



PLATAFORMA DE APRENDIZAGEM DE AUTOMAÇÃO INDUSTRIAL COM CODESYS, MODBUS, ARDUINO E RECURSOS WEB

Tales Silva Rodrigues^{*1}, Fábio Vincenzi Romualdo da Silva¹ e Renato Ferreira Fernandes Júnior¹

¹FEELT - Universidade Federal de Uberlândia

Resumo - Em cursos de graduação e técnicos na área de automação industrial, aulas práticas onde o aluno tem acesso aos equipamentos industriais como transmissores, controladores lógicos programáveis, inversores de frequência, soft-starter, sensores e atuadores, dentre outros dispositivos, são de elevada importância na formação do aluno. No entanto, esses recursos são caros, o que dificulta o processo de ensino e de aprendizagem em automação industrial. Além disso, atualmente, os sites podem disponibilizar conteúdo adicional e de modo diferenciado para que o aluno complemente e amplie o seu conhecimento em um determinado assunto. Nesse sentido, esse trabalho propõe uma plataforma constituída do software CODESYS, do protocolo Modbus, da plataforma aberta de prototipagem Arduino e de recursos audiovisuais, apostilas, programas nas diversas linguagens descritas na norma IEC 61131-3, animações relacionadas a automação industrial, boas práticas de programação na linguagem ANSI C, regras MISRA C, lógica de ligação dos sensores e atuadores que são conectados nas entradas e saídas de um controlador lógico programável, dentre outros recursos digitais que serão disponibilizados a curto e longo prazo em um site especialmente construído para compor a plataforma de aprendizagem proposta.

Palavras-Chave- Aduino, automação industrial, controladores lógicos programáveis, Protocolo Modbus, software CODESYS.

INDUSTRIAL AUTOMATION LEARNING PLATFORM WITH CODESYS, MODBUS, ARDUINO AND WEB RESOURCES

Abstract - In undergraduate and technical courses in the field of industrial automation, practical classes where the student has access to industrial equipment such as transmitters, programmable logic controllers, frequency inverters, soft-starters, sensors and actuators, among other devices, are of high importance in student training. However, these resources are expensive, which makes the teaching and learning process in industrial automation difficult.

*talles.rodriques@ufu.br

Otherwise, nowadays, the sites can provide additional content, in different ways, for the student to complement and expand their knowledge on a given subject. In this sense, this work proposes a platform consisting of the CODESYS software, the Modbus protocol, the Arduino open prototyping platform and audiovisual resources, handouts, programs in the various languages described in IEC 61131-3, animations related to industrial automation, good practices programming in ANSI C language, MISRA C rules, logic for connecting sensors and actuators that are connected to the inputs and outputs of a programmable logic controller, among other digital resources that will be made available in the short and long term on a website specially built to compose the proposed learning platform.

Keywords - Aduino, CODESYS software, industrial automation, programable logic controllers, Modbus protocol.

I. INTRODUÇÃO

O profissional que atua na área de automação industrial necessita de conhecimento em várias áreas como: eletrônica, programação, robótica, acionamentos elétricos, controle de sistemas, deve saber utilizar vários softwares que são proprietários, etc. Nesse sentido, sua formação necessita de vários recursos de hardware/licenças de softwares industriais que são de difícil acesso devido ao custo elevado.

Por isso, muitas universidades, cursos técnicos de curta duração, como os oferecidos pelo SENAI e outras instituições, até os cursos integrados ao nível médio “curso técnico” e superior de curta duração “curso tecnológico”, buscam soluções alternativas para poderem preparar melhor seus alunos para entenderem melhor os conceitos de diversas áreas do conhecimento, bem como visualizarem na prática os fundamentos teóricos, formando assim um cidadão mais capaz de enfrentar as diversas situações práticas que encontrará no mercado de trabalho.

Outro ponto a ser destacado é o acesso da informação pelo aluno. Os dispositivos móveis disponibilizam a informação a qualquer tempo e lugar. Por isso, podem desempen-

har um importante papel no processo de ensino e de aprendizagem fornecendo conteúdo complementar aquele disponibilizado em sala de aula.

Nesse sentido, muitos trabalhos tem sido propostos na área acadêmica. Um projeto cooperativo desenvolvido por uma rede de escolas e empresas de diferentes regiões da Espanha consiste no desenvolvimento de uma casa e sistema de automação predial para ensino prático profissionalizante Treinamento com a plataforma de prototipagem Arduino [1].

