

Programa o de CLPs

Aula 02 - Caracter sticas e introdu o   programa o

Apresentação

Nesta aula serão abordadas as características dos componentes internos que constituem os Controladores Lógicos Programáveis (CLP), a importância que cada componente exerce para constituição e configuração do hardware. As interfaces de entradas, saídas e suas peculiaridades. E no caso de aplicação, como selecionamos o CLP adequado para uma determinada tarefa e, após sua seleção, como é iniciada a lógica de programação utilizando a álgebra booleana para isso.

Objetivos

Ao final desta aula, você será capaz de:

- Conhecer internamente o Controlador Lógico Programável e suas características.
- Identificar as interfaces de entrada e saída.
- Selecionar um Controlador Lógico Programável de acordo com a sua aplicação.
- Entender a lógica de programação;
- Conhecer a lógica Booleana;

Introdução

Quando adquirimos experiência com os Controladores Lógicos Programáveis, essa característica nos permite olhar mais a fundo para o controlador, mesmo que tenhamos mais afinidade com o CLP de um determinado fabricante. Digo isso em relação às configurações que os mesmos já trazem consigo, tipo: Quantidade de entradas, saídas, quais os tipos de cada, se tem entrada analógica, se tem saída analógica, e essas configurações são comuns a todos os fabricantes de CLP.

Nesta aula, abordaremos justamente este tópico, identificar entradas e saídas, saber como elas funcionam através da eletrônica envolvida. E o principal, saber dimensionar um CLP dadas as características de uma aplicação, este, sim, é de suma importância para o técnico de automação.

Ao final, veremos a introdução das linguagens de programação e será dada ênfase à Linguagem Ladder, a qual é considerada uma linguagem universal dentre os CLPs, bem como a lógica de programação, que é uma tarefa de exercitação constante de todo e qualquer programador. Vale ressaltar que para cada aplicação a ser desenvolvida, geralmente existe mais de uma solução na implementação da lógica de programação adotada.

Interfaces de entradas e saídas

Na aula anterior aprendemos a identificar os componentes de um CLP, ou seja, como podemos identificar o CLP de uma máquina que acabamos de adquirir, quais são suas características e, claro, o principal, como funciona a máquina. Nesse contexto, para entendermos como funciona a máquina, tenha sempre em mente que os CLP's "esperam respostas" dos sensores, chaves fim de curso, botoeiras, relés e interfaces homem máquina (IHM), bem como "enviam saídas" para os atuadores, através dos quais o CLP monitora e controla o processo de toda e qualquer máquina automatizada. Por exemplo: imaginemos uma máquina de lavar roupa, quando ao ligá-la você deve selecionar: qual tipo de roupa; qual modo de lavagem; qual tempo de molho; qual o nível de água... e por aí vai. Nesse processo é imprescindível esse

“diálogo homem máquina”. Observe que estamos “ativando ou desativando entradas” que, combinadas, se tornam a configuração de lavagem; após isso, o CLP “compila” esses dados com o programa de usuário (LADDER, funções em diagrama de blocos, lista de instruções, etc.) e pronto! Basta “apertar o start” e a máquina entra em funcionamento. Todo esse processo é muito parecido com tantos outros equipamentos, certamente você consegue se recordar de algum.

Voltando à questão das conexões dos sensores, chaves, etc., o CLP *recebe entradas e saídas tanto digitais quanto analógicas*, as quais controlam e monitoram todo o processo. Esses módulos ou interfaces de entradas e saídas são a seção do CLP onde *sensores e atuadores* estarão conectados.

Todos os sinais provenientes do campo são informados à CPU logo ao serem detectadas pelos “sensores de entrada” e, por sua vez, as ordens geradas pela CPU são comunicadas aos elementos do processo por meio das interfaces de saída.

Após isso tenha em mente que os CLP's são capazes de manejar sinais de tensões e correntes a nível industrial, isto devido ao bloco de circuitos destas interfaces (são os conversores), ou seja, permitem se conectar diretamente com sensores e atuadores do processo, através de suas interfaces de entrada e saída.

Entrada Digital

Este tipo de interface tem a função de informar à CPU, da presença ou ausência de sinal de tensão ou corrente, em um circuito, abertura ou fechamento de um contato, que pode ser de uma botoeira, de um fim de curso, sensor, etc. Por exemplo, quando uma entrada digital está sendo alimentada por um sinal de 24V, a CPU interpreta como nível alto “1” e quando o sinal é desligado é interpretado como nível baixo “0”.

As entradas digitais contemplam uma gama muito grande de opções de operação. Um contato externo ao controlador pode estar conectado a vários níveis de tensão, que pode ir de 5V a 220V.

Sinais digitais ou discretos

É simplesmente um sinal de "1" ou "0", verdadeiro ou falso, ver gráfico 1. Ou seja, as chaves são exemplos destes dispositivos que proporcionam um sinal discreto de tensão ou corrente em que um determinado "range" corresponde a "ON" (verdadeiro), em outro a "OFF" (falso). Por exemplo, um CLP opera com 24Vcc nas interfaces de entrada e saídas onde valores superiores à 22Vcc representam "ON", e valores inferiores a 2Vcc representam "OFF".

Parâmetros de um sinal digital:

- Altura do pulso (nível elétrico);
- Duração (tamanho do pulso);
- Frequência de repetição (velocidade de pulsos por segundo).

No gráfico 1 é possível identificar essas características:

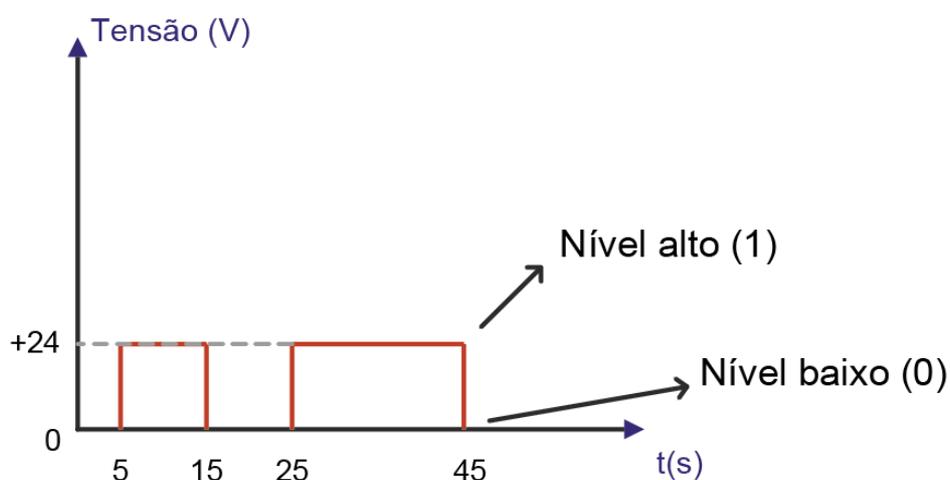


Gráfico 1 - Característica de um sinal digital.

Fonte: Elaborado pelo Autor (2016).

Nos CLP's, as interfaces de entrada digital se encarregam de converter a tensão ou corrente que recebe dos sensores, fim de curso, pulsadores, etc., em níveis apropriados para operação da CPU, conforme diagramas 1 e 2.

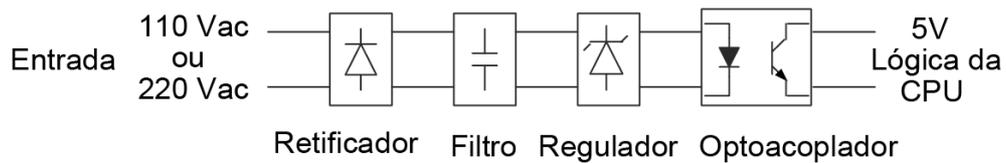


Diagrama 1 - circuito conversor AC-DC de uma entrada digital.

Fonte: Elaborado pelo Autor (2016).

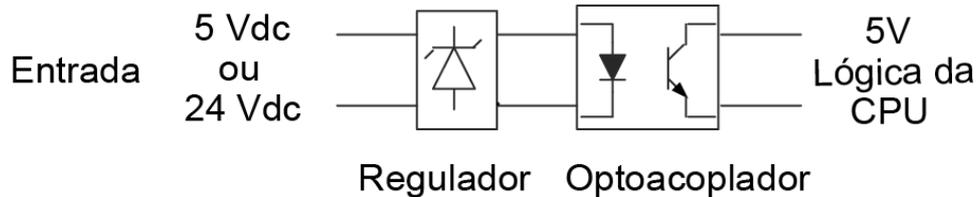


Diagrama 2 - circuito conversor DC-DC de uma entrada digital.

Fonte: Elaborado pelo Autor (2016).

Como exemplo de ligação de uma entrada digital veja uma interpretação do CLP que será estudado nesta aula. Trata-se de um CLP da Siemens (conforme figuras 1 e 2), cuja família é Simatic S7-1200 e, portanto, observe a ligação nos terminais de entrada deste CLP:

Em primeiro lugar observe sempre a referência do CLP, neste caso, Simatic S7-1200, CPU 1214C DC/DC/DC:

Fonte: Adaptado de [https://w3.siemens.com.br/automation/br/pt/automacao-e-controle/automacao-industrial/simatic-plc/s7-cm/s7-1200/Documents/Brochura SIMATIC S7 1200_portugues.pdf](https://w3.siemens.com.br/automation/br/pt/automacao-e-controle/automacao-industrial/simatic-plc/s7-cm/s7-1200/Documents/Brochura_SIMATIC_S7_1200_portugues.pdf) Acesso em: 10 out. 2015.

Observações

1. Como a própria referência já diz, sua alimentação é em 24Vdc;
2. Caso queira conectar sensores, sua ligação também deverá ser a mesma referência 24Vdc do CLP;
3. Existe uma referência para o estado das ligações, ou seja, 1M é ligado em 0V (-) para, assim, a entrada I0 ser ativa com 24V (+) ambos mesmo referencial do CLP;

Mas, atenção! A entrada só será ativada com os referenciais todos interligados. Isso é de extrema importância!

