



CENFIM



+EMPREGO

Automação

Manual de Formação



Ação financiada pela União Europeia.



Ação cofinanciada e gerida pelo Camões, I.P.

Ficha Técnica

Este manual foi produzido pelo CENFIM – Centro de Formação Profissional da Indústria Metalúrgica e Metalomecânica no âmbito do PROJETO MAIS EMPREGO EM PARCERIA PÚBLICO-PRIVADA PARA O EMPREGO DOS JOVENS EM CABO DELGADO (+EMPREGO) financiado pela União Europeia e implementado pelo Camões, I.P. Instituto da Cooperação.

Título: Automação_ Programação de PLC's

Autoria: Célia Santos

Índice

1	ENQUADRAMENTO/INTRODUÇÃO	5
2	AUTOMAÇÃO INDUSTRIAL	7
2.1	Introdução	7
2.2	Evolução histórica	7
2.3	Tipos de lógica	8
3	CONTROLADOR LÓGICO PROGRAMÁVEL – CPL	9
3.1	Tipologia de PLC's – E/S	9
3.2	Tipologia de PLC's – Arquitectura	10
4	DISPOSITIVOS DE ENTRADA E DE SAÍDA – E/S	12
4.1	Definição de Entradas e Saídas (E/S) ou (I/O's)	12
4.2	Dispositivos de Entrada – (Input's)	12
4.2.1	Sensores de deteção eletrónica	12
4.3	Dispositivos de saída – (Output's)	14
4.3.1	Electroválvulas e Relés	15
4.3.2	Motores	15
4.3.3	Sinalizadores	16
5	MEMÓRIA E DADOS	16
5.1	Definições de memória	16
5.2	Endereçamento de variáveis	17
5.3	Tipos de dados	17
5.3.1	Variável do tipo Inteira – “Integer”	18
5.3.2	Variável do tipo Real	18
5.3.3	Variável do tipo Caractere	18
5.3.4	Variável do tipo Cadeia de Caracteres	18

5.3.5	Variável do tipo “Booleana”	18
5.3.6	Quadro tipo de variáveis elementares	18
6	LINGUAGENS DE PROGRAMAÇÃO	19
6.1	Sequential Function Chart – SFC	19
6.1.1	Definição e origem	19
6.1.2	Definição de GRAFCET– norma aplicada	19
6.1.3	Simbologia e regras de GRAFCET	20
6.1.4	GRAFCET de nível 1	24
6.1.5	GRAFCET de nível 2	25
6.2	Diagrama de contactos – (FBD-functional block diagrams).....	26
6.2.1	Enquadramento à linguagem <i>Ladder</i>	26
6.2.2	Instruções de entrada e instruções de saída.....	26
6.2.3	Instruções do tipo – Temporizador	29
6.2.4	Instruções do tipo – Contador.....	30
6.3	Exemplos de GRAFCET.....	31
6.3.1	Exemplo 1 – Carimbo automático	31
6.3.2	Exemplo 2 – Carimbo automático, com duas marcações	32
6.4	Exemplos de Diagrama Ladder.....	33
6.4.1	Exemplo 1 – Carimbo automático.....	33
6.4.2	Exemplo 2 – Carimbo automático, com duas marcações	34
6.4.3	Exemplo 3 – Tabela ou lista de E/S e programa ladder em software TIA Portal.....	36
7	ELETRIFICAÇÃO DE PLC’s	38
7.1	Eletrificação PLC’s modulares	38
7.2	Eletrificação PLC’s compactos	42
8	COMUNICAÇÃO	46
9	ANEXOS	48

9.1	Exercícios de aplicação	48
9.1.1	Elaboração de lista de E/S – em formato papel	48
9.1.2	Elaboração de GRAFCET - em formato papel	48
9.1.3	Elaboração de Ladder em formato papel.....	48
9.1.4	Elaboração de programação Ladder em software TIA Portal	49
9.1.5	Exercício completo, em software TIA Portal – “Separador de embalagens”	49
9.1.6	Exercício completo, em software TIA Portal – “Máquina de café”	49
9.2	Ambiente TIA Portal – Siemens	49
10	FONTES BIBLIOGRÁFICAS	53
11	WEBGRAFIA	53
12	SOFTWARE.....	55
13	GLOSSÁRIO	55

1 ENQUADRAMENTO/INTRODUÇÃO

A União Europeia assinou com o Camões, I.P., em 15 de dezembro de 2020, um Acordo de Contribuição para gestão do projeto “+EMPREGO em parceria público-privada para os jovens de Cabo Delgado”, a qual tem como objetivo global aumentar as oportunidades económicas da população de Cabo Delgado, em particular da sua população jovem, contribuindo para a melhoria do acesso ao trabalho decente e do seu rendimento, em atividades direta ou indiretamente relacionadas com a indústria do gás natural. Pretende, também, estimular parcerias público-privadas com forte impacto no emprego e apoia o setor público da educação profissional e de emprego, as micro, pequenas e médias empresas, visando o aumento da respetiva competitividade e da empregabilidade dos jovens.

A parceria com instituições formadoras de Portugal que foi estabelecida no âmbito do projeto visa, em termos genéricos, assegurar um quadro de cooperação técnica que permita uma intervenção de excelência nos domínios da assistência técnica, formação de formadores multiplicadores e capacitação institucional. Esta cooperação no âmbito permitirá ainda a obtenção, por parte das entidades beneficiárias moçambicanas, de qualificações adequadas e de qualidade e a respetiva sustentabilidade, a promoção de relações de intercâmbio e de colaboração, a inserção em redes internacionais de saber e de investigação e a troca de experiências.

Com os objetivos referidos foi estabelecido um Acordo de Implementação entre o Camões Ip e o CENFIM, Centro que apresenta uma vasta experiência na área da cooperação para o desenvolvimento, nomeadamente em Angola e Moçambique, apresentando condições únicas em termos de equipamentos e recursos humanos através da articulação que estabelece com o Centro de Formação para a Metalomecânica de Maputo (CFPM).

O objetivo essencial do Acordo estabelecido com o Camões IP e o CENFIM é o de assegurar a formação técnica, em domínios especializados e particularmente relevantes para a indústria do gás natural, de formadores multiplicadores dos parceiros moçambicanos do +EMPREGO.

É neste contexto que surge este Manual de apoio ao Formando, da Ação de Formação Técnica de Formadores em Automação que foi desenvolvida ao abrigo do Acordo referido, com o objetivo de permitir aprofundar conhecimentos e competências, a adquirir na sala de aula e no laboratório de automação, pelos formadores que atuam no âmbito do projeto, melhorando o respetivo desempenho profissional e a sua função de multiplicadores de conhecimentos e de boas práticas formativas.

Ressalvamos que, o objetivo e a abrangência dos conhecimentos e competências constantes deste Manual de Formação, enquadrada nos objetivos da Ação de Formação referida anteriormente, não permite a formação de técnicos especialistas em automação industrial, na vertente, programação de PLC's.

O que se pretende atingir com este Manual é apresentar os vários intervenientes da área técnica a que chamamos Automação Industrial.

Neste âmbito, o formando encontrará neste manual de formação, informação sobre sensores, atuadores, autómatos industriais, comunicação industrial, esquemas elétricos de instalações industriais, e finalmente, programação de autómatos/PLC's.

O desenvolvimento da componente principal, abrangida pela ação de formação, e que é a programação de PLC's, terá uma componente teórica e outra prática. Neste manual, encontra-se a componente teórica, que importa explicar, como base e preparação para a realização da componente pratica, em sala de formação.

Em áreas de conhecimentos vastos, como é o caso da automação industrial, um formador de programação de PLC's, deve estar habilitado a programar, em suporte papel ou em suporte informático, via software.

Independentemente do suporte, papel ou software, deve ser transmitido pelo formador ao formando a complexidade que existe na elaboração de um sistema automatizado e programado. Os técnicos que executam a manutenção em sistemas automatizados podem não interagir com todo o processo industrial, e estar restritos apenas a uma parte, por esse motivo e para um melhor desempenho na sua profissão, o conhecimento sobre programação de PLC's, é uma ferramenta muito importante.

Por fim, deixamos um agradecimento especial à empresa Zembe - Distribuição e Soluções de Material Elétrico (www.zembe.pt) que, sendo parceiros do CENFIM, de imediato disponibilizou esquemas e imagens de quadros elétricos, autorizada também pelos seus clientes, tendo em consideração, o fim ultimo desta partilha, contribuir para um melhor resultado da formação ao abrigo do projecto +Emprego.

2 AUTOMAÇÃO INDUSTRIAL

2.1 Introdução

A evolução tecnológica verificada ao longo das últimas décadas, sobretudo a partir de 1970, está fortemente ligada à automação industrial. É a automação industrial a principal responsável pela rapidez de produção de bens, pelo acréscimo de qualidade dos bens produzidos e pela redução dos custos dessa produção. Ultimamente, a automação industrial está associada também aos serviços e não apenas à indústria transformadora.

Podemos associar a automação industrial também à prestação de serviços de manutenção, por exemplo, e também, à prestação de serviços e cuidados de saúde.

A automação pressupõe que tarefas realizadas por seres humanos, possam ser realizadas por equipamentos, máquinas ou dispositivos, de uma forma automática, previsível e autónoma. Evoluindo para um carácter programável, portanto, ajustável às necessidades, de uma forma quase momentânea.

2.2 Evolução histórica

Os primeiros sistemas a evoluir na sua flexibilidade foram os mecânicos e logo após os elétricos. Com a evolução destes dois sistemas, foi possível a combinação entre pneumática (mecânica) e a componente elétrica, resultando na electropneumática e na utilização de relés para impulsionar a automatização de processos industriais. Com a introdução de comutadores, o controlo de um sistema tornou-se menos “manual” e tornou-se mais “programável”.

Desenvolveram-se os relés, os contadores, os temporizadores, na sua vertente elétrica e eletrónica. Esta evolução fez surgir, na indústria automóvel, nos EUA, um primeiro equipamento denominado de PLC (programmable logic controller), no fim da década de 1960, pertencente à empresa General Motors (GM).

O impulso dado pela evolução da eletrónica permitiu que a lógica de relés (lógica cablada) fosse, progressivamente substituída pela lógica programada.

A lógica de relés funcionava, mas apresentava grandes desvantagens, em termos de tempo despendido em cada alteração do processo produtivo.

A aplicação da lógica programada, tendo em conta as suas inúmeras vantagens, forçava o desenvolvimento de tecnologia que suportasse condições de ambiente industrial, em termos de temperatura, pressão, poluição e vibrações, entre outras. Tendo havido uma evolução progressiva, acabou por ser desenvolvido o aspecto modular que ainda hoje se mantém. Sendo o PLC um equipamento programável, também foi sendo desenvolvida a linguagem de programação. A comunicação com o PLC foi também sendo desenvolvida, quer para aceder a dados, quer para a programação ser realizada via software.

Desde essa década de 1960 que a marca Allen-Bradley é uma referência na automação industrial, visto que os primeiros desenvolvimentos significativos foram da sua autoria e que desde esse momento que aposta nesta tecnologia.

2.3 Tipos de lógica

Lógica cablada – O equipamento funciona com base na utilização de cablagem elétrica, tubagem pneumática, etc. Os componentes e equipamentos estão interligados entre eles, o controlo e o funcionamento depende dessas interligações. A cablagem/tubagem ocupa muito espaço físico, permite a existência de erros humanos na execução das ligações, impedindo o funcionamento correto. Qualquer alteração ao funcionamento implica uma alteração física da cablagem/tubagem.

Lógica programada – O equipamento funciona com base numa programação. Todos os equipamentos estão ligados ao autómato (PLC). É o autómato que faz a gestão do funcionamento dos vários equipamentos. Qualquer alteração ao funcionamento de um processo industrial ou máquina, depende de uma alteração do programa e não da cablagem da máquina. Apresenta como vantagens uma maior fiabilidade, maior autonomia, maior manutibilidade.

3 CONTROLADOR LÓGICO PROGRAMÁVEL – CPL

Os CPL's ou PLC's, siglas que se aplicam na língua portuguesa ou na língua inglesa, são também comumente conhecidos, simplesmente como “autómatos”. Podem ser classificados por tipologia ou grupos, em função do número de entradas e de saída (E/S) que possuem, assim como, podem ser classificados na sua forma construtiva ou arquitetura. Os PLC's possuem um componente denominado por CPU (processador) que, em alguns casos é um microprocessador e noutros casos, é um processador com capacidade semelhante ao de um computador, sendo esta tipologia aplicada em indústrias de muito grande dimensão, com níveis de automação industrial extremamente elevadas.

3.1 Tipologia de PLC's – E/S

Os PLC's relativamente ao número de entradas e de saídas podem ser classificados como:

- a. Pequenos;



for the PCD1.M0160E0

Figura 1 – autómato SAIA – modelo PCD1

- b. Médios;



Figura 2 – autómato Omron – modelo CP1L

c. Grandes;



Figura 3- autômato Omron – modelo CP1H

Não deve ser esta divisão ser encarada como absoluta porque cada fabricante de PLC's tem as suas especificações.

3.2 Tipologia de PLC's – Arquitectura

Os PLC's são compostos por cinco elementos principais, nomeadamente: a unidade de processamento CPU, os elementos de entrada e de saída, a fonte de alimentação, os elementos periféricos e a memória.

A fonte de alimentação tem a função de fornecer energia a todos os outros elementos. Recebe energia externa e depois fornece-a aos restantes elementos.

A memória da CPU divide-se entre memória utilizada para a configuração interna do próprio PLC e memória reservada para os programas e restantes dados.

Os elementos periféricos são todas as portas utilizadas para comunicação, download de programas, conexão a HMI e/ou outros PLC's.

Os elementos de entrada e de saída são os locais do PLC onde se realizam as ligações físicas de todas as entradas e de todas as saídas do processo.

A unidade de processamento CPU é a componente hardware que faz a leitura de todas as variáveis de entradas, executa o programa e atualiza, posteriormente, todas as variáveis de saída.

Os PLC's relativamente à sua forma construtiva podem ser classificados como:

a. Compactos;

São equipamentos que concentram todos os componentes necessários à construção de um PLC, num volume único. Desta forma, são limitados na sua modularidade, podendo expandir-

se com limites. Concentram fonte de alimentação, memória, módulo E/S, CPU e periféricos num só volume, sendo exemplo disso o S7-1200 da marca Siemens.



Figura 4 – autómato Siemens – modelo SIMATIC S7-1200

b. Modulares;

São equipamentos cujos componentes são fabricados fisicamente separados. Os fabricantes constroem um suporte adequado, chamado de “bastidor” ou “rack”, permitindo a interligação entre os diferentes componentes e assegurando uma maior modularidade e expansividade dos mesmos, sendo exemplo disso o S7-1500 da marca Siemens.



Figura 5 – autómato Siemens – modelo SIMATIC S7-1500

4 DISPOSITIVOS DE ENTRADA E DE SAÍDA – E/S

4.1 Definição de Entradas e Saídas (E/S) ou (I/O's)

Nos processos industriais, existe sempre “informação” que é fornecida à CPU e “informação” dada pela CPU.

Ao conjunto de informação fornecida à CPU, sob a forma de variáveis de sinais elétricos ou, eletrónicos digitais ou analógicos, é atribuído o nome de “entradas” ou “inputs”, com as siglas “E” ou “I”.

Ao conjunto de informação dada pela CPU, sob a forma de variáveis de sinais elétricos ou, eletrónicos digitais ou analógicos, é atribuído o nome de “saídas” ou “outputs”, com as siglas “S” ou “O”.

Os elementos que estão conectados às variáveis de entrada são chamados de sensores. Os elementos que estão conectados às variáveis de saída são chamados de atuadores.

Estes dispositivos (I/O's) podem ser divididos em duas tipologias, relativamente ao sinal que enviam ou recebem:

Sinais **analógicos**, são sinais que podem assumir qualquer valor entre o valor mínimo e o máximo. Estes dispositivos são considerados mais como transdutores, do que como, sensores.

Como exemplo, indica-se um sensor que lê o nível de um líquido contido num depósito e que converte uma variação de nível de 0 a 5 metros para um sinal elétrico de 0 a 10V.

Sinais **digitais**, são sinais que só assumem dois valores, zero ou um, 0 ou 1. Indica-se como exemplo, um contacto (switch) cujo estado é aberto ou fechado. Também se pode considerar como exemplo, um sinalizador que está ligado ou desligado, sinalizando uma máquina ligada ou desligada.

4.2 Dispositivos de Entrada – (Input's)

Os dispositivos de entrada são muito diversos, mas todos se agrupam sob o tema da instrumentação. A instrumentação é a tecnologia que agrupa todos os instrumentos de medida, sejam sensores, transdutores ou equipamentos de medição e monitorização. A importância desta tecnologia é vital, no dia-a-dia de uma indústria, atualmente.

4.2.1 Sensores de deteção eletrónica

Apresentam-se os tipos de sensores mais comuns, no seu aspeto mais comum. Para mais informações técnicas devem ser consultados os fabricantes ou fornecedores, uma vez que os sensores têm características que têm de ser consideradas, no momento da sua escolha.

4.2.1.1 ReedSwitch

Este tipo de sensor é constituído por uma ampola no interior da qual estão os contactos. A ampola contém no seu interior um gás inerte ou encontra-se sobre vácuo com o objetivo de eliminar o arco elétrico que se cria quando se abre ou fecha os contactos. Pode ter vários aspetos construtivos, para ser utilizados em diferentes ambientes e equipamentos.



Figura 6 - ReedSwitch - tipo ampola

Na electropneumática podem ser utilizados como fins de curso em cilindros que tenham um íman no êmbolo. Não devem ser instalados em locais onde existam fortes campos magnéticos, uma vez que estes podem alterar o comportamento.

