Training Proposal based on MeiA to face Automation Challenges

Article in International Journal of Engineering Education · January 2014

5 authors, including:

A. Burgos
Universidad del País Vasco / Euskal Herriko U...
13 PUBLICATIONS 27 CITATIONS
SEE PROFILE

Maria Luz Alvarez Gutierrez
Universidad del País Vasco / Euskal Herriko U...
5 PUBLICATIONS 9 CITATIONS
SEE PROFILE

Elisabet Estevez
Universidad de Jaén
94 PUBLICATIONS 475 CITATIONS
SEE PROFILE

Marga Marcos
Universidad del País Vasco / Euskal Herriko U...
170 PUBLICATIONS 731 CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Design of Multiaxial Parallel Kinematic Machines with High Dynamic Capabilities (DYNAMESCH) View project

Ultra Flexible Parallel Kinematic Machines (MEDUSA) View project

All content following this page was uploaded by A. Burgos on 30 November 2015.

The user has requested enhancement of the downloaded file.
Training Proposal based on MeiA to face Automation Challenges*

ARANTZAZU BURGOS¹, ISABEL SARACHAGA², MARIA LUZ ALVAREZ¹, ELISABET ESTÉVEZ³ and MARGA MARCOS²
¹ Departamento de Ingeniería de Sistemas y Automática, EUITI de Bilbao, UPV/EHU. Paseo Rafael Moreno ‘Pitxitxi’ 3, 48013 Bilbao, Spain. E-mail: {arantzazu.burgos, marialuz.alvarez}@ehu.es
² Departamento de Ingeniería de Sistemas y Automática, ETSI de Bilbao, UPV/EHU. Alameda de Urquijo s/n, 48013 Bilbao, Spain. E-mail: {isabel.sarachaga, marga.marcos}@ehu.es
³ Departamento de Ingeniería Electrónica y Automática, EPS de Jaén, Campus Las Lagunillas, 23071 Jaén, Spain. E-mail: eestevez@ujaen.es

The new challenges in process control require strengthening the development of design skills and the use of real-world experiences throughout the engineering curricula. So this paper presents a training proposal that aims to incrementally develop competences for implementing the control software of industrial automation systems within engineering curricula. The MEthodology for Industrial Automation systems (MeiA) guides students through the development process from the analysis, through the design, implementation and operation phases. The student competences are developed through three incremental and integrated levels: (1) initial procedure; (2) introduction of design principles and (3) development of methodological skills. This methodological proposal implicitly introduces fundamental concepts of software engineering. This ensures that students are able to develop complex control systems following structured programming concepts. In addition, they generate different type of documentation about the design, reducing the effort from design to implementation and operation. Furthermore, the training proposal focuses on motivating the students through real-world problems, introducing analysis and design issues incrementally through the three levels. At the same time, multi-disciplinary skills are also introduced by integrating the whole development process from the very beginning. The experience after the implementation of the training proposal shows a significant increase in student interest in this engineering field in recent academic years.

Keywords: engineering education; industrial automation; control software; MeiA methodology

1. Introduction

The current manufacturing industry must address a technological change in production processes to evolve towards the so-called ‘innovative processes’ [1] that helps to achieve and maintain leadership in the economic environment of the third millennium. Robotics, automation and computer integrated manufacturing constitute the engine for process innovation, the main keywords used for defining advanced manufacturing technologies [2, 3]. They allow the achievement of process control and automation through modelling and production.

According to the European Commission Ad-hoc Industrial Advisory Group [4], the ‘factories of the future’ must be reusable, flexible, modular, intelligent, virtual, digital, affordable, easy to adapt, easy to operate, easy to maintain and very reliable. So, they require more and more complex, safe and trustworthy automation and control systems. The design of these applications demands control and automation engineers who are able to manage tools and technologies to automate, control and optimise the production processes, ensuring plant availability while providing high quality production with zero defects.

All these requirements have a direct impact on education, as the competences required to operate the factories of the future must be acquired during the students’ academic careers. Curricula need to keep up to date, strengthening the development of design skills and the use of real-world experiences throughout the control and automation engineering curriculum. In this way, the gap between engineering studies and real industrial environment can be reduced.

In this sense, there are initiatives in different engineering areas towards the achievement of these goals. In a recent study on the perception of the role of modelling (related to both the analysis and design processes) within academia, McKenna and Carberry [5] explain the potential changes that might occur in the engineering curriculum in order to support the process of innovation based on the concept of modelling. Strong [6] recognizes the growing need for enhanced design education and multidisciplinary competences for undergraduate students, by means of offering students elective series of courses within the so-called Multidisciplinary Design Stream, introducing different engineering disciplines in different years. Villalobos and González [7] describe how to design project-based
computing engineering programs that address how to incrementally develop competences for understanding complex problems and designing their solutions. Secundo et al. [3] present a case study that focuses on the needs of project-based and action-oriented learning approaches for manufacturing, by means of practical experiences in companies. This aims at developing student capacity for solving problems in the design and manufacturing of complex products. Sell and Seiler [8] propose a design-centric approach for teaching mechatronics and robotics, supported by innovative tools that include on-line engineering environments based on modern web technologies. Dhaouadi et al. [9] explain an integrated learning approach, which consists of a sequence of structured, guided and open-ended design experiences in the area of control engineering at undergraduate level.