Outro trabalho usa a plataforma Arduino para implementar um sistema de automação residencial para o treinamento de alunos de engenharia da Universidade Plovdiv “Paisii Hilenarski” em eletrônica [2].

Em parceria, o instituto federal e uma escola estadual de Florianópolis-SC desenvolveram um conjunto de experimentos de Física nas áreas de Óptica, Termodinâmica e ondas utilizando a plataforma Arduino para fazer a aquisição de dados e o controle de um conjunto de experimentos [3].

Como o objetivo de explorar conceitos básicos de Ciência, Tecnologia, Engenharia e Matemática em aulas práticas de robótica educacional, utilizou-se a plataforma Arduino e impressões 3D para crianças e adolescentes de 10 a 17 anos [4].

Outra proposta é utilizar a robótica educacional, apoiada na plataforma Arduino, para o ensino de conceitos da linguagem de programação C no curso de Aprendizagem Industrial em Informática [5].

Muitos dos trabalhos publicados utilizam recursos de fácil aquisição, como a plataforma Arduino, para o ensino e prática de domótica, eletrônica, física, matemática, ciência, linguagem de programação C, dentre outras áreas do conhecimento. No entanto, quando se trata do ensino e prática na área de automação industrial, deve-se considerar a importância da norma IEC 61131 que padroniza a programação de CLPs, bem como os protocolos de comunicação utilizados na indústria. Além disso, o processo de ensino/aprendizagem/prática, quando direcionados à situações encontradas no chão de fábrica, proporcionam melhor preparo na solução de problemas encontrados na indústria, autoconfiança e estímulo ao aluno que almeja atuar na referida área.

Nesse sentido, este trabalho propõe o uso do software CODESYS, protocolo Modbus, plataforma de prototipagem Arduino, sensores e atuadores de baixo custo como recursos didáticos no processo de ensino e prática de automação industrial. Ademais, busca-se propor experimentos que são situações encontradas em chão de fábrica, bem como conceitos básicos importantes na programação de CLPs referente a problemas encontrados na indústria. O acesso ao material produzido será por meio de site especialmente construído para esse propósito.

II. DESCRIÇÃO DO PROJETO

A norma IEC 61131 padroniza a programação de CLPs e é seguida pelas empresas que produzem equipamentos industriais. Essa norma é dividida em várias partes, sendo que a terceira parte IEC 61131-3 trata das linguagens de programação. Isto é, fornecer metodologias de construção de lógicas de pro-

gramação de forma estruturada e modular, permitindo a quebra dos programas em partes gerenciáveis. Regulamenta abordagem e estruturação *top-down* e *bottom-up*, fundamentada em 3 princípios: modularização; estruturação e reutilização. Considerando essas e muitas outras diretrizes, a IEC 61131-3 define cinco linguagens de programação:

- ST (Structured Text) Texto Estruturado
- IL (Instruction List) Lista de Instruções
- LD (Ladder) Linguagem ladder
- FBD (Function Block Diagram) Diagrama de bloco
- SFC (Sequential Flow Chart) Diagrama de Fluxo

Um dos softwares muito utilizado na aprendizagem de linguagem de programação de CLP é o CODESYS, pois é um ambiente de desenvolvimento para controladores programáveis de acordo com a norma IEC 61131-3. O desenvolvedor do CODESYS disponibiliza uma versão demonstrativa onde pode-se programar e simular os programas de forma 100% operativa. Ele não é orientado a um tipo específico de hardware de controlador programável ou sistema embarcado, inclusive não dependendo do fabricante do hardware. É amplamente utilizado na indústria para diversos tipos de automações, variando de lógicas simples à robótica, controle de movimento e CNC. Vale salientar que fabricantes de equipamentos industriais como a Rockwell, Siemens, Schneider Electric, dentre outros, cobram licença de suas interfaces de desenvolvimento, o que dificulta o treinamento.

