v. 2, n. 8, 2022

Journal of
Engineering
Research

USE OF PROBLEM-BASED LEARNING IN MODELING TEACHING APPLIED TO PROGRAMMABLE LOGIC CONTROLLERS

Rafael Garlet de Oliveira
Instituto Federal Catarinense, *campus* Luzerna
Luzerna – SC
http://lattes.cnpq.br/5694895282692108

Thiago Javaroni Prati
Instituto Federal Catarinense, *campus* Luzerna
Luzerna – SC
http://lattes.cnpq.br/7638101032304275

Luan Cizeski de Lorenzi
Instituto Federal Catarinense, *campus* Luzerna
Luzerna – SC
http://lattes.cnpq.br/2675191651482030

Antonio Ribas Neto
Instituto Federal Catarinense, *campus* Luzerna
Luzerna – SC
http://lattes.cnpq.br/5053976869671217

Abstract: This work presents the experience of applying the Problem-Based Learning method in the discipline of Modeling and Programmable Logic Controllers (PLC) in the Control and Automation Engineering course at Instituto Federal Catarinense - Luzerna campus. The objective of this course is to present concepts on Discrete Event Systems Modeling (SEDs), on PLC architecture and programming and to develop Supervisory Control Systems for PLC controlled systems. Initially the discipline is responsible for passing on the initial basic concepts to students, through an extensive number of exercises oriented to be performed autonomously, and finally challenging students to develop control of a simulated system. In this last step, a simulated industrial plant is proposed as a problem, built by means of a microcontrolled device with inputs, outputs and LED indicators. This device has its programming based on SEDs, whose model is used to build the control to be implemented in the PLC. The development of the control and its implementation in the PLC proved to be a very effective way to establish the initial concepts, in addition to promoting the autonomy of future engineers for teamwork and solving a real practical problem.
Keywords: Problem-Based Learning, Discrete Event Systems, Discrete Event Systems Modeling, Supervisory Control Theory, Programmable Logic Controllers.

INTRODUCTION

The improvement of teaching is a subject widely discussed in the field of engineering. Bazzo and Pereira (1997) suggest that these courses must aim at preparing the individual for the socio-economic reality in which he will work. In this article, a case study will be presented, where the Problem Based Learning (PBL) method is used in teaching Discrete Event Systems Modeling (SED) and Programmable Logic Controllers (PLC). This methodology promotes the active attitude of the student, as a central element in the search for knowledge and not merely informative as is the case with traditional pedagogical practice (BERBEL, 1998).

The experience discussed in this article comprises the curriculum component of the Control and Automation Engineering course at the Instituto Federal Catarinense (IFC) - Luzerna campus, called Modeling and Programmable Logic Controllers.

PROBLEM BASED LEARNING (PBL)

Problem-Based Learning had its genesis in 1969, in the teaching of Health Sciences at McMaster University, Canada.

It is a pedagogical proposal that consists of teaching centered on the student and based on the solution of real or simulated problems. In this method, in order to solve the proposed problems, students resort to previous knowledge, discuss, study, acquire and integrate new knowledge. The integration of concepts, combined with practical application, promotes the absorption of knowledge. In this way, the method values, in addition to the content studied, the way in which learning takes place, reinforcing the active role of the student in this process. PBL promotes the development of autonomy, interdisciplinarity, non-separation between theory and practice, the development of critical thinking and communication skills (BORGES et al, 2014).

SUBJECTS STUDIED IN THE SUBJECT

The discipline of Modeling and Programmable Logic Controllers is mandatory in the Control and Automation Engineering course at IFC – Luzerna campus. Its menu deals with Discrete Event Systems (SEDs), Languages, Regular Expressions, Automata, Petri Nets, Supervisory Control Theory, as well as all the Hardware and Software of Programmable Logic Controllers (PLC). The course is taught by two teachers and is divided into three parts. The first part is intended for the presentation of Petri Nets, Automata and Supervisory Control Theory. The second part includes content about PLCs, their physical hardware construction and their software programming languages and forms. The third part of the discipline is reserved for the development of a complete project cycle, from modeling and obtaining the Supervisory Control to its programming in a PLC.