Following this stream, this paper presents a training proposal related to the design and development of control software for industrial automation systems. The final goal is to train students to face complexity-bounded real problems through the development phases, from analysis, and design, to implementation and operation. In order to achieve this goal, the MEthodology for Industrial Automation systems, MeiA, is applied [10]. It offers design guidelines and templates in order to methodologically guide developers through the definition of the ultimate design of industrial control systems, using well known methods and guides within the automation and software engineering disciplines.

This methodology is based on lessons learned over more than 20 years of teaching the development of control systems for real automation processes, final engineering degree projects (most of them for companies), as well as master and research projects. This cooperation between academia and industry leads to the need for a new training approach based on industrial expectations, putting special emphasis on a multidisciplinary approach. The implementation of the methodology over three levels allows the student to acquire constructive learning and progressive abstraction capacities and, as a consequence, the acquisition of competences by the student is facilitated.

As the goal is to train students to be able to apply this methodology in their future professional life (regardless of the technology to be used), the subjects are complemented with laboratory exercises that consist of practical implementations of designs using PLCs (Programmable Logic Controllers). This allows design validation and deals with practical issues that are also very important [11]. Furthermore, given the multidisciplinary nature of the development teams and the need for collaboration with other people outside the team, communication skills are also addressed within this learning approach.

The paper is structured as follows. Section 2 briefly describes the domain specific modelling methods used in the training proposal, as well as giving an overview of the MeiA methodology. Section 3 introduces the training proposal structured on the three levels mentioned above: initial, design and methodological levels, while Section 4 presents the deployment of this training proposal in the University of the Basque Country (UPV/EHU). Section 5 presents, as a proof of concept, an assessment restricted to the second level. It includes the resolution of a typical design problem as well as student opinion surveys in the last five academic years. Finally, Section 6 gives some concluding remarks. This last section also presents future perspectives with respect to the development of a tool based on the MeiA methodology.

2. MEthodology for Industrial Automation systems (MeiA)

The MeiA methodology proposes a development process for automation projects that relies on GEMMA (Guide d’Etude des Modes de Marches et d’Arrêts) [12], UML (Unified Modeling Language) use case diagrams [13] and GRAFCET (GRAph Fonctionnel de Commande, Etapes, Transitions) [14] using the lexicon, syntax and semantics that the final users commonly use. GEMMA and use case diagrams are applied during the analysis phase and GRAFCET is used during the design phase. In particular:

- UML use case diagram is used to document the system behaviour from the user point of view. Use cases describe the functional requirements of the system.
- GEMMA helps in the collection of correct specifications, ensuring that inconsistencies and unexpected situations are avoided. It is a guide for performing a systematic search of the possible states of an automated process from a control perspective.
- GRAFCET allows describing the behaviour of sequential systems. This international standard IEC-60848 [15] is a modelling language for specifying automation systems.

GRAFCET and GEMMA are widespread methods in the automation field. GRAFCET is broadly used in industry and in education by automation engineers. GEMMA is less widespread, although it was conceived to complement the first method. However, the synergy of both graphical formalisms together with UML, use case diagrams represent a
powerful combination to face the development of control software for industrial automation systems.

Within Spanish universities, GEMMA, GRAFCET, or both, are traditionally used in courses related to industrial process automation. As a result, there are classic publications that constitute the basic bibliography of current degree courses [16–20]. However, more recent publications [21], free access teaching material with practical examples [22–24] and on-line courses [25] are also available. From an international perspective, numerous universities have also incorporated the combination of these methods in courses related to automation [26–33].

But, if the goal is to replace traditional intuitive methodologies for systematic methodologies that cover potential weaknesses in the students training in terms of system modelling, most of these attempts are far from the objective. Some of them establish relationships between GEMMA states and GRAFCET designs in a very accurate and clear way [21, 34]. However, the analysed works do not consider all possible states and how to consider them in the designs.

The MeiA methodology includes the best practices of the analysed works together with new proposals coming from the ideas and experiences of the authors. Some other works also discuss the convenience of including design considerations about the operator panels, the monitoring system, the security system, etc. These have also been taken into account in the MeiA methodology.

As with the GEMMA guide, all analysed works suggest open approaches to applying knowledge in a practical way with many possibilities for interpretation. This forces developers to create their particular methodology. Conversely, the MeiA methodology guides the designers to correctly define the industrial control systems using modelling languages that manage the concepts they commonly use. The operation modes are identified by means of GEMMA. UML use case diagrams are used to identify the actors participating in the operation modes. From these, the evolution conditions in GEMMA states can be derived. Finally, a set of pre-defined GRAFCET templates assist during the design of the use cases.

Although the MeiA methodology is presented in Alvarez et al. [10], a brief summary is presented here, highlighting the key parts related to the incremental methodology implementation on three levels.

The first phase establishes the main sequence of the automation system. It organizes the control system start-up in automatic operation mode, as well as the safe stop. It also generates the control signals that report to the production procedures: when the system is ready to start the production or a stop at the end of cycle has been requested. For this purpose, the steps to be performed are illustrated in Table 1.

The second phase analyses if manual operation is needed. Commonly this means to verify certain individual movements or process parts out of the usual cycle sequence; for instance, sensor or actuator calibration, preventive maintenance operations, etc. Generally these operations will be performed under maintenance personnel supervision. When manual operation mode is required, the steps to be performed are illustrated in Table 2.

The third phase analyses the need to verify certain movements (or process parts) either step by step or continuously. Generally these operations will be performed according to the usual cycle sequence under the control of the personnel in charge of the task. Tables 3 and 4 present the steps to be performed for step by step verification and functional block verification, respectively.