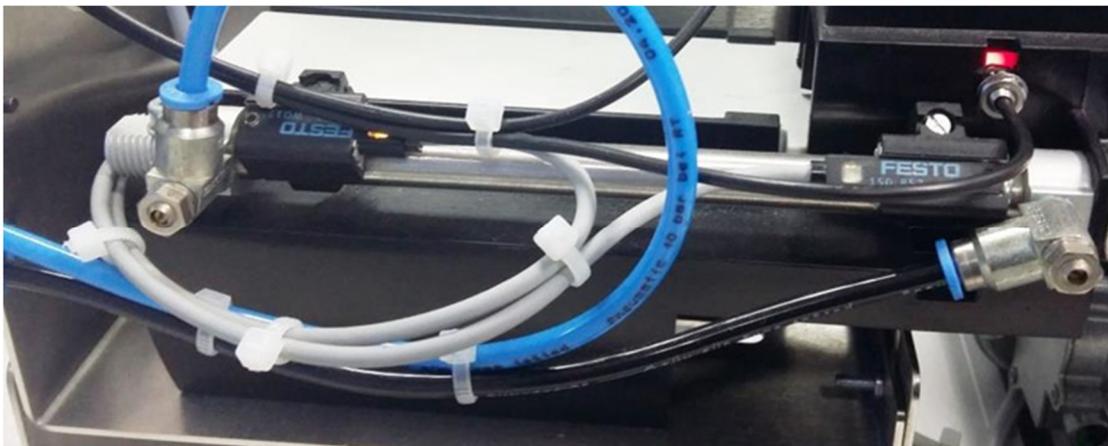


Figura 7 - Reedswitch aplicados como sensores fins de curso magnéticos

4.2.1.2 Indutivo

Este sensor tem como principais características:

- ✓ actuar na proximidade de metais,
- ✓ não existe nenhum movimento mecânico
- ✓ longa duração
- ✓ elevada velocidade de comutação,



Figura 8 - Sensor do tipo indutivo

4.2.1.3 Capacitivo

Os sensores capacitivos são semelhantes aos indutivos, mas detectam qualquer tipo de material: plástico, metal, madeira, entre outros.



Figura 9 - Sensor do tipo capacitivo

4.2.1.4 Ótico

Os sensores óticos funcionam segundo o princípio de emissão e recepção de luz, existe um emissor que é a fonte de luz, podem esta ser luz visível ou invisível (infravermelho), e existe um receptor que irá receber a luz enviada pelo emissor, o modo como a luz é encaminhada do emissor para o receptor é que distingue o tipo de sensor.

Podem ser classificados em sensores de barreira, com reflector e de reflexão directa.

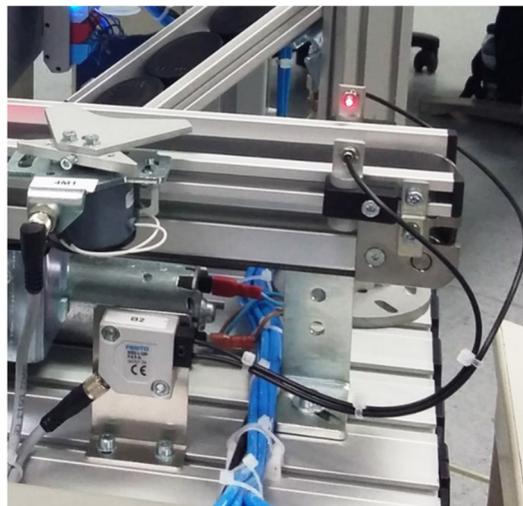


Figura 10 - Sensor ótico de barreira

4.3 Dispositivos de saída – (Output's)

Os dispositivos de saída são também chamados de atuadores. Estes equipamentos ou dispositivos atuam de diretamente no processo ou comandam esses.

4.3.1 Electroválvulas e Relés

Estes dois tipos de atuadores são sobretudo elétricos. Ambos funcionam com base numa bobine que alimentada cria um campo eletromagnético. Este efeito é muito comum em diversos componentes eletromecânicos, amplamente utilizados em instalações e processos industriais.

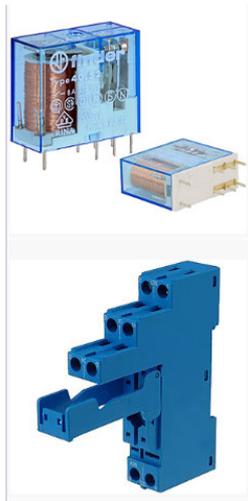


Figura 11 - Relé e Base



Figura 12 - Eletroválvula



Figura 13 - Eletroválvula

4.3.2 Motores

Os motores elétricos sejam de corrente alternada ou de corrente contínua, independentemente da sua construção e utilização são passíveis de serem controlados por um PLC. Será por vezes necessário recorrer a outros componentes elétricos como o caso dos contactores, para controlar o seu arranque e paragem. Noutras situações, se for necessário controlar a velocidade de rotação, recorreremos a variadores de frequência.



Figura 14 - Motor e Variador de frequência

4.3.3 Sinalizadores

Os dispositivos de sinalização são a forma, por excelência, de avisar os operadores e demais utilizadores de um local, de forma luminosa ou sonora. Em situações normais, situações de aviso e alarme, e por ultimo, situações de emergência e perigo iminente, são das formas mais rápidas de comunicar com o ser humano. São dispositivos que podem ser facilmente controlados de diversas formas.



Figura 15 - Sinalizadores luminosos



Figura 16 - Sinalizadores sonoros

5 MEMÓRIA E DADOS

A memória do PLC que está dedicada a todos os dados que compõem um programa, é considerada uma matriz de duas entradas, que permite organizar a informação (dados) sob a forma binária de 0 ou 1. Esta forma binária transforma toda a informação numa linguagem que o “microprocessador” ou “CPU” entende. De forma simples, podemos afirmar que o estado 0 (zero) corresponde à ausência de “carga” elétrica, enquanto que o estado 1 (um) significa a presença de “carga” elétrica”, nesse local da matriz.

A menor quantidade de dados (memória) que é possível armazenar é chamado de bit. A esse bit é associado um estado/valor, de 0 ou 1, conforme já mencionado.

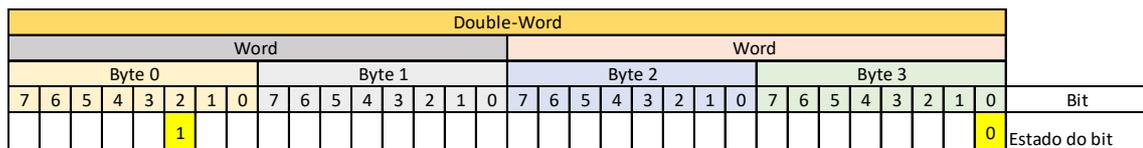
Os “bits” agrupam-se em quantidades, permitindo obter valores de “bytes”.

5.1 Definições de memória

Em programação de PLC, importa definir dentro da memória o bit, o “byte”, a “word” e a “double-word”.

Ao conjunto de 8 bits de informação definimos como um byte. Ao conjunto de 16 bits de informação definimos como uma “word” ou palavra. Como conjunto de 32 bits de informação, a transmitir ou a processar, teremos uma “double-word” ou palavra dupla.

De forma gráfica, podemos representar da seguinte forma:



Byte mais significativo

Byte menos significativo

Figura 17 - Quadro representativo da área da memória

5.2 Endereçamento de variáveis

Em programação de PLC, o endereçamento das variáveis são um aspeto de grande importância. A localização de uma variável, em determinado local da memória deve ser inequívoca para que a leitura e envio, no momento do processamento da mesma, decorra sem erros.

Utilizando uma comparação mais simples, podemos imaginar que o endereçamento de uma variável é semelhante à atribuição de um “código-postal”, que se coloca numa carta para ter a certeza que se chega ao destinatário.

Dessa forma, torna-se obrigatório chegar ao “número do bit”, onde a variável será guardada, ou ao “numero da word”, caso se trate de uma variável que necessite de maior quantidade de memória para ser guardada.

Na figura 17, conseguimos:

- Identificar o bit 3.0, está num estado 0, desativado. Pertence ao byte 3, número de bit 0.
- Identificar o bit 0.2, está no estado 1, ativo. Pertence ao byte 0, número do bit 2.

Importa ainda referir que a memória está dividida entre zonas, isto é, temos a zona atribuída às variáveis de entrada, a zona atribuída às variáveis de saída, a zona atribuída a memórias internas, entre outras. Assim, além do número do bit, temos de definir a zona de memória onde esse bit está alocado. Iremos então encontrar junto do número do bit, uma letra. Será a definição da zona de memória. Por exemplo: “I 3.0”, “O 0.2”, “M10.0” ou também “Q 8.7”

PLC tags									
	Name	Tag table	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	S0	Default tag table	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	FCs	Default tag table	Bool	%I0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	S1	Default tag table	Bool	%I0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	S2	Default tag table	Bool	%I0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	FCi	Default tag table	Bool	%I0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	s3	Default tag table	Bool	%I0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	s4	Default tag table	Bool	%I0.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	L_verde	Default tag table	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	L_vermelha	Default tag table	Bool	%Q0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	Motor_Tapete_av	Default tag table	Bool	%Q0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	EV1	Default tag table	Bool	%Q0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figura 18 - Declaração de variáveis em ambiente software

5.3 Tipos de dados

Os dados, informação ou variáveis referidos neste manual, são os necessários à programação de PLC's. Uma vez que os dados são referentes a um processo industrial, ou uma máquina que pretendemos controlar, programar, monitorizar, torna-se importante utilizar todas as variáveis disponíveis no mundo da programação informática.

A declaração das variáveis a utilizar numa programação PLC é imperativa. Ao fazer a declaração, sob a forma de uma lista de I/O's, é necessário saber identificar os diferentes tipos de variáveis que podemos utilizar.

As variáveis podem ser declaradas de forma local ou global, conforme se aplicam à totalidade do programa ou apenas a uma parte.

Relativamente aos atributos das variáveis, podem ser também consideradas retentivas, se for necessário que guardem o seu valor, em caso de falha de energia no sistema.

A declaração das variáveis e alguns parâmetros associados depende do tipo de variável a declarar. O modelo de autómato, a marca do autómato e também a linguagem de programação que estiver ao alcance do programador, em termos de operacionalização, obriga a ajustes no momento da programação. No entanto, de uma forma generalizada é possível identificar os seguintes tipos de variáveis:

5.3.1 Variável do tipo Inteira – “Integer”

É uma variável exclusivamente numérica, ou seja, unicamente números inteiros, sem o uso de casas decimais. Inclui números negativos.

5.3.2 Variável do tipo Real

É uma variável exclusivamente numérica, utilizada para declarar números com casas decimais. Inclui números negativos. Também permite a utilização de números exponenciais.

5.3.3 Variável do tipo Caractere

É uma variável utilizada para declarar caracteres diversos, ou seja, letras e símbolos. Ocupa 8 bits e aceita os caracteres definidos pelo código ASCII.

5.3.4 Variável do tipo Cadeia de Caracteres

É uma variável que representa uma cadeia ou conjunto de caracteres, entre o limite de 0 a 254 caracteres, com o número de bits definido pelo programador. Os caracteres aceitáveis estão definidos no código ASCII.

5.3.5 Variável do tipo “Booleana”

É uma variável do tipo lógica, que aceita apenas dois valores possíveis: Verdadeiro ou Falso. Em termos de espaço de memória, necessita apenas de um bit.

5.3.6 Quadro tipo de variáveis elementares

Na figura 19 constam informações sobre este tipo de variáveis elementares e podemos ver no quadro um resumo das variáveis anteriormente indicadas, e que se podem aplicar ao modelo S7-300 da marca Siemens.

Tipo de Dados	Tamanho	Descrição	Exemplo de notação
BOOL	1 (bit)	Bit	Falso/Verdadeiro
BYTE	8 (bits)	8 bit número hexadecimal	B#16#00, 16#00 B#16#FF, 16#FF
CHAR	8 (bits)	um caractere ASCII	A'
WORD	16 (bits)	16 bit número hexadecimal	W#16#0000, #16#0000 W#16#FFFF, 16#FFFF
		16 bit número binário	2#0000_0000_0000_0000 2#1111_1111_1111_1111
		Valor para contador	C#000
INT	16 (bits)	números inteiros, + / -	-32 768 / + 32 767
REAL	32 (bits)	números decimais, + / -	+ 23,12 ; -355,5896

Figura 19 - Tipo de variáveis - quadro resumo

6 LINGUAGENS DE PROGRAMAÇÃO

Devido a questões de mercado internacional e de normalização, ao longo da evolução da automação, deu-se o desenvolvimento, a revisão e atualização de normas internacionais relacionadas com esta área da industria. Atualmente, internacionalmente a norma IEC 611313 – Parte 3 regula as diversas linguagens de programação de PLC's.

As linguagens de programação de PLC's são diversas e baseiam-se em diferentes tipos de código. Para que o fabricante de PLC's, e de outros componentes relacionados com a automação industrial, possa produzir equipamento a instalar, programar e utilizar, em qualquer país, torna-se forçoso que siga as normas internacionais de fabrico e de programação.

6.1 Sequential Function Chart – SFC

6.1.1 Definição e origem

A norma que actualmente regulamenta e define o SFC é a IEC 611313 (Programmable Controllers – Part 3: Programming Languages), que se traduz em Autómatos Programáveis – Parte 3: Linguagens de Programação.

Interpretando a normal em vigor, verificamos que o SFC é uma linguagem que permite transformar programas complexos, com partes sequenciais e com partes paralelas, em blocos de programação menores e menos complexos.

6.1.2 Definição de GRAFCET– norma aplicada

Em sistemas que funcionam de forma sequencial, e até repetitiva, a metodologia GRAFCET aplica-se com grande sucesso, uma vez que recorre a etapas e transições, de forma alternada e ordenada. Esta linguagem de programação, hoje possível em diversas marcas e modelos de autómatos, surgiu pela primeira vez, na França, no final da década de 1970. Actualmente, é a norma IEC 60848:2013 regula a metodologia GRAFCET.

IEC 60848:2013 define as especificações da linguagem GRAFCET aplicáveis a sistemas que apresentam funcionamento sequencial. Especifica e define regras e símbolos, para a representação gráfica da linguagem e para a sua interpretação.

6.1.3 Simbologia e regras de GRAFCET

Existem três regras simples associadas ao GRAFCET e que são:

- a) A etapa inicial define o estado em que o funcionamento do sistema começa.
- b) Uma transição, para ser transposta, pressupõe a realização com sucesso das etapas anteriores. Pressupõe ainda que a expressão que define a transição seja ela própria satisfeita, com valor lógico 1.
- c) Assim que uma transição é satisfeita, a etapa anterior é desativada e a seguinte é ativada.

6.1.3.1 Definição de Etapa Inicial

A etapa inicial é definida por um quadrado de moldura dupla. Marca o estado inicial do programa. Existem três tipologias principais: etapa inicial sem retorno; etapa inicial com retorno; etapa inicial com activação forçada. Pode apresentar ou não, ações associadas.

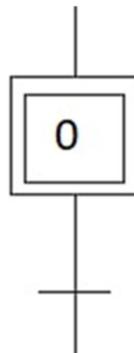


Figura 20 - Etapa Inicial

6.1.3.2 Definição de Etapa

A etapa define um estado do programa, associando-se imediatamente a um estado do sistema que está a ser programado. Uma etapa pode definir-se como activa, caso o sistema programado esteja nesse ponto. Quando se diz inactiva, o sistema está posicionado noutra etapa. Uma etapa é numerada, sendo a numeração definida pelo programador. Entre etapas existe uma transição.

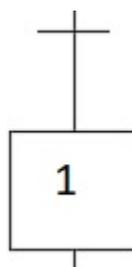


Figura 21 - Transição e Etapa 1

6.1.3.3 Definição de Transição

A transição é definida como a condição ou condições necessárias para transitar entre uma etapa e a etapa seguinte. Representa-se, em termos gráficos, por um traço horizontal, entre duas etapas. Sendo a transição, em termos práticos, uma expressão ou condição lógica, só quando essa expressão logica for verificada é que se transita de uma etapa para a próxima.

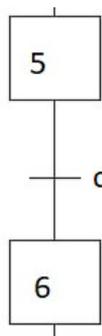


Figura 22 - Transição entre duas etapas, com condição “c” a satisfazer

6.1.3.4 Definição de Ações (Ordens)

As ações ou ordens são a manifestação física real, no sistema programado, quando uma etapa é ativada. Esta afirmação significa que as ações estão sempre associadas às etapas. Em termos gráficos, são representadas por um rectângulo colocado à direita da etapa. Cada etapa pode ter associadas várias ações ou ordens.

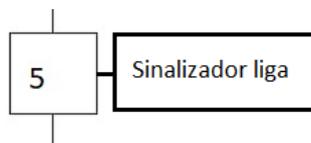


Figura 23 - Etapa 5 com ação associada

As ações ou ordens podem ser do tipo contínuas, do tipo condicionais, ou do tipo memorizadas.

6.1.3.5 GRAFCET Sequencial

Um GRAFCET sequencial entende-se por um GRAFCET que é composto por uma sequencia de etapas, no qual não existem derivações verticais nem horizontais.

