	X	C	C	O	C	O	O	O	C	O
Value of the input word (16 bits)	1	1	1	0	0/1	0	0/1	1	1	0	1	0	0	0	1	0
Syntax of the conditional statement (pattern of contact)	0 WAITAND 0 CCCOXOXCCOCOOOCO 1 1 OUTPUT OCXCCXCO															
Output pattern	/	/	/	/	/	/	/	/	O	C	X	C	C	X	C	O
Value of the output word (8 bits)	/	/	/	/	/	/	/	/	0	1	1	1	1	0	1	0
No. of the output address									7	6	5	4	3	2	1	0

Fig. 6. The concept of processing the pattern of contacts, where: C – high state, O – low state, X – any state

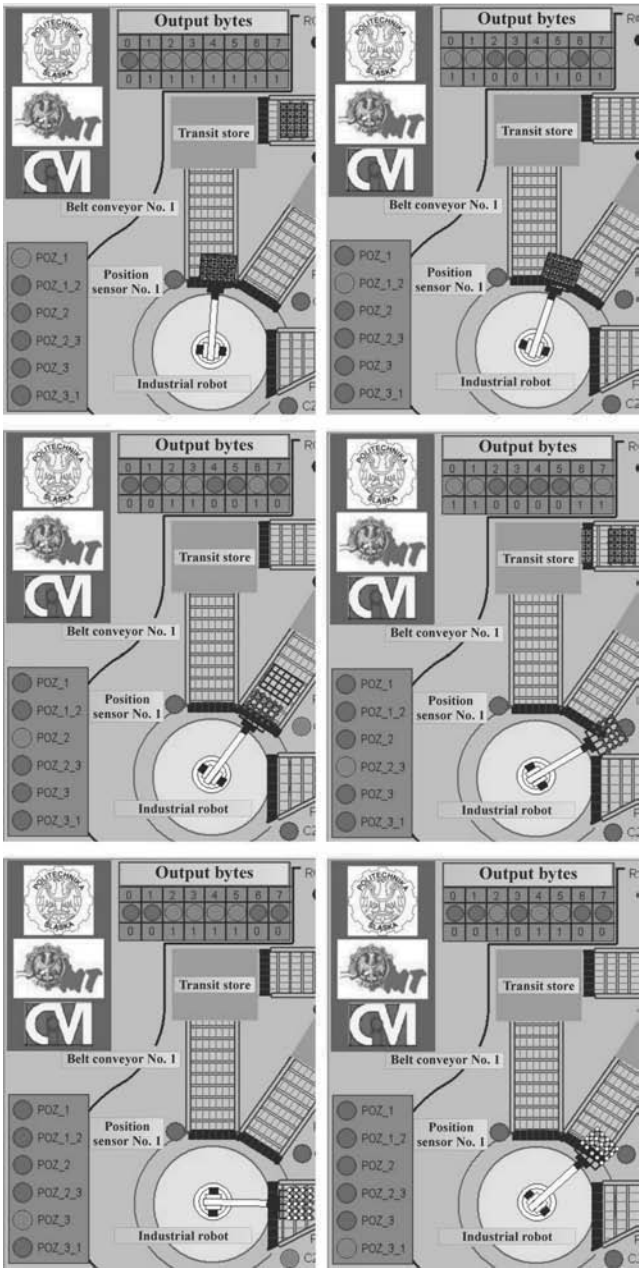

Fig. 7. The operational mode of the programmed DC block

The defined vector consists of the states of the sensors designating the presence of manipulated objects, permission for the robot to move, the state of the gripping device, etc.

By inserting the memory values of the controller to SCADA database (for example: two-state input/output blocks) a byte word is derived, defined by the input addresses (Fig. 5).

Six patterns of contacts were identified in the described block with the use of the WAITAND processing instruction (Fig. 6). The instruction defines the time required for the fulfilment of the condition [11]. The program searches through the contents of the block verse by verse, in accordance with the refresh time imposed by the user.

Upon detecting the consistency between the initial sequence and the pattern, the next step to the programming line is made, followed by the adjustment of the states of the actuators to which particular bytes were assigned.

The use of the option of alerts makes it possible to construct a system of diagnosing equipment to supervise its appropriate operation not only in view of the logical state of the byte cells but also from the point of view of the time during which the values specified in the WAITAND instruction are reached [12].

One of the advantages of the pattern is the consideration of the states of all control elements in each position (for example in position 3, the sensors of the robot presence numbered 1 and 2 are in a low state; accordingly, in the pattern they are represented as two bytes of the value of 0), which enables fast detection of a faulty element or causes of failures.