In the fourth phase, the process failures are identified, analysed and evaluated, distinguishing two types: those that continue the production (even accepting product quality degradation) and those that require a controlled stop. For each failure, the steps to be performed are illustrated in Table 5.

The fifth phase evaluates the emergencies and the actions to reach a safe state. Table 6 presents the steps to be performed.

In the sixth phase, the process actions must be studied and the production cycle must be defined. For this purpose, the steps to be performed are illustrated in Table 7.
Guidelines for the user have been defined for each methodology phase. During the analysis procedure, these guidelines support the generation of GEMMA and UML use case diagrams. Furthermore, they allow the identification of the control and monitoring requirements, as well as the configuration of the operator panel and other required auxiliary panels.

During the design procedure, two types of GRAFCET diagrams have been defined: the so-called decision GRAFCETs and the production GRAFCETs. The former organize and coordinate

---

### Table 2. Phase II: manual

<table>
<thead>
<tr>
<th>Steps</th>
<th>Synopsys</th>
<th>L1</th>
<th>L2</th>
<th>L3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>To establish the activation procedure for this operation mode, as well as the system states from which this operation mode can be requested.</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Step 2</td>
<td>To determine if manual operation mode has higher priority than automatic mode. If so, its activation implies an immediate process stop that includes all tasks needed to evolve to a safe situation from either a production point of view or a personnel safety point of view.</td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Step 3</td>
<td>To identify the available control elements to perform the process actions. Each action must be analysed to determine whether safety procedures are required.</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Step 4</td>
<td>To define the de-activation procedure for this operation mode.</td>
<td></td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Step 5</td>
<td>To analyse the tasks to reach the initial state after manual mode is stopped. Alternatively, the procedure of initial and safety conditions can be adapted.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 6</td>
<td>To analyse the need to adapt the shutdown process with the actions required by the stop of this operation mode.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 7</td>
<td>To define the activation procedure for automatic operation mode, if manual operation mode has higher priority.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>

### Table 3. Phase III: test (step by step)

<table>
<thead>
<tr>
<th>Steps</th>
<th>Synopsys</th>
<th>L1</th>
<th>L2</th>
<th>L3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>To establish the activation procedure for this operation mode, as well as the system states from which this operation mode can be requested.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 2</td>
<td>To identify the available control elements to perform each step.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 3</td>
<td>To define the de-activation procedure for this operation mode.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>

### Table 4. Phase III: test (functional block)

<table>
<thead>
<tr>
<th>Steps</th>
<th>Synopsys</th>
<th>L1</th>
<th>L2</th>
<th>L3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>To establish the activation procedure for this operation mode, as well as the system states from which this operation mode can be requested.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 2</td>
<td>To determinate the procedure of initial and safety conditions for the functional block. Alternatively, the system procedure of initial and safety conditions can be adapted.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 3</td>
<td>To analyse the need of a start-up procedure for the functional block. Alternatively, the system start-up procedure can be adapted.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 4</td>
<td>To identify the available control elements to perform the functional block execution.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 5</td>
<td>To define the de-activation procedure for this operation mode.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 6</td>
<td>To analyse the need of a shutdown procedure for the functional block. Alternatively, the system shutdown procedure can be adapted.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>

### Table 5. Phase IV: failures

<table>
<thead>
<tr>
<th>Steps</th>
<th>Synopsys</th>
<th>L1</th>
<th>L2</th>
<th>L3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>To establish the activation procedure for this operation mode, as well as the system states from which this operation mode can be requested.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 2</td>
<td>To determinate the procedure of initial and safety conditions for the functional block. Alternatively, the system procedure of initial and safety conditions can be adapted.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 3</td>
<td>To analyse the need of a start-up procedure for the functional block. Alternatively, the system start-up procedure can be adapted.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 4</td>
<td>To identify the available control elements to perform the functional block execution.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 5</td>
<td>To define the de-activation procedure for this operation mode.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Step 6</td>
<td>To analyse the need of a shutdown procedure for the functional block. Alternatively, the system shutdown procedure can be adapted.</td>
<td></td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>
the possible system states, such as system start-up, automatic or manual operation mode, emergency, etc.; the latter correspond to the actions related to the production cycle. In this case, the guidelines support the generation of the decision GRAFCETs and the control signals to be used in the production GRAFCETs.

Furthermore, when the designs procedures require the coordination of different tasks, coordination requires special attention. In this sense, a guideline has been established: vertical coordination (also called hierarchical coordination) is used for decision GRAFCETs, while horizontal and vertical coordination are used for production GRAFCETs.

3. Training proposal

The new training proposal aims at making students incrementally develop competences for implementing the control software of industrial automation systems. The MeiA methodology guides students to face these developments at the three levels, addressing all the development phases. A collection of real-world examples allows gradually incorporating system states, and complex process operations are introduced incrementally.

During laboratory sessions, the specifications of a real process (of different complexity, depending on the methodology level) are given to the student who has to face the design and development of the control system. The methodology encourages the student to take into account the following aspects:

- To use a tool for generating GRAFCET designs. For instance, the SFCEdit tool [35]. It is recommended to use one compliant with the IEC 60848 standard [36].
- To define parallel sequences, when possible, in order to minimize process delays.
- During code generation, the students are required to separate the activation/de-activation sequence of the GRAFCET from the outputs to the process generation. The former is transformed into sequential code. The latter is programmed as separate code and is in charge of generating the outputs to the process based on the current active steps.
- The automation projects must be IEC 61131-3 standard compliant [37].
- The unitary and integrated tests of the implemented designs must be documented using pre-defined templates.
- A set of templates is provided for the student in order to generate the complete documentation of the implementation.
- It is also recommended to use a PLC programming tools that support the PLCopen XML format [38].