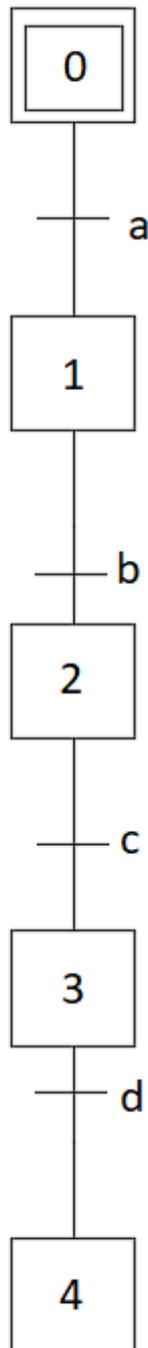


Figura 24 - GRAFCET Sequencial

6.1.3.6 GRAFCET com divergência

Num GRAFCET, a ocorrência de uma divergência é representada graficamente conforme a fig.14. Uma divergência é sinónimo de um momento, em programação e no funcionamento real, em que existe a necessidade de seguir um de dois caminhos possíveis, após a verificação da transição.

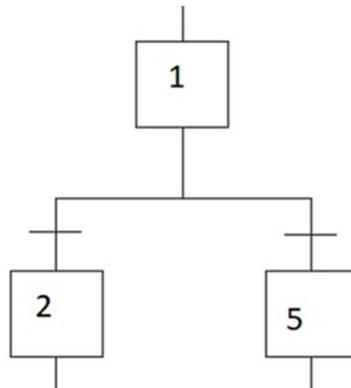


Figura 25 - GRAFCET com zona de divergência

6.1.3.7 GRAFCET com convergência

Num GRAFCET, a ocorrência de uma convergência é representada graficamente conforme a fig.15. Uma convergência é sinónimo de um momento, em programação e no funcionamento real, em que a necessidade de escolha entre caminhos termina, após a verificação das transições.

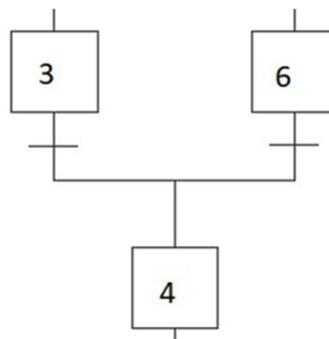


Figura 26 - GRAFCET com zona de convergência

6.1.3.8 GRAFCET com salto, com repetição de etapas e com retorno à etapa inicial

Neste exemplo, da fig. 16, encontramos entre a etapa #1 e a etapa #2, um salto entre etapas, após a verificação da transição “b”. Encontramos também, uma repetição de etapas, entre a etapa #2 e a etapa #3, sendo ser possível retornar à etapa #1, caso a transição “c” não seja verificada. Em relação ao retorno à etapa inicial #0, a mesma ocorre após a etapa #4 e a verificação da transição “z”, configurando uma situação de funcionamento cíclico.

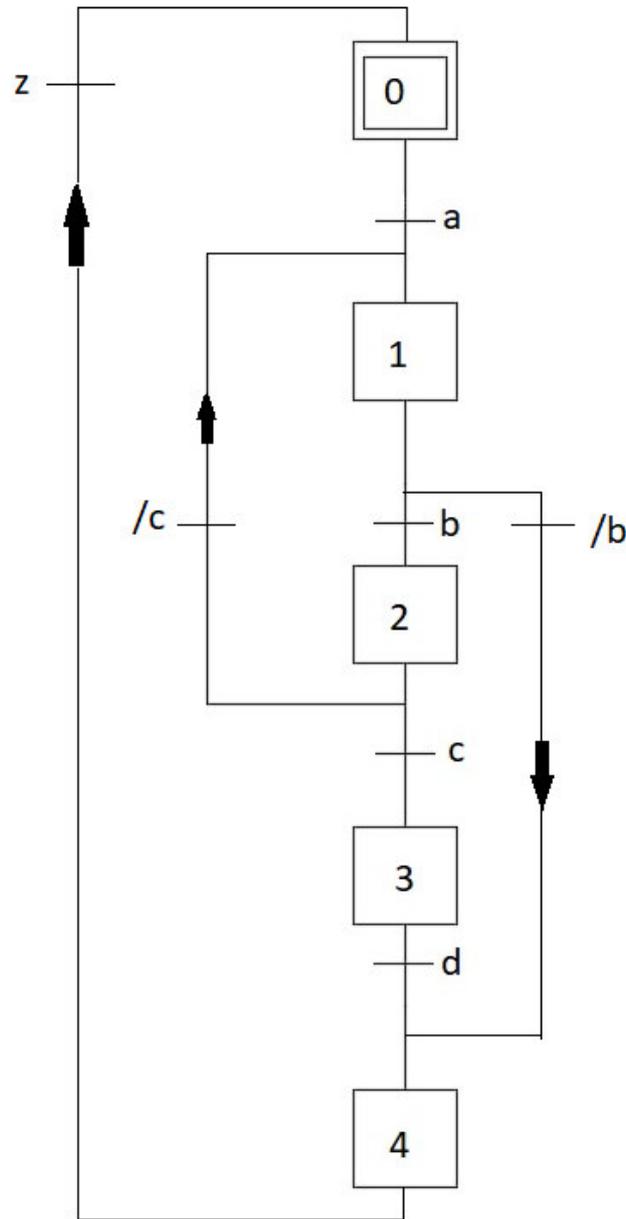


Figura 27 - GRAFCET com retorno, repetição e salto entre etapas

6.1.4 GRAFCET de nível 1

Quando visualizamos um GRAFCET de nível 1, encontramos uma descrição textual, do funcionamento real do equipamento, em termos de transições e de ações.



Figura 28 - GRAFCET nível 1

6.1.5 GRAFCET de nível 2

O GRAFCET de nível 2 tem as mesmas informações que o GRAFCET de nível 1, no entanto, não se visualiza uma descrição textual. No nível 2, encontramos nas transições e nas ações, uma identificação mais próxima do tipo de equipamento e da tecnologia real. Nomeadamente, a numeração dos sinalizadores atualmente, na industria segue a terminologia “H”. Neste caso, “H1” é o primeiro sinalizador (cuja côr não é significativa em programação). Cabe ao programador, utilizar uma linguagem que enquadre o programa dentro dos parâmetros aceites pela industria na qual será utilizada a máquina ou equipamento que está a programar.

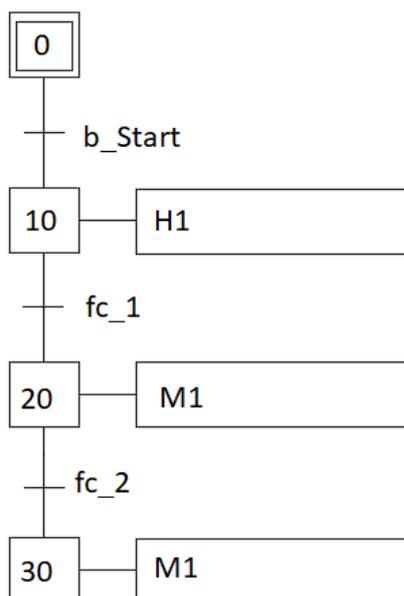


Figura 29 - GRAFCET de nível 2

6.2 Diagrama de contactos – (FBD-functional block diagrams)

6.2.1 Enquadramento à linguagem Ladder

A linguagem de diagrama de contactos - FBD é uma linguagem comumente utilizada em programação de PLC's, sendo uma das linguagens com mais antiguidade. Tornou-se muito presente e até, de alguma forma, preferida por parte da indústria, uma vez que, graficamente é muito semelhante a esquemas elétricos. Dentro do diagrama de contactos - FBD, o "Ladder" é sobejamente a mais conhecida e utilizada, sobretudo em programação de PLC's de "média" dimensão.

Sendo uma linguagem sobretudo gráfica, baseia-se em simbologia própria e regras de programação, que obedecem às normas IEC já mencionadas anteriormente.

6.2.2 Instruções de entrada e instruções de saída

6.2.2.1 Instrução - Contacto NA

O contacto normalmente aberto (NA) é uma instrução que está associada a um endereço na memória. Quando esse endereço está ativo, no estado "1", então verifica-se que a instrução NA é verdadeira. Em termos práticos, existindo um botão *start* composto fisicamente por um contacto NA, ligado a uma entrada do PLC, posteriormente, ao associarmos o endereço dessa entrada a esse botão *start*, quando o botão for premido pelo utilizador, fechando o contacto, fecha também em termos visuais a instrução contacto aberto NA, no programa.



Figura 30 - Instrução Contacto aberto - NA

6.2.2.2 Instrução - Contacto NF

O contacto normalmente fechado (NF ou NC) é uma instrução que está associada a um endereço na memória. Quando esse endereço está inativo, no estado "0", então verifica-se que a instrução NF é verdadeira.

Em termos práticos, existindo um botão *stop* composto fisicamente por um contacto NF, ligado a uma entrada do PLC, posteriormente, ao associarmos o endereço dessa entrada a esse botão *stop*, quando o botão for premido pelo utilizador, abrimos o contacto, e abre também em termos visuais a instrução contacto fechado NF, no programa. A instrução terá então valor *false*.

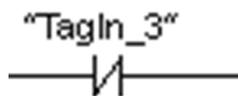


Figura 31 - Instrução Contacto fechado - NF

6.2.2.3 Instrução E – “And”

A instrução E ou “and” representa graficamente, em linguagem de programação, uma associação em série de duas ou mais instruções. O resultado dessa associação será *verdadeiro* quando todas as instruções que estão em série forem, também elas, *verdadeiras*.

No caso da figura 32, é uma associação série de duas instruções do tipo NA, quando ambas forem *verdadeiras*, o resultado da associação é também *verdadeiro*.



Figura 32 - Instrução E (And)

6.2.2.4 Instrução OU – “OR”

A instrução OU ou “or” representa graficamente, conforme figura 33, em linguagem de programação, uma associação em paralelo de duas ou mais instruções. O resultado dessa associação será *verdadeiro* quando uma das instruções que estão em paralelo forem, também elas, *verdadeiras*.



Figura 33 - Instrução Ou (Or)

6.2.2.5 Instrução de saída – tipo contacto – (coil)

Uma instrução de saída do tipo *coil* (figura 34) inscreve, na memória associada a essa mesma saída, o resultado logico da programação realizada com determinadas instruções de entrada. Se o resultado for *verdadeiro*, a instrução de saída é *activada*. Quando o resultado for *falso*, a instrução de saída é *desactivada*.

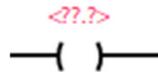


Figura 34 - Instrução saída - coil

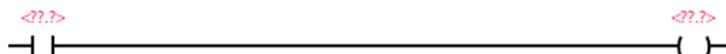


Figura 35 - Linha de programação ladder com instrução de entrada e instrução de saída

6.2.2.6 Instrução de saída – tipo Set – (S)

Esta instrução permite ao programador activar uma saída, colocando o seu “bit” de memória sempre no estado *ON*, até que, por efeito da programação o mesmo seja colocado em *OFF* e a saída correspondente seja desactivada.

Utilizando a instrução *Set*, a saída mantém-se activa sempre, caso não seja posteriormente desactivada.

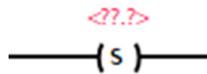


Figura 36 - Instrução de saída tipo Set

6.2.2.7 Instrução de saída – tipo Reset – (R)

Esta instrução permite ao programador desactivar uma saída, colocando o seu “bit” de memória sempre no estado *OFF*, até que, por efeito da programação o mesmo seja colocado em *ON* e a saída correspondente fique activa.

Utilizando a instrução *Reset*, a saída mantém-se desactivada sempre, caso não seja posteriormente activada. Importa ressaltar que em caso de conflito, de programação, a instrução *Reset* sobrepõe-se à instrução *Set*.

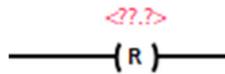


Figura 37 - Instrução de saída tipo Reset

6.2.2.8 Instrução de saída – tipo Set/Reset – (SR)

A instrução "Set/reset" - "flip-flop" activa ou desactiva determinado “bit” de saída, dependendo do estado das duas condições de entrada, conectadas aos “inputs” “S” e “R1”.

Se a condição de entrada em “S” for verdadeira, com valor “1”, e a condição de entrada em “R1” for falsa, com valor “0”, o resultado da instrução verificado em “Q” será “1”.

Caso ambas as condições, em “S” e em “R1”, estiverem com valor “1”, então em “Q” o resultado será “0”, porque a instrução *Reset* sobrepõe-se à instrução *Set*.

Com ambas as condições em valor “0”, não acontece nenhuma alteração de estado em “Q”.

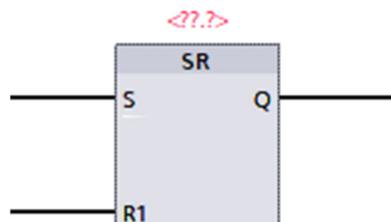


Figura 38 - Instrução de saída Set/Reset - SR

6.2.2.9 Instrução de saída – tipo Reset/Set – (RS)

A instrução "Reset/Set" - "flip-flop" activa ou desactiva determinado "bit" de saída, dependendo do estado das duas condições de entrada, conectadas aos "inputs" "S" e "R1".

Se a condição de entrada em "R1" for verdadeira, com valor "1", e a condição de entrada em "S1" for falsa, com valor "0", o resultado da instrução verificado em "Q" será "1".

Caso ambas as condições, em "S1" e em "R", estiverem com valor "1", então em "Q" o resultado será "1", porque a instrução Set sobrepõe-se à instrução Reset.

Com ambas as condições em valor "0", não acontece nenhuma alteração de estado em "Q".

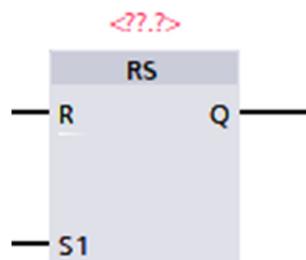


Figura 39 - Instrução de saída Reset/Set - RS

6.2.3 Instruções do tipo – Temporizador

A instrução "temporizador" apresenta, em termos de software, muitas variantes, porém, podemos reduzir a duas opções mais utilizadas de forma geral. Estas duas opções são bastante semelhantes a dois dispositivos, bem conhecidos, em termos de eletromecânicos, o temporizador "ON-Delay" e o temporizador "OFF-Delay".

A instrução "temporizador", em programação de PLC, necessita da seguinte parametrização em software. De notar que o parâmetro "ET", pode ser facultativo.

	Parâmetro	Tipo de dados	Área da memória	Descrição
entrada	IN	BOOL	I, Q, M	entrada que dá início - Start input
entrada	PT	Tempo	I, Q, M	duração do temporizador
saída	Q	BOOL	I, Q, M	saída que será activada após a duração definida em PT
saída	ET	Tempo	I, Q, M	valor de tempo a decorrer

Figura 40 - Parametrização de temporizador - em software TIA Portal para Siemens SIMATIC S7-300

Funcionamento "ON-Delay" - Quando o valor da instrução que está conectada ao parâmetro "IN" for "verdadeira", a instrução inicia a contagem do tempo que foi definido no parâmetro "PT". Terminado esse tempo, o parâmetro "Q" fica ativado. Enquanto o tempo decorre, o parâmetro "Q" apresenta valor "0", está desactivado.

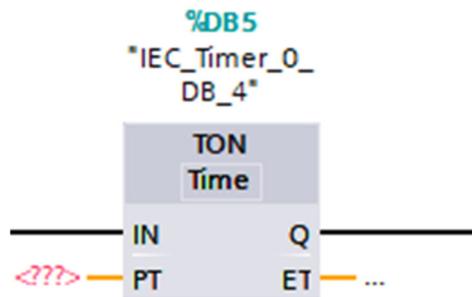


Figura 41 - Grafismo de TON - software

Funcionamento “OFF-Delay” - Quando o valor da instrução que está conectada ao parâmetro “IN” for “verdadeira”, a instrução inicia a contagem do tempo que foi definido no parâmetro “PT”. Terminado esse tempo, o parâmetro “Q” é desactivado. Enquanto o tempo decorre, o parâmetro “Q” apresenta valor “1”, está activo.

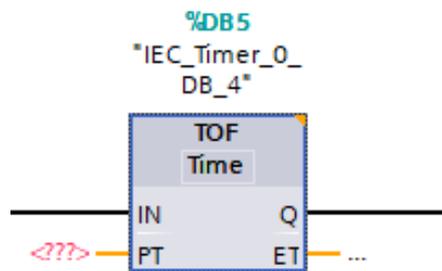


Figura 42 - Grafismo de TOFF - software

6.2.4 Instruções do tipo – Contador

A instrução “contador” apresenta, em termos de software, algumas variantes, porém, podemos reduzir a duas opções mais utilizadas de forma geral, o contador “UP” e o contador “DOWN”.

Funcionamento “contador-UP” – Quando o sinal da entrada conectada em “CU” muda de “0” para “1”, o valor do parâmetro em “CV” é incrementado numa unidade. Repete-se esta incrementação até que seja atingido o valor pré-definido em “PV”. Quando “CV” ≥ “PV”, obtemos no parâmetro “Q” o valor “1”. O valor de “CV” é reposto a zero, sempre que o parâmetro tiver valor “1”.

	Parâmetro	Tipo de dados	Área da memória	Descrição
entrada	CU/CD	BOOL	I, Q, M	entrada que dá início - Start input
entrada	R	BOOL	I, Q, M	entrada que desliga - Reset input
entrada	PV	INT	I, Q, M	valor que activa o output Q
saída	Q	BOOL	I, Q, M	estado do contador
saída	CV	INT	I, Q, M	valor do contador

Figura 43 -Parametrização de contador UP - em software TIA Portal para Siemens SIMATIC S7-300

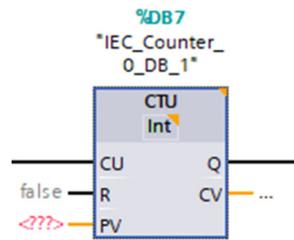


Figura 44 - Grafismo - Contador Up

Funcionamento “**contador-Down**” – Quando o sinal da entrada conectada em “CD” muda de “0” para “1”, o valor do parâmetro em “CV” é decrementado numa unidade. Repete-se esta ação até que seja atingido o valor pré-definido em “PV”. Quando “CV” ≤ 0(zero), obtemos no parâmetro “Q” o valor “1”. O valor de “CV” é reposto para o mesmo valor de “PV”, sempre que o parâmetro LD tiver valor “1”.