The first two parts of the course are based on the resolution of guided exercises, in which students must resort to bibliographic references and support materials for their resolution and understanding of each topic. The third part of the course aims to use the PBL methodology, where students must seek the basic concepts set out in the first two parts to be able to prepare a complete project. In this step, the proposed problem is the control of an industrial plant, simulated by means of a microcontrolled device built for this purpose. Students are challenged to propose a model of an SED for the simulated plant, obtain the supervisory control based on this model and carry out its programming and implementation in a PLC, resulting in complete control of the system.

PETRI NETS

The use of Petri Nets allows a better understanding of the functioning of distributed systems, as it is a graphical tool (CARDOSO; VALETE, 1997). According to CASSANDRAS and LAFORTUNE (2006), the definition of this language is based on four elements: the places (P), the transitions (T), the arcs (A) and the weight of the arcs (w). Places represent available resources for use, or conditions for transitions. Arcs are the graph's connectors, between transitions and places. They can include their weight as additional information, representing both a condition for the occurrence of a transition and an output value of a transition to a place.

$$R = (P, T, A, w)$$

In the application of Petri Nets for control, the desired behavior of the plant must be described in a model of Petri Nets. This model can be obtained through several approaches. One way is the description of the open-loop plant and the constraints that define the closed-loop behavior separately followed by the fusion of transitions from the different Petri Nets.

AUTOMATA AND SUPERVISORY CONTROL THEORY (TCS)

Automata consist of a language that describes the behavior of discrete event systems (SEDs), representing regular expressions through a graph. In addition, they are devices capable of representing a language according to well-defined rules (CASSANDRAS; LAFORTUNE, 2006).

The representation of an automaton can be done through five elements: the system states (Q), the set of events (Σ), the transition functions (f), an initial state (q0) and the marked states (Qm), which represent complete system tasks.

$$\mathbf{G} = (Q, \Sigma, f, q_0, Q_m)$$

TCS consists of modeling the different subsystems of the plant through automata, as well as the specifications that determine the behavior in closed loop. With these models in hand, it is possible to obtain the ultimate controllable sub-language for the controlled system. This sub-language is as permissive as possible given the chosen specifications and the final result is a supervisory control that can be represented by an automaton and implemented by a PLC.

PROGRAMMABLE LOGIC CONTROLLERS (CLP)

The PLC is an industrial computer that can be programmed to execute instructions that control devices with machines and process operations. It is capable of storing instructions for implementing control functions, such as control logic, sequencing, arithmetic operations, among others for Automated Systems. It has in its structure a CPU (Central Processing Unit) composed of memory and processor, being responsible for the controller activities. The input and output modules are responsible for the communication between the CPU and the external environment, guaranteeing the isolation and protection of the central unit. The Ladder language is one of the most used in these devices. It is described graphically and based on symbols similar to those found in electrical schematics provided with contacts and coils (GEORGINI, 2009).

APPLICATION OF PROBLEM-BASED LEARNING (PBL) IN THE SUBJECT

As an application of the PBL method in the course, an activity to be developed by the students is presented in which a plant to be automated must be modeled and, using this model, build a simulated version of the plant in question. Also using this model, propose a solution based on SED to be implemented in a PLC.

METHODOLOGY FOR DEVELOPING THE PROPOSED WORK

At first, the student must propose a plant to be automated, which can be described by an SED, which requires the modeling of each subsystem independently.

Within the scope of the discipline, an embedded system was developed, based on a microcontroller, to simulate the plant modeled by the students and be able to interact with a PLC (Figure 1). In its inputs, opto-couplers were used in order to electrically isolate the control signals that come from the PLC, typically operating at 24V. In order for the microcontroller to be able to send signals equivalent to the sensor values to the PLC, its outputs are also electrically isolated.