The three levels of the methodology are described in the following subsections.

<table>
<thead>
<tr>
<th>Table 6. Phase V: emergencies</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Steps</strong></td>
</tr>
<tr>
<td>Step 1</td>
</tr>
<tr>
<td>Step 2</td>
</tr>
<tr>
<td>Step 3</td>
</tr>
<tr>
<td>Step 4</td>
</tr>
<tr>
<td>Step 5</td>
</tr>
<tr>
<td>Step 6</td>
</tr>
<tr>
<td>Step 7</td>
</tr>
<tr>
<td>Step 8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 7. Phase VI: normal production</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Steps</strong></td>
</tr>
<tr>
<td>Step 1</td>
</tr>
<tr>
<td>Step 2</td>
</tr>
<tr>
<td>Step 3</td>
</tr>
</tbody>
</table>
3.1 Initial level

The main objective of this initial level focuses on consolidating the concept of a control system to ensure that the student is able to establish the relationship between the control program execution and the process behaviour.

The basic concepts are studied through simple examples that have enough entity to allow the student to:

- Identify the different components of the physical system (machines, sensors, actuators, pre-actuators, operation panels, etc.).
- Distinguish between the signals that provide process information (i.e. outputs from the process and inputs to the control system) and those signals that act directly on the process (i.e. outputs from the control system and inputs to the process).
- Design control systems for simple case studies using GRAFCET as the design tool. The final goal is to design a control system that includes the main sequence with automatic operation mode request. Thus, the verification of the initial and safety conditions, a stop at the end of cycle request during production and production stop are required. The emergency state is also taken into account and the corresponding safe stop must be triggered. The analysis is performed according to the steps of the corresponding MeiA phases, while pre-defined GRAFCET templates assist the student during the design.
- Implement the corresponding automation projects in order to validate the designs.
- Document all the development phases.

To illustrate the type of basic design problem that must be used in this phase, a case study consisting of a warehouse that supplies parts by gravity (illustrated in Fig. 1) is presented. The process consists of two cylinders and five sensors. A double-acting cylinder (A) extracts a part from the warehouse and a second double-acting cylinder (B) moves the piece to the transfer point. Each cylinder has two limit-switch sensors that detect if the cylinder is retreated or expanded. The warehouse has one sensor, which indicates that there is a part at the bottom of the warehouse. The operation panel consists of three pushbuttons (Start, Stop and Reset), an alarm indicator and an Emergency stop pushbutton.

The student must design and implement the control system (shown in Fig. 2) that consists of the following parts:

- Main Sequence: As commented above, it organizes the control system start-up in automatic operation mode, as well as the system stop. It also generates the control signals that report to the production procedures if the system is ready to start the production (NormalProd) or a stop at the end of cycle has been requested (StopEC). In this case, the verification of the initial and safety conditions is required before starting production.
- Initial Conditions: It takes the process to a stop state with the cylinders in the initial position, i.e.

![Fig. 1. An initial level case study: a warehouse that supplies parts by gravity.](image-url)
cylinder A retreated (a0) and cylinder B retreated (b0), and no parts present in the system.

- **PartToTrans**: When the system is in normal production (NormalProd) and a part is detected (s1), the cylinders move the part to the output position. The request of a stop at the end of cycle stops the procedure at the initial step (X20).

- **Emergency**: The activation of the emergency stop pushbutton performs an immediate process stop with the de-activation of all Grafcets. Once the emergency stop pushbutton is unlocked and the reset pushbutton is pressed, the initial steps of all Grafcets are activated.

In summary, during this initial level, the terminology of the discipline is introduced to the student through simple examples and design problems. Although GEMMA has not been formally presented, the minimal operation states of a manufacturing system are introduced. The student uses GRAFCET to define the system operation and the operator interface. The student is thus prepared to go into the design level after having acquired the basics of automation system design.

### 3.2 Design level

The main objective of this second level focuses on consolidating the operation modes and concurrency to ensure that the student is able to associate the parallel operations of the process with the concurrent tasks that control them, addressing the coordination and synchronization problems.

The complexity of the examples is increased by introducing some concurrent operations and new operation modes (manual, step by step or production in presence of failure operation modes). The main sequence is extended to include new procedures (start-up process, shutdown process). At this stage GEMMA is introduced and an overview of the MeiA methodology is presented.

All this is intended to ensure that the student is able to:

- Identify all possible states in which a control system may operate.
- Design and implement control systems that include these states.
- Establish the information required by the monitoring system, the control elements (pushbuttons, selectors, etc.) and indicators for operation panels, security systems, etc.
- Detect the need to adapt or modify the operational part in order to facilitate, simplify or improve the control system development.
- Write documentation about every phase of the development cycle.

The use of case studies allows the working of all these aspects. To illustrate this, a case study is presented: a system for processing bales (see Fig. 3). The process is composed of a pressing system, an incoming conveyor of bales, a priming system, a packing system and the operation panel. The operation panel consists of three pushbuttons (Start, Stop and Reset), an automatic/manual selector (Auto/Man), an alarm indicator, an emergency stop pushbutton and a pushbutton for each manual action.