Exemplary patterns of the sequence of output bytes dependent on the position of the robot are shown in Fig. 7.

Discrete inputs or outputs can inform on the status of majority parameters of the concerned system.

The SCADA system enables an user to collect data from remote devices, send instructions and analysis of the course of events in deterministic time. Such applications allows also to connect equipment and systems separated by large distances.

2.4. Visualization of a continuous process on the ground of characteristic points

The visualization of the motion of the robot between its designations is performed independently from the sensor of the vertical axis of the kinematic chain [13]. On the basis of the velocity of the motion of the robot the time of its passage through particular points is determined; whereas the condition of finishing the generation of the motion values is a high state of the sensor that detects the position of the robot.

All programming functions may be replaced by a program written in an implemented programming language, or by means of program blocks. The program blocks secure complete functionality of the programming language; however, they present more simplicity of the algorithm notation.

The essence of the programmable block lies in writing down the program (instruction lines) and such configuration of the options that will secure the set of the block to scan the database blocks (in a manner selected by the user).

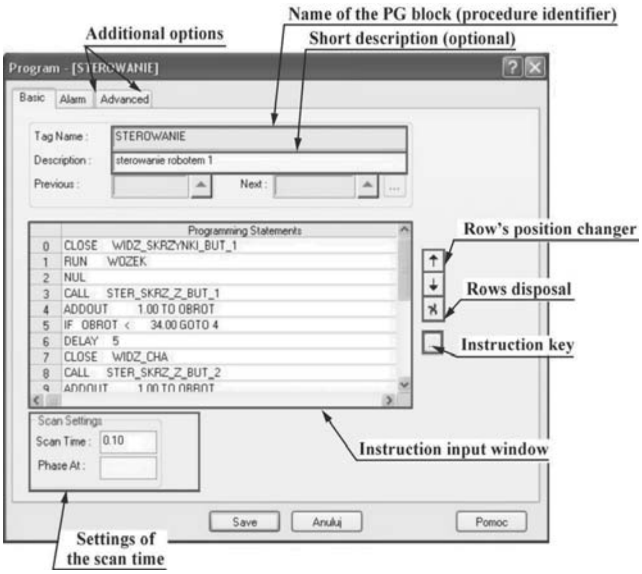

Fig. 8. The view of the PG block window of iFIX program

The programmable block (PG) enables writing of 20 instruction (indices 0-19), implemented in the sequence of increasing values of the number of verses (Fig.8). It is possible to open other PG blocks, so the user is not limited to uncomplicated programs [12].

Furthermore, there is a possibility of adapting the graph of the actual time required to create the cyclogram of the operation of a stationary robot (Fig.9) on the basis of its occupied position. The recorded variable is the content of the analogue register characterising the rotation of the robot around its own axis.

The user can configure a PG block to send messages using the MSG command. When this command executes, the PG block sends a message to all alarm destinations in the block's alarm area.

Instead of PG block, it is possible to use a recipe, the schedule or a combination of specified tools.

SCADA systems base on personal computers, on the following reasons [14, 15]:

- the computer can record and store a very large amount of data stored into an arbitrary database,
- the data can be displayed in any way the user requires,
- an operator station displays the data and allows to perform remote control tasks,
- the performed system can be adapted to the user requirements.

After reaching the particular positions, in which the robot is required to collect (or discard) the manipulated object, a horizontal line is plotted on the graph. This corresponds to the constant value of the register at gradually changing time of running the application.

The representation in the form of the bytes sequence facilitates the assignment of each intermediate position (or destination position) to the initial value starting up the code responsible for the control of two-state devices, as a number that is the multiple of the number of bytes of the output word.

Likewise, the problem of identifying the states of the portal manipulator is solved (Fig. 10).

It should be emphasised that the plotted cyclogram does not collect any external data, but is only derived on the basis of several defined parameters (i.e. the time in which the robot passes through the destination points, periods between high states of the sensors).

Increases in angular displacement (the robot) and linear paths (the portal manipulator) are made with the use of the signals generator securing the assumed ease of motion.