Portanto, a ligação ilustrada serve para ativar a entrada digital (I0). Mais adiante, veremos que podemos usar essa configuração de entrada para “ligar uma lâmpada”, por exemplo.

Saída digital

Da mesma maneira que ocorre na entrada, a interface de saída permite que sinais em baixa amplitude, originada na CPU, comandem contatores, solenoides, relés, etc. Basicamente são três os tipos de saídas: A triac (diagrama 3), a transistor (diagrama 4) e a relé (diagrama 5), esta última mais utilizada:

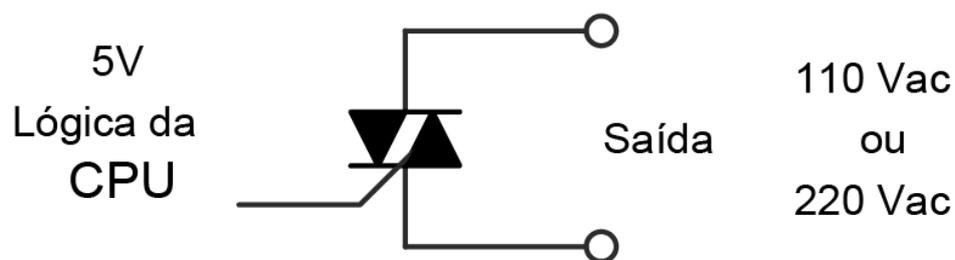


Diagrama 3 - Circuito conversor DC-AC de uma saída digital.

Fonte: Elaborado pelo Autor (2016).

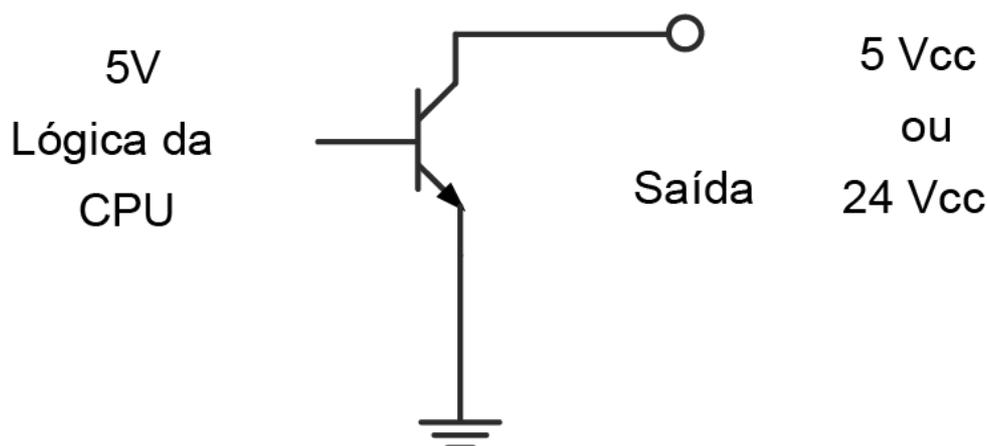


Diagrama 4 - Circuito DC-DC de uma saída digital a transistor.

Fonte: Elaborado pelo Autor (2016).

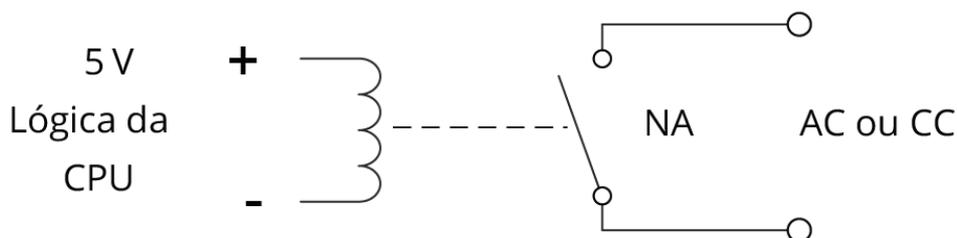


Diagrama 5 - Circuito conversor DC-AC de uma saída digital.

Fonte: Elaborado pelo Autor (2016).

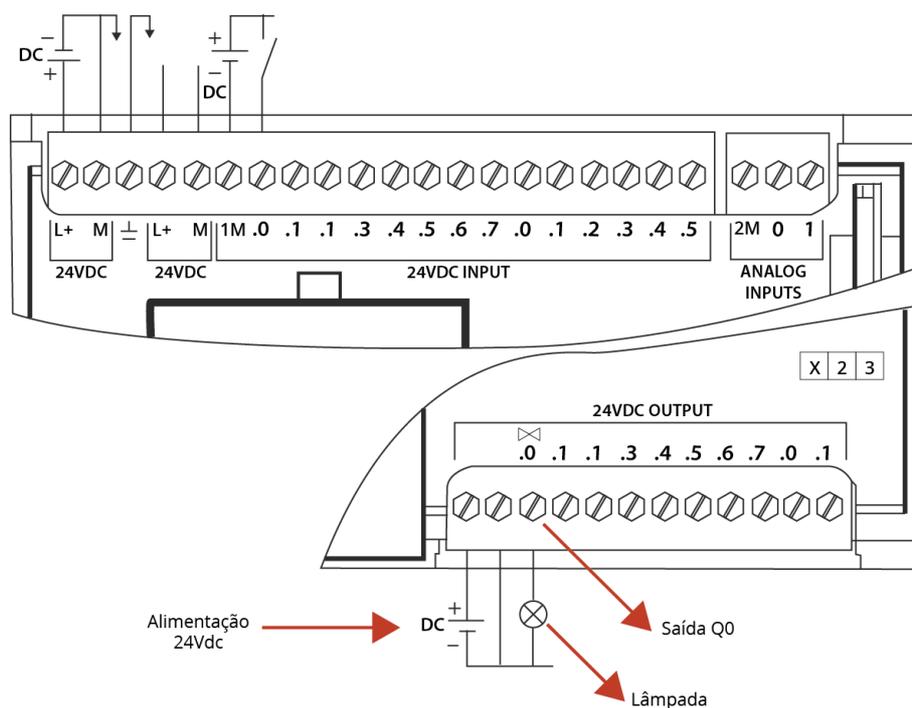
Observações

1. No caso das saídas a relé, o bloco “normalmente aberto (NA)” se comporta como um contato seco, ou seja, quando a saída for ativada o contato fecha. Observe que podemos colocar uma carga CC ou AC;
2. Em todos os aspectos as saídas dos CLP's são usadas para implementação do comando da máquina, tendo em vista que essas saídas suportam no máximo 8A. *“Portanto, não é recomendado conectar cargas elevadas nas saídas dos CLPs para não correr o risco de queimá-las”;*
3. http://materiais.imd.ufrn.br/prog_clp/aula02.php. As características, tanto das entradas quanto das saídas, estão indicadas no modelo do CLP (é a referência), basta consultar o manual do fabricante e então terá acesso às suas características;

Cabe ao programador decidir qual o melhor custo benefício na escolha da melhor saída, a qual lhe proporcione confiabilidade do sistema a ser automatizado.

Como exemplo de ligação, observe a figura 3, uma continuação do CLP S7-1200, só que desta vez com uma ligação de uma lâmpada utilizando uma saída Q_0 :

Figura 03 - Esquema de ligação de uma saída do CLP S7-1200.



Fonte: Adaptado de https://w3.siemens.com.br/automation/br/pt/automacao-e-controle/automacao-industrial/simatic-plc/s7-cm/s7-1200/Documents/Brochura SIMATIC S7 1200_portugues.pdf Acesso em: 10 nov. 2015.

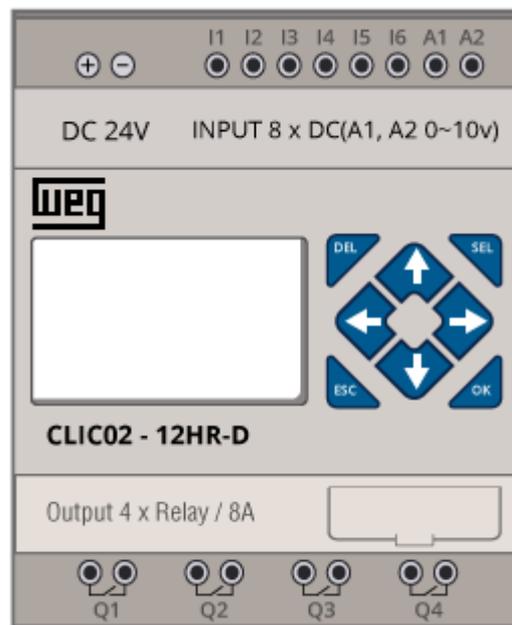
Observações

1. Como mostra a ilustração, a saída também é 24Vdc;
2. Para ativar o sinal de saída é preciso alimentar a saída com o mesmo referencial do CLP;
3. Obrigatoriamente a lâmpada da ilustração é 24Vdc;
4. Caso necessite ligar uma carga com uma tensão em 220V é fundamental o uso de relés.

Atividade 01

1. Dada a ilustração da figura 4 de um CLP WEG, identifique suas entradas e saídas e suas características. Observe que suas saídas são a relé, como poderíamos ligar uma lâmpada cuja tensão é 220Vac?

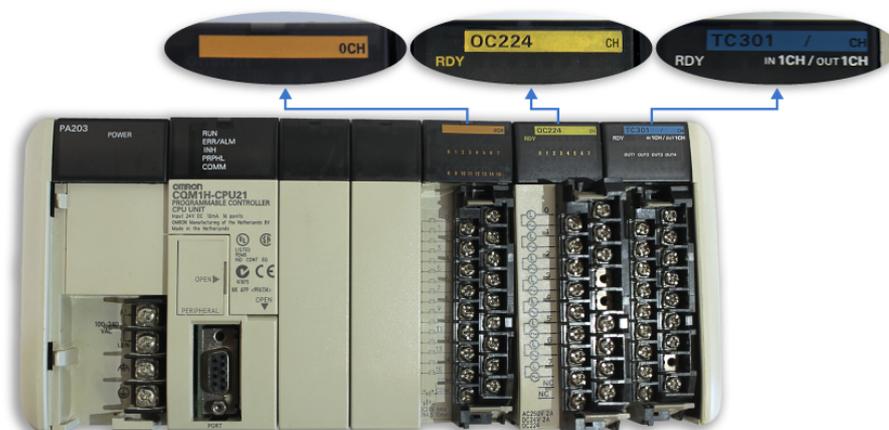
Figura 04 - Detalhe ilustrativo do CLP WEG;



Fonte: Adaptado de: <http://ecatalog.weg.net/files/wegnet/WEG-rele-programavel-clic-02-3rd-manual-portugues-br.pdf> Acesso em: 29 fev. 2016.