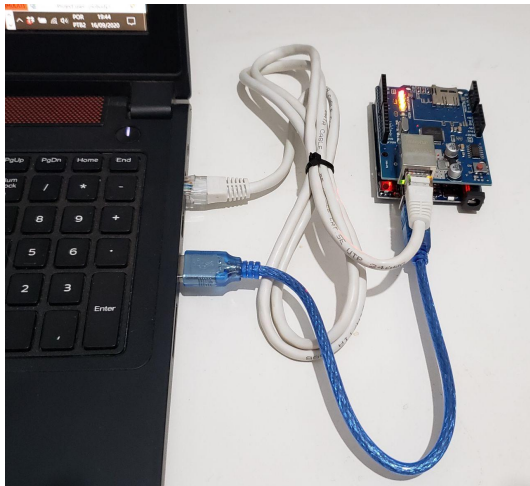
O CODESYS também pode operar como um Soft PLC, isto é, pode transformar um computador em um controlador lógico totalmente funcional e programável. Nesse caso, a comunicação entre o computador e outro dispositivo pode ser realizada por meio do protocolo Modbus. É um Protocolo de comunicação de dados utilizado em sistemas de automação industrial. Criado em 1979 pela fabricante de equipamentos Modicon, atual Schneider Electric, que posteriormente transferiu os direitos do protocolo para a Organização Modbus em 2004. A utilização do protocolo é livre de taxas de licenciamento. O Soft PLC é uma ferramenta muito útil na prática de programação de linguagens de CLPs, pois em se tratando de treinamento, podem substituir a compra de CLPs que costumam ser equipamentos caros.

Além de CLPs, os sensores, atuadores, transmissores e inversores de frequência também fazem parte da automação de processos industriais. Como os CLPs, também são equipamentos caros que totalizariam milhares de reais. Uma maneira de contornar o investimento elevado em equipamentos industriais em se tratando de treinamento, seria adotar sistemas embarcados, sensores, interruptores e atuadores de baixo custo. Nesse caso reproduz-se a lógica empregada no processo industrial utilizando-se recursos mais acessíveis.

A plataforma proposta possui uma configuração básica de *hardware*, como mostra a Figura 1. Ela é composta de um Shield Ethernet W5100 acoplado a um Arduino UNO. O shield é conectado ao laptop por meio de um cabo de rede com conector RJ45 e o Arduino é conectado ao laptop com cabo

USB. A comunicação Modbus com o CODESYS (que está operando como Soft PLC) é feita por meio do Shield Ethernet. Essa configuração é denominada básica porque está presente em todos os experimentos realizados.

Figura 1: Configuração Básica.



O CODESYS possui um canal para leitura de dados que serve para fazer a aquisição de informação e outro para a escrita de dados com o propósito de enviar informação.

O Soft PLC implementado pelo CODESYS é o mestre na comunicação Modbus e o Arduino o escravo. Somente um mestre Modbus pode solicitar informação para um escravo Modbus. Um escravo Modbus nunca pode solicitar informação para um mestre, nem mesmo para outro escravo que esteja presente na rede.

Para que ocorra a comunicação Modbus, de acordo com as especificações do referido protocolo, é necessário implementar no dispositivo escravo (Arduino) *holding registers* que são um bloco de memória destinado a armazenar os valores de entrada e saída (I/O), dentre outros valores necessários para a automação de um determinado sistema industrial. Desse modo, o mestre não pode ler diretamente uma entrada analógica/digital/ou qualquer outra variável do dispositivo escravo. É necessário que o dispositivo escravo disponibilize os valores solicitados pelo mestre nos *holding register* para que o mestre tenha acesso e vice versa. Toda informação enviada pelo mestre é armazenada nos *holding register* do escravo antes que ele possa utilizá-la. A Figura 2 mostra o diagrama do banco de memória de leitura e escrita implementada no CODESYS (mestre Modbus) e do banco de memória reservada no Arduino (escravo Modbus) [6]. Para que a comunicação ocorra, cada *holding register* de leitura e de escrita implementado no Arduino precisa ter o seu correspondente no canal de leitura e escrita do CODESYS.