Once the plant is defined as a set of subsystems described in SED, its programming for the embedded simulator is possible. For this, the SEDs must be translated into the C language, used in the programming of the microcontroller used. The translation can be done using Automata Oriented Programming, as through the approach of SHALYTO (1991). When implementing the plant for simulation, the values that the plant outputs (sensors) must have in each state must be programmed. In the end, there is a system that interacts with the PLC receiving control signals and sending signals from sensors, as shown in Figure 2.

Once the simulated plant is programmed, it is possible to develop a control method based on SED. As examples, there is the application of Petri Nets for control, the use of Automata to describe a desired behavior for the plant or the application of Supervisory Control Theory (TCS) for the most permissible control possible for the plant.

With the calculated or modeled controller, it is possible to implement the control in a PLC. The translation of Petri Nets to the Ladder language consists of programming the places for internal variables of the PLC while the transitions are lines of code to be executed. The transformation of a supervisory control to a Ladder language is not as straightforward as the transformation of an automaton to a programming language. This is because the

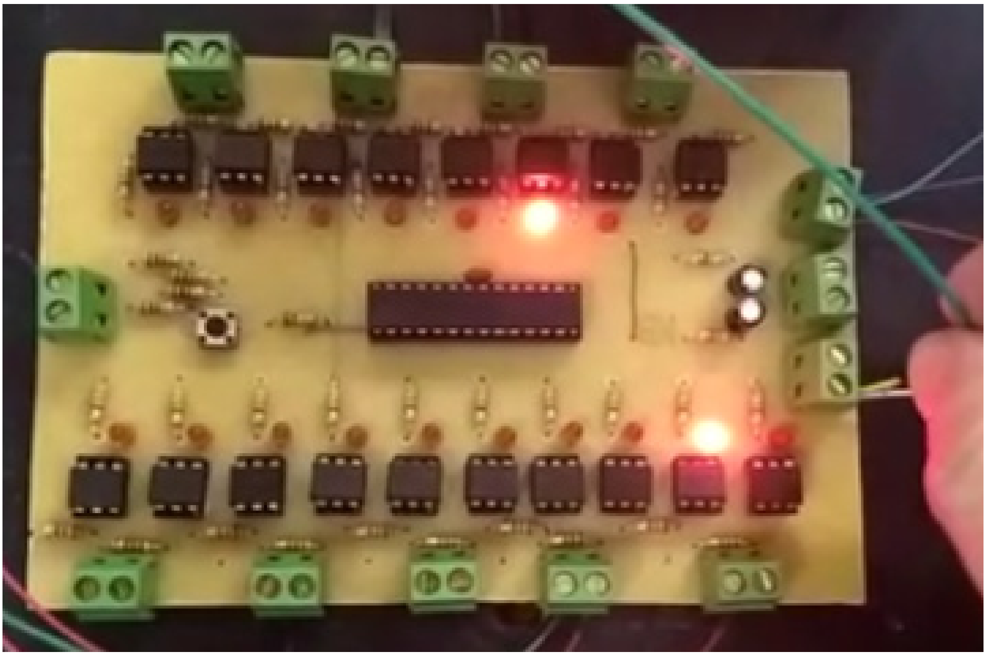

Figure 1: Embedded system for simulating the plant to be controlled.