2. Faça uma pesquisa a respeito do CLP OMRON da figura 5 abaixo, no que diz respeito às características das entradas e saídas (em destaque). Quais os tipos de sinais compatíveis com o CLP que estas entradas e saídas proporcionam, bem como as suas ligações?

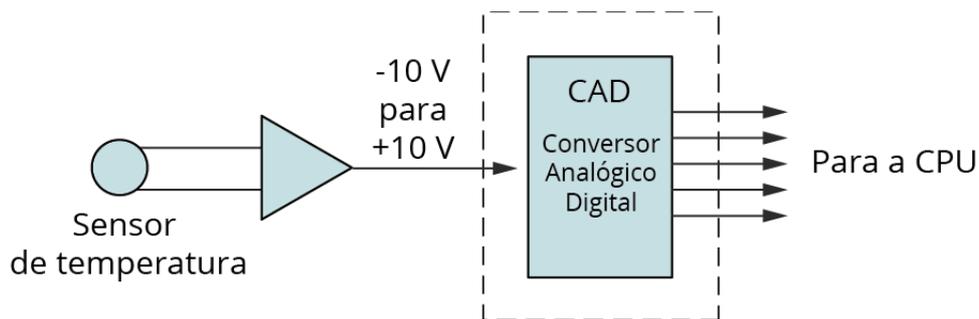
Figura 05 - Ilustração do CLP Omron.



Entrada Analógica

As entradas analógicas permitem que os controladores trabalhem com atuadores de controle analógico e leiam sinais analógicos, como é o caso da leitura de temperatura mostrada na figura 6. Além da temperatura, outras variáveis também trabalham com sinais analógico tais como pressão, vazão, nível, tensão, intensidade de corrente, etc.

Figura 06 - Representação de um sinal analógico, oriundo de um sensor de temperatura.



Fonte: CARRILHO (2005)

Os módulos de entradas analógicas convertem um sinal analógico em um número que fica depositado na memória interna do CLP, onde essa conversão é feita através de conversores AD dedicados, tendo em vista que a CPU só trabalha com sinais digitais. Essa conversão é realizada com precisão ou resolução (dependendo do número de bits) a cada intervalo de tempo. O processo de aquisição do sinal analógico segue as seguintes etapas:

1. **Filtro:** Em ambientes industriais ocorre muitas interferências harmônicas, o que pode ocasionar ruído no sinal enviado pelo sensor, para tanto é necessário filtrar esse sinal;
2. **Conversão:** Uma vez que o sinal está limpo, ou seja, sem interferência, um circuito eletrônico se encarrega de converter o sinal analógico em sinal digital para que, assim, possa ser enviado à memória;
3. **Memória interna:** Perceba que cada fração da onda do sinal analógico é convertido em sinal digital e que cada fração obtida é armazenada para que a CPU possa fazer a interpretação do mesmo.

Os sinais analógicos mais comumente usados são:

- Sinais de tensão
 - 0 a 10V
 - -10 a +10V
- Sinais de corrente
 - 4 a 20mA
 - 0 a 20mA

Um exemplo disso é o controle de velocidade de um motor através de um “potenciômetro” onde sua escala varia de “0 a 10V”.

Sinal analógico

À medida que a complexidade vai aumentando, é necessário contar com outro tipo de interface que possa interpretar **sinais analógicos** provenientes do processo e emitir como saídas.

Esses sinais são como “controles de volume” com um range de valores entre “0” e o “topo da escala”, normalmente representados com valores inteiros pelo CLP, com vários ranges de precisão, dependendo dos dispositivos utilizados. Variáveis como pressão, temperatura, vazão e peso, são normalmente representados como sinais analógicos. Os sinais analógicos podem usar tensão ou intensidade de corrente (4 a 20mA) com magnitude proporcional ao valor do sinal que é processado, conforme mostra o gráfico 2.

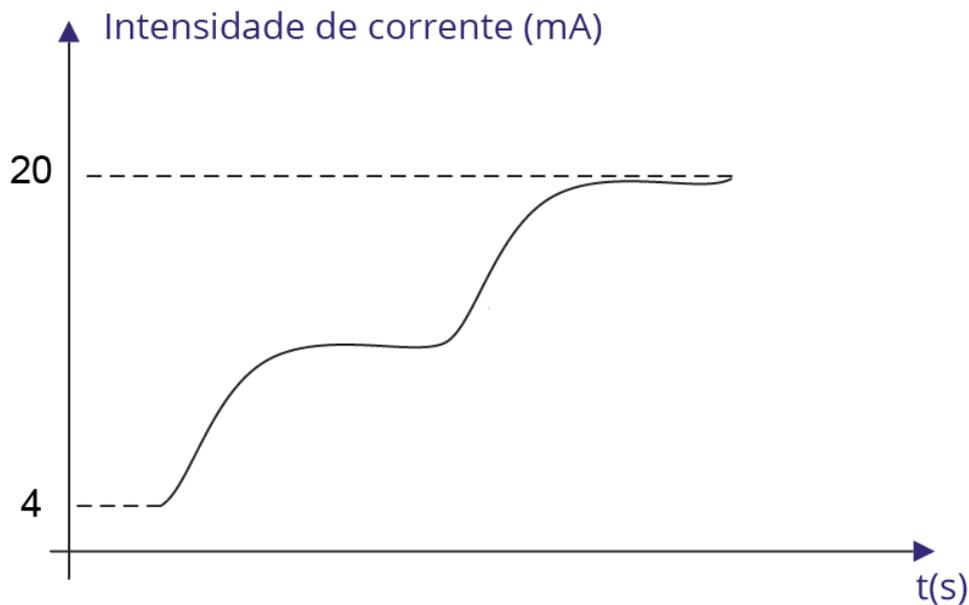


Gráfico 2 - Características de um sinal analógico.
Fonte: Elaborado pelo Autor (2016).

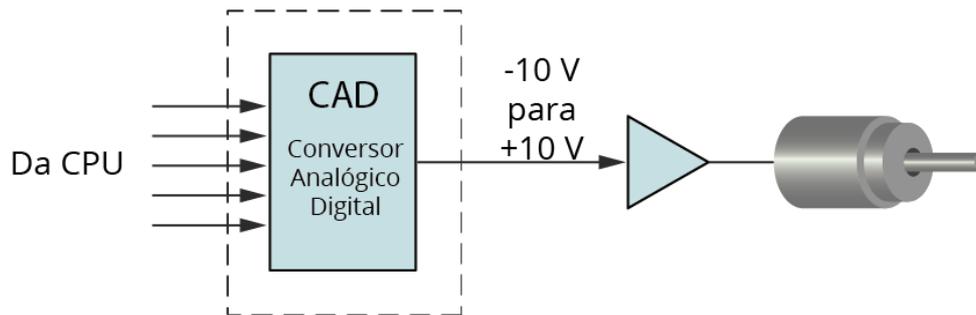
Saída Analógica

Os módulos de saída analógicos permitem que o valor de uma unidade numérica, armazenada e processada pela CPU do CLP, converta-se em tensão ou intensidade de corrente através de uma conversão digital analógica (D/A). Geralmente, quando a entrada analógica foi em nível de tensão "0 a 10V", a saída também é processada em "0 a 10", o mesmo ocorre com intensidade de corrente, conforme mostra a figura 7.

O processo de envio do sinal analógico consta das seguintes etapas:

1. **Conversão digital analógica:** Uma vez que a CPU pegou o sinal contido na memória ele processa esse sinal e em seguida entrega sua resposta para o Conversor Digital Analógico;
2. **Circuitos de amplificação:** O conversor recebe o sinal em baixa potência. Para elevar o sinal a níveis desejados há um circuito eletrônico que amplifica esse sinal;
3. **Proteção eletrônica da saída:** Para evitar sinais de retorno por interferência, uma proteção eletrônica é requisitada a fim de evitar a queima do circuito de saída do CLP.

Figura 07 - Representação de um sinal analógico, oriundo de um sensor de temperatura.



Fonte: CARRILHO (2005)

Módulos de funções especiais

Todas as interfaces (entradas ou saídas) descritas têm a tarefa de converter sinais de entrada em valores aceitáveis à CPU, seja ele digital ou analógico, ou converter os sinais entregues pela CPU em valores convenientes para os atuadores de saída.

Porém, existem processos ou máquinas que requerem tarefas mais complexas, por exemplo, controle de velocidade superior à varredura do equipamento, resolução de equações matemáticas para sincronismo de motores, dentre outros controles. Essas tarefas podem resultar em limitações que são dadas pela falta de capacidade dos módulos encontrados nos CLP's. Para esses tipos de módulos são atribuídas funções especiais, específicas a cada tipo de aplicação. Esses módulos possuem seu processamento embarcado, possibilitando, assim, alto desempenho de trabalho. Podemos citar alguns exemplos, tais como:

- Módulo de entrada de pulsos de alta velocidade:
Permitem conectar ao sistema dispositivos que produzem trem de pulso rápido, para que o CLP possa reagir ou realizar contagens. Por exemplo, em turbinas, tacômetros, etc.;
- Módulo de controle de eixos:
Permite controlar a posição ponto a ponto de servomotores. Por exemplo, em sistema de empacotamento de produtos;
- Módulo de entrada termoacoplada:
Fornecem alta precisão na leitura de temperaturas usando termopare. Podem ser usadas em grau Celsius ou Fahrenheit;

- Módulo de entrada de termoresistencia:
Também usados para medir temperaturas, onde mediante a leitura de um valor de resistência há uma conversão também para graus Celsius, Fahrenheit, Ohm ou valor numérico.