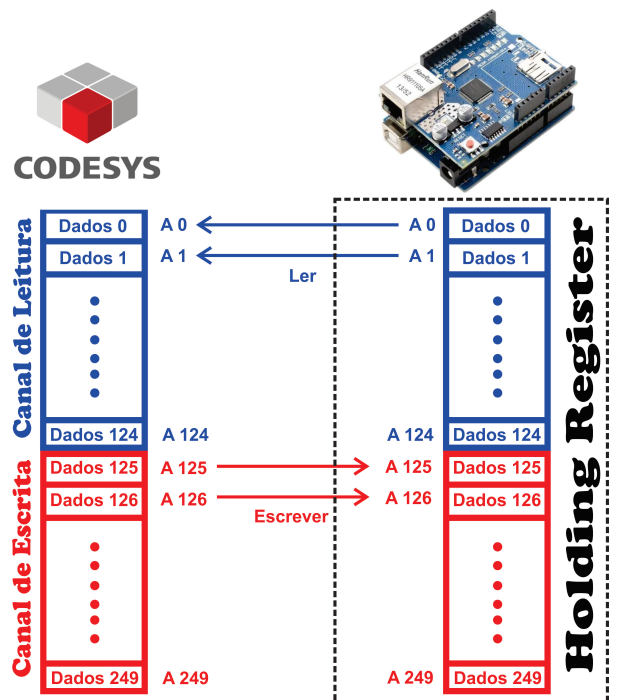
O CODESYS possui todos os recursos para operar como mestre ou escravo Modbus. Na função de mestre pode comandar vários escravos, como ocorre em muitas automações industriais onde um CLP mestre Modbus gerencia vários transmissores e atuadores Modbus.

Cada *Holding Register* é do tipo `uint16_t` (número inteiro

sem sinal de 16 bit), sendo que um bit pode representar o valor atual de uma entrada ou saída digital. Por isso, **saber realizar operação bit a bit é importante** na programação do Arduino. Isto é, ler ou alterar o valor de um único bit no registrador sem mudar o valor dos demais.

No entanto, para que o aluno se acostume com os padrões de programação utilizados na indústria, recomenda-se o uso do padrão ANSI C ao invés de funções particulares da plataforma Arduino, como ocorre com a função `bitSet()`, por exemplo.

Figura 2: Holding Register.



III. OBJETIVOS DO PROJETO

O objetivo geral deste projeto é propor uma configuração básica de software/hardware em conjunto com experimentos na área de automação industrial de modo a favorecer o compartilhar de conhecimento e de treinamento na referida área.

Os objetivos mais específicos são:

- Utilizar softwares de uso gratuito ou que possam ser de fácil aquisição;
- Utilizar as linguagens de programação especificadas pela norma IEC 61131-3;
- Quando possível na programação do Arduino, fazer uso do padrão ANSI C, regras MISRA C e de boas práticas de programação, que são mais adequadas a ambientes industriais.
- Hardware utilizado deve ser de fácil aquisição;
- Propor experimentos que estejam relacionados a automação industrial;

- Utilizar protocolo de comunicação industrial;
- Disponibilizar documentação para a fácil instalação e utilização do software de programação, do hardware e execução dos experimentos sugeridos além de outros exemplos;
- O material será disponibilizado a curto (janeiro de 2021) e longo prazo (indefinido) no site que está sendo especialmente construído para esse propósito. O domínio www.sirius.guru ou www.trix.guru estará funcional para dar acesso ao referido site.

A indústria possui uma diversidade enorme de situações que necessitam de automação. Desse modo, a curto prazo será disponibilizado material explicativo de como utilizar a configuração básica proposta, bem como vários experimentos de situações encontradas na indústria. A longo prazo, serão acrescentadas ao material mais situações envolvendo automação industrial nas diversas linguagens regulamentadas pela norma IEC 61131-3.

IV. ALGUMAS SUGESTÕES DE EXPERIMENTOS

Essa seção possui o objetivo de proporcionar uma visão geral dos experimentos que podem ser explorados com a plataforma proposta. Os detalhes de como montar, como programar e realizar os testes estarão disponíveis no site. Os experimentos propostos não possuem a pretensão de cobrir todos os detalhes, processos e conhecimento necessário para realizar automação em qualquer processo industrial, visto que esse é um tema a ser explorado a longo prazo devido as diversas situações existentes na prática.

Além disso, o CODESYS disponibiliza uma gama enorme de recursos, dentre os quais podemos citar: operadores booleanos, operadores matemáticos, blocos de funções, elementos específicos de cada linguagem, dentre outros operadores. Os exemplos podem ser implementados em qualquer uma das linguagens da norma IEC 61131-3.

A seguir, em cada experimento, um painel de controle deve ser montado com os seguintes comandos em comum:

1. Botão pulsador BT1 para acionar o equipamento e BT2 para pará-lo. Um botão retentivo BT3 de emergência para interromper o funcionamento de tudo;
2. Uma luz verde no painel deve indicar que o sistema está energizado;
3. Uma luz branca para indicar que o equipamento está em operação;
4. Se o botão de emergência (BT3) for pressionado uma lâmpada vermelha de ser energizada. Nesse caso e na situação de parada (BT2) a lâmpada branca deve permanecer apagada.