	Parâmetro	Tipo de dados	Área da memória	Descrição
entrada	CD	BOOL	I, Q, M	entrada que dá início - Start input
entrada	LD	BOOL	I, Q, M	entrada LOAD
entrada	PV	INT	I, Q, M	valor que activa o output Q, se LD=1
saída	Q	BOOL	I, Q, M	estado do contador
saída	CV	INT	I, Q, M	valor do contador

Figura 45 - Parametrização de contador Down - em software TIA Portal para Siemens SIMATIC S7-300

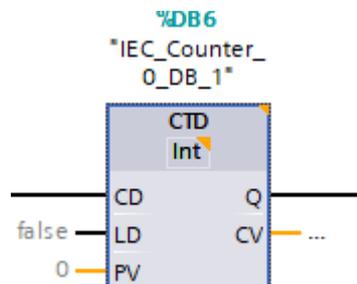


Figura 46 - Grafismo - Contador Down

6.3 Exemplos de GRAFCET

6.3.1 Exemplo 1 – Carimbo automático

Princípio de funcionamento – Um equipamento carimba, de forma automática caixas de cartão. O técnico carrega no botão S1 e liga-se um sinalizador H1. Se o cilindro do carimbo estiver recuado, o cilindro avança para carimbar. Quando o cilindro estiver avançado, a caixa já foi carimbada. Depois da caixa estar carimbada, o cilindro recua e desliga o sinalizador H1. Quando for necessário carimbar nova caixa, o técnico terá de voltar a carregar no botão S1.

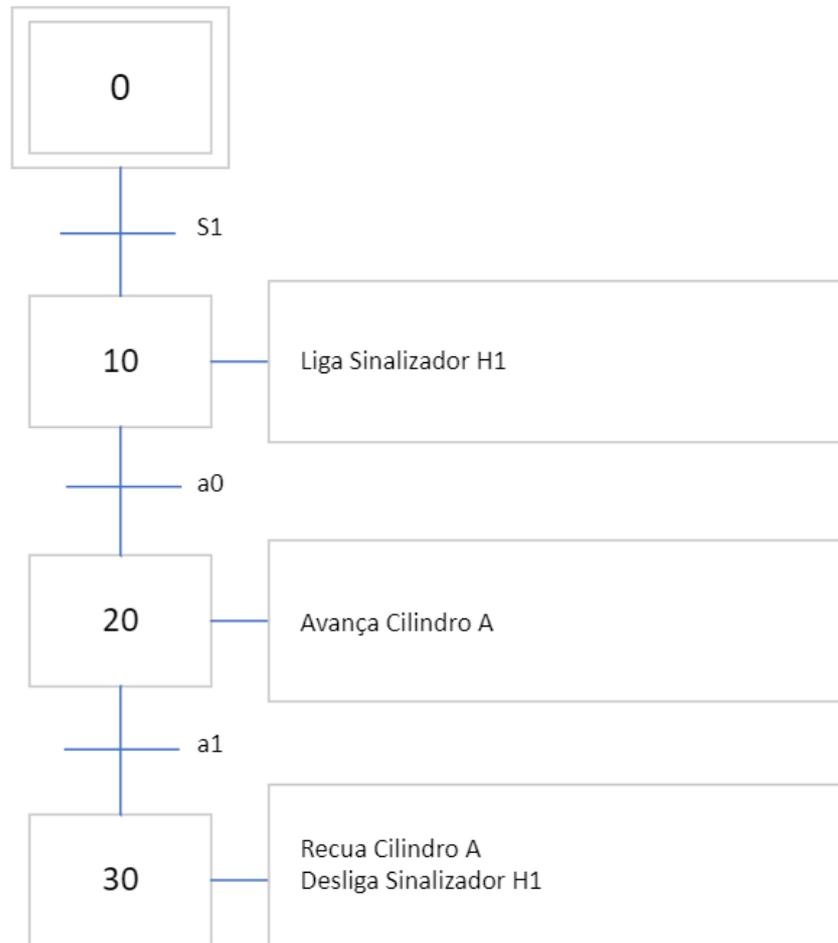


Figura 47 - GRAFCET EXE_1

6.3.2 Exemplo 2 – Carimbo automático, com duas marcações

Princípio de funcionamento – Um equipamento carimba, de forma automática, caixas de cartão, com duas marcações independentes, realizadas pelo cilindro A e pelo cilindro B. O técnico carrega no botão S1 e liga-se um sinalizador H1. O cilindro A que está recuado, avança e recua, realizando a primeira marcação. Feita essa primeira marcação, o sinalizador H1 desliga-se. Nesse momento, o cilindro B avança e liga-se o sinalizador H2. Quando o cilindro B estiver avançado, a caixa já foi carimbada com a segunda marcação. Nesse momento, o equipamento regressa ao seu estado inicial, com ambos os cilindros recuados e ambos os sinalizadores desligados. Quando for necessário carimbar nova caixa, o técnico terá de voltar a carregar no botão S1.

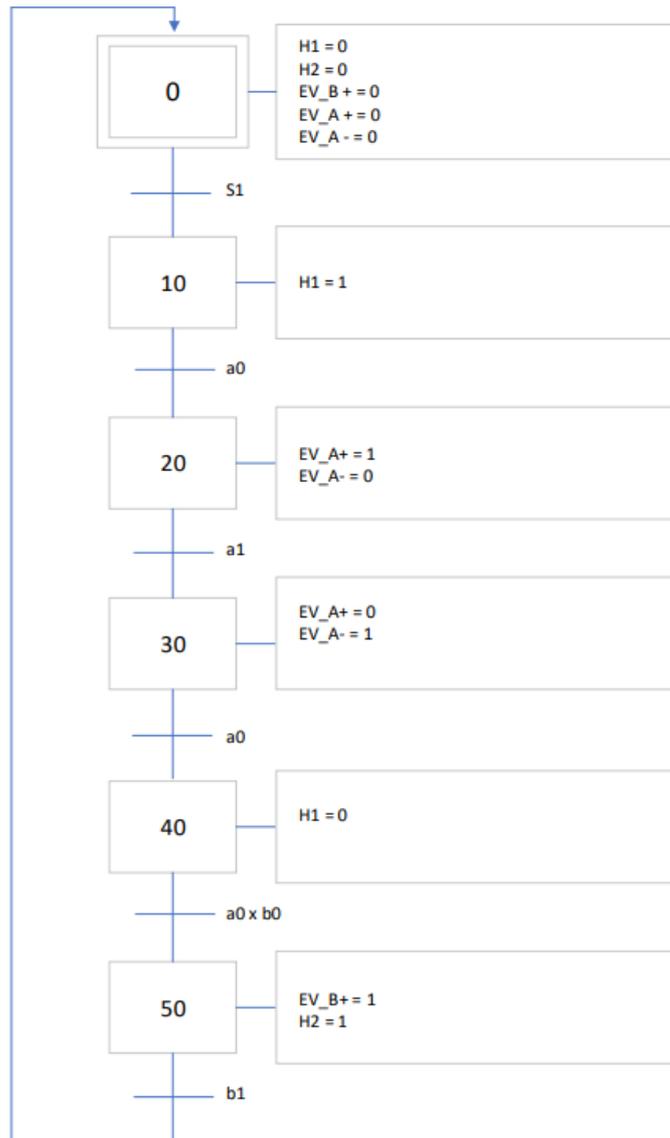


Figura 48 - GRAFCET EXE_2

6.4 Exemplos de Diagrama Ladder

6.4.1 Exemplo 1 – Carimbo automático

Princípio de funcionamento – Um equipamento carimba, de forma automática caixas de cartão. O técnico carrega no botão S1 e liga-se um sinalizador H1. Se o cilindro do carimbo estiver recuado, o cilindro avança para carimbar. Quando o cilindro estiver avançado, a caixa já foi carimbada. Depois da caixa estar carimbada, o cilindro recua e desliga o sinalizador H1. Quando for necessário carimbar nova caixa, o técnico terá de voltar a carregar no botão S1.

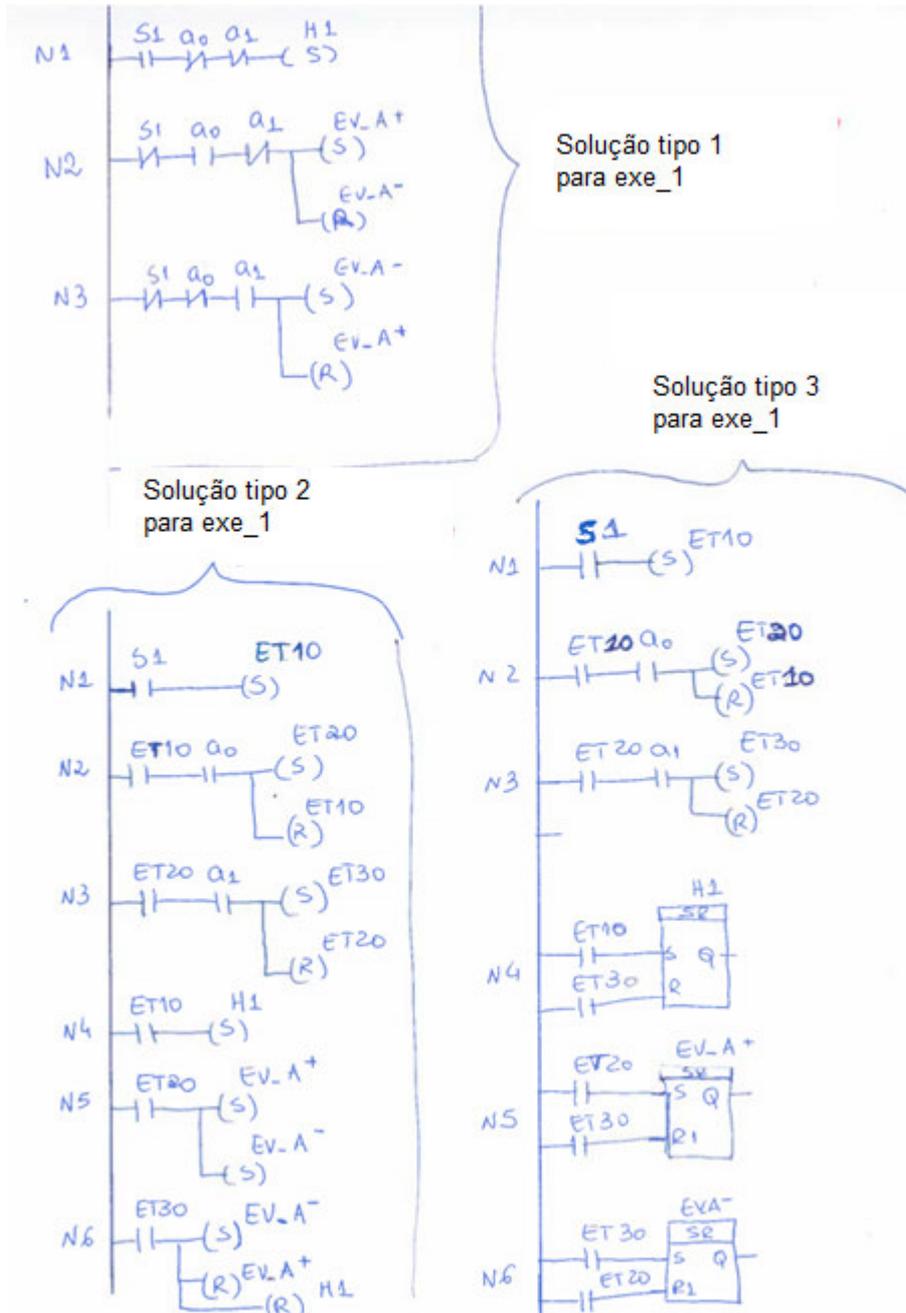


Figura 49 - Soluções EXE_1 Ladder

6.4.2 Exemplo 2 – Carimbo automático, com duas marcações

Princípio de funcionamento – Um equipamento carimba, de forma automática, caixas de cartão, com duas marcações independentes, realizadas pelo cilindro A e pelo cilindro B. O técnico carrega no botão S1 e liga-se um sinalizador H1. O cilindro A que está recuado, avança e recua, realizando a primeira marcação. Feita essa primeira marcação, o sinalizador H1 desliga-se. Nesse momento, o cilindro B

avança e liga-se o sinalizador H2. Quando o cilindro B estiver avançado, a caixa já foi carimbada com a segunda marcação.

Nesse momento, o equipamento regressa ao seu estado inicial, com ambos os cilindros recuados e ambos os sinalizadores desligados. Quando for necessário carimbar nova caixa, o técnico terá de voltar a carregar no botão S1.

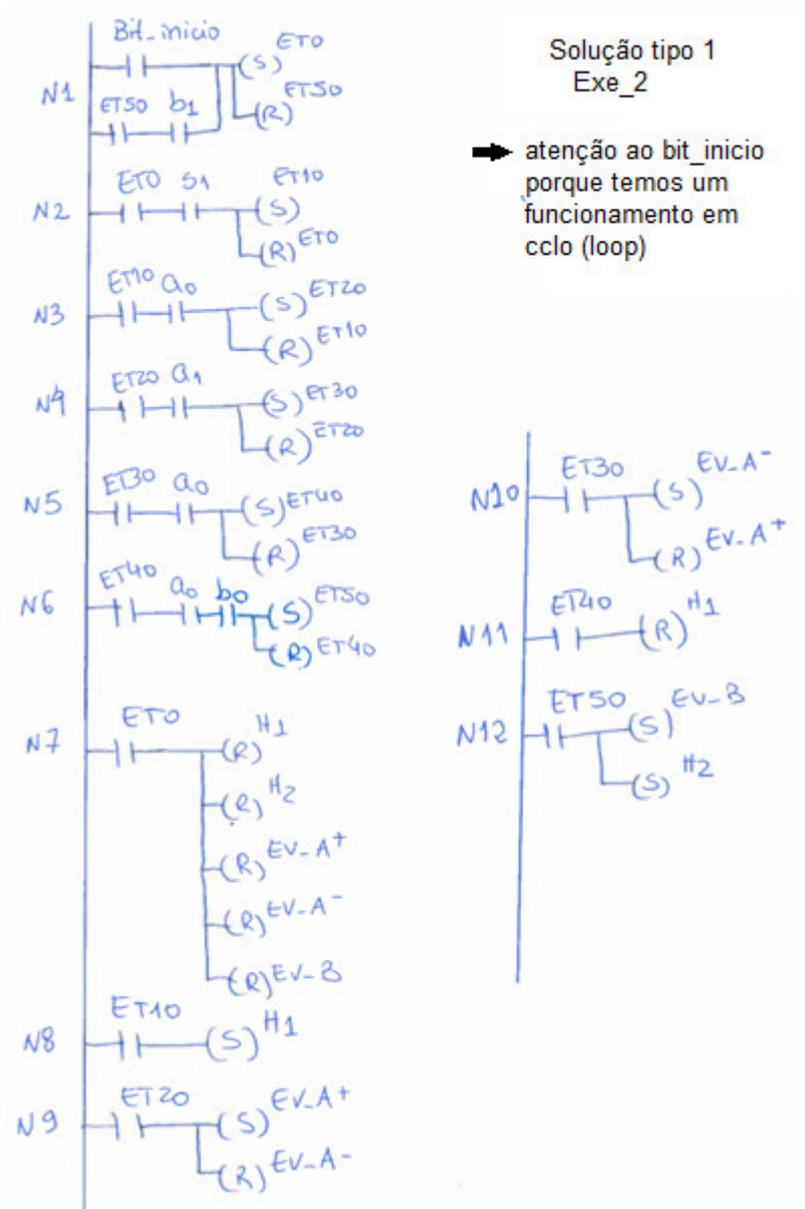
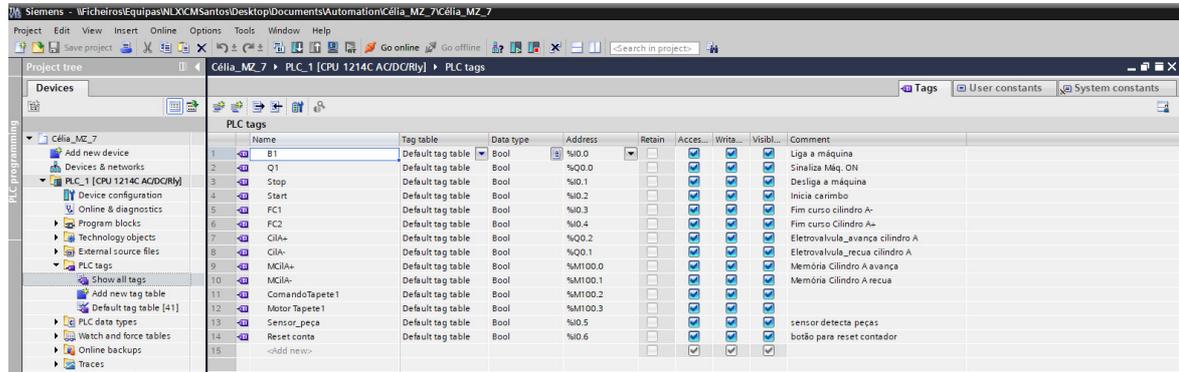