Módulos de expansão

É comum em processos de automação a instalação ocorrer em etapas, tendo em vista que o custo poderá ser significativo a ponto de eleger as etapas de instalação e, em momento propício, ir acrescentando funções ao processo. Através dos módulos de expansão, como mostra a figura 8, a estrutura do sistema pode ser adaptada em função das necessidades de cada aplicação, podendo ser facilmente expandido. Desta forma os custos acabam sendo reduzidos sem perdas de funcionalidades. A combinação de módulos pode ser feita livremente. Essa é uma das grandes vantagens de sistemas automatizados.

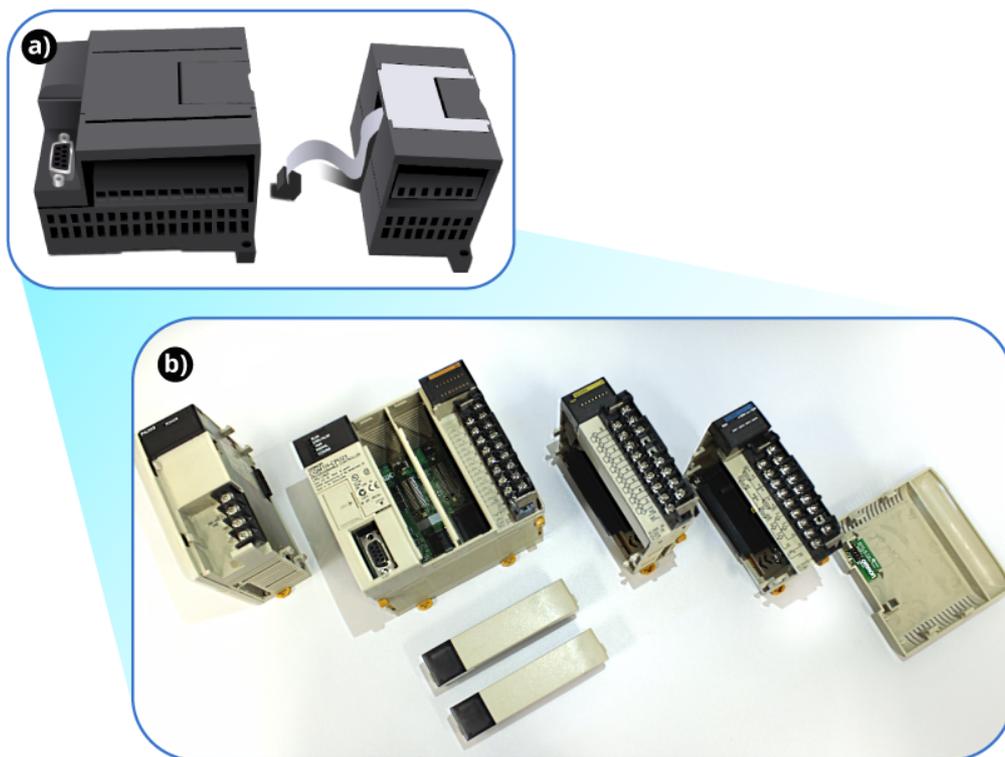


Figura 8a – Unidade básica do CLP simatic S7 juntamente com seu módulo de expansão. **Figura 8b** - Unidade básica do CLP Omron juntamente com seus módulos de expansão.

Fonte: Figura 8a- Adaptado de: <http://w3.siemens.com.br/automation/br/pt/seguranca-de-maquinas/interfaces-de-seguranca/plc-de-seguranca/simatic-s7-200/pages/simatic-s7-200.aspx>

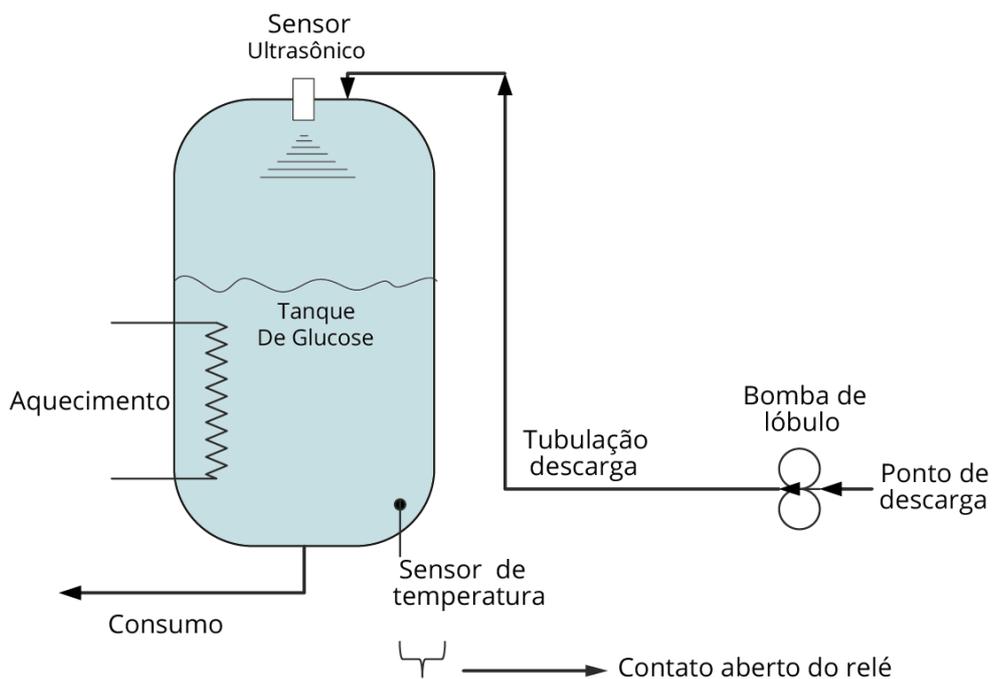
Acesso em: 15 nov. 2015.

Como Selecionar um CLP

Sempre em projetos de automação de máquinas é necessária a “compra” ou aquisição do CLP e comumente nos deparamos com a seguinte pergunta: Qual CLP devo comprar? Qual CLP é o mais apropriado para as funções da máquina que eu quero que sejam desempenhadas? Lembre-se: Você é o responsável pelo projeto, portanto é obrigatório saber como a máquina opera ou irá operar. Para entendermos melhor como funciona essa seleção vejamos o exemplo de projeto a seguir:

Deseja-se projetar um sistema para armazenamento de glucose em um tanque elevado cuja planta do processo é ilustrada na figura 9:

Figura 09 - Detalhe do layout da planta de um tanque de aquecimento.



Fonte: Elaborado pelo Autor (2016)

Na planta temos:

- A bomba para carregamento do tanque de glucose;
- Uma válvula solenoide para consumo dessa glucose;

- Sensor de nível para indicar o nível da glucose;
- Resistência elétrica para aquecimento;
- Sensor de temperatura.

Modo de funcionamento da planta:

1. O carregamento de glucose é feito por meio de uma bomba de lóbulos que é acionada por chave de partida direta. O acionamento acontece quando chega uma "carreta" carregada de glucose e precisa descarregá-la, então o operador "aciona" a bomba;
2. O sensor de nível serve para indicar (em modo de alarme luminoso) que o tanque está cheio ou seco. Ele trabalha com sinal de 4 a 20 mA, ou seja, quando ele envia 4mA o tanque está seco e 20mA o tanque está cheio;
3. O sistema de aquecimento serve para manter a temperatura da glucose em um valor constante (tipo 60°C) e funciona com um sensor tipo termopar em conjunto com uma resistência elétrica. Quando a temperatura da glucose está fora do "set point" a resistência "liga" ou "desliga";
4. E, por último, quando é preciso consumir essa glucose, uma válvula solenoide permite a passagem ou o bloqueio da mesma. Ela está aberta totalmente ou fechada totalmente.

Dessa forma, para o projeto de um sistema de armazenamento de glucose em um tanque elevado, iremos precisar dos seguintes elementos:

- Para o acionamento da bomba precisaremos de **duas entradas digitais** (Liga/desliga) e uma **saída digital (bomba)**;
- Para o alarme do nível: de **uma entrada analógica de 4 a 20mA** para receber o sinal proveniente do sensor de nível e **duas saídas digitais** para os alarmes ("tanque cheio" e "tanque seco");
- Para o controle de temperatura: **uma entrada para termopar** para o sensor de temperatura e **uma saída digital** para liga/desliga da resistência;
- E por último, para consumir a glucose, **duas entradas digitais e uma saída digital** para a válvula solenoide.

- Portanto, para o projeto acima utilizaremos um CLP com no mínimo 6 entradas digitais, 5 saídas digitais, 1 entrada analógica de 4 a 20mA e uma entrada (termoacoplada) para termopar.

Programação de CLP

Hoje em dia, com o avanço da tecnologia, cada vez mais novos fabricantes de CLP's estão se instalando no mercado, e para isso é preciso padronizar a forma como o CLP pode ser programado. Quando falamos de linguagem de programação referimos a diferentes formas que se pode "escrever" o programa do usuário.

Deste modo, de acordo com a norma IEC 61131-3 é definido cinco linguagens de programação:

- Lista de Instruções (IL)
- Texto Estruturado (ST)
- Diagramas de Blocos Funcionais (FBD)
- Funções Gráficas de Sequenciamento (SFC)
- Ladder Diagrams (LD)

A seguir, vamos descrever cada uma dessas linguagens. Vamos lá?

Lista de instruções (IL)

A lista de Instruções é uma linguagem de baixo nível baseado em operações booleanas e cuja aparência é similar à linguagem assembly. Também se pode afirmar que é a representação em forma de texto da linguagem gráfica "Ladder". É a mais antiga das linguagens de programação para CLP's. Era muito usada quando os computadores não tinham telas gráficas.

Nesta linguagem cada instrução se baseia nas definições de álgebra:

LD Início da operação, contato aberto.