A. Esteiras

A esteira é um equipamento muito utilizado no ambiente industrial. Considere uma esteira A e uma esteira B, na sequência da A. Ao final de cada esteira existe um sensor fotoelétrico

barreira. Cada esteira é comandada por um motor. Elaborar um programa para realizar as funções:

1. Realizar a programação usando selo, botões (BT1, BT2 e BT3) NA e sensores fotoelétrico barreira (S1 e S2) NA. O sensor S1 está posicionado ao final da esteira A e S2 ao final de B;
2. As esteiras devem ser desligadas caso não passe objeto pelos sensores por 10s para economizar energia;
3. Utilizar o sensor da esteira (A) para contar (contador CTU) as caixas que estão passando. O contador necessita de um botão para reiniciar a contagem;
4. Utilizar o contador decrescente (contador CTD) para desligar as esteiras quando um determinado número de caixas passar pelo sensor. Implementar um botão de reset para reinicializar o contador;
5. Utilizar a função set/reset ao invés do selo e usar botões (BT1, BT2 e BT3) NF e sensores fotoelétrico barreira (S1 e S2) NF;
6. Implementar um gerador de pulso para que a cada 3 segundos acione um mecanismo para colocar uma caixa na esteira;
7. Encapsule o gerador de pulso em um *function block*;
8. Um dispositivo de impressão precisa ser acionado por 100ms toda vez que uma borda de descida do sensor S1 é detectada (indicando o final da caixa). O dispositivo de impressão registra na parte superior da caixa a data de fabricação, validade e lote de fabricação.

B. Tanques

Os tanques industriais são recipientes estáticos utilizados em diversos processos, geralmente destinados ao armazenamento de líquidos. Os tanques fazem parte de diferentes operações no setor, como: produção, tratamento, transporte, refino, distribuição, reserva, etc. Algumas das indústrias que utilizam tanques são: química, cosméticos, alimentícias, farmacêutica, usina sucroalcooleira e petrolífera.

Um tanque possui duas válvulas. Uma válvula para encher (V1), localizada na parte superior do tanque. Outra localizada na região inferior do tanque para esvaziá-lo (V2). Também há dois sensores de nível do tipo NA. Um para indicar que o tanque está cheio (S1) e outro para indicar que o tanque está vazio (S2).

Elaborar um programa para realizar as funções:

1. Adicionar um botão BT4 para encher o tanque durante 15s e um botão BT5 para esvaziar por um período de 15s. Resolver o problema várias vezes, usando temporizadores diferentes (TON, TOFF, TP);
2. Mudar a função do botão BT4 para encher o tanque até atingir o nível máximo indicado pelo sensor S1;
3. Alterar a função do botão BT5 para esvaziar o tanque até atingir o nível mínimo indicado pelo sensor S2;

4. Implementar controle de nível PID de modo que o fluxo de saída permaneça constante (implementar um modelo matemático do tanque no Arduino).

C. Linha de Produção

Uma linha de produção é composta por uma esteira de entrada A com um sensor óptico barreira NF (S1) posicionado no final dela. Quando a peça passa por esse sensor um braço robótico pega a peça e coloca dentro de uma máquina CNC (Comando Numérico Computadorizado) com programação ISO, linguagem G. Quando a peça fica pronta, o braço robótico coloca a peça na esteira de saída B que possui um sensor óptico do tipo barreira NF no seu início (S2) e um sensor (S3) do mesmo tipo no seu final.

Além da configuração padrão, o painel de controle possui um contador de peças ligado no sensor da esteira de saída B e um botão pulsador para reiniciar a contagem.

Elaborar um programa para realizar as funções:

1. Quando o botão BT1 for pressionado a esteira A entra em operação e a luz branca do painel deve ser ligada. No momento em que o sensor S1 detectar o fim da peça (gerar uma borda de subida) a esteira A deve ser desligada. Assim que o sensor S2 detectar uma peça a esteira A entra em operação novamente e para quando o sensor S1 detectar o fim de uma nova peça. Esse ciclo irá se repetir até que o botão BT2 ou BT3 seja pressionado;
2. Quando o sensor S2 detectar o começo de uma peça ela deverá permanecer ligada até que o sensor S3 detecte o seu fim;
3. Toda vez que o sensor 2 detectar uma peça o contador no painel deve incrementar uma unidade.