As shown in Fig. 4, the complexity of the control system increases. It has a more complex main
sequence, new operation modes, and increasing complexity of the production operations. Furthermore, the signal number also increases: there are 10 signals coming from the sensors, 12 signals are sent to the actuators, and 18 signals are related to the operation panel. These are the signals coming from/to the process and operator. The control system also manages signals from/to the supervision system but these are taken into account while developing the control system.

In this example the control system consists of 12 procedures:

- **Main Sequence:** As commented on at the initial level, it organizes the automatic operation mode and generates the control signals for production procedures. At the design level the examples add complexity by introducing a start-up process before starting production and a shutdown process after a production stop.
- **Initial Conditions:** It takes the process to a stop state where all elements are taken to its initial position and there are no parts within the system.
- **Start-up Process:** It tests if the bale feeder and the priming tank are full using Load Bale and Load Tank, respectively. These procedures are also used during Normal Production when the bale feeder or the priming tank is empty.
- **Shutdown Process:** It empties and cleans the priming tank after a production stop.
- **Normal Production:** It consists of four production operations: the incoming conveyor of bales controlled by Input Conveyor, the pressing system controlled by Press Bale, the priming system controlled by Prime Bale, and the packing system controlled by Output Conveyor-Packing.
• Manual: It performs individual process movements by means of the push-buttons in the operation panel. This operation mode is not introduced in the initial level.
• Emergency: It performs an immediate process stop.

In summary, during this level the student is introduced in concurrent tasks during normal production procedure, i.e., he/she must deal with parallel GRAFCETs. More complex examples are studied, as he/she is familiar with the terminology and the design of very basic designs. Finally, in order to manage the designs of different operation states that are typical in the real world, the GEMMA guide is introduced. At this stage, the student is prepared to study and to apply the whole MeiA methodology.

3.3 Methodological level

The main objective of this third level focuses on consolidating the concept of hierarchy to ensure that the student is able to analyse, design and implement systems of different complexity with different control levels.

Since students are already aware of the need of an engineering approach to develop industrial software, fundamental aspects are introduced about the phases of the software life cycle, the development models, the techniques and methods for specification, analysis and design, and the control software architectures conform to the standards for automation projects. Thus, new concepts from the engineering software domain customized for the automation domain are introduced. The formal models and modelling languages are discussed, although the student has used them as tools at previous levels. Multi-disciplinary design and model integration is introduced as a means for managing complexity. In this sense, separation of concerns issues is highlighted.

The MeiA methodology is applied in detail, including all the aspects to consider during the control software development for discrete processes. The objective aims to ensure that the student is able to:

• Capture the data needed to provide the specification for the control system.
• Carry out the analysis, design and implementation of individual systems, as a starting point to introduce integration aspects and operation in conjunction.
• Design the control system for a process of medium complexity, such as a cell controller.
• Write documentation about every phase of the development cycle.

At this level, the student works with systems of medium complexity. To illustrate a typical case, an example is presented that consists of a flexible assembly cell FMS-200 [39] that is in charge of mounting a set of four pieces (base, bearing, shaft and cap). The cell consists of four stations and a modular conveyor system (Transport Station). In
the first station, the base is placed on the pallet located on the transportation system. In the second, the bearing and shaft are placed on the base, while the cap is put in the third station. In the last station, the set mounted on the pallet is stored in the warehouse.

Figure 5 presents the detail of the first station named the Base Location. The base is extracted from the base warehouse and, if its position is correct, it is placed on the pallet. After that, the pallet is placed on the conveyor system and once the base is on the pallet, this is moved to the following station (Transport Station). When the position is not correct, the base is rejected.

The control system development has, as a main requirement, that every station may operate independently or in conjunction with the other stations. Thus, when developing one station, the student can reuse the work done during the initial and design levels. If the student comes from another training plan, the individual control of every station is the first task to develop as introductory work to the matter. A partial view of the control system is presented in Fig. 6, where the decision and production GRAFCETs are represented.

When the system works as a whole, every station’s main sequence is cancelled and it is substituted by the main sequence of the general control that organizes the start-up of all stations in automatic operation mode, and their programmed stop. It also generates the control signals that report to the production procedures whether the system is ready to start the production (CG_NormalProd) or a stop at the end of cycle has been requested (CG_StopEC). The verification of the initial and safety conditions (Fig. 7) and the start-up process are performed for all stations before starting production and the shutdown processes after a production stop. Normal production executes the production plan, sequencing operations and coordinating all stations.

In summary, at this level, the student is introduced to the whole aspects of a design of an automation system. Coordination aspects are very important at this level as well as the methodological aspects for developing software systems. As a natural stream, the methodology is introduced and the student works the complete phases acquiring procedures that he/she will be able to use in his/her professional life.

4. Deployment of the training proposal in the UPV/EHU

In the fall semester of 2007 and in order to give a satisfactory answer to the Bologna Process reforms, the ‘System engineering and automatic control’ Department of the University of the Basque Country (UPV/ EHU) started the design of a new proposal for a master’s degree in the area. The
master’s degree in Control Engineering, Automation and Robotics (60 ECTS—European Credit Transfer System) aims to provide advanced training in the design, implementation, operation and maintenance of control and automation systems for monitoring, controlling and managing production processes, which require high performance, energy saving, pollution reduction, efficiency and safety. In addition, it is oriented towards research activities as a prelude to the associated Ph.D. programme. This one year master’s degree has two elective (but not exclusive) branches of expertise: automation and control theory.