LD NOT	Contato em série aberto
AND (Y)	Contato em série aberto
OR (O)	Contato paralelo aberto
AND NOT	Contato série fechado
OR NOT	Contato paralelo fechado
OUT	Bonina de saída do relé
TMR	Temporizador
CNT	Contador

Como exemplo de programação veja o acionamento de uma lâmpada:

LD	Botão partir
OR	Lâmpada
AND NOT	Botão parar
OUT	Lâmpada

Texto Estruturado (ST)

Como o próprio nome já diz, o Texto Estruturado é uma linguagem de alto nível, é estruturada em blocos e sua sintaxe é semelhante à linguagem Pascal. Geralmente é usada em aplicações que envolvem manipulação de dados, aplicações matemáticas, inteligência artificial, entre outros.

Como exemplo, vejam como fica a programação utilizando a mesma lâmpada para acionamento:

IF (Botão partir **OR**) **AND NOT** Botão parar **THEN**

Lâmpada := TRUE

ELSE

Lâmpada := FALSE

END_IF

Diagrama de blocos funcionais

A linguagem de Diagramas de Blocos Funcionais permite ao usuário construir procedimentos mediante a conexão de blocos funcionais predefinidos. Essa linguagem também descreve uma função entre as variáveis de entrada e variáveis de saídas, semelhante a um conjunto de blocos. Esses blocos são conectados por *linhas de conexão* igual a um circuito elétrico.

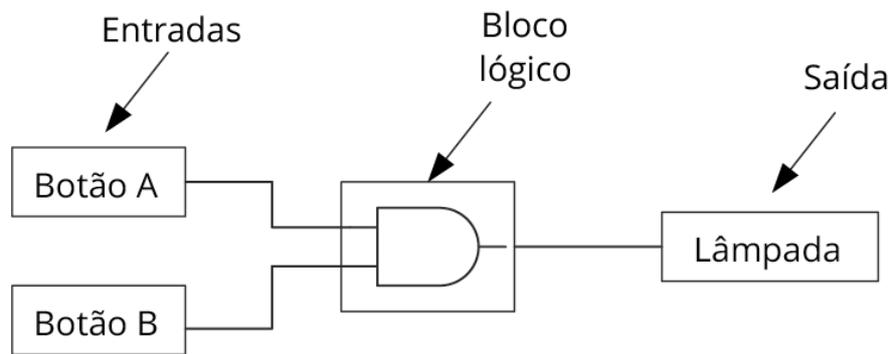
Geralmente os fabricantes utilizam simbologia de blocos lógicos (funcionais) para representar a função. Esse “diagrama” de funções lógicas se assemelha à sintaxe com portas lógicas onde um “algoritmo” processará entradas e saídas do CLP.

Características

- As saídas dos blocos de funções não se conectam entre si;
- A avaliação de uma conexão será validada logo após a sua interligação antes da conexão seguinte.

Como exemplo, conforme figura 10, vejamos o acionamento da mesma lâmpada mostrado anteriormente, porém agora a lâmpada só acenderá se o “Botão A” e “Botão B” estiverem selecionados:

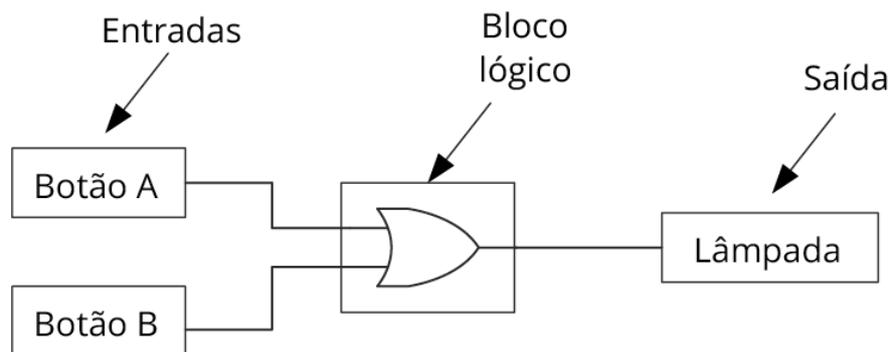
Figura 10 - Circuito lógico para acionamento de uma lâmpada utilizando Diagrama de Blocos Funcionais.



Fonte: Elaborado pelo Autor (2016)

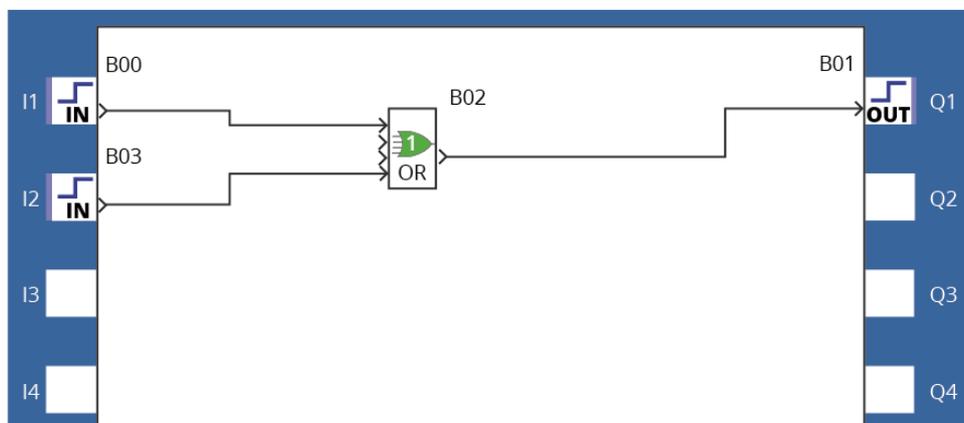
Vejam agora mudando a condição de acendimento, ou seja, agora pressionando Botão A ou Botão B, a lâmpada acende, observe as figuras 11 e 12:

Figura 11 - Circuito lógico para acendimento de uma lâmpada.



Fonte: Elaborado pelo Autor (2016)

Figura 12 - Circuito lógico para acendimento de uma lâmpada utilizando a ferramenta de programação Zelio soft da Schneider.



Fonte: Elaborado pelo Autor (2016)

Funções Gráficas de Sequenciamento (SFC)

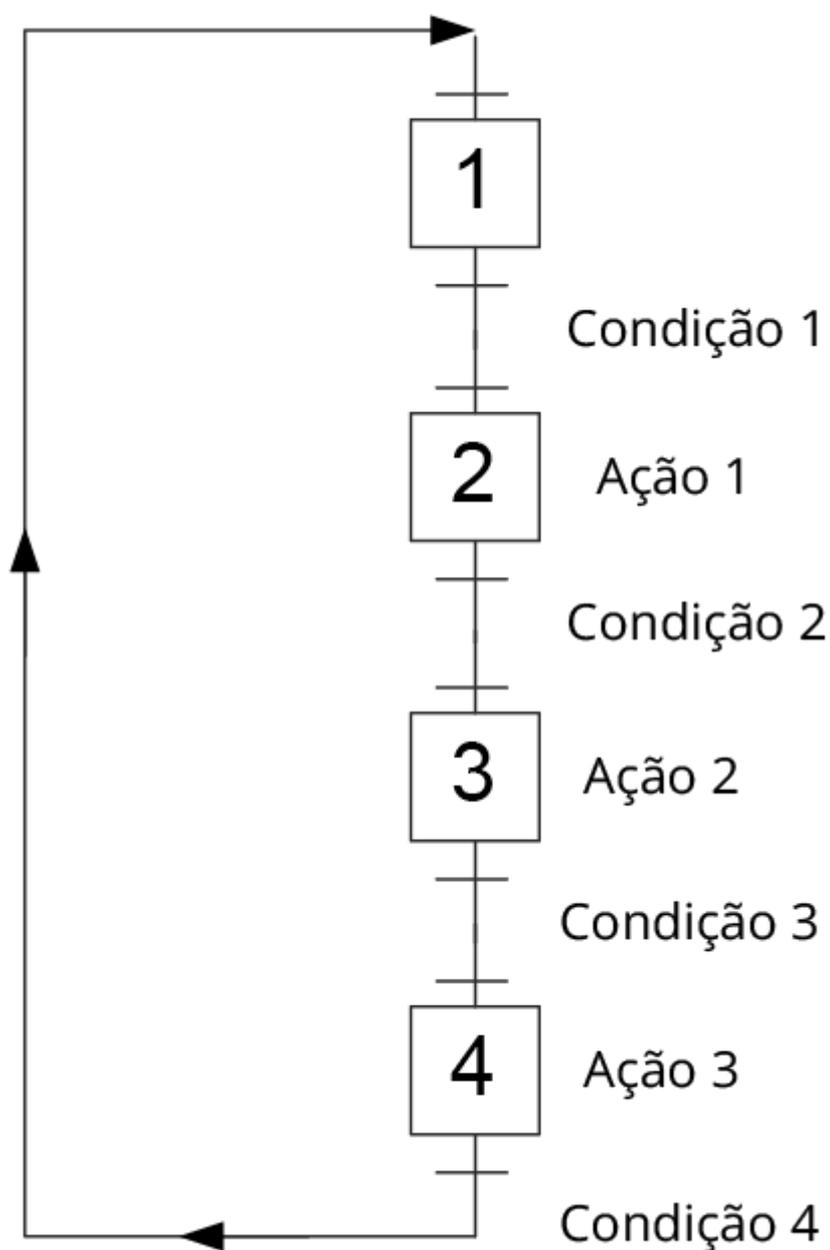
A SFC Surgiu na França nos anos 1970, através de um grupo de desenvolvedores de CLP's, sendo também chamada de linguagem Grafcet. É baseada em linguagem de gráficos, os quais fornecem uma representação diagramática de sequências de controle.

As características de programação SFC contêm três principais elementos que organizam o programa de controle:

- Etapas
- Transições
- Ações

O programa irá ativando cada uma das etapas e desativando a anterior, conforme vai processando cada uma das condições. O acionamento se realizará em função da etapa ativa as quais estão associadas, de acordo com a figura 13:

Figura 13 - Ilustração do diagrama de operação da linguagem de programação SFC.