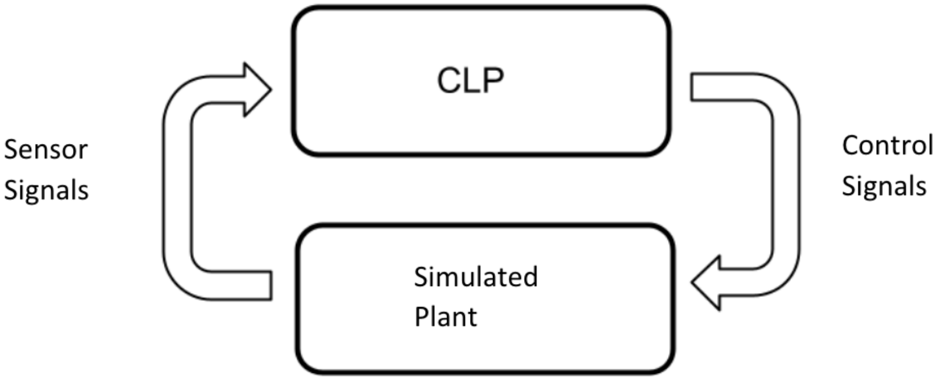

Figure 2: Interaction between PLC and simulated plant for the proposed work.

information contained in a supervisory are event disablements (PLC actions sent to the plant) that must not occur. In this way, the PLC programming must take into account, not which command to send to the plant, but which commands must not be sent in order to prevent the plant from reaching a bad state. The implementation of supervisory control in a PLC can be performed using the methodology presented in VIEIRA et al. (2017).

APPLICATION OF THE METHODOLOGY

To exemplify the proposed approach, an example was developed and is presented as one of the possible approaches to solve the proposed work. In this example, the system to be automated consists of a crane with a vertical actuator (up and down), a horizontal one (moving left and right) and a grapple to hold the load while it is transported from one side to the other. the other (Figure 3). The reference of the point where the load is initially is to the right and they must be transported to the left. For the system to carry out the transport, a button, indicating the presence of load, must be pressed. The transport cycle consists of lowering, holding the load, lifting, going to the left, lowering again, opening the grapple to leave the load in the final position, climbing and waiting for another press of the button to start a transport cycle again. The button can be pressed at any time while a load is being transported, indicating that a new load is in position for transport. The system has 4 sensors to indicate the limit switches of the vertical and horizontal movement actuators.

To program the plant simulator, it is necessary to model the behavior of each of the subsystems, vertical, horizontal and gripper movement, taking into account their real behavior. That is, the evolution of the model must be based on control signals coming from the PLC and the evolutions in a physical system are not instantaneous. For the plant in question, models were created that presented a small time between the sending of the signal and the actual occurrence of the corresponding event. For example, the system represented in Figure 4 presents the behavior expected by the vertical actuator, so, after receiving the activation signal, which causes the actuator to go down, it is necessary that a time elapses before it is considered that the actuator has actually lowered.

It is necessary to note that in states S0 and S1 the actuator is in the upper (retracted) and lower (extended) positions respectively. These states must be taken into account as a condition for activating the corresponding outputs (sensor signals) of the simulated plant. Still, the event went down and up only occurs after a predetermined time after the automaton is in states S2 and S3 respectively, being necessary to implement this behavior in the code of the simulated plant.

The other actuators were modeled with the same approach, with a time being established for each one in order to try to approximate the behavior of the simulated system to that of a real system.

In terms of models of the plant to be controlled, low-level modeling is not necessary, considering each aspect of the system such as temporal aspects, for example, it is possible to create high-level models that are just representations of the behavior of the process. For this plant, 3 models were created, one for the gripper, another for the vertical actuator and another for the horizontal one. The models can be seen in figure 5. The events ag, fg, ie, fe, id and fd are, respectively: open claw, close claw, command to go to the left, event that the system has reached the left, command to go right and event that the system has reached the right.