D. Modelos Matemáticos

Vale destacar que modelos matemáticos de tanques de vários formatos e dimensões, sensores, atuadores, dentre outros dispositivos industriais podem ser implementados no Arduino de modo a proporcionar uma experiência de automação mais próxima daquela encontrada no chão de fábrica. Controladores PID, Fuzzy, além de outros, desde que não exijam muito processamento, também podem ser implementados de modo a fornecer uma experiência mais próxima da real.

V. RESULTADOS EXPERIMENTAIS

Uma empresa vende concreto usinado e consome em média 211 litros de água por m^3 de concreto usinado. Na empresa existe um tanque de água com capacidade de 2110 litros. Ele possui um sensor de nível com sinal de saída 4-20ma/0-5v/0-10v (tensão de saída $\leq 510\Omega$ e alimentação de 12 ~ 36VDC). Uma bomba de 2 cv é utilizada para encher o reservatório e uma válvula de controle por solenóide NF de duas polegadas é utilizada para escoar o líquido.

No escritório da empresa existe um SoftPLC (Mestre Modbus) que possui a interface gráfica mostrada na Figura 3. Uma chave seletora de dez posições determina a quantidade de água

que será bombeada para o tanque. Na posição (1) a caixa receberá 211 litros de água, na posição (2) 422 litros, na posição (3) 633 litros e assim por diante.

Quando o botão “MOTOR” é pressionado a bomba liga e enche a caixa até atingir o nível especificado pela chave seletora. Quando o botão “VÁLVULA” é pressionado a válvula abre até a caixa ficar totalmente vazia.

Figura 3: SoftPLC implementado com CODESYS.



Em campo, o controle do tanque é feito por um dispositivo Modbus escravo com dois botões pulsadores, caso seja necessário acrescentar mais água a mistura. Um dos botões aciona a bomba e o outro a válvula.

O experimento foi montado com um Arduino Uno, um Ethernet Shield W5100, cabo ethernet RJ45, potenciômetro de 2k2, software CODESYS. O potenciômetro foi utilizado para fornecer na entrada ADC0 do Arduino um sinal de 0 a 5V, fazendo assim, o papel do sensor de nível.

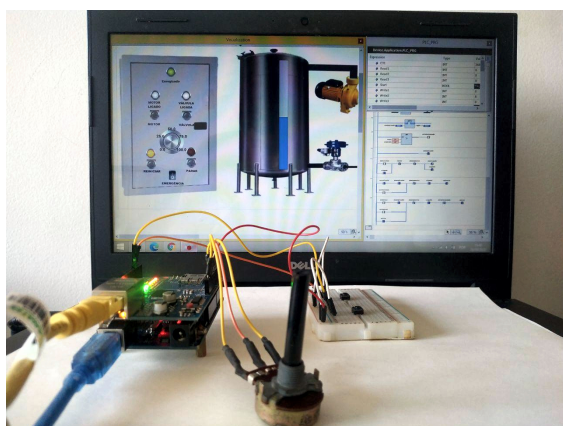
Após a configuração da rede Modbus no software CODESYS, as variáveis do projeto foram mapeadas como mostrado na Figura 4. Em seguida, foi criado no CODESYS uma lógica, em linguagem LADDER.

Figura 4: Variáveis e Registradores.

Variáveis CODESYS	CODESYS Modbus Master IPV4: 192.168.2.74
ADC0 : INT = ReadData0;	ReadData0 - %IX0.0...%IX0.7 : %IX1.0...%IX1.7
-	ReadData1 - %IX2.0...%IX2.7 : %IX3.0...%IX3.7
BTF_Valvula : BOOL = %IX4.0;	ReadData2 - %IX4.0...%IX4.7 : %IX5.0...%IX5.7
BTF_Motor : BOOL = %IX6.0;	ReadData3 - %IX6.0...%IX6.7 : %IX7.0...%IX7.7
Reiniciar : BOOL = %QX0.0;	WriteData0 - %QX0.0...%QX0.7 : %QX1.0...%QX1.7
-	WriteData1 - %QX2.0...%QX2.7 : %QX3.0...%QX3.7
-	WriteData2 - %QX4.0...%QX4.7 : %QX5.0...%QX5.7
-	WriteData3 - %QX6.0...%QX6.7 : %QX7.0...%QX7.7