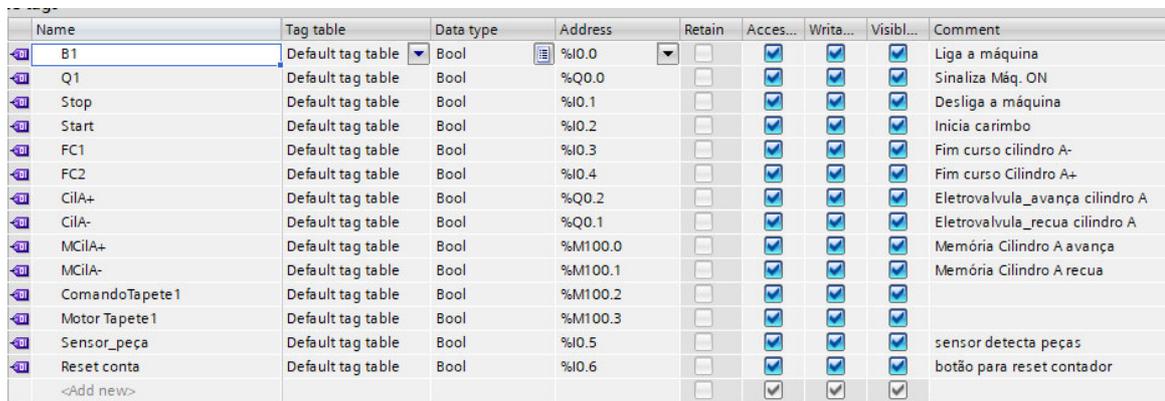
Figura 50 - Solução EXE_2

6.4.3 Exemplo 3 – Tabela ou lista de E/S e programa ladder em software TIA Portal



Name	Tag table	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
B1	Default tag table	Bool	%I0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Liga a máquina
Q1	Default tag table	Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Sinaliza Mq. ON
Stop	Default tag table	Bool	%I0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Desliga a máquina
Start	Default tag table	Bool	%I0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Inicia carimbo
FC1	Default tag table	Bool	%I0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Fim curso cilindro A-
FC2	Default tag table	Bool	%I0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Fim curso Cilindro A+
CilA+	Default tag table	Bool	%Q0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Eletravalvula_avanca cilindro A
CilA-	Default tag table	Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Eletravalvula_recura cilindro A
MCilA+	Default tag table	Bool	%M100.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Memória Cilindro A avança
MCilA-	Default tag table	Bool	%M100.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Memória Cilindro A recua
ComandoTapete1	Default tag table	Bool	%M100.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Motor Tapete1	Default tag table	Bool	%M100.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Sensor_peça	Default tag table	Bool	%I0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor detecta peças
Reset conta	Default tag table	Bool	%I0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	botão para reset contador
<Add new>					<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figura 51 - Tabela E/S - TIA Portal



Name	Tag table	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
B1	Default tag table	Bool	%I0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Liga a máquina
Q1	Default tag table	Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Sinaliza Mq. ON
Stop	Default tag table	Bool	%I0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Desliga a máquina
Start	Default tag table	Bool	%I0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Inicia carimbo
FC1	Default tag table	Bool	%I0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Fim curso cilindro A-
FC2	Default tag table	Bool	%I0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Fim curso Cilindro A+
CilA+	Default tag table	Bool	%Q0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Eletravalvula_avanca cilindro A
CilA-	Default tag table	Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Eletravalvula_recura cilindro A
MCilA+	Default tag table	Bool	%M100.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Memória Cilindro A avança
MCilA-	Default tag table	Bool	%M100.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Memória Cilindro A recua
ComandoTapete1	Default tag table	Bool	%M100.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Motor Tapete1	Default tag table	Bool	%M100.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Sensor_peça	Default tag table	Bool	%I0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor detecta peças
Reset conta	Default tag table	Bool	%I0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	botão para reset contador
<Add new>					<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figura 52 - Pormenor tabela E/S - TIA Portal

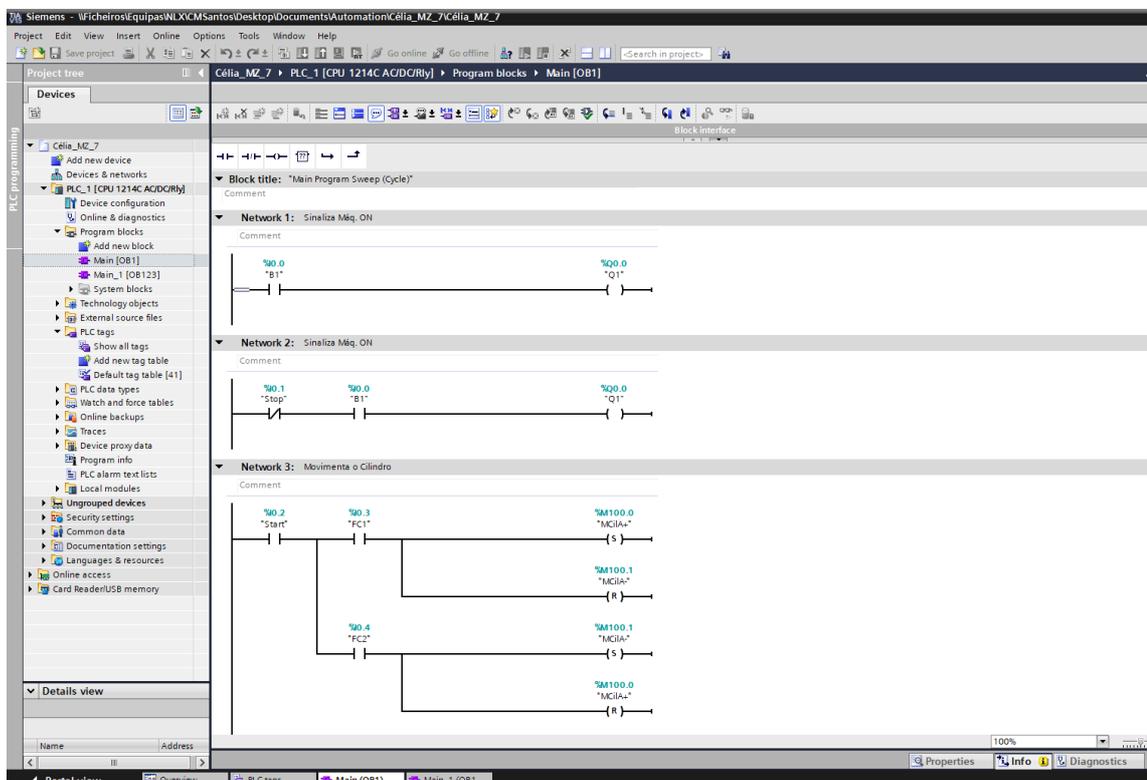
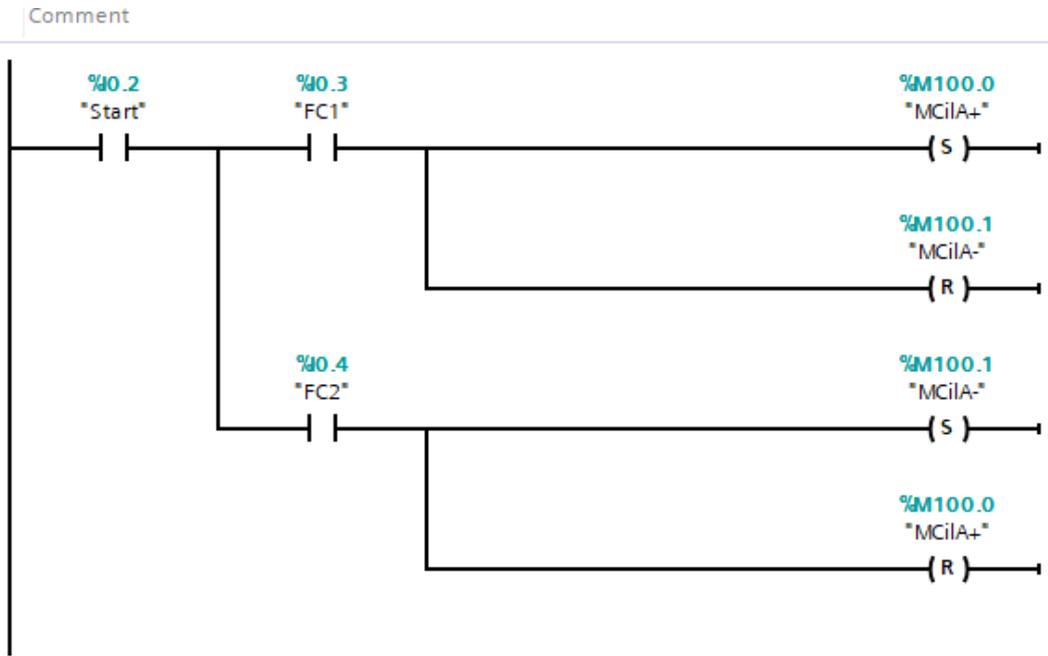


Figura 53 - Aspecto geral de um programa ladder, em software TIA Portal



Network 4:

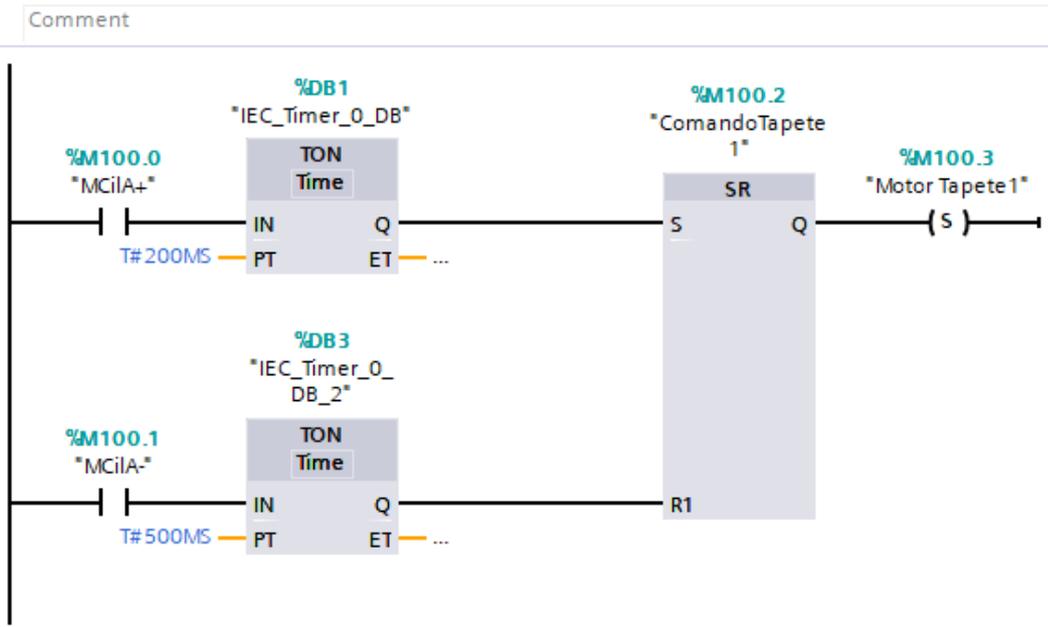


Figura 54 - Pormenor de ladder, em TIA Portal, com temporizadores

7 ELETRIFICAÇÃO DE PLC'S

A eletrificação de um quadro elétrico de automação implica também a alimentação de todos os componentes que estão fisicamente ligados ao PLC. Isto quer dizer que, se for um PLC modular, podem haver vários componentes físicos que necessitam de alimentação elétrica.

As ligações estão intrinsecamente relacionadas com o próprio *hardware*, ou seja, devem sempre ser consultados os manuais e esquemas de ligação do fabricante, caso não se disponha de qualquer esquema elétrico.

Apresentam-se em seguida várias imagens e esquemas de ligações de alguns equipamentos da marca Siemens.



Figura 55 - Imagem Siemens S7-300



Figura 56 - Imagem Siemens S7-1200

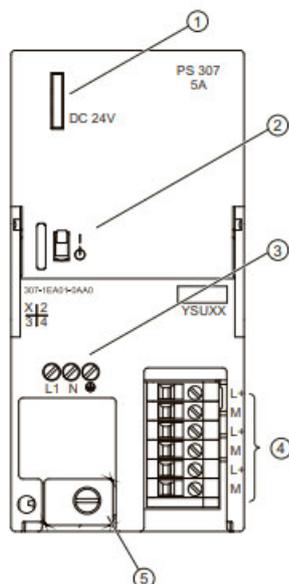
7.1 Eletrificação PLC's modulares

No caso dos PLC modulares, nesta situação particular, iremos recorrer ao S7-300. Para o podermos eletrificar será necessária uma fonte de alimentação, uma CPU, módulos de entradas e de saídas.

Em seguida, teremos os esquemas do fabricante para a eletrificação destes componentes. A língua inglesa, em automação é de certa forma, um dado adquirido, porque existe pouca informação dos fabricantes em língua portuguesa.



Figura 57 – Imagem de fonte de alimentação Siemens PS_307

Wiring diagram of PS 307; 5 A


Tradução:

1 – LED/sinalização para uma saída efetiva de 24 V DC

2 – Interruptor da fonte de alimentação

3 – Terminais de alimentação da fonte – CA monofásica

4 – Terminais de saída da fonte em CC, 24V

5 – Parafuso fixação

O fabricante informa que a fonte deve estar previamente protegida com um disjuntor 6A 230V AC, curva característica tipo C

- ① Display for "Output voltage DC 24 V present"
- ② 24 V DC On/Off switch
- ③ Mains and protective conductor terminals
- ④ Terminals for 24 V DC output voltage
- ⑤ Strain-relief

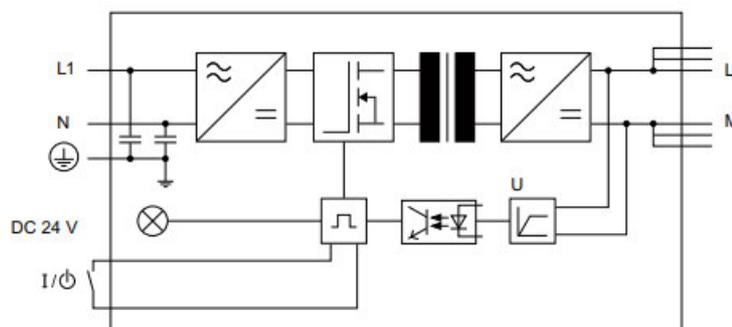
Block diagram PS 307; 5 A


Figure 2-2 Block diagram of the PS 307; 5 A power supply module

Line protection

The mains supplies of the PS 307; 5 A power supply module should be protected with a miniature circuit-breaker (for example Siemens 5SN1 series) of the following rating:

- Rated current at 230 V AC: 6 A
- Tripping characteristics (type): C.

Figura 59 - Esquema de ligações PS_307 5A

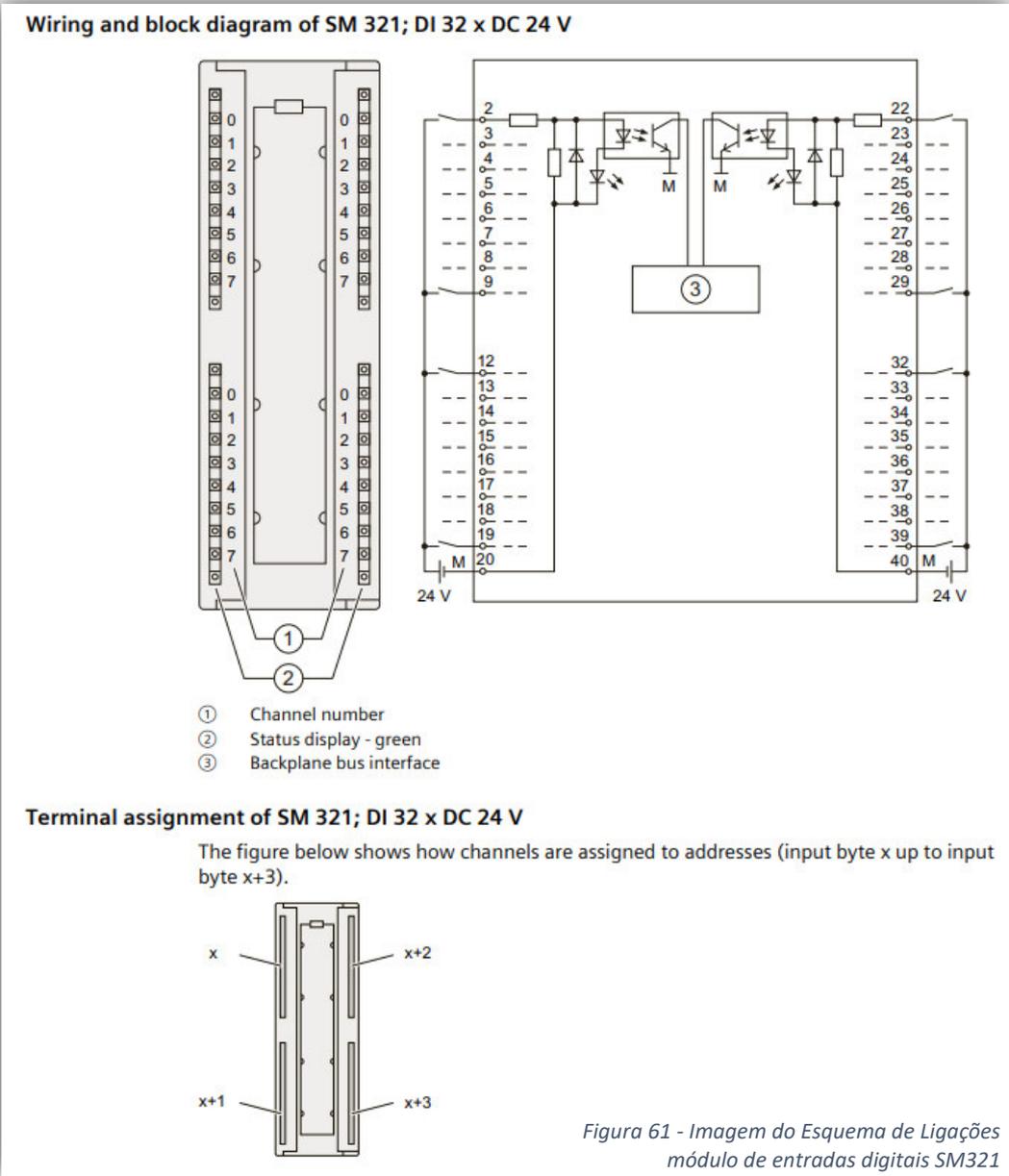
Módulo de Entradas Digitais

Os módulos que serão posteriormente anexados à CPU, necessitam também de alimentação elétrica.