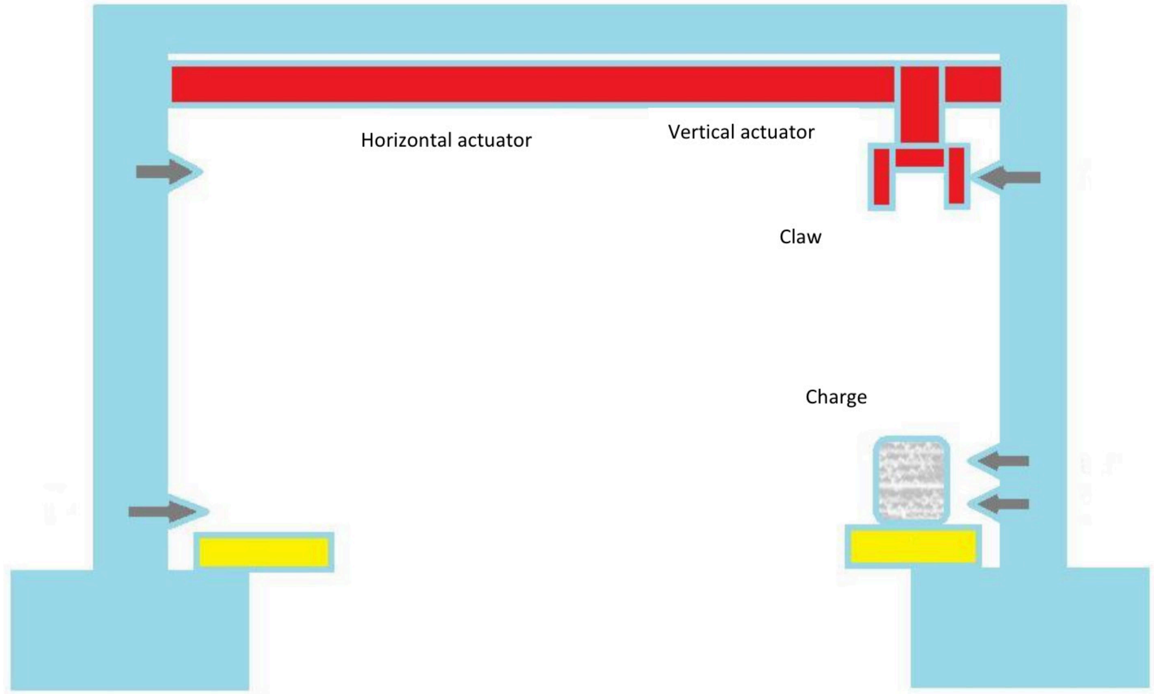

Figure 3: Sketch of the crane proposed as an industrial plant.

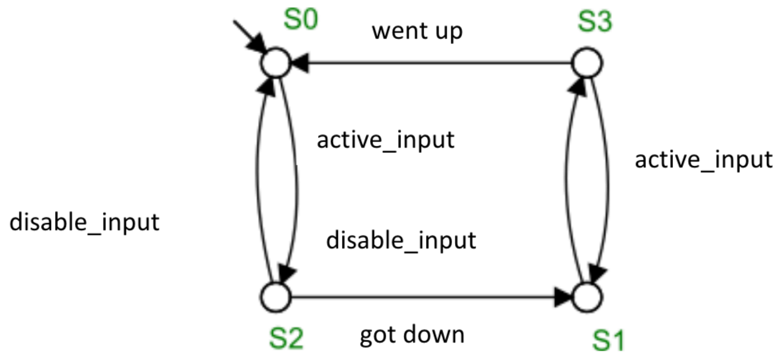

Figure 4: Simulation model of the vertical actuator in the microcontrolled system.