Fonte: Elaborado pelo Autor (2016)

Diagrama Ladder

O Diagrama Ladder foi introduzido nos Estados Unidos e dentre os tipos de linguagens vistos anteriormente a linguagem Ladder, atualmente, é a mais usada por “programadores de CLP’s”, devido à sua simplicidade de identificação de

parâmetros, fácil assimilação da “lógica de programação” e por se basear em *esquemas elétricos de contatos (clássicos)*, os quais são utilizados na lógica a relé. É por esse motivo que a partir de agora será dada total atenção a esta linguagem.

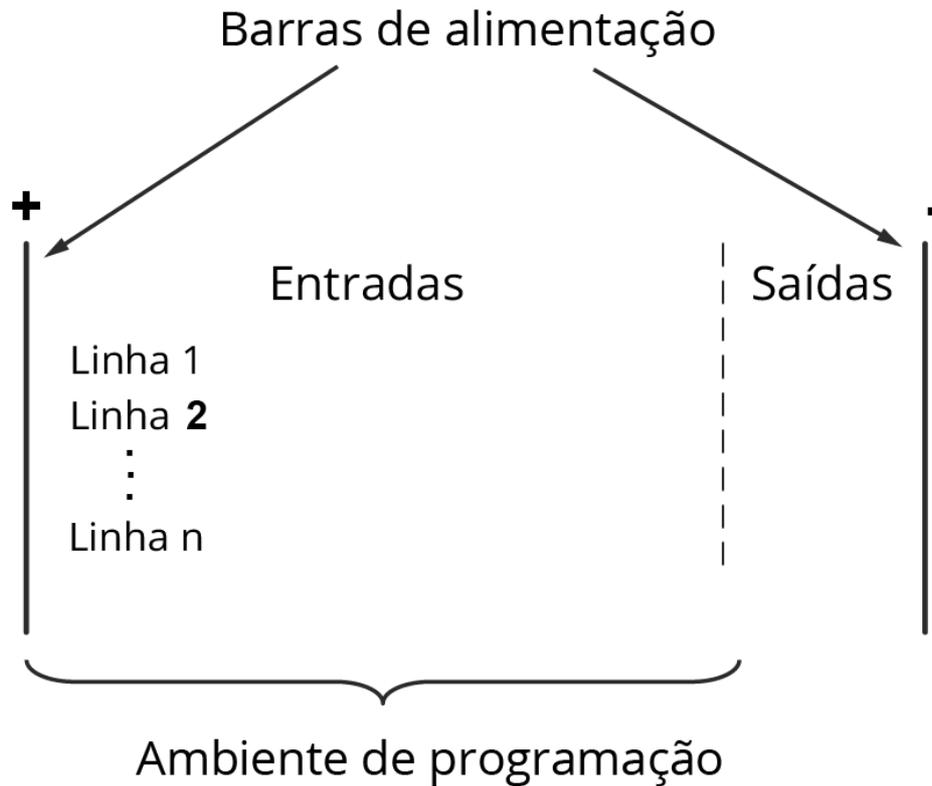
Desse modo, com os conhecimentos técnicos adquiridos em nossas aulas, é muito fácil adaptar-se à programação em Ladder. Uma das vantagens é a norma NEMA no que rege à simbologia dos contatos elétricos que são empregados pelos fabricantes de CLP. Sendo assim, uma vez que o programador se “*familiarizar*” com a linguagem ladder, estará apto para programar todo tipo de CLP, uma vez que os diagramas em Ladder são esquemas de uso comum para representar a “lógica de controle” de sistemas automatizados. Então, vamos lá!

Ambiente de programação em Ladder

O ambiente de programação em Ladder é formado por duas barras verticais (barras de alimentação), onde serão inseridas linhas de programação (de acordo com a figura 14). Essas linhas são onde o circuito de controle é montado de acordo com a lógica de funcionamento da máquina e suas funções.

Toda a programação em Ladder de determinado equipamento estará constituído entre as barras de alimentação.

Figura 14 - Características do ambiente de programação em linguagem Ladder;



Fonte: Elaborado pelo Autor (2016)

O “fluxo lógico” se dá da esquerda para a direita e de cima para baixo.

O diagrama de programação de um processo automatizado pode conter uma ou várias linhas de programação. Vai depender da lógica de programação utilizada.

Entrada

É o local onde são inseridos botoeiras, sensores, fim de curso, contatos de relés, contatos de temporizadores, contatos de contadores, etc. Obedecendo, claro, à lógica de programação.

Nos CLP's, as entradas podem ser representadas pelas seguintes nomenclaturas:

- $E_0, E_1, E_2, \dots, E_n$. (Entrada 1, Entrada 2, ...; Entrada n)
- I_1, I_2, \dots, I_n . (Input 1, Input 2, ..., Input n)
- $\$I_{\{0,0\}}, I_{\{0,1\}}, \dots, I_{\{0,n\}}$. (Input 01, Input 02, ..., Input 0n)

- DI_1, DI_2, \dots, DI_n . (Digital Input 1, Digital Input 2, ..., Digital Input n)
- AI_1, AI_2, \dots, AI_n . (Analog Input 1, Analog Input 2, ..., Analog Input n)

Observações

Se um CLP tem 4 entradas digitais, então será E_0, E_1, E_2, E_3 ou I_1, I_2, I_3, I_4 . E assim por diante. Depende de qual nomenclatura o fabricante optou.

Saídas

É o local onde são inseridos os elementos finais de um “sistema de acionamento”, ou seja, é o local onde são alimentadas as “bobinas de saídas” dos dispositivos utilizados na programação.

Bobinas são as variáveis independentes dos relés, contadores, etc. Se a bobina for energizada, seus contatos mudam de estados.

As saídas podem ser representadas pelas seguintes nomenclaturas:

- S_1, S_2, \dots, S_n . (Saída 1, Saída 2, ...; Saída n)
- . (Output 1, Output 2, ..., Output n)
- Q_1, Q_2, \dots, Q_n . (Quit 1, Quit 2, ..., Quit n)
- DO_1, DO_2, \dots, DO_n . (Digital Output 1, Digital Output 2, ..., Digital Output n)
- AO_1, AO_2, \dots, AO_n . (Analog Output 1, Analog Output 2, ..., Analog Output n)

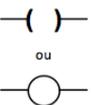
Observações

Da mesma maneira das entradas, se um CLP tem 4 saídas digitais, então será S_1, S_2, S_3, S_4 ou Q_1, Q_2, Q_3, Q_4 . E assim por diante. Depende de qual nomenclatura o fabricante optou.

Simbologia

O CLP não entende terminologias como chaves, relés, etc., porém interpreta simbologias de entradas, bobinas de saídas, entre outros. Devido a isso a linguagem Ladder converte a simbologia em uma linguagem que a CPU entenda.

A tabela a seguir ilustra os principais contatos utilizados na linguagem Ladder:

Símbolo	Nome	Descrição
	Contato aberto (NA)	Contatos normalmente abertos são aqueles contatos que em seu estado de repouso não "deixa" conduzir energia;
	Contato Fechado (NF)	Contatos normalmente fechados são aqueles contatos que em seu estado de repouso "deixa" conduzir energia;
	Bobina	É ativada quando a combinação de contatos, que há na entrada, permite. Sua análise é a seguinte: "1" para bobina energizada e "0" para bobina desenergizada.
	Bobina (SET)	Uma vez ativada não pode ser desativada, a não ser que seja dado um comando de "RESET".

Símbolo Nome**Descrição**

—(R)— Bobina (RESET) Permite desativar uma bobina "SET" previamente ativada.

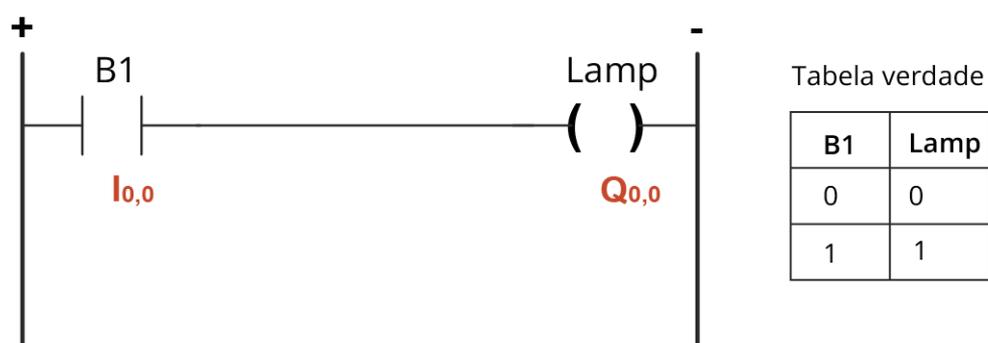
É importante lembrar que o estado ativado (ligado) de uma entrada ou saída corresponde a "1". Já para o estado desativado (desligado) de uma entrada ou saída corresponde a "0".