Em qualquer situação, entre as saídas 24V DC ou CC da fonte de alimentação, e a alimentação elétrica de módulos, sejam entradas ou saídas, devem existir régua de bornes, com separadores e vários bornes interligados com “shunts” próprios. Desta forma, a organização da cablagem e a substituição de componentes, assim como, a deteção de falhas e avarias relacionadas com a alimentação elétrica do circuito, torna-se mais eficaz e rápida.



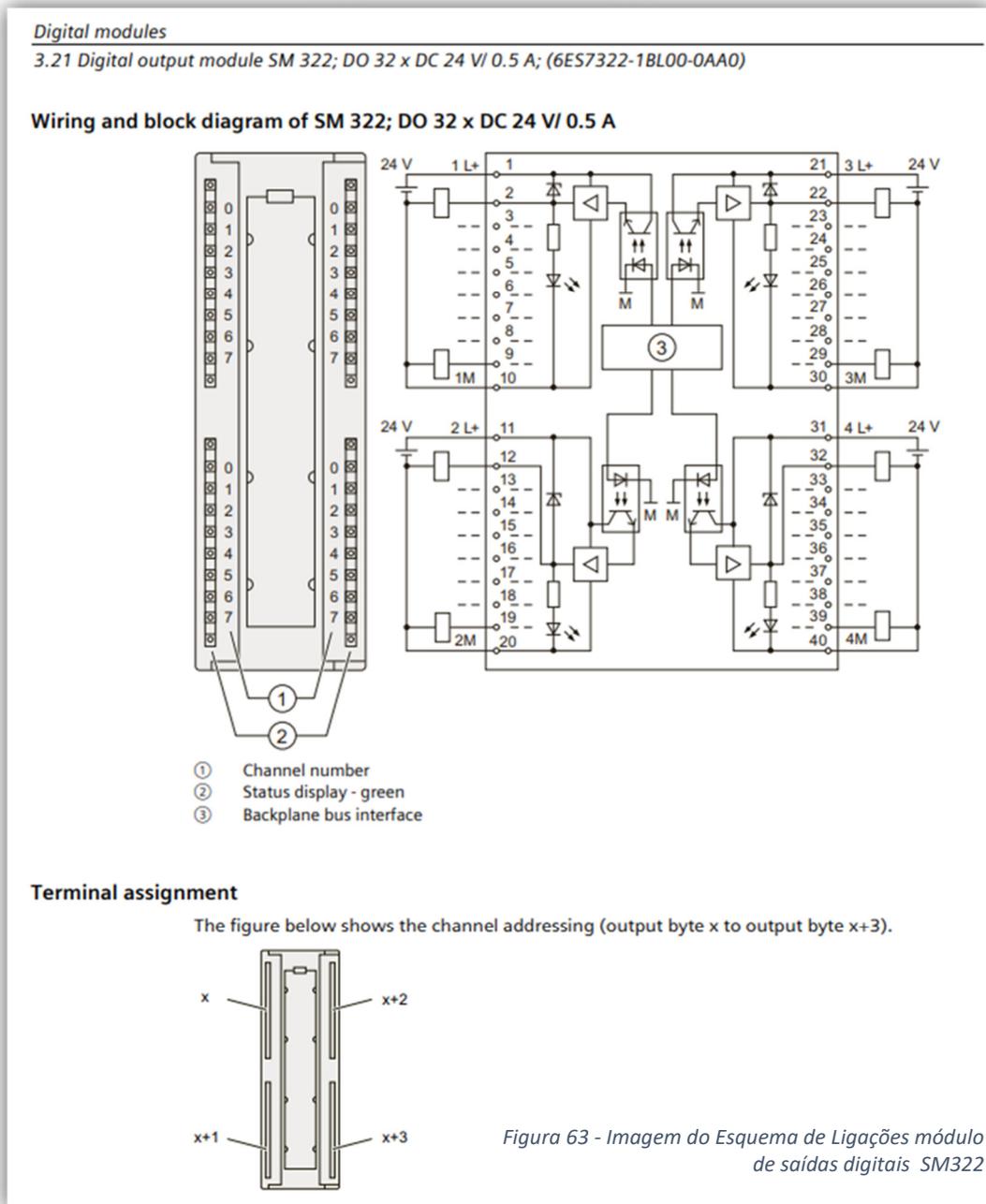
Figura 60 - Imagem módulo entradas digitais SM321 – Order 6ES7321-1BL00-0AA0



Módulo de Saídas Digitais



Figura 62 - Imagem módulo entradas digitais SM322 – Order 6ES7322-1BL00-0AA0



7.2 Eletrificação PLC's compactos

Para PLC compactos, como é o caso do S7-1200, apresenta-se o esquema de ligações da versão CPU 1214C AC/DC/RLY.

Este modelo de PLC é alimentado com CA, possui fonte de alimentação interna e tem terminais que fornecem 24V DC.

Também estes terminais de 24VDC devem primeiro alimentar réguas de bornes, antes de serem ligados aos restantes terminais M e L, das áreas das entradas e das saídas, que estão fisicamente no mesmo *hardware*.

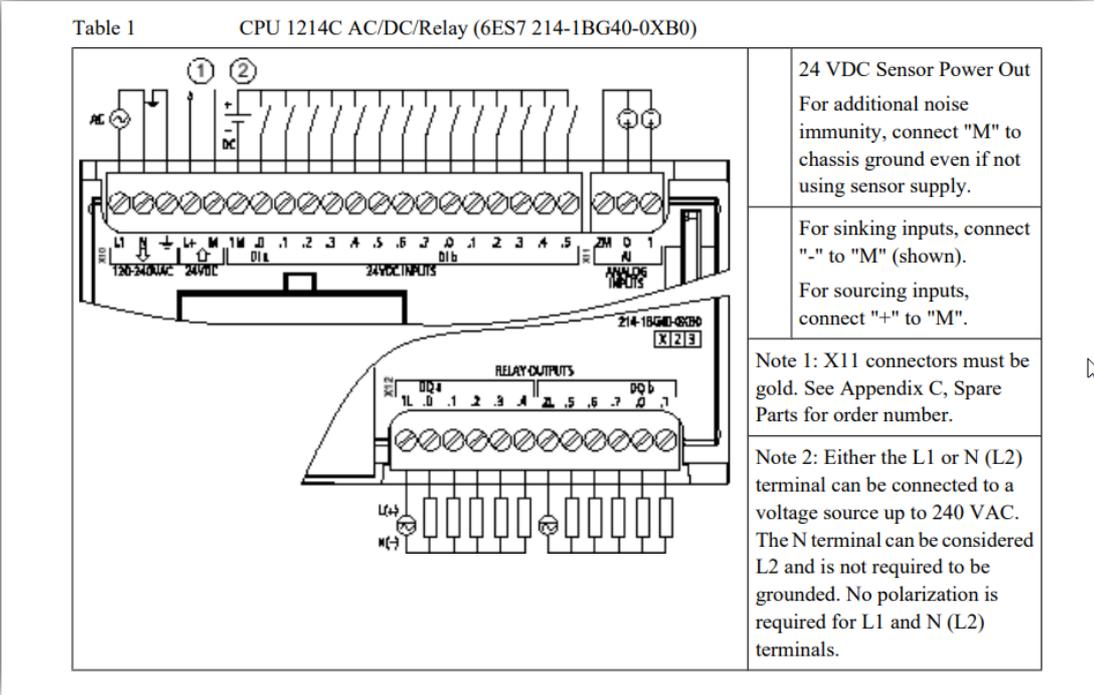


Figura 64 - Esquema de ligações PLC CPU S7-1214C AC/DC/RLY



Figura 65 - Imagem fotográfica de PLC CPU S7-1212C AC/DC/RLY com extensão DI/DO SM1223DC/DC



Figura 66 - Imagem da eletrificação do PLC
S7-1200 - Entradas

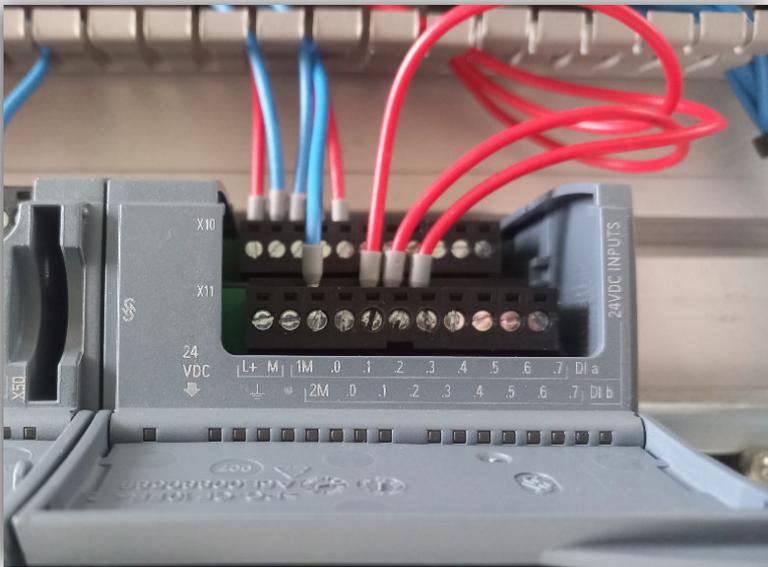


Figura 67 - Imagem da eletrificação da extensão DI/DO - Entradas



Figura 69 - Imagem da eletrificação do PLC S7-1200 e da extensão DI/DO - Saídas

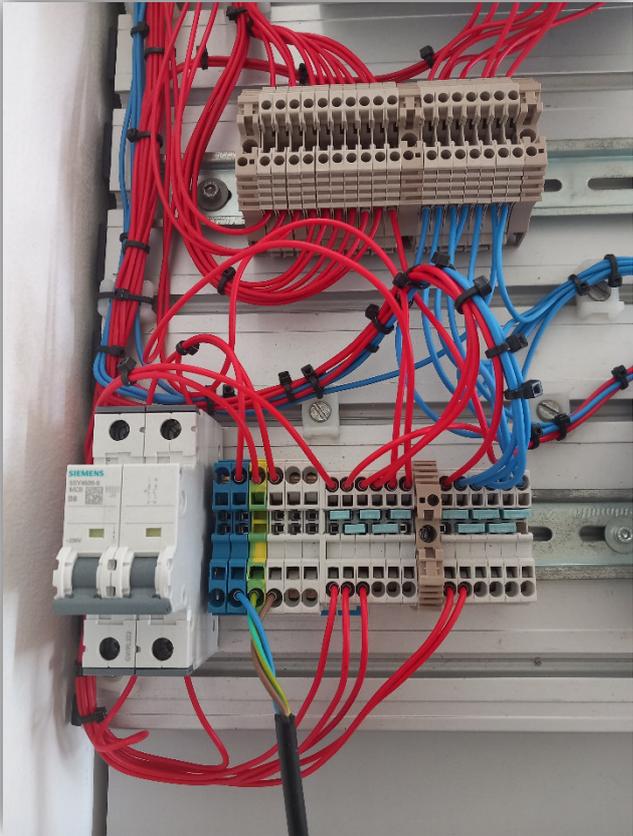


Figura 68 - Componentes elétricos para eletrificação de PLC's

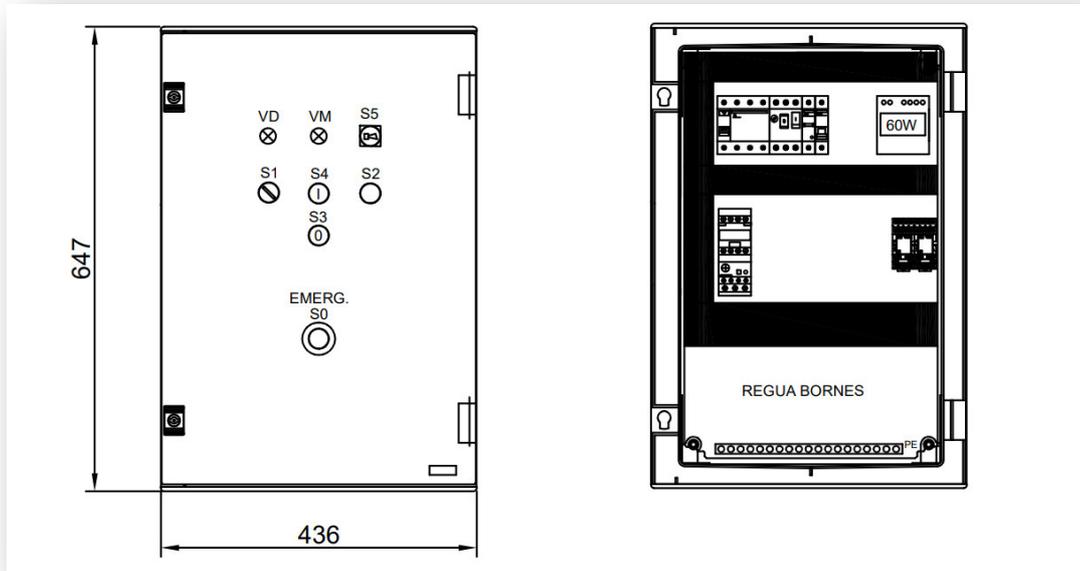


Figura 70 - Peça desenhada QE - Vista de frente exterior e interior

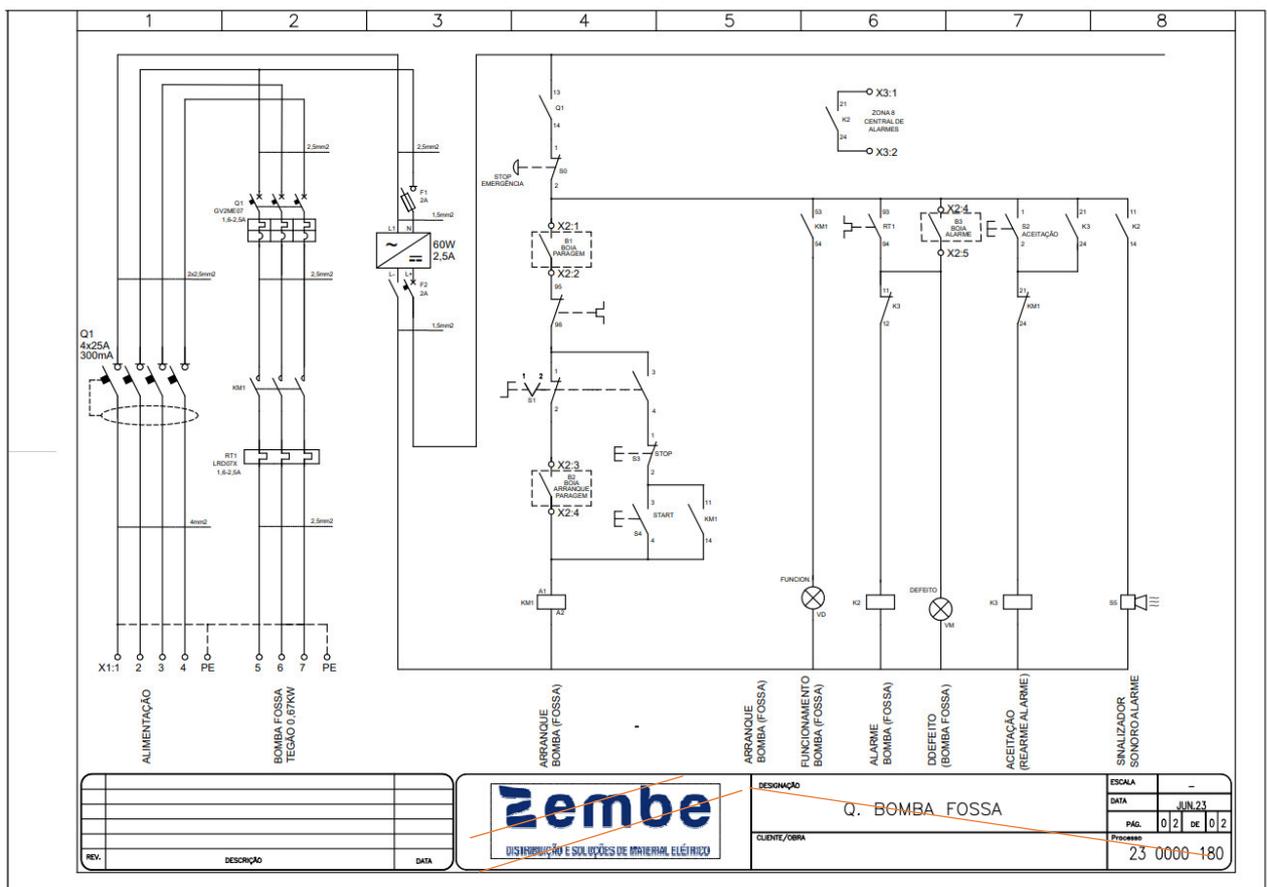


Figura 71 - Esquema elétrico de bombagem

8 COMUNICAÇÃO

A comunicação entre máquinas, assim como, entre o ser humano e a máquina são dois aspectos importantes da automação. No que respeita à comunicação entre ser humano e máquina, existem incontáveis dispositivos e equipamentos, sob a designação de *human-machine interface* - HMI ou interface homem-máquina (IHM).