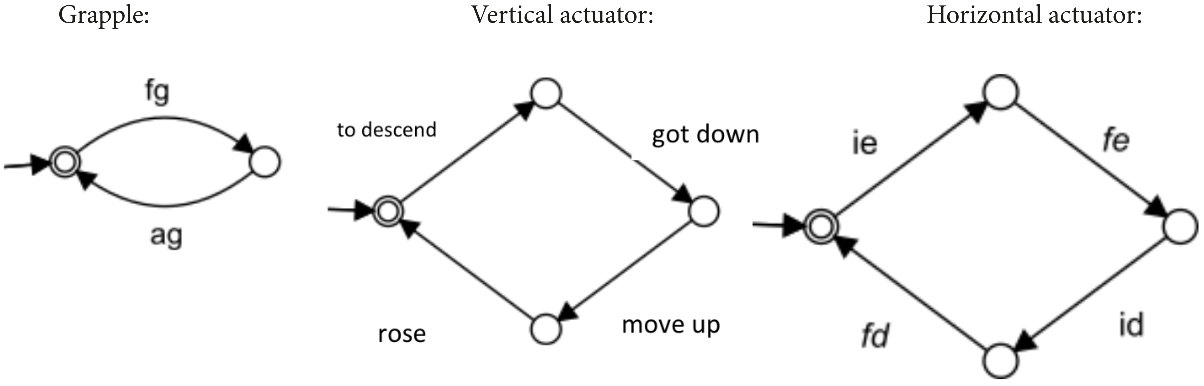

Figure 5: Models that make up the plant.

Five specifications were designed for the behavior of the plant, whose representative models are presented in Figure 6. E0 guarantees that the button must be pressed at least once for the crane to pick up the part, E1 guarantees that the button is pressed once for the first movement, E2 ensures that the grapple goes up right after releasing or holding the load, E3 prevents horizontal movement with the crane extended, and E4 ensures that the grapple opens or closes in the correct positions.

From the specifications and models of the plant, the monolithic supervisory control represented in Figure 7 is obtained.

The implementation of supervisory control in PLC could then be done, using the methodology developed by VIEIRA et al. (2017), which presents a systematic way to translate the automata into Ladder language, explaining the supervisory control disablements in each state.

FINAL CONSIDERATIONS

This article presented the application of the PBL teaching methodology and its results in an engineering discipline. This course is divided into three parts: the teaching of SED, CLP and a work that aims to apply the knowledge seen during the course uniting the first two parts.

The proposed problem reproduces real control conditions of a simulated plant, built on the basis of a microcontrolled device, through a PLC that implements a controller based on SED. This approach proved to be very effective for fixing concepts, in addition to promoting the autonomy of future engineers for teamwork and solving a real practical problem.

It is observed, however, that the discipline has a reduced workload for the great content covered (CLP and SED). Ideally, a higher load would allow for a deeper understanding of the contents or approaches used for modeling and controlling discrete systems.