Within the automation branch, the ‘Design of Automation Systems’ subject (6 ECTS) aims to train students to face the development of industrial control systems. For achieving this objective, this subject must combine the maturity of the software engineering disciplines with the well-spread methods and standards of the industrial automation field. Furthermore, given the importance of understanding the rationale behind an automation solution for achieving efficient knowledge transfer [40], the MeiA methodology was designed in order to manage the strategic knowledge underlying the steps to solve automation projects, integrating well-spread standard methodologies and techniques.

The master’s degree in Control Engineering, Automation and Robotics was approved by the Spanish National Agency for Quality Assessment and Accreditation (ANECA) and the teaching activities started in the academic year 2009/2010. At the end of this first academic year, the ‘Design of Automation Systems’ subject was evaluated in relation to the acquisition of student skills using the MeiA methodology and its application to case studies having growing complexity (4 ETCS over 6 ETCS of the subject are dedicated to practical issues). The results indicated that the students had acquired problem-solving skills leading to quality designs. However, the assessment of the subject revealed the need for a different approach due to the different student’s level of maturity in the subject, as the access to the master degree may come from different bachelors. The assessment also allowed us to identify the potential changes that might be included in the future Industrial Electronics and Automatic Engineering bachelor degree for achieving whole competences in the master’s degree in a less forced manner. In this way, the training proposal presented in this paper was conceived and launched.

Currently, the competences of the Initial level are addressed in the ‘Automatism and Control’ subject, while the ‘Industrial Informatics’ subject deals with the competences of the Design level. These subjects are taught in the second and third year of the Industrial Electronics and Automatics Engineering bachelor degree, respectively. The competences of the Methodological level are worked in the ‘Design of Automation Control Systems’ subject, which is taught in the master in Control Engineering, Automation and Robotics.

Table 8 summarizes the deployment of this training proposal. The introduction of the different levels is presented chronologically identifying the degree, the year, the subject, the number of ETCS of the subject, the number of ETCS specifically used for the application of the MeiA methodology from the total ETCS of the subject, and the development of a group project.

4.1 Student assessment

The student assessment has been developed for the second level. Concretely, the resolutions of a design

Fig. 7. Verification of initial conditions of the general control (CG_Initial Condition).
problem and the student opinion surveys have been used to perform the assessment.

On the one hand, the design problem has been evaluated in the 'Industrial Informatics' subject (6 ECTS). This problem corresponds to 20% of the evaluation vector. The design problems are worked in the classroom using different teaching–learning techniques. Roughly, the working dynamic is the following:

- **First session.** The design problem is presented. The student works on it and he/she must develop a solution for the next session. Three hours is the estimated time that the student must dedicate.
- **Second session.** Teams of four students are organized. The team discusses the individual solutions worked by every student for 30 minutes, looking for a common solution. Thus collaborative learning strategies are promoted, which increases self-efficacy for learning the subject material [41]. The final solution can be one of the proposals or a combination of them. It is delivered on paper with the names of the team members and the team number.
- **Third session.** During the classroom the lecturer discusses with the students the good points of the designs as well as the incorrect parts. In this way, the teams with an incorrect or incomplete solution can modify their solution accordingly before delivering it again.

This working dynamic is evaluated positively. The team members discuss the different proposals and reach an agreement about the solution to deliver. Despite this, only 20% of the groups deliver a new solution, which merges contributions from the members, and 80% of them choose the solution of one member and stop working on the problem. In any case, the students work on the problem and get involved with the discussion about the correct alternatives and the reported errors. Thus, lecturers can identify the gaps and the evolution of the student designs, which compensates the significant workload.

In order to evaluate the design problem, several aspects are taken into account: the structure of the solution, the consideration of all the required operation modes and the way they are solved, the use of

### Table 8. Chronological deployment of the training proposal.

<table>
<thead>
<tr>
<th>Level</th>
<th>Group</th>
<th>Years</th>
<th>Subjects</th>
<th>Credits</th>
<th>Design</th>
<th>Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Master's degree</td>
<td>Control Engineering, Automation and Robotic (for five-year graduates)</td>
<td>1&lt;sup&gt;st&lt;/sup&gt;</td>
<td>Design of Automation Systems</td>
<td>6</td>
<td>Start</td>
</tr>
<tr>
<td>1-2</td>
<td>Engineering degree</td>
<td>Technical Industrial Engineering mention in Industrial Electronics (previous career)</td>
<td>3&lt;sup&gt;rd&lt;/sup&gt;</td>
<td>Industrial Informatics</td>
<td>7.2 (equivalent)</td>
<td>Start</td>
</tr>
<tr>
<td>1</td>
<td>Bachelor degree</td>
<td>Industrial Electronics and Automation Engineering</td>
<td>2&lt;sup&gt;nd&lt;/sup&gt;</td>
<td>Automation and Control</td>
<td>6</td>
<td>Start</td>
</tr>
<tr>
<td>2</td>
<td>Bachelor degree</td>
<td>Industrial Electronics and Automation Engineering</td>
<td>3&lt;sup&gt;rd&lt;/sup&gt;</td>
<td>Industrial Informatics</td>
<td>6</td>
<td>Start</td>
</tr>
<tr>
<td>3</td>
<td>Master's degree</td>
<td>Control Engineering, Automation and Robotic II (adapted for four-year graduates)</td>
<td>2&lt;sup&gt;nd&lt;/sup&gt;</td>
<td>Design of Automation Systems</td>
<td>3</td>
<td>Start</td>
</tr>
</tbody>
</table>
parallel sequences for optimising the production and minimizing process delays, and the coordination of decision and production GRAFCET. The mark ($p$) is considered from zero to ten:

- Without design ($p < 3$). The design is not structured and it has fatal errors. It does not meet all functional requirements.
- Insufficient design ($3 \leq p < 5$). The design presents a vague structure that only addresses parts of the functional requirements. Furthermore, it has serious coordination failures.
- Correct design ($5 \leq p < 7$). The design covers all functional requirements with minor errors and parts that can be optimised.
- Good design ($7 \leq p < 9$). Some parts of the design can be improved.
- Excellent design ($p \geq 9$). The design takes into account the all specifications correctly and they have been solved in a structured way, considering the coordination aspects properly and optimising the process.

The evaluation results for the last five academic years are summarized in Fig. 8. For each year, a graphic presents the number of students, the punctuation distribution, and the percentage failed and passed. One more graphic gathers the punctuation distribution for all the years. It is necessary to highlight the significant improvement in the last two academic years. In the last year, it should be noted that not only the number of students who have passed the design problem has increased, but also the average with a larger number of designs marked higher than 7, as showed in Figs 9 and 10.

On the other hand, Table 9 summarizes the results from the opinion survey to the students on their initial interest in the subject at the beginning and their interest at the end of the subject for the last five academic years. These questions are part of the
student opinion surveys about the lecturer at the end of the subject, which are managed by the Teaching Assessment Service of the University. The student opinions range from 1 (Very low interest) to 5 (Very high interest), but they can also select 6 (Do not know/Do not answer). The graphical representation of these results (illustrated in Fig. 11) points out the improvement of the student interest at the end of the subject in the last two academic years. Students are more motivated because they face real-world problems and learn the practical aspects of engineering.

This significant increase in student interest has provoked a significant increase in the number of students who apply for a Final Degree Project in this field. Moreover, either the Final Degree Projects or the team projects are based on more complex systems. In any case, the students improve their ability to identify the data needed to solve each control part. As a result, the designs are more structured, better documented as they use a template that guides the documentation process, and is developed in less time because the number of errors in the analysis and design phases decreases. In addition, the students become more independent because the methodology guides them when facing the analysis and design of industrial control systems. This translates into a teaching load reduction by decreasing the questions and doubts.

<table>
<thead>
<tr>
<th>Academic year</th>
<th>Very Low</th>
<th>Low</th>
<th>Medium</th>
<th>High</th>
<th>Very High</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008-09</td>
<td>8.1%</td>
<td>45.9%</td>
<td>35.1%</td>
<td>8.1%</td>
<td></td>
</tr>
<tr>
<td>At the beginning</td>
<td>8.1%</td>
<td>32.4%</td>
<td>37.8%</td>
<td>18.9%</td>
<td></td>
</tr>
<tr>
<td>At the end</td>
<td>8.7%</td>
<td>58.5%</td>
<td>30.4%</td>
<td>4.3%</td>
<td></td>
</tr>
<tr>
<td>2009-10</td>
<td>8.2%</td>
<td>32.4%</td>
<td>37.8%</td>
<td>18.9%</td>
<td></td>
</tr>
<tr>
<td>At the beginning</td>
<td>8.7%</td>
<td>52.2%</td>
<td>30.4%</td>
<td>8.7%</td>
<td></td>
</tr>
<tr>
<td>At the end</td>
<td>8.7%</td>
<td>52.2%</td>
<td>30.4%</td>
<td>8.7%</td>
<td></td>
</tr>
<tr>
<td>2010-11</td>
<td>3.2%</td>
<td>6.5%</td>
<td>45.2%</td>
<td>32.3%</td>
<td>9.7%</td>
</tr>
<tr>
<td>At the beginning</td>
<td>3.2%</td>
<td>41.9%</td>
<td>32.3%</td>
<td>16.1%</td>
<td></td>
</tr>
<tr>
<td>At the end</td>
<td>3.2%</td>
<td>41.9%</td>
<td>32.3%</td>
<td>16.1%</td>
<td></td>
</tr>
<tr>
<td>2011-12</td>
<td>4.5%</td>
<td>13.6%</td>
<td>54.5%</td>
<td>9.1%</td>
<td>13.6%</td>
</tr>
<tr>
<td>At the beginning</td>
<td>4.5%</td>
<td>36.4%</td>
<td>50.0%</td>
<td>9.1%</td>
<td></td>
</tr>
<tr>
<td>At the end</td>
<td>4.5%</td>
<td>36.4%</td>
<td>50.0%</td>
<td>9.1%</td>
<td></td>
</tr>
<tr>
<td>2012-13</td>
<td>30.6%</td>
<td>61.1%</td>
<td>2.8%</td>
<td>0.0%</td>
<td></td>
</tr>
<tr>
<td>At the beginning</td>
<td>2.8%</td>
<td>22.2%</td>
<td>58.3%</td>
<td>13.9%</td>
<td></td>
</tr>
<tr>
<td>At the end</td>
<td>2.8%</td>
<td>22.2%</td>
<td>58.3%</td>
<td>13.9%</td>
<td></td>
</tr>
</tbody>
</table>
This training proposal attempts the shortcomings of traditional approaches in order to address the growing need for an enhanced design education and the use of real life industrial problems.