Lógica de programação Ladder

Para o programador de CLP é extremamente necessário que o mesmo tenha sempre um "raciocínio lógico" para entendimento e desenvolvimento de um programa. Deste modo, as funções lógicas são estudadas em todos e quaisquer elementos e a combinação entre os contatos NA e NF servem de importante orientação para o projetista e programador de circuitos lógicos. Vejamos um exemplo de programação, bem como sua "tabela verdade" dos estados lógicos das entradas e saídas, para clarear nossas ideias:

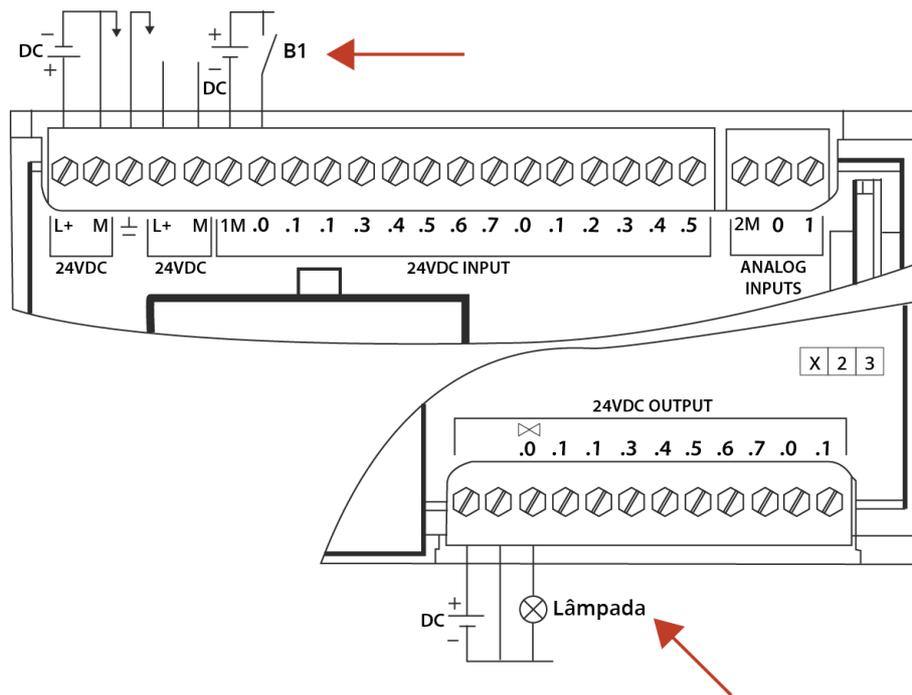
Veja o exemplo de um acionamento simples de uma Lâmpada (*Lamp*) através de uma botoeira de impulso (B_1) conforme figura 15, bem como seu esquema de ligação para esta lâmpada conforme figura 16:

Figura 15 - Ambiente de programação em Ladder para acionamento simples de uma lâmpada.



Fonte: Elaborado pelo Autor (2016)

Figura 16 - Montagem do hardware do CLP para o acionamento simples de uma lâmpada.



Fonte: Elaborado pelo Autor (2016)

Observações

Sem muita complicação, vemos que a botoeira é representada pelo bloco de contato aberto (NA), que por sua vez está interligada com a entrada digital ($I_{0,0}$). Já a lâmpada está representada por uma saída digital do CLP que também está representada pela bobina ($Q_{0,0}$).

De acordo com a tabela verdade no estado “0” da botoeira a saída também é “0”, ao se mudar o estado da botoeira para “1” a saída também muda para “1”.

Portanto, Lê-se da seguinte forma:

“Ao pressionar a botoeira B1 o contato permite conduzir energia alimentando, assim, a bobina Q1, que acenderá a lâmpada (estado 1). Ao soltar a botoeira a energia deixa de ser conduzida e a lâmpada apagará (estado 0).”

Contudo, para programar deve-se interpretar. Interpretar significa obedecer a sequência de funcionamento do sistema, senão vejamos:

- Como se faz para manter a saída (Lamp) ativada?
- Se impormos condições tais como: a necessidade de pressionar não uma, mas duas botoeiras para ativar a saída, como a colocamos no ambiente de programação?
- Uma vez ativada, como desativamos a saída (“desligamos a lâmpada”)?

Para **viabilizar** estas respostas é que precisamos entender a lógica de programação. Para isso é estudada a “lógica booleana”. É a combinação entre os contatos NA e NF que servem de importante orientação para o programador de circuitos lógicos.

A seguir estudaremos um pouco sobre algumas dessas funções lógicas.

Função “E” (AND)

É a associação de botoeiras NA em série, ou seja, para que o evento ocorra é preciso que as botoeiras estejam no seu estado ativo “1”, caso contrário a saída não será ativada. Observe a programação para acendimento de uma lâmpada com a função “E”, conforme a figura 17:

B1	B2	Lamp
-----------	-----------	-------------

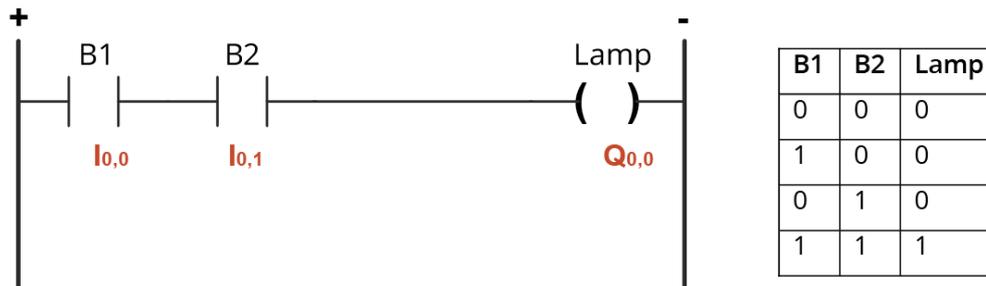
0	0	0
---	---	---

0	1	0
---	---	---

1	0	0
---	---	---

1	1	1
---	---	---

Figura 17 - Sistema de acendimento simples de uma lâmpada usando duas botoeiras B1 e B2.



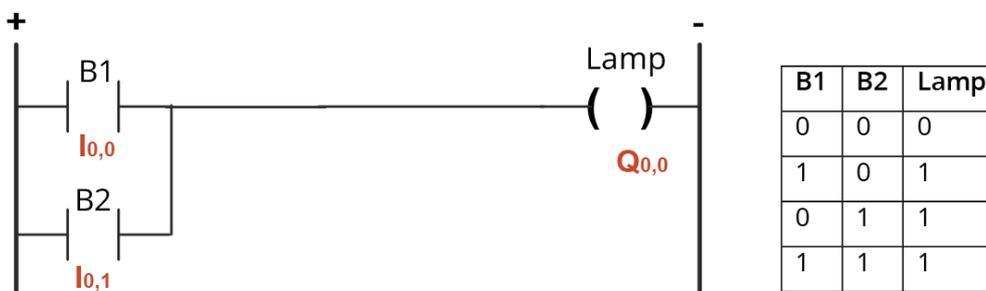
Fonte: Elaborado pelo Autor (2016)

Observando o programa e a tabela verdade se observa que para “acionar” a saída é preciso pressionar as duas botoeiras simultaneamente (ao mesmo tempo), “B1 e B2”, caso contrário a saída não será ativada.

Função “OU” (OR)

É a associação de botoeiras NA em *paralelo*, ou seja, para que o evento ocorra é preciso que qualquer uma das botoeiras esteja no seu estado ativo “1”, para que a saída seja ativada. Observe a programação de um exemplo de aplicação para acendimento de uma lâmpada, conforme mostra a figura 18:

Figura 18 - Sistema de acendimento simples de uma lâmpada usando duas botoeiras B1 e B2.



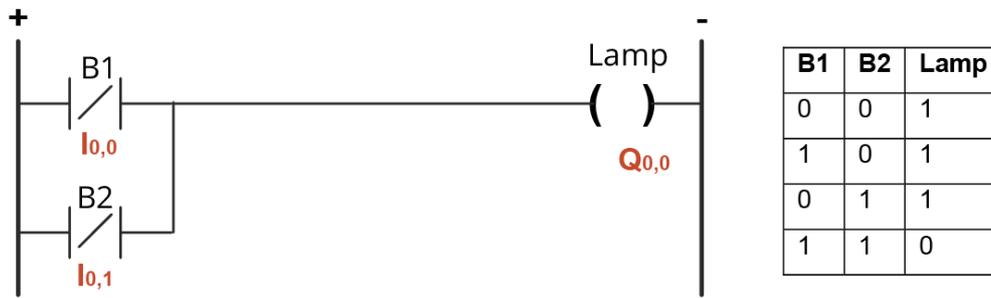
Fonte: Elaborado pelo Autor (2016)

Observando o programa e a tabela verdade percebe-se que para “acionar” a saída basta pressionar qualquer uma das botoeiras, “B₁ ou B₂”.

Função “Não E” (NAND)

É a associação de botoeiras NF em *paralelo*, ou seja, o evento também inicia com seus estados ativados “1”. Para que a saída seja desativada é preciso pressionar “B₁ e B₂”. Observe o exemplo de programação, da figura 19:

Figura 19 - Sistema de acendimento simples de uma lâmpada usando duas botoeiras B1 ou B2 na lógica NAN.



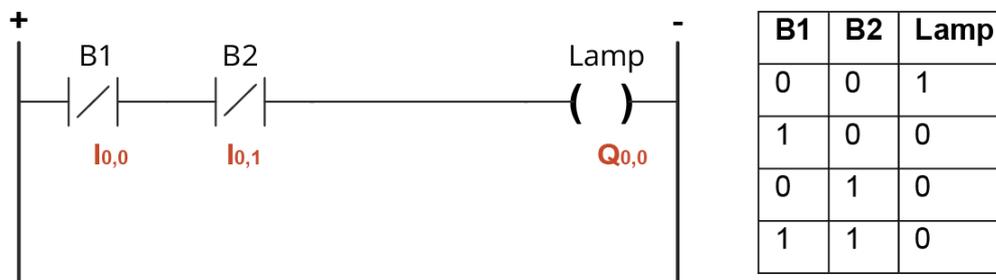
Fonte: Elaborado pelo Autor (2016)

Analisando a aplicação se observa que no estado de repouso das botoeiras a saída está ativa "1" e que pressionando "B1 e B2" a saída será desativada "0".

Função "Não OU" (NOR)

É a associação de botoeiras NF em *série*, ou seja, o evento já inicia com seus estados ativados "1". Para que a saída seja desativada basta pressionar " B_1 ou B_2 ". Observe o exemplo de programação da figura 20:

Figura 20 - Sistema de acendimento simples de uma lâmpada usando duas botoeiras B1 e B2 na lógica NOR.



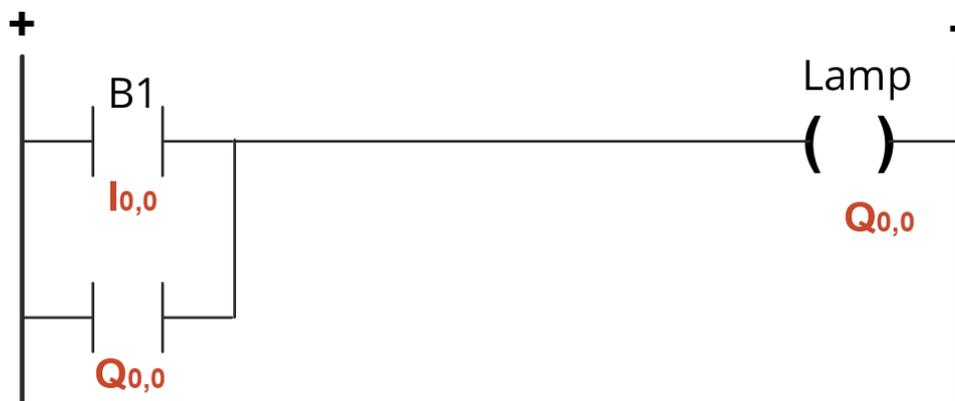
Fonte: Elaborado pelo Autor (2016)

Analisando o programa se observa que no estado de repouso das botoeiras a saída está ativa "1" e que pressionando "B1 ou B2" a saída será desativada "0".

Circuito de selo (retenção)

É uma combinação básica entre os elementos para manter uma saída ligada quando são utilizadas botoeiras NA. Ou seja, como mantemos a saída (lamp) ativada mesmo após soltarmos a botoeira? Lembre-se que para toda bobina de saída de elementos pertencentes ao circuito tem-se seus **contatos auxiliares** NA e NF, serão eles que farão a “retenção do sinal”. Como exemplo, observe o acionamento da lâmpada conforme a figura 21:

Figura 21 - Sistema de acendimento simples de uma lâmpada usando a retenção da bobina.



Fonte: Elaborado pelo Autor (2016)

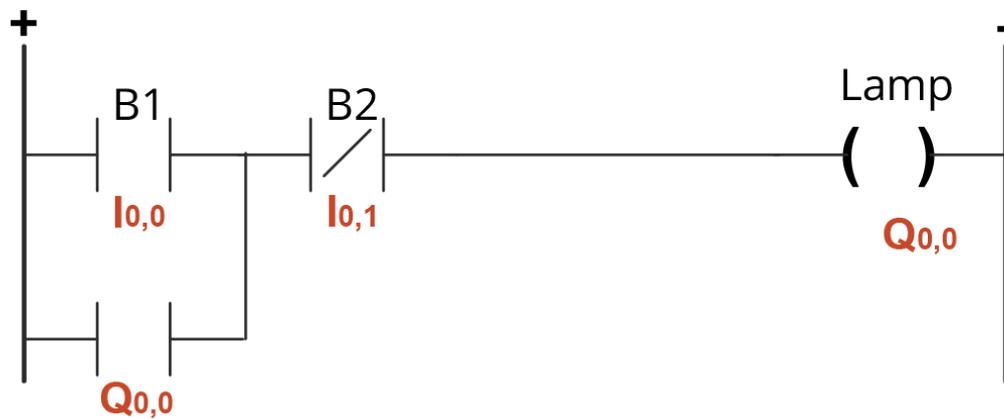
Para a aplicação da figura 21, lê-se:

“Ao pressionar B_1 , a saída é ativada, que por sua vez, fecha o contato NA da bobina $Q_{0,0}$ e mesmo soltando a botoeira B_1 a bobina continuará energizada através do contato de selo (retenção)”.

E como a saída (Lamp) é desligada?

Bom! Para desligar uma saída que esteja alimentada pela retenção se usa uma segunda botoeira, desta vez, NF para realizar o desligamento do sistema. Observe o exemplo da figura 22:

Figura 22 - Sistema de acendimento simples de uma lâmpada usando a retenção da bobina e com desligamento por botoeira.



Fonte: Elaborado pelo Autor (2016)

Para a aplicação da figura 22, lê-se:

“Ao pressionar B_1 a energia é conduzida para botoeira B_2 . Como a mesma está com seu contato fechado permite a passagem do sinal para ativar a saída (Lamp), que em seguida fecha seu contato de retenção e ao soltar a botoeira B_1 o sistema fica ativo pela retenção e pelo contato NF da botoeira B_2 . Para desligar o sistema basta pressionar B_2 , que interrompe a passagem do sinal para a bobina, desligando tudo.”

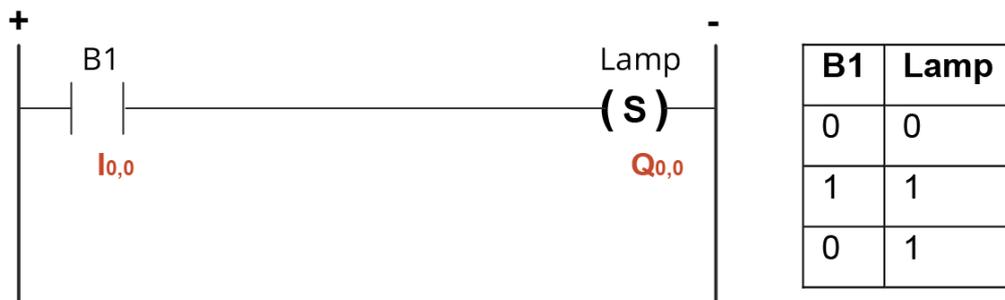
Observe também que B_2 está em série com a bobina, o que facilita o entendimento de interrupção do sinal.

Esse método de acionamento apresentado na figura 22 é uma das metodologias mais importantes em acionamentos.

Instrução SET

A instrução SET liga uma saída e a mantém ligada mesmo que a alimentação da entrada seja retirada, observe na aplicação da figura 23:

Figura 23 - Sistema de acendimento simples de uma lâmpada usando uma bobina SET.

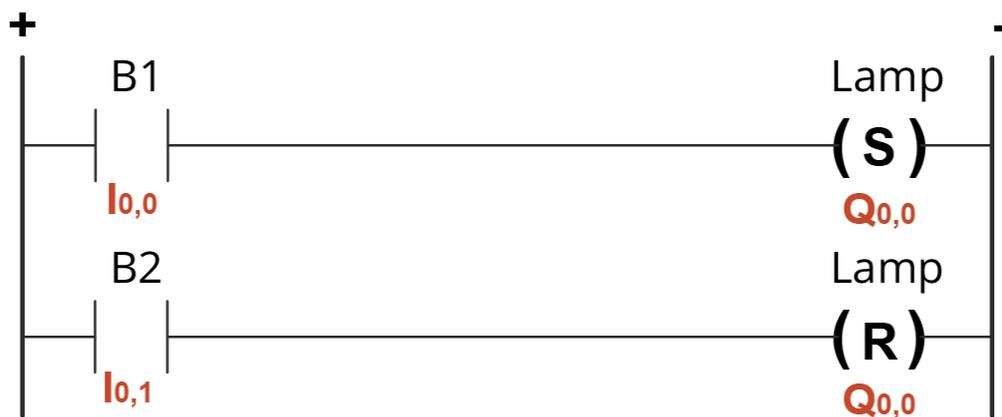


Fonte: Elaborado pelo Autor (2016)

Instrução RESET

A instrução RESET desliga uma saída que estava em SET, observe no acionamento da figura 24:

Figura 24 - Sistema liga/desliga de uma lâmpada usando bobinas SET e RESET.



Fonte: Elaborado pelo Autor (2016)

Observações

- Sempre que se dá um "RESET" é aconselhável usar botoeiras de impulso ou "contatos sem selo";
- Quando se utilizam as funções SET e RESET é preciso muita atenção, pois nestas funções é comum o "erro do programa" pelo fato da bobina continuar ativada ou desativada. Na maioria das programações é suficiente a utilização da forma convencional das saídas.

Portanto, apesar de existirem essas linguagens de programação para CLP's, nem todos os fabricantes disponibilizam a opção de programar em uma das cinco opções.

Cada fabricante enfrenta uma verdadeira maratona a fim de simplificar cada vez mais a interface de programação para que cada vez mais técnicos da área possam ser atraídos pela sua metodologia.

Basta observar que se um técnico prefere programar em lista de instruções, então ele terá que procurar CLP's que sejam compatíveis com essa linguagem de programação.

Resumo

No seguimento do curso foi apresentado como perceber as características básicas dos CLP's e, tão importante quanto, a sua programação. Apresentamos a forma de programar um controlador lógico programável bem como as linguagens de programação empregadas nestes controladores. Vimos que estas linguagens permitem ao usuário inserir lógicas de controle utilizando sintaxes pré-estabelecidas.

Após o dimensionamento do CLP é o programa que vai definir o automatismo que é constituído por uma série de instruções e funções onde são operados os bits de memória. Essas instruções e funções serão introduzidas na memória do CLP, através de um software específico para o respectivo CLP via um terminal de programação, que em geral é um computador.

Autoavaliação

1. Em uma empresa de logística contém uma Esteira Transportadora composta de cinco motores, onde todos os motores são ligados em partida direta. Faça uma pesquisa e dimensione um CLP para fazer esta aplicação (5 partidas diretas para os motores). Considere que serão utilizados na aplicação:
 - Para cada motor deve haver 2 botoeiras (uma para ligar e outra para desligar).

Referências

CAPELI, Alexandre. **CLP: Controladores Lógicos Programáveis na Prática**. 1. edição. [S.l.]: Ed. Antenna, 2007.

CARRILHO, Eduardo. **Automação de Sistemas e Instrumentação Industrial / Controladores Lógicos Programáveis CLPs**, 2005. Notas de Aula disponível em: <<http://aquarius.ime.eb.br/~aecc/>>

GEORGINI, Marcelo. **Automação aplicada: descrição e implementação de sistemas sequenciais com PLC's**. 3. ed. [S.l.]: Ed. Érica, 2007.

MAITELLI, André L. **Controladores lógicos programáveis**. Disponível em: . Acesso em: 19 out. 2015.

OMRON. CLP de médio porte CJ1 e CJ2. Disponível em: . Acesso em: 25 nov. 2015.

SIEMENS. Sistemas de automação industrial. Controlador SIMATIC S7 (PLC). Disponível em: . Acesso em: 23 nov. 2015.

WEG, CLP's e controle de processos. Clic 02. Disponível em: . Acesso em: 19 out. 2015.