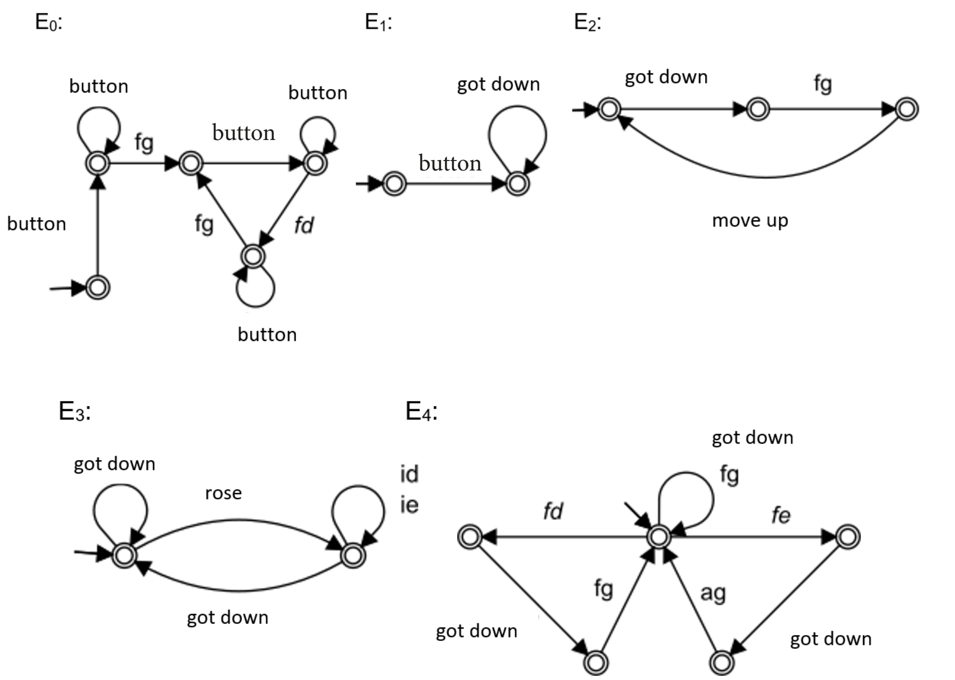

Figure 6: Models that represent the languages of the specifications.

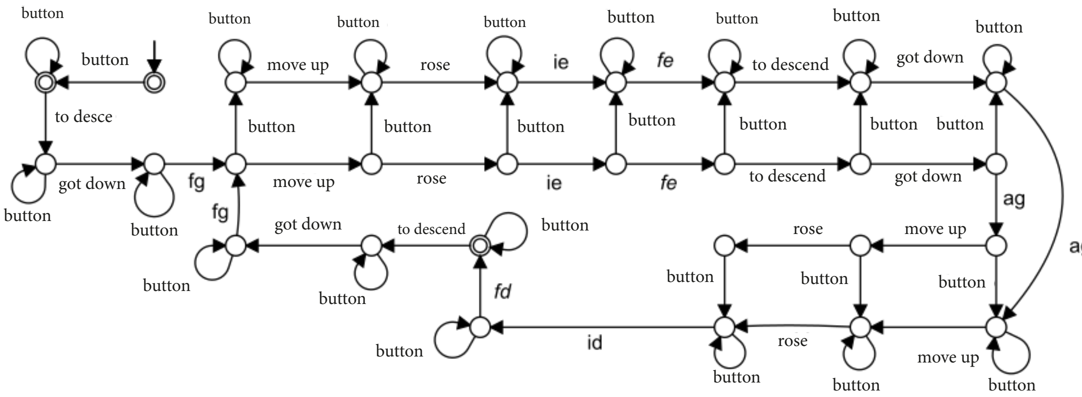

Figure 7: Monolithic supervisory control.

REFERENCES

BAZZO, W. A.; PEREIRA, L. T. V. **Ensino de Engenharia, na Busca do seu Aprimoramento.**[S.l.]: Florianópolis, Editora da UFSC., 1997.

BERBEL, N. N. **"problematization" and problem-based learning: different words or different ways?** Interface — Comunicação, Saúde, Educação, v. 2, n. 2, 1998.

BORGES, M. et al. **Aprendizado baseado em problemas. Tópicos fundamentais para a formação e desenvolvimento docente para professores dos cursos da área da saúde.** Capítulo VIII. Medicina. Ribeirão Preto, São Paulo., Jun 2014.

CARDOSO, J.; VALETTE, R. **Redes de Petri.** Editora da UFSC. Florianópolis: [s.n.], 1997. v. 1.

CASSANDRAS, C. G.; LAFORTUNE, S. **Introduction to Discrete Event Systems. Secaucus**, NJ, USA: Springer-Verlag New York, Inc., 2006.

GEORGINI, J. M. **Automação Aplicada - Descrição e Implementação de Sistemas Sequenciais com PLCS**. São Paulo: [s.n.], 2009. v. 1.

SAVIANI, D. **Escola e democracia.** São Paulo: Cortez, 1991.

SHALYTO, A. **Programmatic implementation of control automata. Automation and remote control series,** n. 13, p. 41–42, 1991.

VIEIRA, A. D.; SANTOS, E. A. P.; QUEIROZ, M. H. de; LEAL, A. B.; PAULA NETO, A. D. de; CURY, J. E. R. **A Method for PLC Implementation of Supervisory Control of Discrete Event Systems.** IEEE Transactions on Control Systems Technology, v. 25, p. 175–191, 2017.