Os protocolos de comunicação, como tecnologia de comunicação entre máquinas, acompanharam a evolução dos computadores, dos *chips* e da própria *internet*, assim como a conhecemos hoje, por *www* – *world wide web*.

As redes de comunicação industriais estão em franco desenvolvimento, actualmente, devido ao desenvolvimento da Indústria 4.0 e da IOT, forçando a necessidade de melhoria contínua dos protocolos.

Os PLC's (hardware) e os computadores (software), assim como, outros dispositivos de instrumentação e controlo de processo industrial, comunicam entre si, sobretudo, via *EtherNet/IP*. Existem outras soluções de comunicações, via wireless, que obriga ao acréscimo de módulos especiais e, que são a solução ideal em locais remotos e isolados.

Apresenta-se aqui, algumas tipologias de comunicação em rede e wireless, em ambiente de automação industrial, sendo certo que a forma de comunicação mais comum, para aquisição e envio de dados, é seguramente a *EtherNet/IP*.

- a) CANopen
- b) DeviceNet – IEC 62026-3 – está relacionada com a tecnologia CAN
- c) ControlNet é uma tipologia fieldbus – IEC 61784/61158. Tipologia de rede muito utilizada em comunicação, entre PLC's e também entre PLC, entradas e saídas.
- d) EtherNet/IP – Protocolo de rede industrial que junta e adapta o CIP à Ethernet standard.
- e) HART Protocolo - “HART” significa “Highway Addressable Remote Transducer” – é um protocolo de comunicação que permite dados analógicos e dados digitais, em simultâneo. A sua aplicação é junto da instrumentação de campo, para recolha e envio de dados entre o software e o instrumento.
- f) HostLink – protocolo de comunicação, entre PLC's da marca Omron. Baseado no Sistema de numeração ASCII, utiliza comunicação RS-232 ou RS-422.
- g) INTERBUS – IEC 61158 – sistema de transmissão de dados, desenvolvido pela marca Phoenix Contact, para comunicação entre computadores, PLC's, variadores de frequência, e outros dispositivos.
- h) IO-Link – IEC61131-9 – rede de comunicação industrial de curta distância, digital, cablada ou wireless, estabelecida entre entradas e saídas e o PLC.
- i) PROFIBUS - IEC 61158/61784-1 – *fieldbus de comunicação entre instrumentação de processo, entradas e saídas, e o Sistema de control, ou, PLC. É possível ser utilizado em locais ATEX – apresenta-se por uma cablagem lilás, com um conector próprio.*
- j) PROFINET - IEC 61158/61784-2 – *network/rede de comunicação entre instrumentação de processo, entradas e saídas, e o sistema de controlo, ou, PLC. Apresenta-se por uma cablagem verde, com um conector próprio, com aspecto semelhante a um cabo de rede de internet comum.*

- Allen Bradley (RockwellAutomation) – módulos de comunicação

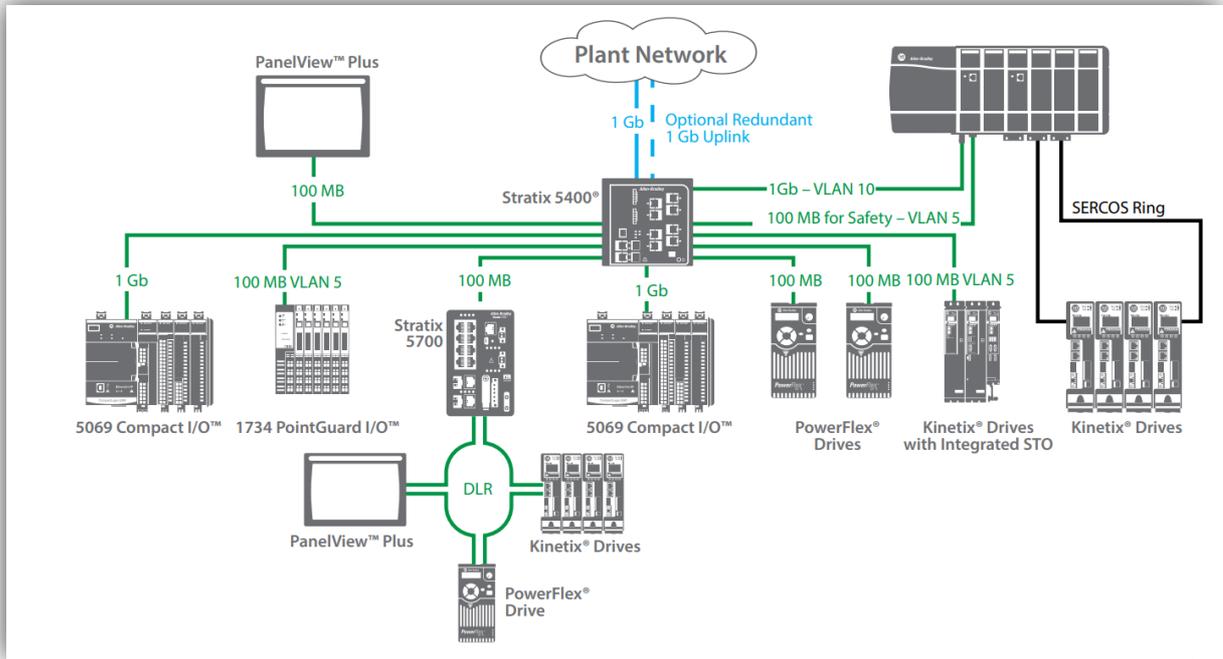


Figura 72 - Estrutura de comunicação industrial AllenBradley

- PROFIBUS e PROFINET solução conjunta para processo de engarrafamento

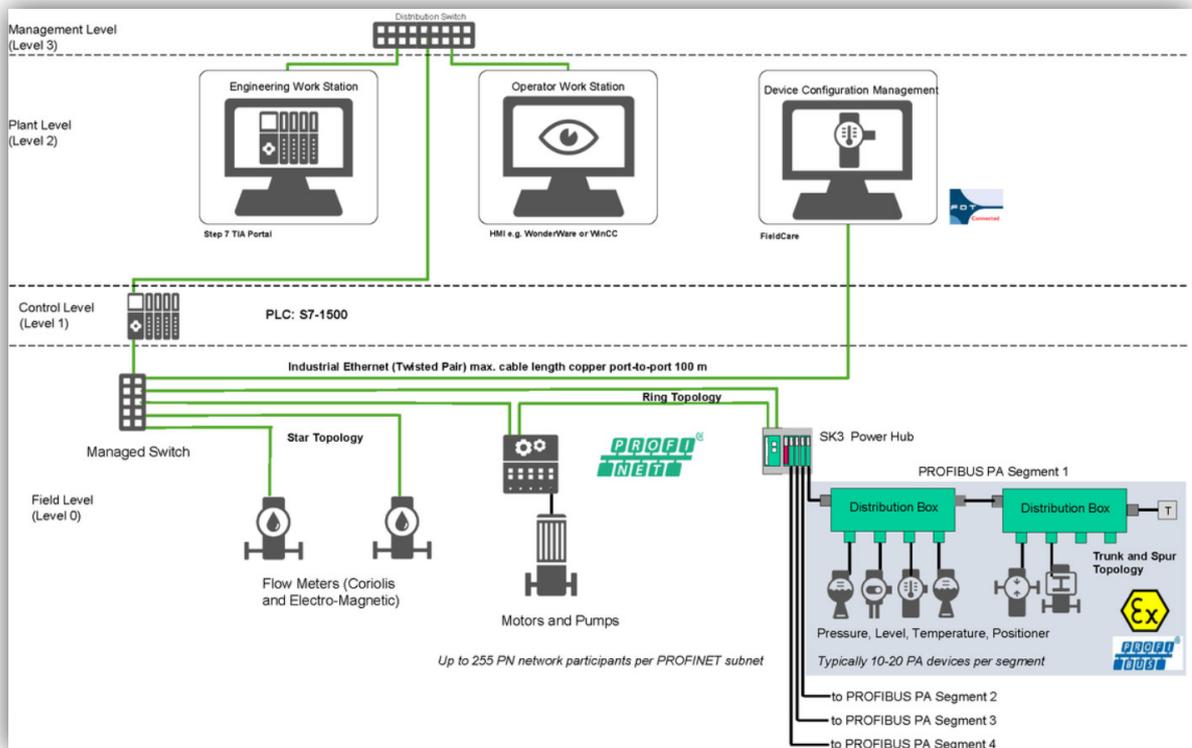


Figura 73 - PROFIBUS - caso de estudo

9 ANEXOS

Os sete exercícios aqui solicitados, devem ser realizados em sala, recorrendo ao formato papel e ao formato software quando indicado. Para a sua realização devem recorrer ao manual e às informações dadas pelo formador, no decorrer da formação.

9.1 Exercícios de aplicação

9.1.1 Elaboração de lista de E/S – em formato papel

Considere um sistema que apresenta os seguintes componentes:

Um depósito para água, uma bomba para fazer circular a água, um botão on/off para desligar o sistema na sua totalidade, 3 botões de pressão para escolher um produto, vários leds.

O princípio de funcionamento é: o utilizador deve ligar o equipamento no botão on/off (Start) dando início ao aquecimento de uma resistência (R1). Essa resistência aquece durante 10 segundos, enquanto um led (H1) pisca a 1Hz. Terminado esse tempo, a máquina indica que está pronta a funcionar, porque o led (H1) desliga-se e liga-se um outro verde (H2) de forma permanente. O utilizador escolhe um dos três modos de funcionamento, correspondendo cada um a uma temporização diferente:

Modo A – temporização de 5 segundos para a bomba (B1) fazer circular a água no sistema

Modo B – temporização de 8 segundos para a bomba (B1) fazer circular a água no sistema

Modo C – temporização de 12 segundos para a bomba (B1) fazer circular a água no sistema.

Após cada temporização terminar a bomba desliga-se.

Quando um dos modos está em funcionamento o led (H2) pisca a 1Hz.

O sistema quando ligado em Start, fica em suspensão, desligando-se na totalidade, quando estiver mais de 60 segundos sem ser utilizado.

9.1.2 Elaboração de GRAFCET - em formato papel

Elabore o grafcet para o sistema indicado no exercício anterior. Não considere funcionamento cíclico. Acrescente um botão stop para parar o funcionamento a qualquer instante.

9.1.3 Elaboração de Ladder em formato papel

Carimbo automático, com duas marcações

Princípio de funcionamento – Um equipamento carimba, de forma automática, caixas de cartão, com duas marcações independentes, realizadas pelo cilindro A e pelo cilindro B. O técnico carrega no botão S1 e liga-se um sinalizador H1. O cilindro A que está recuado, avança e recua, realizando a primeira marcação. Feita essa primeira marcação, o sinalizador H1 desliga-se. Nesse momento, o cilindro B avança e liga-se o sinalizador H2. Quando o cilindro B estiver avançado, a caixa já foi carimbada com a segunda marcação. Nesse momento, o equipamento regressa ao seu estado inicial, com ambos os cilindros recuados e ambos os sinalizadores desligados. Quando for necessário carimbar nova caixa, o técnico terá de voltar a carregar no botão S1.

NOTA: recorra a instruções do tipo SR, para programar as saídas.

9.1.4 Elaboração de programação Ladder em software TIA Portal

Elabore a programação Ladder, para o exercício 7.1.3

9.1.5 Exercício completo, em software TIA Portal – “Separador de embalagens”

Equacione um elemento de cartão, um elemento de plástico e um elemento metálico.

Equacione um princípio de funcionamento para um equipamento ou processo que separe esses três elementos. Elabore a lista de E/S, elabore o GRAFCET e o respectivo programa em Ladder.

Deve realizar as tarefas lista de E/S e Ladder, em software TIA Portal.

9.1.6 Exercício completo, em software TIA Portal – “Máquina de café”

Elabore a programação Ladder, para o exercício 7.1.1.

9.2 Ambiente TIA Portal – Siemens

Neste capítulo dos anexos constam algumas imagens ilustrativas, com indicações breves para guiar o formando no início da utilização do software da Siemens, o “TIA-Portal”.

Este é o software que utilizaremos nesta formação, na componente mais prática. Naturalmente, existem vários softwares que estão ligados às diferentes marcas de PLC’s.

É importante lembrar que os softwares, apesar de diferentes no aspecto, no que respeita à programação serão muito idênticos, uma vez que devem respeitar a norma internacional.

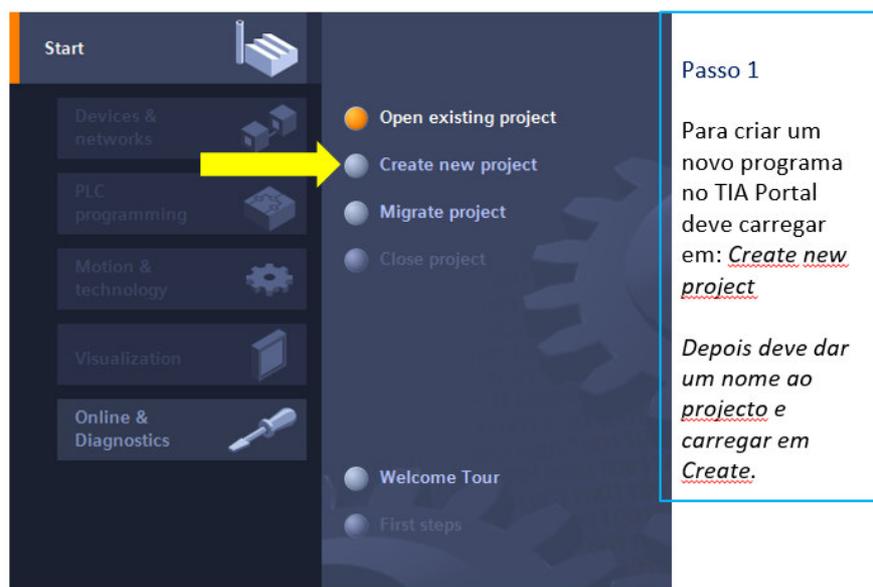


Figura 74 – Imagem ilustrativa da abertura do TIA-Portal

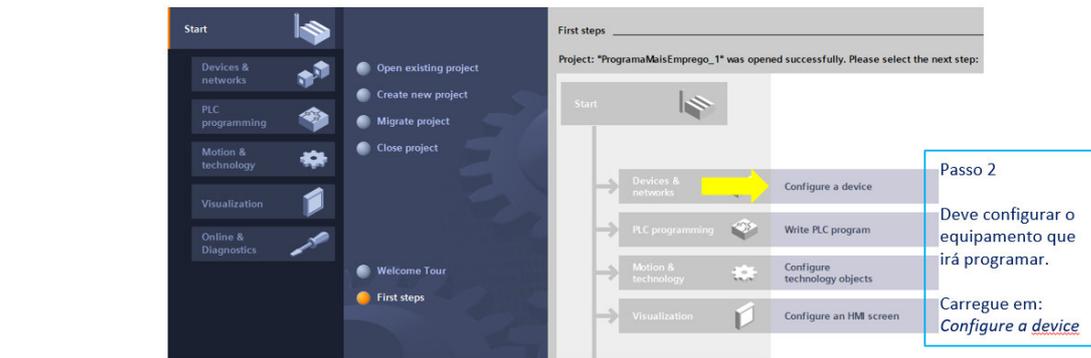


Figura 75 – Imagem ilustrativa da página onde se configura o hardware

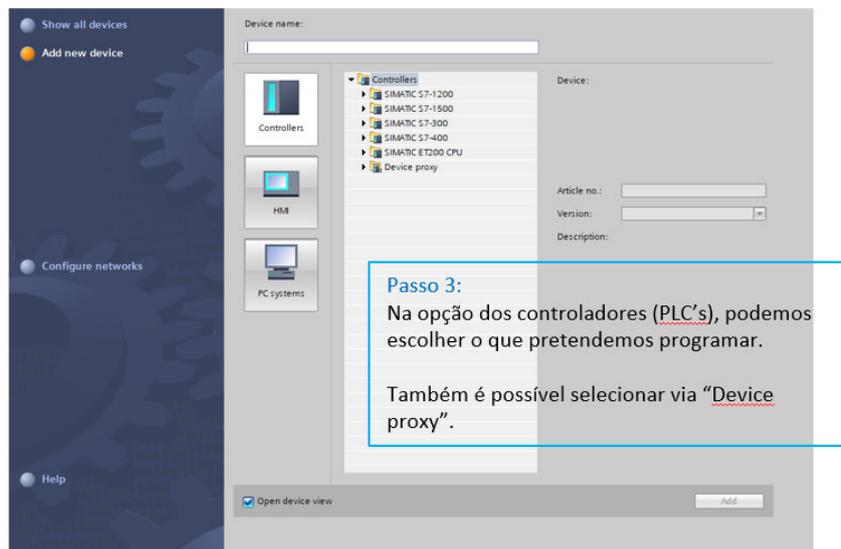


Figura 77 – Imagem ilustrativa da página onde se configura o hardware (cont.)

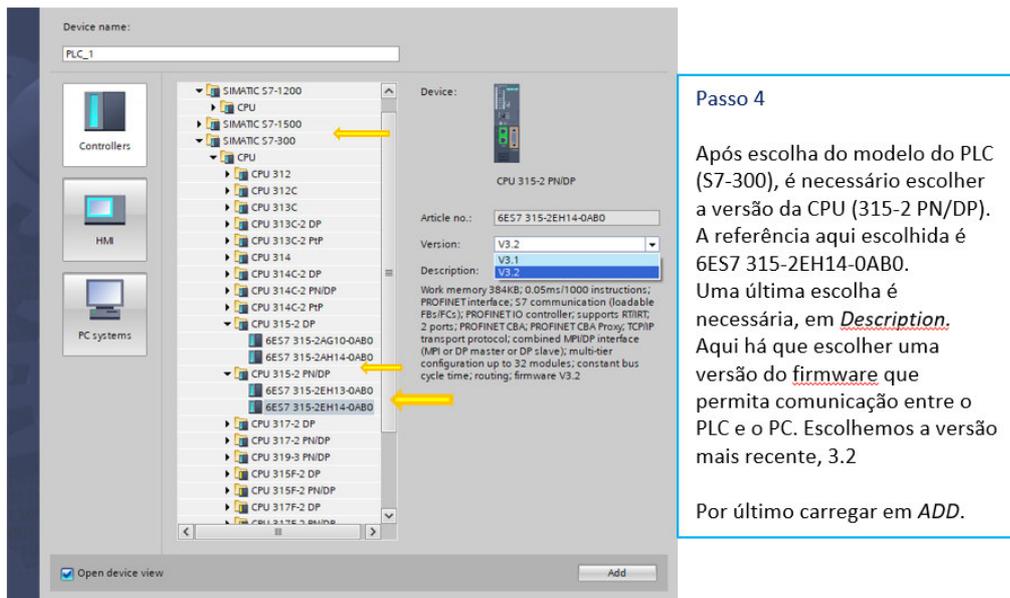


Figura 76 - Imagem ilustrativa da página onde se configura o hardware (cont.1)

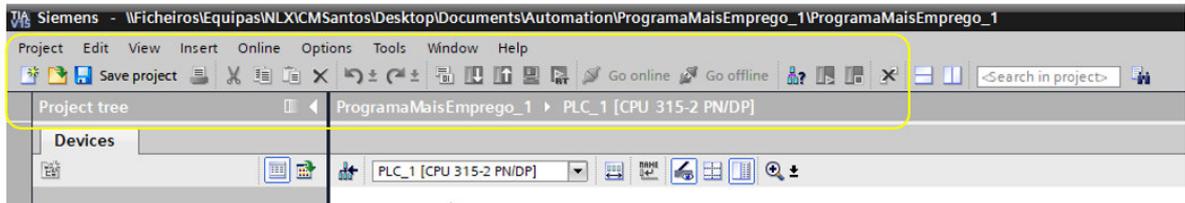


Figura 78 – Menus e atalhos – TIA Portal - de grande importância para a gestão do programa, a comunicação com o PLC e a monitorização

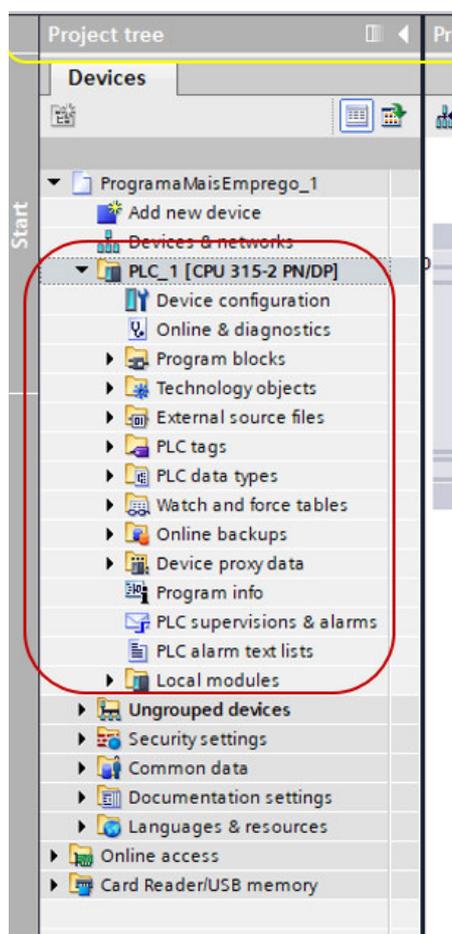


Figura 80 – “Project Tree” – local onde visualizamos todos os pormenores relacionamos com o programa

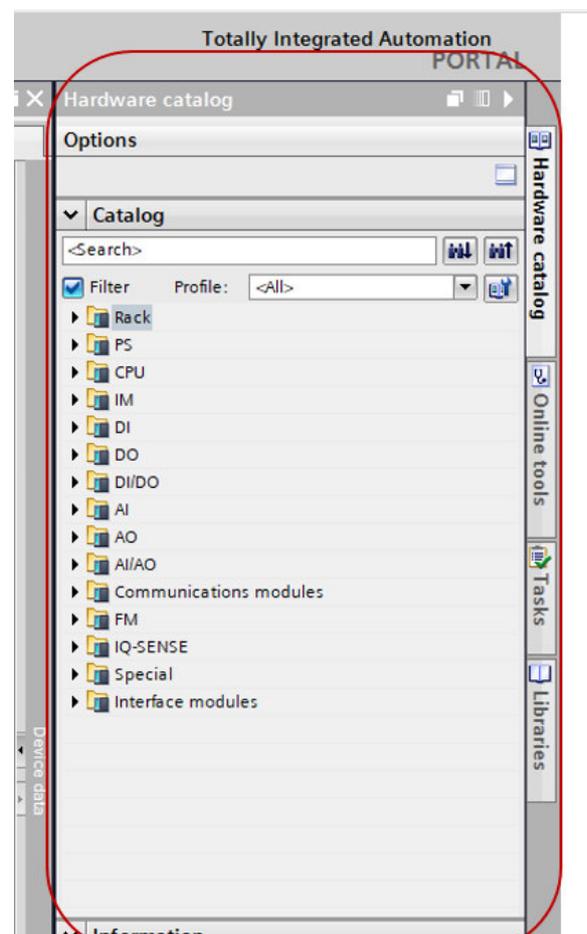


Figura 79 – separador “hardware” – local onde alteramos e manipulamos a parte física do PLC e adicionamos tipos de software.

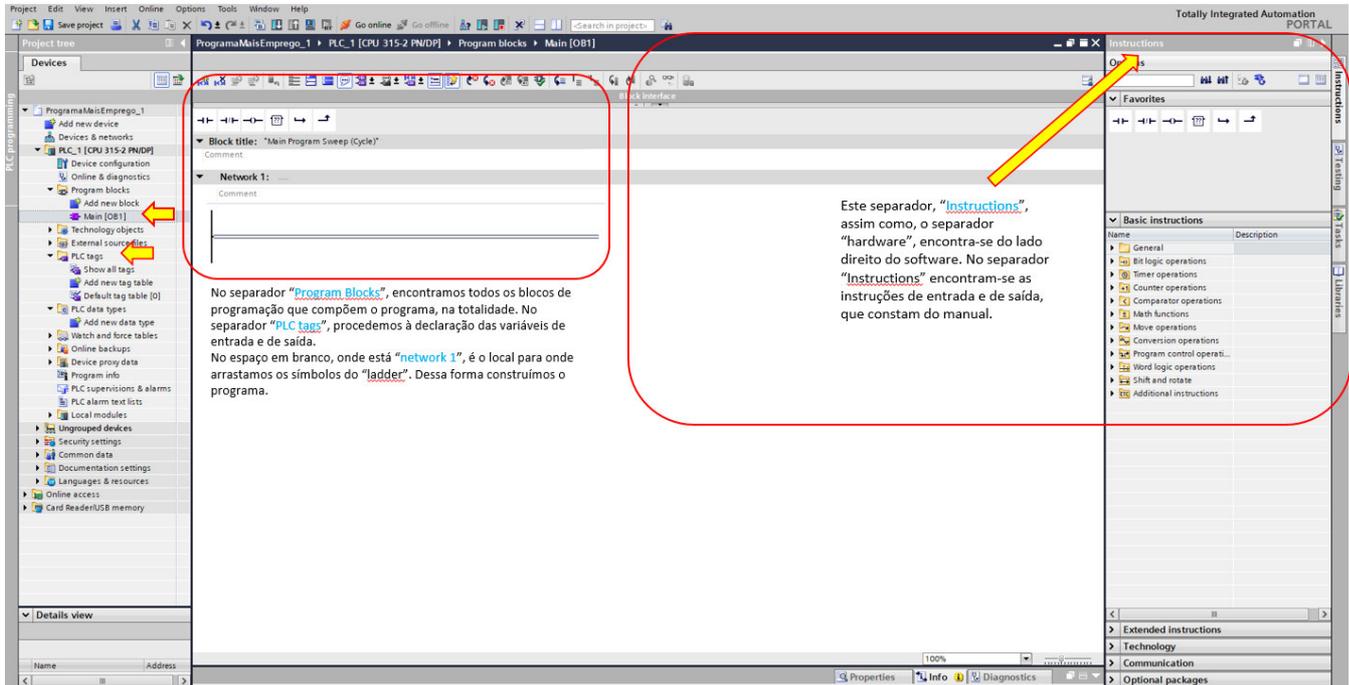


Figura 81 - Imagem ilustrativa geral - do ambiente de programação TIA-Portal

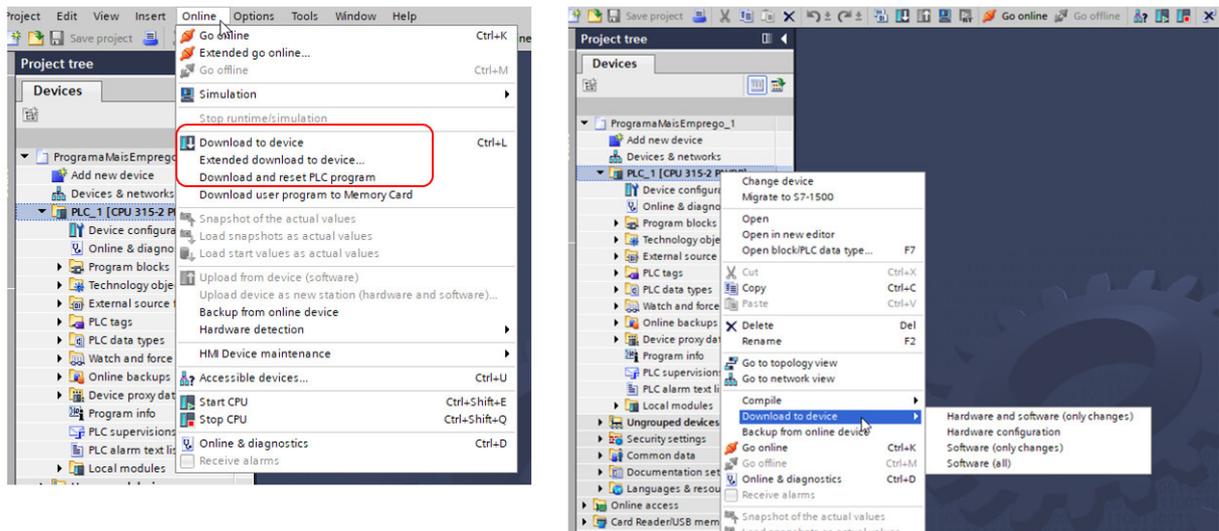


Figura 82 - Dois caminhos possíveis para a comunicação PC-PLC

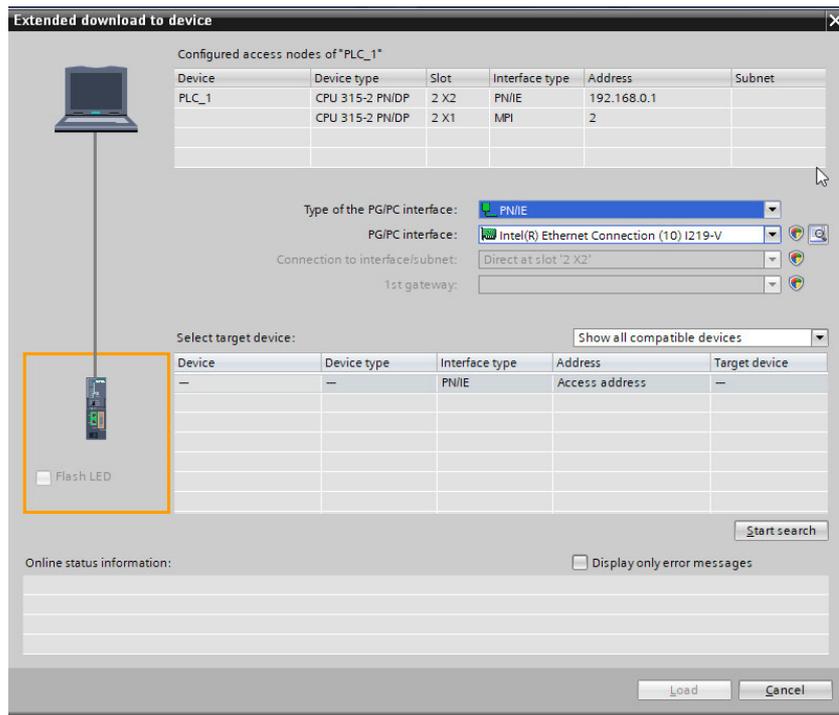


Figura 83 - Imagem ilustrativa da janela inicial do download em TIA-Portal

10 FONTES BIBLIOGRÁFICAS

Os esquemas elétricos, ilustrações, imagens, fotografias, tabelas, e demais informações necessárias à elaboração deste manual foram retiradas das seguintes obras:

SANTOS, Adriano Almeida; SILVA, António Ferreira da – Automação Integrada. 1ª edição. Porto, 2012. ISBN:978-989-723-002-8

BERGER, Hans – Automating with STEP 7 in LAD and FDB, 5ª edição. Erlangen, 2012. (Conteúdo totalmente em inglês) – ISBN 978-3-89578-410-1

JOHN, Karl-Heinz; TIEGELKAMP, Michael - IEC 61131-3 Programming Industrial Automation Systems. 2ª Edição. Springer, 2010 – ISBN 978-3-642-12014-5

11 WEBGRAFIA

Para mais informação técnica e comercial sobre sensores, transdutores, actuadores, motores, electroválvulas, PLC's de variadas gamas e marcas, diversos equipamentos de automação, sugerem-se os sites das marcas dos equipamentos mencionados no manual de formação, entre outros.

	<h2>Automação</h2>	
---	--------------------	---

Alguns sites são também importantes para o conhecimento da comunicação industrial. É de notar que grande parte da informação será encontrada, principalmente, em língua inglesa.

Site Rockwell

<https://www.rockwellautomation.com/en-us/products/hardware/allen-bradley/programmable-controllers.html>

<https://www.sensiglobal.com/Measurement> - Instrumentação

Site OMRON

<https://www.ia.omron.com/products/category/automation-systems/programmable-controllers/>

Site Schneider

<https://www.se.com/pt/pt/all-products/>

Site Festo

https://www.festo.com/pt/pt/c/produtos-id_pim1/

Site Siemens

<https://www.siemens.com/global/en/products/automation/systems/industrial/plc.html>

<https://mall.industry.siemens.com/goos/WelcomePage.aspx?regionUrl=/pt&language=pt>

Site Mauser

https://mauser.pt/catalog/index.php?cPath=324_2683&atalho=2669

Site Telemecanique

<https://tesensors.com/pt/pt>

Site Saia

<https://sbc-support.com/en/product-category/programmable-controller/>

Site WEG

<https://static2.weg.net/medias/downloadcenter/h32/hc5/WEG-motores-eletricos-guia-de-especificacao-50032749-brochure-portuguese-web.pdf>

Site Zembe

<https://www.zembe.pt/quadrista-certificado>

Site PROFIBUS

<https://www.profibus.com/>

Site EMERSON

<https://www.emerson.com/pt-br>

Site FIELDCOMMGROUP – tecnologia de comunicação HART

<https://www.fieldcommgroup.org/technologies/HART-IP>

12 SOFTWARE

O software utilizado para a produção do manual, TIA-Portal da Siemens, versão 15.1, faz parte de licença educacional, utilizada apenas para efeitos de formação profissional. As licenças educacionais não podem ser partilhadas com os formandos.

Para a elaboração de GRAFCET existem softwares, também sujeitos a licença, porém, neste manual, o software utilizado foi o Microsoft Visio.

13 GLOSSÁRIO

Neste capítulo colocam-se algumas definições que se encontram também, dentro do manual.

PLC – Programmable logic controller

CLP – Controlador lógico programável

CPU – Unidade central de processamento

PC – Computador pessoal. Termo utilizado para se referir ao suporte onde corre o software, no momento da programação.

Manutibilidade – De acordo com a Norma Portuguesa NP 13306, define-se manutibilidade como a “aptidão de um bem, sob condições de utilização definidas, para ser mantido ou restaurado, de tal modo que possa cumprir uma função requerida, quando a manutenção é realizada em condições definidas, utilizando procedimentos e recursos prescritos”.

Entrada ou Input – E ou I – Conjunto de informação fornecida à CPU, sob a forma de variáveis de sinais elétricos ou, eletrónicos digitais ou analógicos.

Saída ou Output – S ou O – Conjunto de informação dada pela CPU, sob a forma de variáveis de sinais elétricos ou, eletrónicos digitais ou analógicos.

Contacto ou switch – Dispositivo cuja função é seccionar partes de um circuito elétrico, abrindo ou fechando o mesmo, permitindo ou impedindo a passagem de corrente elétrica.

SFC – Sequential function chart (SFC) é uma linguagem de programação gráfica.

IEC – Comissão Eletrotécnica Internacional

MPI – Interface Multiponto (propriedade Siemens) é um dispositivo de comunicação PLC-PC.

Etapa – estado de um programa.

Transição – a condição ou condições necessárias para transitar entre duas etapas.

Common Industrial Protocol (CIP) – é um protocolo de comunicação para a industria.