5. Conclusions

This paper has presented a training proposal aimed to incrementally develop student competences for constructing the control software of industrial automation systems. The proposal is based on methods and standards widely used in the automation field. It is based on the MeiA methodology that combines GEMMA, UML use case diagrams and GRAFCET for assisting the designer, as well as on a collection of real-world examples that gradually introduces more complex processes through the three levels of implementation.

This methodological proposal implicitly introduces fundamental concepts of software engineering required for the development of such systems, which are difficult to introduce in engineering curricula, mainly due to time constraints. Thus, this proposal ensures that the student is able to develop complex control systems in a structured and well documented manner with fewer errors in the analysis and design phases—in short, quality designs that require shorter times for implementation and operation.

The designs use the pre-defined GRAFCET templates that capture the designer experience and so, they include not only key structural aspects but also aspects related to flexibility, modularity and extensibility in the designs that can be used for a high number of systems. Design reuse is achieved at earlier development phases; this in turn assures an effective productivity in the development of the automation system.

In order to implement the designs, the use of domain standards is mandatory. Students develop automation projects compliant to the IEC 61131-3 standard during the three levels, and they use PLC programming tools that support the PLCopen XML format at least in the methodological level. When generating a PLCopen XML file, it provides portability of applications, software reuse at different levels and a powerful interface to other software tools.

Current work is focused on the development of a tool based on the MeiA methodology. This tool guides the user through the different steps and phases in order to customize generic models for a particular control system, offering the terminology they commonly use.

Acknowledgements—This work was supported in part by the Government of Spain under projects DPI2012-37806-C02-01, the Government of the Basque Country (GV/EJ) under Project IT719-13 and by UPV/EHU under grant UFI11/28.

References


17. O. Boix, A. Sudria and J. Bergas, Automatización industrial amb GRAFCET, Coleccion Aula practica 12, Ediciones UPC (Universitat Politècnica de Catalunya), Barcelona, 1993.


Arantzazu Burgos is a permanent associate professor of Automatic Control and Systems Engineering at the University of the Basque Country. She has her BSc in Computer Science from the University of Deusto in 1991, Master’s in Robotic and Automation in 1992, Specialist in Industrial Computing in 1992 and Ph.D. in Telecommunication Engineering from the University of the Basque Country in 1997. She started her work at a Technology Centre where she participated as a researcher in different communication projects related to CAD/CAM/CAE systems and control systems for flexible manufacturing systems. Between 1992 and 1998, she worked at the Faculty of Engineering as coordinator of the Advanced Manufacturing Technology Master. Since 1992, she has worked as a researcher on different projects related to communications, industrial automation systems, software engineering and bioelectronics, some of them in collaboration with companies in the Basque Country. Since 2006, she has also been a member of the GCIS research group.

Isabel Sarachaga is a permanent associate professor of Automatic Control and Systems Engineering at the University of the Basque Country. She graduated with BSc in Computer Science from the University of Deusto in 1990 and obtained her Ph.D. degree from the same University in 1999. She started her work at a Technology Centre where she participated as a researcher in different communication projects related to SCADA systems, industrial networks and fieldbuses. Between 1993 and 1997, she worked at the Faculty of Engineering as coordinator of the Industrial Software Engineering Master, but also acted as a researcher on different projects related to communications, control systems for flexible manufacturing systems and software engineering. Since 2001, she has been a member of the GCIS research group and acted as a researcher in different projects related to distributed industrial control systems.

Maria Luz Álvarez obtained her Physic Degree from the University of the Basque Country in 1990 and her Master’s in Advanced Manufacturing Technology in 1994. She has worked as a teacher and researcher in the Automatic Control and Systems Engineering Department at the University of the Basque Country. She is working on a thesis on the development of a methodology for the automatic generation of software reconfigurable distributed control. Since 2011, she has been a member of the GCIS research group and acted as a researcher on different projects related to distributed industrial control systems.

Elisabet Estévez is a Lecturer of Electronic and Automatic Engineering at the University of the Jaen. She obtained her degree in Telecommunications Engineering in 2002 from the University of the Basque Country (UPV/EHU). In 2007 she obtained her Ph.D. in Automatic Control from UPV/EHU. She has co-authored more than 70 technical papers in international journals and conference proceedings in the field of distributed industrial control systems. She has also been involved in several research projects funded by National and European R&D programmes. Her main expertise deals with applying software engineering concepts to the industrial control field. She has carried out review work for various conferences and technical journals.
Marga Marcos received her BSc in Physics, MSc and Ph.D. in Control Engineering from the University of the Basque Country in 1983, 1984 and 1988, respectively. She is professor in Automatic Control and Systems Engineering at the University of the Basque Country, where she was Vice-Dean of the Faculty of Engineering from 1990 to 1993. She was chairman of the Automatic Control and Systems Engineering Department for more than 10 years (1995–2005). She has authored and co-authored more than 140 technical papers in international journals and conference proceedings. She has acted as the main researcher and researcher of more than 80 research projects funded by National and European R&D programmes. She has carried out review work for various conferences and technical journals. She has served on technical committees of IFAC (AARTC, WRTP, CC), IEEE (NBCS) and she has been a member of the European Control Association Council, the National Organizing Committee of IFAC Spain and Publication Co-Chair for the IEEE Control and Decision Conference (CDC2005). She has been the General Co-Chair of the *IEEE International Conference on Emerging Technologies for Factory Automation* 2010. Her research interests include model based design of distributed automation and embedded systems and real-time control of manufacturing and robotic systems.