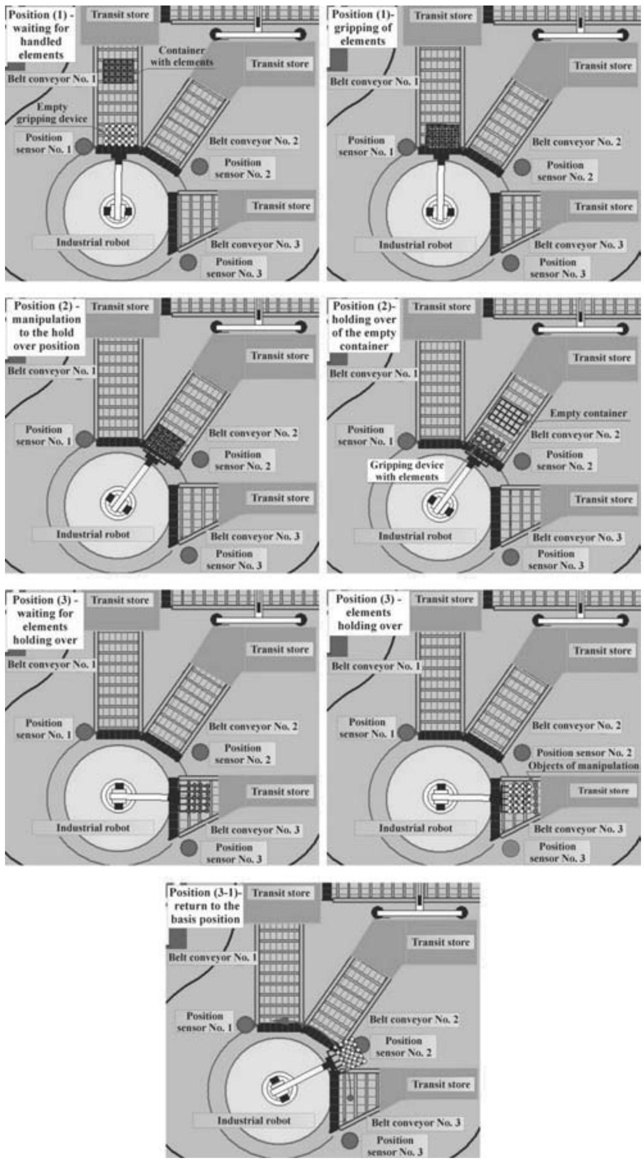

Fig. 9. The position of a stationary robot in the conveyors service centre (the positions generating control sequences)

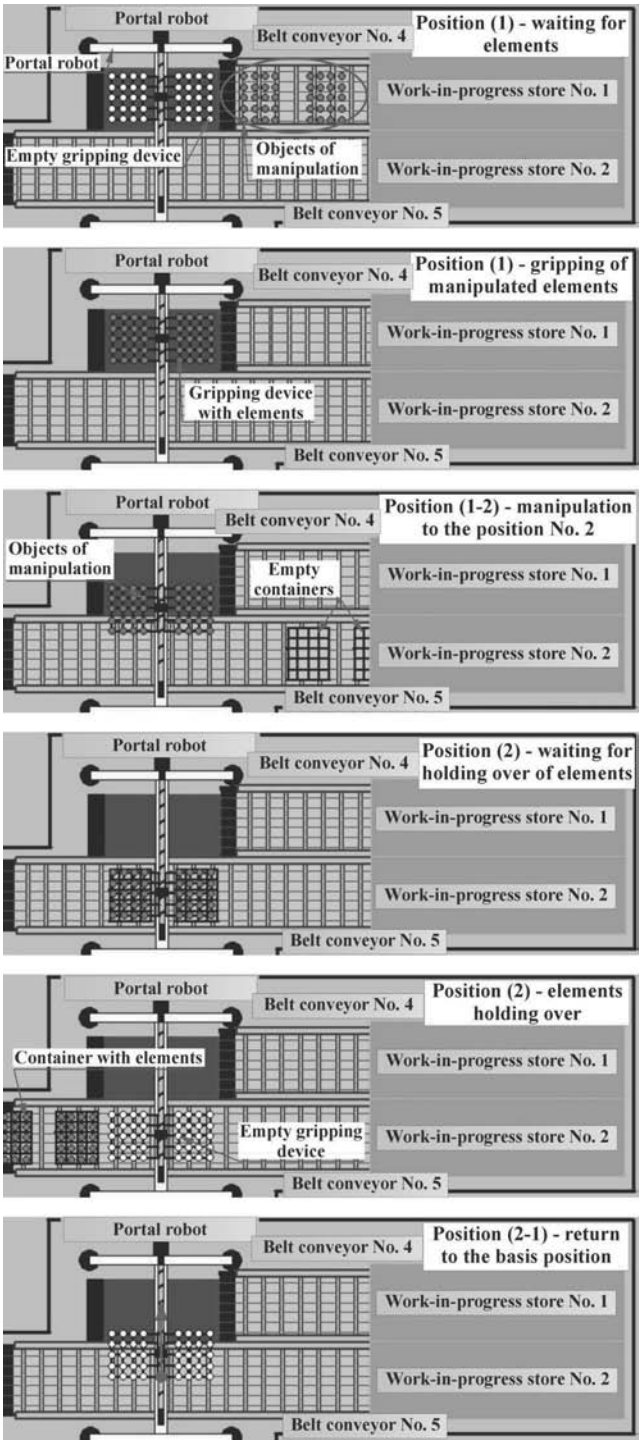

Fig. 10. Positions of a portal robot

3. Advantages and disadvantages of the proposed solution

Advantages of the proposed solution include:

- the minimization of the complexity of PLC algorithms;
- the improvement of the level of data transmission by means of customized communication software (reduction of the quantity of data in the form of real numbers);
- the opportunity of applying more advanced tools for the visualization of the state of actuators [16, 17];
- increased ease of operation at the concurrent maintenance of a high level of representation of the actual conditions;
- the possibility of detecting the states of failure on the grounds of checking the time of the performance of particular tasks,
- the proposed system maybe adapt to cooperation with an external Programmable Logic Controller (PLC),
- exist a possibility of development.

The disadvantages of the concept involve:

- the requirement of using the programming language implemented in the visualization program or enabling direct feedback (i.e. by means of OPC server);
- the increase of the time required for the performance of the visualization (in comparison with the preparation of the control algorithm work outlays are insignificant);
- not every real system may be transformed to a discrete form.

4. Conclusions

SCADA systems are very useful in many branches of production. The software described in this paper has been chosen among other due to popularity and potential. The iFIX application is evolving dynamically and pretends to become standard at the Polish market.

The use of the described functions in combination with PLCs offers a better opportunity for the functioning of the created applications and for reducing costs (the option of transferring a part of logistics operations to the zone of SCADA software).

The discussed approach offers simple control of industrial applications without the necessity of the knowledge of the syntax of high level programming languages, yet, at the same time, it remains open to the application of high level languages.

References

[1] S. Boyer, SCADA: Supervisory Control and Data Acquisition, ISA, USA, 2004.

[2] J. Fraden, Handbook of modern sensors –physics, design, applications, Springer, 2004.

[3] T. Legierski, PLC Controllers Programming, Jacek Skalmierski Publishing, 1998, (in Polish).

[4] J. Świder, G. Kost, D. Reclik, The signal connections in robot integrated manufacturing systems, Journal of Achievements in Materials and Manufacturing Engineering 23/1 (2007) 87-90.

[5] J. Świder, K. Foit, G. Wszołek, D. Mastrowski, The system for simulation and off-line, remote programming of the Mitsubishi Movemaster Rv-M1 robot, Journal of Achievements in Materials and Manufacturing Engineering 25/1 (2007) 7-13.

[6] W. De Silva, Clarence, Mechatronics: an integrated approach, CRC Press LLC, USA, 2005.

[7] N. Bulusu, S. Jha, Wireless Sensor Networks, Artech House, London, 2005.

[8] N. V. Kirianaki, S. Y. Yurish, N. O. Shpak, V. P. Deynega, Acquisition and Signal Processing For Smart Sensors, John Wiley & Sons, England, 2002.

[9] P. S. Naidu, Sensor Array Signal Processing, CRC Press, London, 2001.

[10] H. R. Harrison, T. Nettleton, Principles of Engineering Mechanics, R. Harrison & T. Nettleton, London 1994.

[11] D. Gacek, O. Geynisman, Member, D. Proudfoot, K. Minnick,Migrating from SCADA to Automation, Transmission and Distribution Conference and Exposition, IEEE/PES 1 (2001) 343-348.

[12] www.gefanuc.com

[13] T. R. Kurfess, Robotics and Automation Handbook, CRC PRESS, London, 2005.

[14] J. Świder, K. Foit, The use MITSUBISHI PLC systems in student`s preparation for realization of industrial tasks, Journal of Achievements in Materials and Manufacturing Engineering 14 (2006) 190-196.

[15] G. Clark, D. Reynders, E. Wright, Practical moderns SCADA protocols: DNP3, 60870.5 and Related Systems, Elsevier Linacre House, Oxford, 2004.

[16] J. Świder , P. Michalski, G. Wszołek, Laboratory suport for the didactic process of engineering processes automation at the Faculty of Mechanical Engineering, Journal of Achievements in Materials and Manufacturing Engineering 15 (2006) 199-206.

[17] J. Świder, G. Wszołek, D. Reclik, Didactic model of the high storage system, Journal of Achievements in Materials and Manufacturing Engineering 16 (2006) 237-240.