Variáveis Arduino	Arduino Modbus Master IPV4: 192.168.2.12
int : ADC0 = analogRead(A0);	= Mb.MbData[0]
-	= Mb.MbData[1]
int : BTF_Valvula;	= Mb.MbData[2]
int : BTF_Motor;	= Mb.MbData[3]
bool : BTF_Reiniciar;	= Mb.MbData[4]
-	Mb.MbData[5]
-	Mb.MbData[6]
-	Mb.MbData[7]

Figura 5: Slave Modbus



VI. CONCLUSÕES

O trabalho proposto sugere o uso do software CODESYS, plataforma Arduino, protocolo Modbus e Website com instruções, exemplos e códigos diversos de modo a compor uma plataforma de estudo e de prática das linguagens descritas na norma IEC 61131-3. A programação é realizada no software CODESYS que não é orientado a um tipo específico de hardware de controlador programável ou sistema embarcado e é amplamente utilizado na indústria para diversos tipos de automações. Na plataforma proposta o CODESYS é utilizado como SoftPLC e desempenha a função de Master Modbus.

A plataforma de prototipagem Arduino realiza o papel de um dispositivo Slave Modbus e pode ser utilizado para ler sinais analógicos, realizar I/O digital, executar modelos de dispositivos industriais, controles PID, fuzzy, dentre outras funções.

O fato dos softwares que devem ser executados no Arduino, CODESYS, configurações, ligações do hardware, in-

formações sobre sensores, atuadores, conceitos sobre automação industrial, vídeos, dentre outras informações importantes e necessárias para a realização dos experimentos, estarem disponíveis em um site, facilita o acesso dos interessados e a atualização das informações por parte dos desenvolvedores.

A plataforma Arduino não substitui o CLP no processo de automação industrial, porém pode proporcionar recurso valioso no processo de ensino e de aprendizagem quando o propósito é a prática de linguagens da norma IEC 61131-3, protocolo Modbus, I/O digital, leitura de sinais analógicos e programação da linguagem C.

Considerando o custo do Arduino Uno e do Shield Ethernet se comparado ao custo de um CLP, a plataforma proposta pode ser utilizada por alunos interessados em automação industrial para a realização de projetos e/ou prática de programação em ambiente domiciliar. Favorece também a montagem de redes industriais Modbus com vários dispositivos escravos Modbus construídos com Arduino Uno e Shield Ethernet, o que amplia as possibilidades de estudo e de aprendizagem em automação industrial.

REFERÊNCIAS

- [1] M. C. Currás-Francos, J. Diz-Bugarín, J. R. García-Vila, A. Orte-Caballero, "Cooperative Development of an Arduino-Compatible Building Automation System for the Practical Teaching of Electronics", *IEEE Revista Iberoamericana de Tecnologías Del Aprendizaje*, vol. 9, no. 3, pp. 91-97, August 2014.
- [2] S. Lyubomirov, D. Shehova, K. Asparuhova, *Arduino-based Hardware Platform for the Realization of an Intelligent Home Through the Focus of Engineering Education*, in *Proc. X National Conference with International Participation - Electronica 2019*, Sofia, Bulgaria, May 16 - 17, 2019.
- [3] C. A. Petry, F. S. Pacheco, D. Lohmann, G. A. Correa, P. Moura, "Project Teaching beyond Physics: Integrating Arduino to the Laboratory", in *Latin American Robotics Symposium (LARS), Brazilian Symposium on Robotics (SBR) and Workshop on Robotics in Education (WRE)*, pp. 407-410, 2019.
- [4] T. L. de Souza, L. S. Elisiário, "Educational robotics teaching with Arduino and 3D print based on STEM projects", in *Latin American Robotics Symposium (LARS), Brazilian Symposium on Robotics (SBR) and Workshop on Robotics in Education (WRE)*, pp. 407-410, 2019.
- [5] W. Titon, A. R. G. Ramirez, "Teaching Programming Concepts Using Educational Robotics, Supported by the Arduino Platform: An Application in the Industrial Computer Learning Course", in *XIII Latin American Conference on Learning Technologies (LACLO)*, pp. 100-106, 2018.
- [6] Modbus *MODBUS Protocol Specification*. Acesso em 17 de Setembro de 2020, em: https://modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf.