

Introdução à Automação Industrial

Aula 03 - Princípios de funcionamento do CLP

Apresentação

Já conhecemos um pouco sobre os principais níveis da automação industrial. A partir de agora, nossas aulas irão se concentrar nos três primeiros níveis, abordando tópicos como CLP (Controlador Lógico Programável) e sistemas supervisórios. O assunto sobre redes industriais será visto mais adiante numa disciplina específica. “Trocando em miúdos”, vamos trabalhar com as tecnologias e dispositivos mais próximos do processo a ser controlado. Hoje, começaremos com um dispositivo eletrônico conhecido como CLP. Ele apresenta inúmeras vantagens frente aos seus concorrentes, dentre elas, o seu tamanho reduzido e a robustez física. Além disso, a flexibilidade em atuar nos dois segmentos da automação torna esse dispositivo uma ferramenta muito poderosa na indústria. Vamos trabalhar de duas formas para construir os diagramas Ladder. A primeira delas se baseia na conversão de circuitos elétricos em Ladder. Apesar de ser um procedimento mais direto, requer do usuário uma experiência prévia sobre circuitos de comando (projeto e/ou análise). A segunda, mais comum, utiliza o método do GRAFCET, apresentado na aula passada. Ainda nesta aula, serão propostas duas aplicações muito comuns na indústria: o avanço e recuo de um cilindro de dupla ação e o controle de nível em um reservatório. A ideia é introduzir novas técnicas de comando e novos elementos da linguagem de programação Ladder, dentre eles, o bloco temporizador.

Objetivos

- Descrever o Controlador Lógico Programável (CLP) e identificar as suas aplicações na indústria;
- Identificar o hardware do CLP
- Listar as principais linguagens de programação do CLP.
- Conhecer os circuitos elétricos de comando.
- Distinguir os elementos principais de um diagrama Ladder.
- Descrever o funcionamento de um programa em Ladder
- Apontar os princípios básicos de programação em Ladder e dar exemplos de diagramas
- Aprofundar o conhecimento sobre o método do GRAFCET
- Desenvolver algumas aplicações em Ladder: o avanço e recuo de um cilindro industrial, e o controle de nível em um reservatório

Relé eletromecânico: o antecessor do CLP

Por muitos anos, um dispositivo eletromecânico conhecido como relé (Figura 1) dominou a automação de processos do tipo discreto. De construção simples e fácil manipulação, ele foi (e ainda é) muito utilizado em diversas atividades no meio industrial, desde o acionamento de máquinas elétricas, até a implementação de processos sequenciais mais elaborados. Sem ele, a indústria não teria avançado tão significativamente como avançou e possivelmente ainda estaria presa a técnicas rudimentares de produção. Apesar do seu belo histórico, o relé apresenta alguns problemas na linha de produção. Composto por partes mecânicas móveis, o seu desgaste prematuro é inevitável. Em fábricas de cimento, por exemplo, a poeira está sempre presente e com o tempo provocava falhas em dispositivos como o relé. Além disso, a complexidade de muitas aplicações industriais exige a instalação de painéis metálicos com dezenas ou centenas de relés, gerando aquecimento e produzindo uma quantidade enorme de fios e conexões. Por isso, é muito complicado realizar modificações em um sistema com a presença de muitos relés.

Figura 01 - Exemplo de um relé industrial.

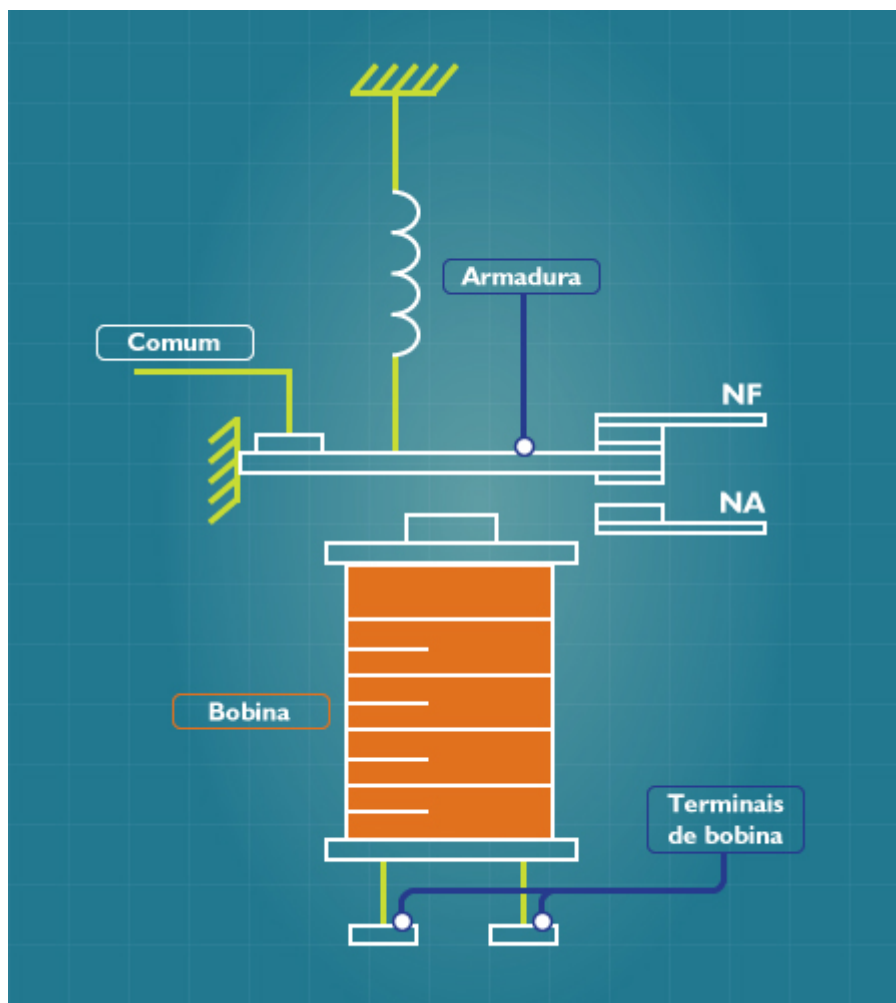


Fonte: <http://2.bp.blogspot.com/-rblyrflqV-U/UUr0Fmndqdl/AAAAAAAAQA/ez7c-wEXPug/s1600/455px-RELEcomat.jpg> Acesso em: 24 abr. 2014

Para exemplificar essa dificuldade, imagine que uma pessoa deseja mudar a posição de algumas tomadas em casa. O primeiro passo seria desligar a chave geral da residência, o que não é muito agradável nos dias de hoje, pois vários eletrodomésticos seriam desligados. Chato, né? Se a fiação for aparente, ou seja, instalada sobre as paredes, a reconfiguração é mais simples e menos demorada. Porém, se a fiação for interna, o proprietário vai ter uma pequena dor de cabeça, pois algumas paredes serão quebradas para se ter acesso à fiação. O grau de dificuldade dessa manutenção é semelhante ao apresentado pela reconfiguração de relés numa indústria de grande porte. Além de paralisar a produção (gerando perda de receita), é preciso lidar com uma quantidade significativa de fios e conexões. A documentação com a descrição de todas as ligações e conexões do sistema deve estar bem atualizada e ser de fácil acesso. Caso contrário, a identificação dos relés pode se tornar um pesadelo, aumentando ainda mais o tempo de parada e os gastos com a manutenção. Por fim, os relés apresentam um alto consumo de energia e devem ser evitados em ambientes com a presença de gases inflamáveis, pois o seu funcionamento pode produzir faíscas.

O principal elemento interno do relé corresponde à sua bobina, responsável por abrir ou fechar os contatos principais e auxiliares desse dispositivo. Considere o esquema da Figura 2. Quando alimentada, a bobina cria um campo magnético (semelhante ao de um ímã) que atrai um elemento metálico chamado de armadura. Este, por sua vez, ao se movimentar, conecta o terminal comum (C) a um dos terminais de saída do relé. Se o contato estiver naturalmente ligado ao terminal C (ou seja, sem a alimentação da bobina), o contato é chamado de NF (Normalmente Fechado). Caso contrário, ele é chamado de NA (Normalmente Aberto). Ao cessar a alimentação, a armadura volta à sua posição inicial pela ação de uma mola. Note que não existe conexão entre o circuito da bobina e o circuito que vai ser conectado aos terminais C, NA e NF. Dessa forma, é possível separar os circuitos de comando (baixa potência) dos circuitos de alimentação de carga (alta potência).

Figura 02 - Estrutura interna de um relé eletromecânico.



Fonte: http://4.bp.blogspot.com/-vF6oug55Oio/TXOOMi0sFFI/AAAAAAAAAPA/8aE8TmVzl2g/s1600/Reles_01.jpg Acesso em: 24 abr. 2014

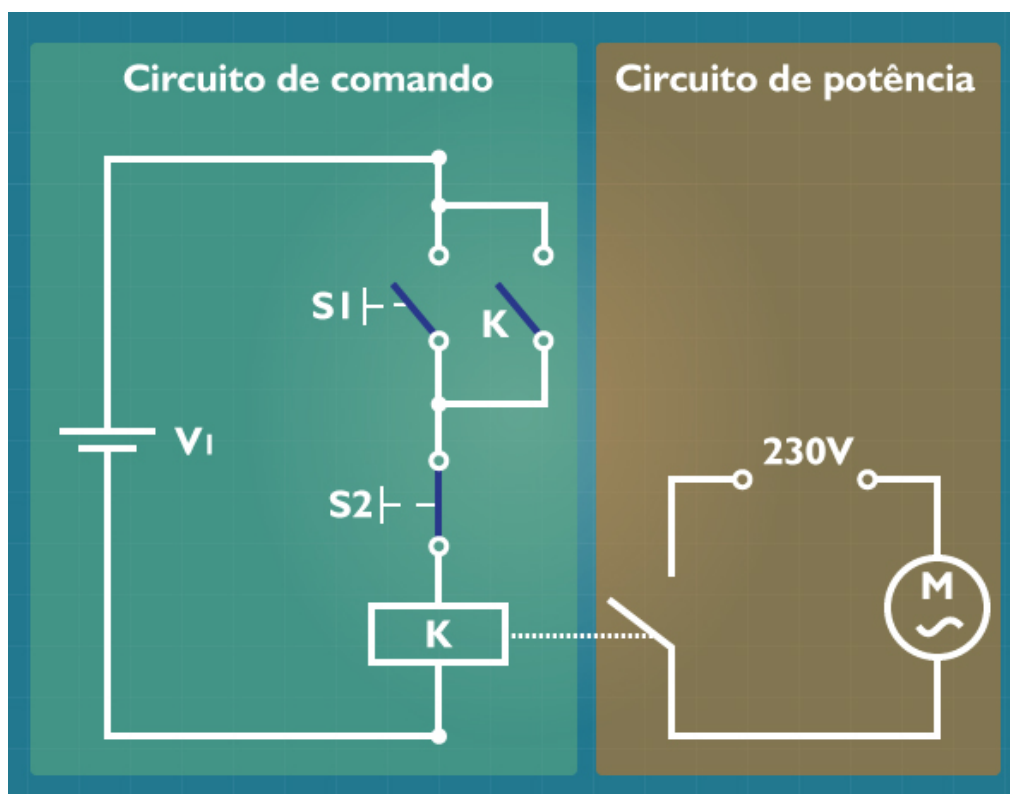
Atividade 01

1. Pesquise sobre os diversos tipos de relé. Apresente as suas características básicas e as respectivas aplicações na indústria. Liste alguns fabricantes de cada tipo.

Exemplo de uma aplicação com relé

Suponha que se deseja ligar e desligar uma determinada carga (um motor, por exemplo) através de duas botoeiras diferentes, denominadas de S1 e S2. Ao se pressionar S1, a carga deve ser ligada, permanecendo assim até que a botoeira S2 seja pressionada. Para tal, podemos utilizar um relé eletromecânico com dois contatos do tipo NA, um para atuar no circuito de comando e outro para conectar a carga com a fonte de alimentação alternada. É interessante ressaltar que os dois contatos NA fecham ao mesmo tempo, particularmente, no instante em que a bobina K é energizada. Note que, apesar de isolados eletricamente, eles são controlados pela mesma bobina K. A Figura 3 apresenta o circuito de comando e de potência para essa aplicação.

Figura 03 - Exemplo de um circuito de comando e de potência utilizando relés.



Fonte: Corradi Júnior

Com base no circuito apresentado, observe que a botoeira S1 está em paralelo com um dos contatos NA do relé (ver chave K na Figura 3). Quando pressionamos S1, a bobina K do relé é energizada, fechando os dois contatos NA, sendo um deles,

o da carga. Ao liberar S1, a bobina K permanece energizada (graças a chave K), mantendo, assim, a carga também energizada. O contato K é conhecido como contato de selo ou retenção. Essa situação só é alterada quando a botoeira S2 for pressionada. Nesse caso, a alimentação da bobina é interrompida, resultando na abertura dos dois contatos NA, um do circuito de comando (chave K) e outro do circuito de potência (chave da carga).

O controlador lógico programável (CLP)

Com o avanço da tecnologia no final da década de 1960, principalmente com o desenvolvimento do transistor na forma de CI (Circuito Integrado), foi possível a criação de dispositivos eletrônicos mais sofisticados, como o controlador Lógico Programável, CLP (ver Figura 4). Desde então, esse elemento passou a substituir os relés e outros dispositivos eletromecânicos na linha de produção. Apesar de mais complexo, o CLP apresenta inúmeras vantagens em relação ao seu antecessor: maior confiabilidade, menor quantidade de fios e conexões, reconfiguração fácil, tamanho compacto, robustez física, inexistência de faíscas e documentação sempre atualizada.

Figura 04 - CLP de pequeno porte do fabricante Allen-Bradley.



Fonte: http://literature.rockwellautomation.com/idc/groups/literature/documents/pp/2080-pp004_-en-p.pdf Acesso em: 24 abr. 2014

Para que esse dispositivo realize uma dada tarefa, é necessário que o usuário desenvolva um programa e depois “carregue-o” no CLP. Tal procedimento fornece diversas vantagens ao usuário. Em primeiro lugar, o processo de programação já realiza uma documentação prévia do sistema, pois toda a lógica de comando e as respectivas conexões estarão descritas no código. Em segundo, diversas alterações físicas nas conexões e fiação podem ser evitadas. Para exemplificar, considere o circuito de comando apresentado na Figura 3. Desde que as botoeiras S1 e S2 estejam conectadas ao CLP, podemos implementar qualquer lógica de comando através de um programa. Caso seja necessária uma reconfiguração (por exemplo, ligar o motor através de S2, ao invés de S1), é preciso apenas que o usuário altere o programa original e depois carregue-o novamente no CLP. Nenhuma alteração física é necessária.

Mas, nem tudo é vantagem. A complexidade do CLP necessitava de uma mão de obra mais qualificada, particularmente no que se refere à programação do dispositivo. No princípio, a programação não era tão bem-vinda assim, e parecia uma barreira difícil de ser superada. Antes, tão acostumado aos esquemas elétricos dos relés, o usuário teria que se adaptar a essa nova realidade. Pensando nisso, os fabricantes criaram uma maneira mais fácil de programar o CLP. Em vez de um programa tradicional (ou seja, um arquivo com várias linhas de código), o usuário passou a desenvolver algo muito próximo ao projeto de um esquema elétrico. No lugar de instruções, símbolos são organizados de forma similar ao/um circuito elétrico, num ambiente de programação amigável e simples de se utilizar. Essa forma de programar é conhecida como a linguagem de programação Ladder, e suavizou significativamente a transição entre o “velho” e o “novo” mundo.

O CLP na indústria

Atualmente, o CLP pode ser encontrado em diversos setores da indústria, desde empresas petroquímicas, até empresas do setor automobilístico. Aliás, a sua primeira geração foi concebida nesta última. A famosa e poderosa GM (General Motors), assim como as demais empresas do setor na época, enfrentava inúmeras dificuldades com o uso de relés e outros dispositivos eletromecânicos. Tal fato motivou uma equipe de engenheiros da GM a desenvolver um dispositivo capaz de superar esses obstáculos, resultando na primeira geração de CLPs. Daí por diante, não só as empresas do setor automobilístico, mas também as demais empresas de fabricação por lotes passaram a utilizar o CLP, muito embora ele só fosse capaz de trabalhar com variáveis digitais (ou seja, variáveis que apresentassem apenas dois

estados, ligado ou desligado, aberto ou fechado, etc.). Com o advento de novas tecnologias, o CLP passou a trabalhar com variáveis analógicas, dentre elas, vazão, pressão e temperatura, aumentando a quantidade de aplicações na indústria.

O Hardware do CLP

De maneira geral, existem dois tipos de CLPs: o modular e o compacto. O primeiro pode ser encontrado na maioria das indústrias de grande porte, pois apresenta uma característica bem interessante para essa aplicação: a capacidade de expandir rapidamente o seu hardware. Imagine a seguinte situação. Uma indústria petroquímica do RN adquiriu vários sensores sem fio (sensores de fumaça) e deseja gerenciá-los através de CLPs com a seguinte dinâmica: se o sensor identificar fumaça, ele deve enviar uma informação ao CLP, que por sua vez, encarrega-se de acionar o sistema de incêndio e alertar os responsáveis. A empresa deseja ainda aproveitar os CLPs que já estão funcionando na fábrica, mas que atualmente não possuem comunicação sem fio. A primeira pergunta é: qual é o tipo dos CLPs envolvidos? São modulares? Em caso afirmativo, podemos incluir módulos de comunicação sem fio, mesmo que esse hardware não esteja presente na configuração original do CLP. Analogamente, é como se alguém comprasse um computador e depois quisesse melhorar as suas configurações iniciais de hardware (memória, HD, monitor etc.). É preciso apenas adquirir novos dispositivos e conectá-los ao computador. O CLP modular funciona da mesma forma. De acordo com a disponibilidade do fabricante e do modelo, podemos obter diversas funcionalidades ao adquirir novos módulos e conectá-los ao CLP. Assim, a partir de novas demandas de produção, é possível aumentar o “poder de fogo” do nosso CLP. Entretanto, essa flexibilidade tem um preço. Quando comparado ao CLP do tipo compacto, o modular é extremamente caro. Sendo assim, o seu primo “pobre” (o CLP compacto) pode ser uma opção mais indicada na maioria das aplicações de pequeno e médio porte.

Atividade 02

1. Pesquise sobre os diversos modelos de CLP. Para dois deles (um modular e outro compacto), apresente as suas características básicas e realize comparações. Liste alguns fabricantes.

Elementos Básicos que Compõe um CLP

A seguir, temos uma breve descrição dos elementos básicos que compõem um CLP, seja ele do tipo modular ou compacto:

- a. **Fonte de alimentação:** é o elemento responsável por fornecer uma tensão adequada ao funcionamento do CLP. Esse elemento converte a tensão de alimentação AC da rede (220V, por exemplo) para uma tensão DC compatível com os dispositivos internos do CLP (5V, por exemplo), como o processador e os módulos de entrada e saída.
- b. **Unidade Central de Processamento (CPU):** corresponde ao elemento que contém o microprocessador principal do CLP. Essa unidade é responsável por executar um dado programa desenvolvido pelo usuário, processando as informações de entrada de acordo com as instruções descritas no programa.
- c. **Unidade de memória:** corresponde ao elemento de armazenamento de dados do CLP. Entende-se por “dados” as informações oriundas dos módulos de entrada, bem como o próprio programa a ser executado pelo dispositivo (as instruções que devem ser processadas pelo CLP), além dos dados que já foram processados.
- d. **Módulos de entrada e saída (E/S):** são os principais dispositivos de interação com o mundo exterior. Normalmente, muitos elementos em campo precisam ser conectados fisicamente aos CLPs através de fios. O hardware responsável por se comunicar com esses elementos são os módulos de entrada e saída (E/S). A maior parte dos sensores fornece um sinal de tensão ou corrente com o objetivo de representar alguma informação. Já os atuadores recebem um sinal de tensão ou corrente capaz de acionar um dado dispositivo, como um motor ou uma válvula. Nesse contexto, o objetivo dos módulos de entrada é adequar o sinal recebido dos sensores para o uso interno do CLP, enquanto que os módulos de saída têm como objetivo gerar o sinal capaz de acionar os elementos externos (atuadores em campo). Os módulos de E/S podem ainda ser classificados como digitais ou analógicos, de acordo com os sinais que chegam dos sensores e saem para os atuadores. Caso as variáveis de entrada ou saída sejam digitais, os módulos são classificados como digitais. Se as variáveis de entrada ou saída forem analógicas, os módulos são classificados como analógicos.

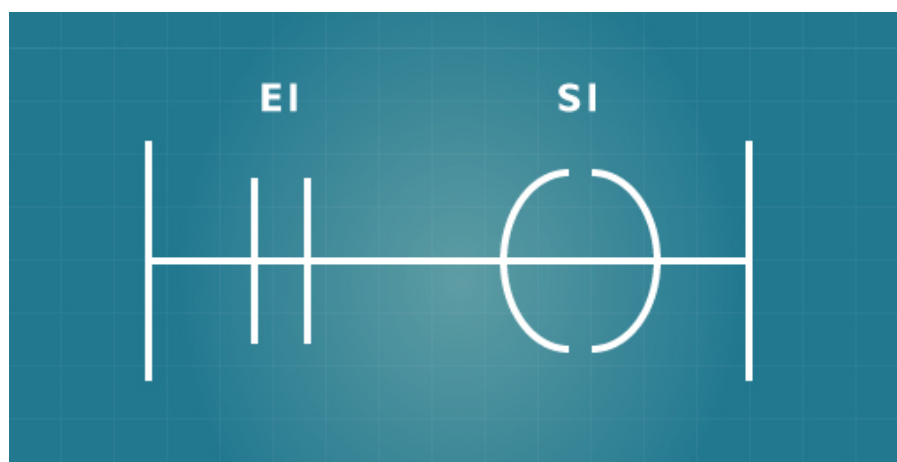
- e. **Dispositivo de programação:** elemento responsável por carregar o programa desenvolvido pelo usuário no CLP. O processo de programação começa com o desenvolvimento de um programa num computador convencional e termina com a utilização de um dispositivo de programação. Para tal, o usuário interage com um software fornecido pelo fabricante capaz de acionar o dispositivo de programação.
- f. **Módulos de comunicação:** são responsáveis por receber/transmitir informações de/para outros CLPs (ou dispositivos) que estejam conectados em rede.

Linguagens de programação

Basicamente, as linguagens de programação mais comuns são: a linguagem de programação Ladder e a linguagem de programação Statement List. A seguir, você verá uma breve descrição de cada uma delas.

- a. **Linguagem Ladder:** essa linguagem se baseia na construção de um diagrama elétrico semelhante ao desenvolvido com relés. Todo o processo de programação é realizado em um ambiente de programação amigável utilizando um software específico fornecido pelo fabricante. O usuário seleciona os símbolos que ele deseja incluir no diagrama e depois os organiza de forma a representar a lógica de comando desejada. A Figura 5 apresenta um diagrama Ladder básico. A ideia de funcionamento é bem simples: a saída digital S1 do CLP só será energizada se a entrada digital E1 for energizada, e se mantiver nessa condição. É como se E1 fosse uma chave normalmente aberta, e S1 fosse a bobina de um relé. Além disso, as duas barras laterais das extremidades representam o positivo e o negativo de uma fonte de tensão fictícia.

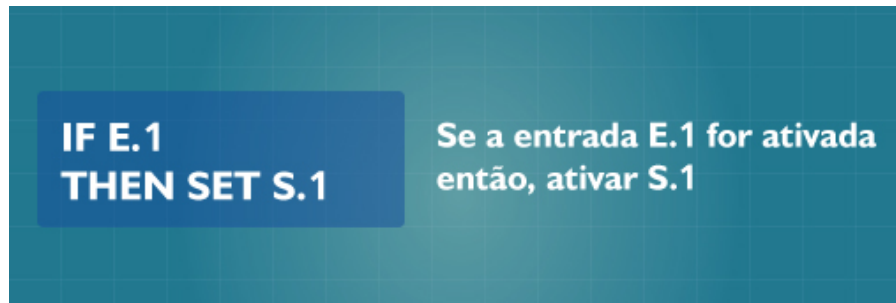
Figura 05 - Exemplo de um diagrama Ladder.



Fonte: Bolton (2009).

- b. **Linguagem Statement List:** em vez de símbolos, essa linguagem lida com uma descrição literal do programa. Cada linha corresponde a uma instrução individual a ser executada pelo CLP. A Figura 6 apresenta um trecho de código equivalente ao diagrama Ladder apresentado na Figura 5.

Figura 06 - Exemplo de um diagrama Ladder.



Fonte: Bolton (2009).

Atividade 03

1. Para dois modelos de CLP, apresente a sua arquitetura interna (através de um diagrama de blocos) e as possíveis linguagens de programação desses dispositivos.

Os circuitos elétricos de comando

Antes de começarmos a trabalhar com a linguagem de programação Ladder, vamos falar um pouco sobre os circuitos elétricos de comando. No passado, esses circuitos dominavam o comando das válvulas e cilindros. Porém, com o avanço das tecnologias, eles foram substituídos pelos Controladores Lógicos Programáveis (CLP), o que gerou uma grande revolução na forma de lidar com os sistemas eletropneumáticos. Para evitar problemas de adaptação, os desenvolvedores criaram uma maneira de suavizar a transição entre o velho e o novo: a linguagem de programação Ladder. Muito similar aos circuitos elétricos, corresponde a uma das formas mais comuns de programação do CLP. Vale salientar que não é a única, porém é a mais intuitiva, principalmente para profissionais que estão migrando ou iniciando a programação desse dispositivo.

Quem sabe analisar e projetar os circuitos de comando, faz isso também com os diagramas Ladder. Dessa forma, vamos utilizar a seguinte estratégia na aula de hoje: estudaremos inicialmente os circuitos a relé; depois, obteremos os respectivos diagramas Ladder, para, em seguida, examiná-los com base no método do GRAFCET.

Para começar, considere, como exemplo, a aplicação da Figura 7, a seguir. Já sabemos que a sequência, com quatro passos,

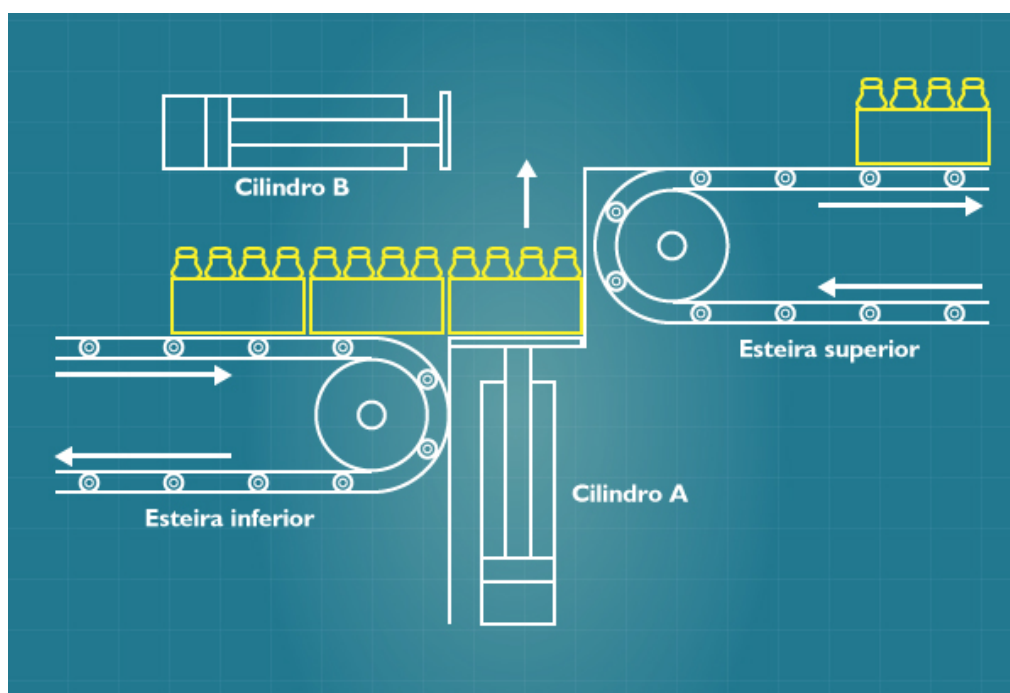
$$A^+ \rightarrow B^+ \rightarrow A^- \rightarrow B^-$$

é a chave para movimentar os cilindros e que o acionamento de cada cilindro está ligado a um dos sensores de fim de curso (para mais detalhes, consultar o material da aula passada). Em suma: o cilindro A só deve avançar (A^+), se o cilindro B estiver recuado (b_0); o cilindro B só deve avançar (B^+), se A estiver avançado (a_1); o cilindro A só deve recuar (A^-), se B estiver avançado (b_1); por fim, o cilindro B só deve recuar (B^-), se A estiver recuado (a_0). Os sensores de fim de curso podem funcionar como uma chave normalmente aberta (NA) ou normalmente fechada (NF). Inicialmente, vamos considerar sensores do tipo NA. Ou seja, quando o sensor é ativado, os seus respectivos contatos se fecham.

A Figura 8 apresenta o circuito de comando para o sistema. De acordo com o procedimento da aula passada, as válvulas escolhidas possuem dois solenoides, um para o avanço do cilindro e outro para o seu recuo. No circuito de comando, observe que cada sensor é representado por uma chave NA e cada solenoide, por quadrados. Note, ainda, a presença de uma chave S, responsável por iniciar todo o processo.

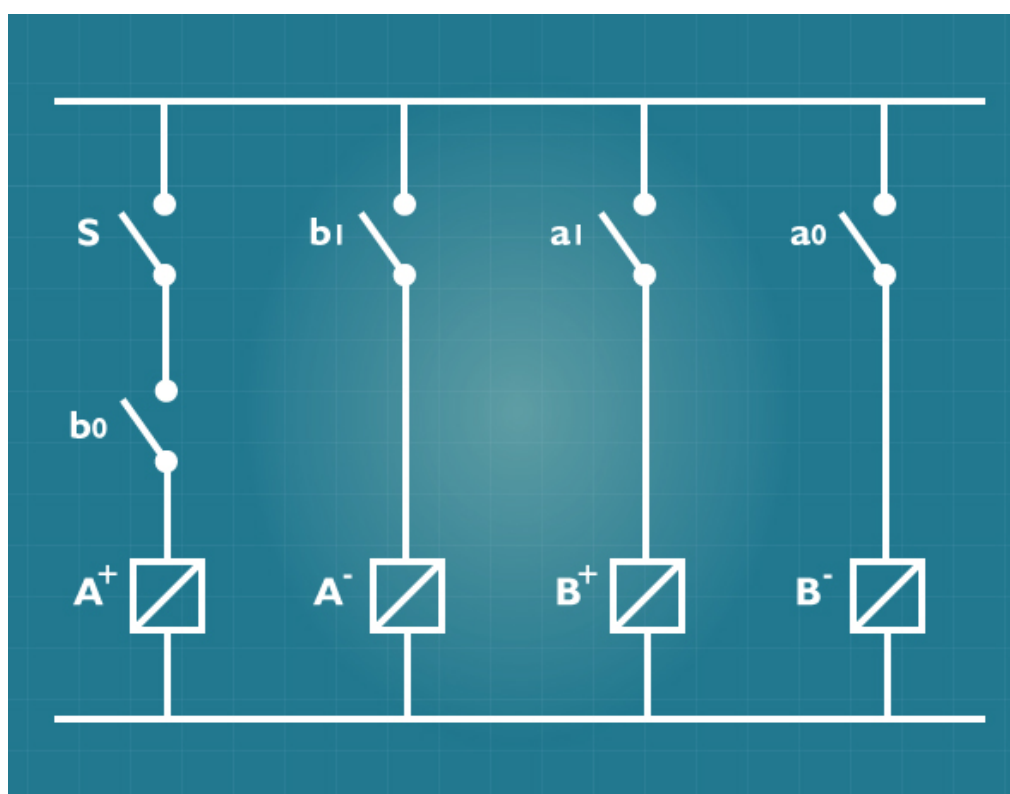
Há de se fazer, porém, uma pergunta: se estamos lidando com um circuito, onde está a fonte de alimentação? Qual a sua representação no esquema? Inicialmente, é importante esclarecer que estamos lidando com um circuito DC (corrente contínua). Sendo assim, precisamos identificar onde estão os terminais positivo e negativo da fonte. O objetivo principal é simplificar a representação do circuito. Em vez de desenhar uma fonte, que deve ser conectada a todos os ramos do circuito, vamos utilizar duas linhas horizontais: uma na parte superior (terminal positivo) e outra, na inferior (terminal negativo). Pela convenção de análise de circuitos DC, a corrente flui no sentido vertical, de cima para baixo, ou seja, na direção dos solenoides.

Figura 07 - Aplicação do uso de cilindros na indústria.



Fonte: Maitelli (2005).

Figura 08 - Circuito de comando para o exemplo da Figura 1.



Fonte: Elaborado pelo Autor (2014).

Análise do circuito

Vamos começar pelo primeiro ramo vertical, da esquerda para a direita. Caso a chave S (chave de início) seja fechada - e caso o cilindro B esteja recuado (chave b_0 fechada) -, o solenoide A^+ será energizado, fazendo com que A avance até acionar o seu sensor de fim de curso (fechamento da chave a_1). Na sequência, o solenoide B^+ será energizado (terceiro ramo, da esquerda para a direita), fazendo com que o cilindro B avance até acionar (fechar) a chave b_1 . Nesse momento, o solenoide A^- será energizado (segundo ramo, da esquerda para a direita), fazendo com que o cilindro A recue até acionar (fechar) a chave a_0 . Esta, por sua vez, energiza o solenoide B^- , fazendo com que o cilindro B recue até acionar (fechar) a chave b_0 . Caso a chave S se mantenha fechada, o ciclo se inicia novamente. Caso contrário, o sistema fica aguardando até que a chave de início seja fechada.

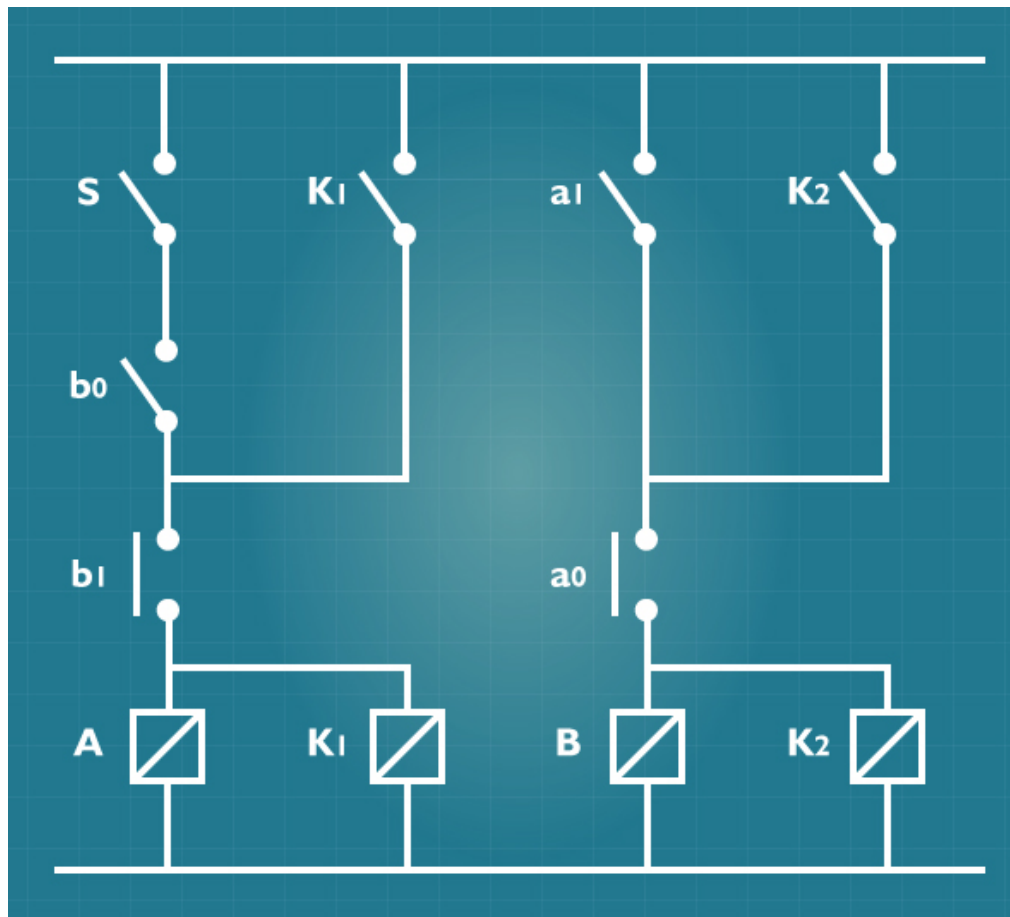
Já sabemos lidar com circuitos de comando para sistemas eletropneumáticos. Porém, considere agora a seguinte situação: imagine que não existam válvulas com dois solenoides disponíveis. A empresa que vai utilizar o sistema da Figura 7 deseja conter gastos, decidindo usar somente válvulas com um único solenoide (o estoque da empresa está cheio desses equipamentos). Como devemos proceder? É possível desempenhar a mesma atividade apenas com válvulas de um único solenoide? A resposta é sim. Porém, teremos um pouco mais de trabalho; precisaremos de dois relés auxiliares nessa abordagem.

O circuito da Figura 9 apresenta a nossa solução. Em relação ao primeiro passo da sequência $A^+ \rightarrow B^+ \rightarrow A^- \rightarrow B^-$, considere o primeiro ramo na vertical (Figura 3), da esquerda para a direita. Temos a chave de início S, o sensor de fim de curso b_0 (tipo NA) e o sensor de fim de curso b_1 (tipo NF). Adicionalmente, temos dois solenoides: um referente à nossa válvula de comando (A) e outro, ao relé auxiliar (K_1). Note que a lógica de comando para o cilindro A continua a mesma, uma vez que o fechamento de S, em conjunto com b_0 (cilindro B recuado), faz com que o cilindro A avance.

No segundo passo da sequência, quando o cilindro B avançar, precisamos manter o solenoide A energizado. Porém, temos um pequeno problema. Ao avançar o cilindro B, a chave b_0 é aberta, cortando a alimentação de A. Nesse momento, entra o nosso relé auxiliar. Observe que, no primeiro passo (avanço de A), a bobina K_1 também é energizada de acordo com a Figura 3, fechado o contato K_1 , que é do

tipo NA (contato de selo). Dessa forma, no segundo passo, quando o cilindro B avançar e a chave b_0 for aberta (B está se movimentado), o solenoide A se manterá energizado através de K_1 . Quando o cilindro B avançar totalmente, o seu sensor de fim de curso b_1 abrirá (considere agora esse sensor como do tipo NF), cortando a alimentação dos dois solenoides (A e K_1). Comportamento similar é observado para o cilindro B.

Figura 09 - Circuito de comando modificado para trabalhar com válvulas que possuem um único solenoide.



Fonte: Elaborado pelo Autor (2014).

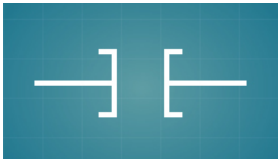
Atividade 04

1. Consulte a Figura 9 desta aula e explique como funciona o circuito em relação ao comando do cilindro B.

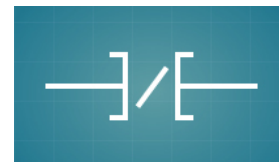
Linguagem de programação Ladder

Como já mencionado, os diagramas Ladder são bem semelhantes aos circuitos de comando. Considere o diagrama da Figura 10 a seguir, por exemplo. Ele corresponde ao diagrama referente ao circuito da Figura 8. Notou as diferenças e as semelhanças? De imediato, parece que o nosso circuito original (Figura 8) foi rotacionado em 90 graus. Além disso, os símbolos das chaves e solenoides foram substituídos por novos (ver Quadro 1) e as posições dos ramos foram alteradas. É interessante ressaltar que esse programa será executado em um CLP. Dessa forma, os itens de sua descrição são, na verdade, representações das entradas e saídas do dispositivo (com exceção das variáveis internas, como veremos logo mais). Todas as chaves e solenoides físicos do nosso sistema (Figura 7) estão conectados ao CLP, sendo representados via software pelos símbolos do diagrama Ladder.

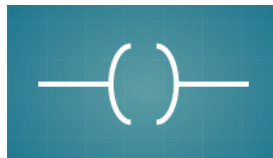
Em relação às entradas da Figura 10, a dinâmica do sistema é a seguinte: o CLP percebe que a chave S fechou, se um valor de tensão não nulo, padronizado, for observado em seus terminais (normalmente 24V DC), o que resulta na mudança do estado da chave no diagrama Ladder. Em contrapartida, se a entrada for nula (0V), o dispositivo entenderá que a chave está aberta fisicamente. Em relação às suas saídas, quando são acionadas, um valor de tensão padronizado não nulo (normalmente 24 V) é apresentado em seus terminais. Resumidamente, cada chave externa e cada solenoide tem uma ligação física (por meio de fios) e lógica (por meio dos diagramas) com o CLP. Assim, podemos, para simplificar, fazer uma associação direta dos símbolos no diagrama Ladder com os dispositivos externos (chaves e solenoides).

Descrição	Simbologia Padrão	Simbologia Alternativa
Chave normalmente aberta - NA (Entrada)		

Chave normalmente fechada - NF
(Entrada)



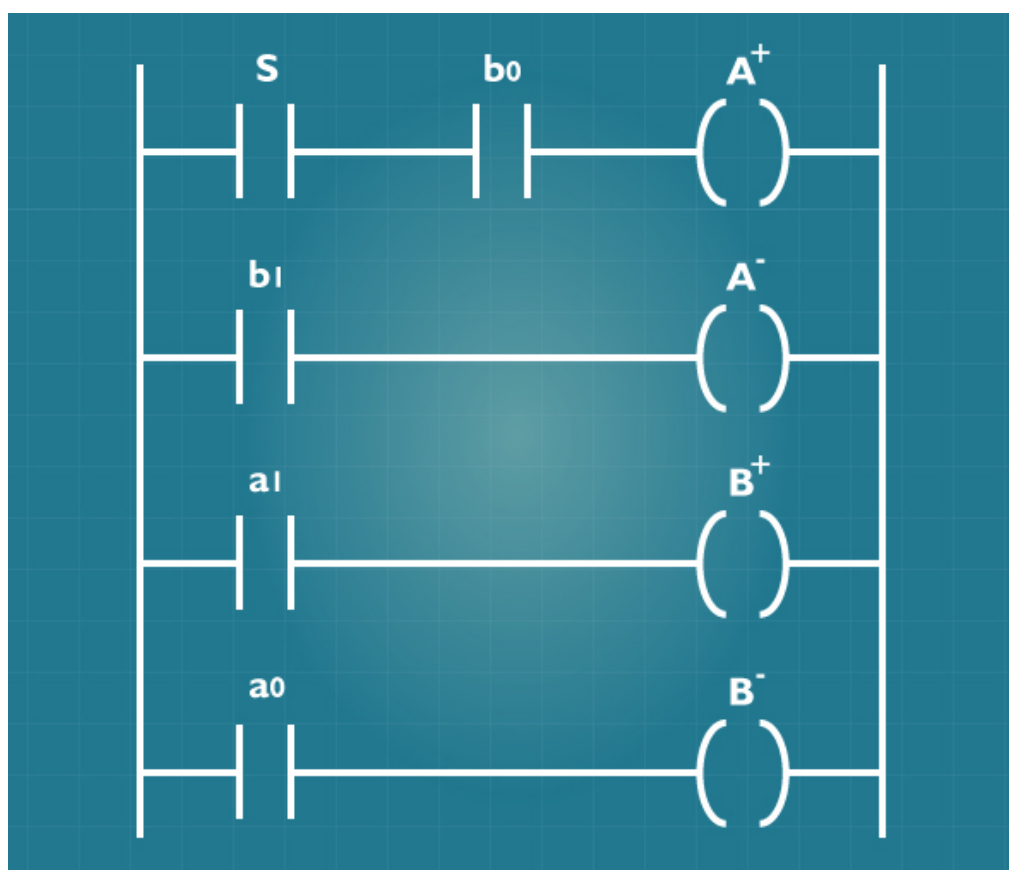
Solenóide (Saída)



Quadro 1: Simbologia básica da linguagem de programação Ladder.

Fonte: Adaptado de PARKER TRAINING (2005) e Maitelli (2005).

Figura 10 - Diagrama Ladder para o exemplo da Figura 2.



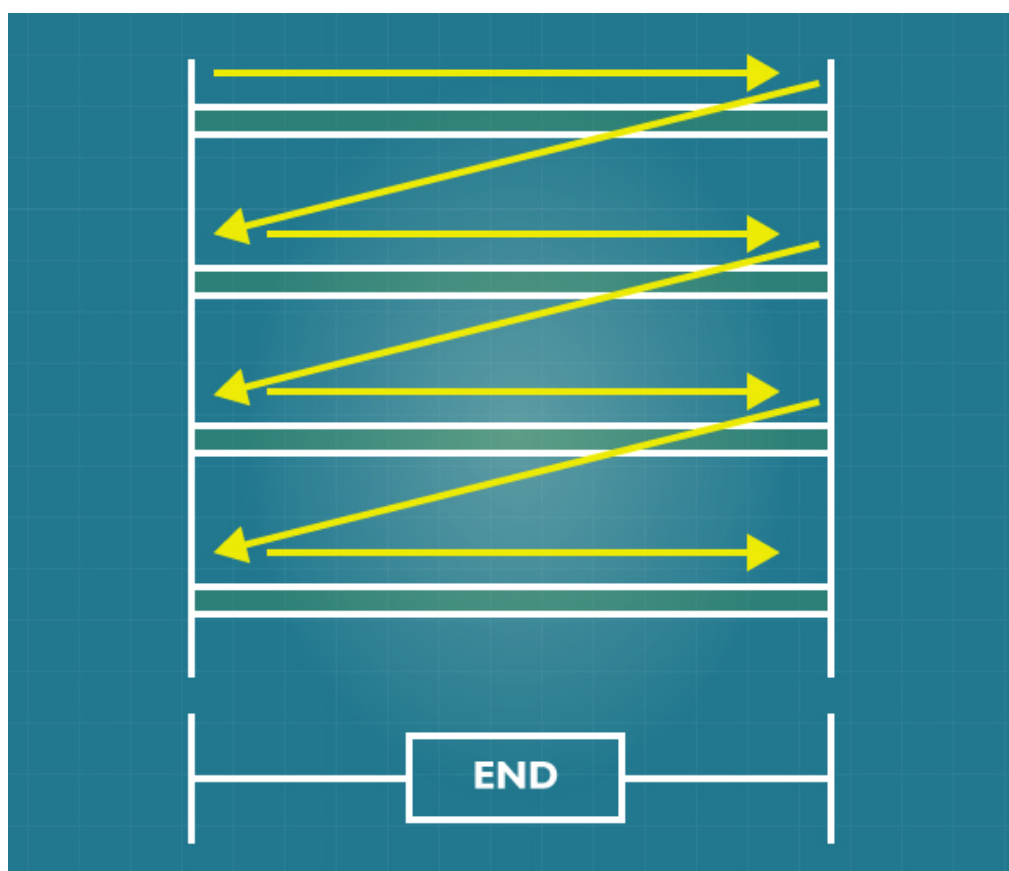
Fonte: Elaborado pelo Autor (2014).

O termo em inglês **Ladder** quer dizer escada; como em qualquer escada, ela é formada por **degraus** (linhas horizontais). Cada degrau define uma operação a ser executada. Porém, precisamos saber qual operação deve ser executada primeiramente. Se estivermos lidando com a linguagem Ladder, o CLP **sempre**

executa o programa da mesma forma: da esquerda para a direita, de cima para baixo - como apresentado na Figura 11, adiante. Perceba que o diagrama da figura 5 apresenta, ainda, um degrau (linha) adicional que não mencionamos. Localizado na parte inferior (última linha horizontal), ele corresponde à identificação do fim do programa (note o END entre as duas barras horizontais). Uma varredura completa do programa (até a linha END) é chamada de ciclo. O tempo de cada ciclo varia de acordo com o fabricante e com o modelo. Obviamente, quanto mais rápido, mais caro é o dispositivo. Por fim, é interessante ressaltar que as saídas físicas do CLP não são atualizadas à medida que o programa é executado. A ideia básica é a seguinte:

1. Inicialmente, o dispositivo verifica o estado de todas as entradas, salvando essas informações na memória.
2. Depois, de posse das informações do estado de cada entrada, ele executa o programa e obtém o estado de cada saída. Em seguida, essas informações são salvas na memória.
3. Ao final (linha END), todas as saídas são atualizadas de uma só vez.

Figura 11 - Sentido de execução de um programa Ladder.



Fonte: Bolton (2009).

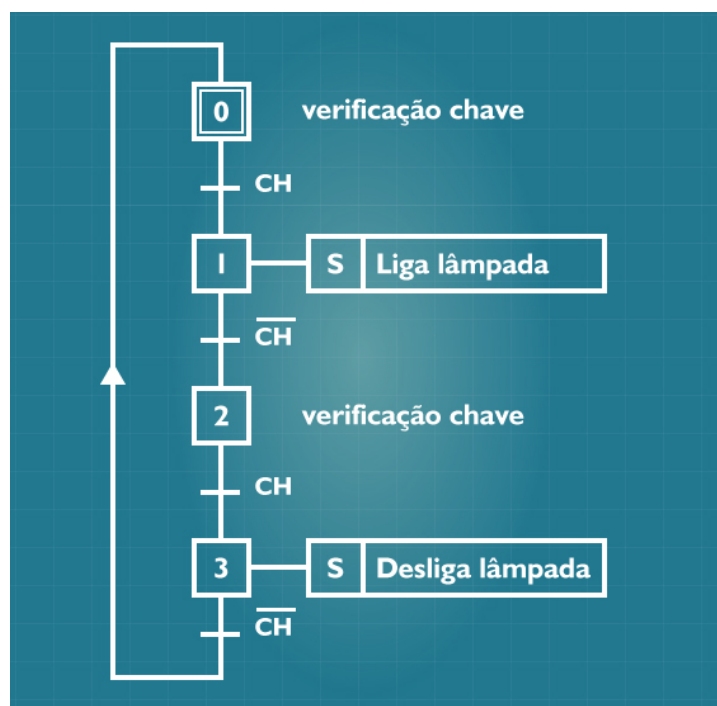
Atividade 05

1. Pesquise sobre a linguagem de programação *Statement List* (SL). Escreva um programa em SL, equivalente ao diagrama Ladder apresentado na Figura 10 para o nosso exemplo.

Método de GRAFCET

Até agora, já somos capazes de obter um diagrama Ladder a partir de circuitos de comando. Apesar de ser uma estratégia muito comum, existem outras abordagens que podemos utilizar. Uma delas corresponde ao desenvolvimento de diagramas Ladder com base no método do GRAFCET. Em vez de desenvolver um circuito e depois convertê-lo para Ladder, podemos utilizar o GRAFCET para construir o nosso próprio diagrama. Para tal, vamos considerar o diagrama da Figura 12, a seguir, como exemplo. Basicamente, ele representa a sequência de operações a ser executada por etapas.

Figura 12 - GRAFCET para ligar e desligar uma lâmpada utilizando-se uma botoeira.

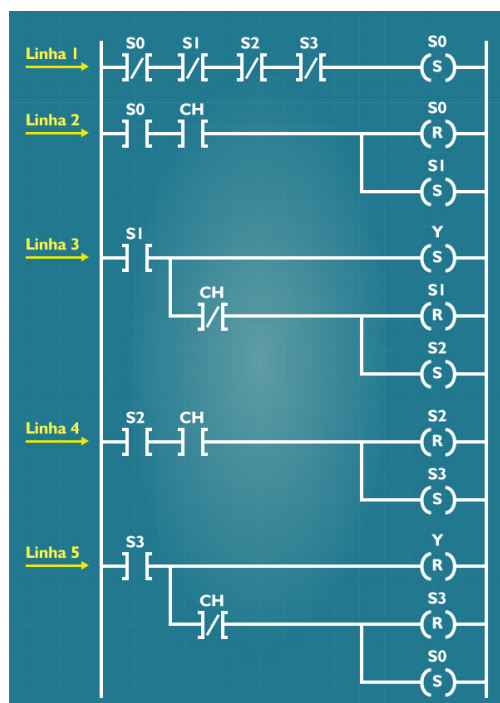


Fonte: Maitelli (2005).

Suponha que estamos na etapa inicial, identificada no diagrama por 0. Caso a chave CH seja acionada, o sistema vai da etapa 0 para a etapa **1**, acendendo a lâmpada. Em seguida, se a chave for aberta (condição representada por CH), o sistema sai da etapa **1** para a etapa **2**. Esta, por sua vez, serve apenas para verificar se a chave será novamente acionada. Em caso positivo, a lâmpada é desligada na etapa **3**. Nesta última, caso a chave seja novamente aberta, o sistema retorna para a etapa **0**, na qual permanece aguardando CH ser fechada. Em resumo: se CH for uma botoeira, precisamos pressioná-la para ligar a lâmpada; para desligá-la, precisamos soltar CH e pressioná-la novamente.

A Figura 13, a seguir, apresenta o diagrama Ladder para a situação da Figura 6. Nesse diagrama, é interessante ressaltar que nem todos os seus elementos estão associados a dispositivos físicos, como chaves e solenoides. Alguns são apenas variáveis auxiliares, como as que vamos utilizar na identificação das etapas do GRAFCET. Podemos alterar essas variáveis (que são binárias) da mesma forma que fazemos com uma saída convencional. Precisamos apenas satisfazer as condições necessárias para a mudança - ou seja, fechar todas as chaves associadas ao respectivo solenoide.

Figura 13 - Diagrama Ladder para o exemplo da lâmpada.



Fonte: Maitelli (2005).

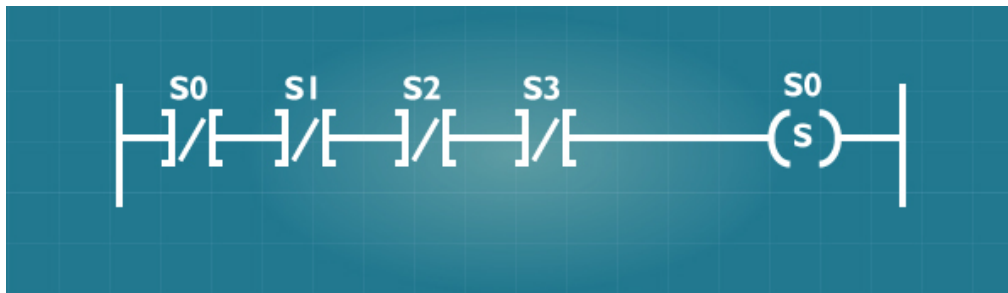
Outra novidade é o símbolo definido para a saída (solenoide) no Quadro 1. Note que agora na Figura 13 existe um S ou R no seu interior. Isso significa que vamos forçar o valor **1** (S) ou **0** (R) numa dada saída (ou variável), caso as condições da respectiva linha sejam satisfeitas (ou seja, chaves fechadas). Mesmo após mudanças nessas condições (abertura das chaves), o valor da saída se mantém o mesmo (**0** ou **1**, dependendo do último comando executado). É importante ressaltar que quando uma determinada saída no diagrama Ladder recebe **1**, a tensão de saída nos terminais do CLP será 24V; caso receba **0**, essa tensão será 0V. Em contrapartida, se uma variável auxiliar do programa receber **0** ou **1**, apenas o seu valor na memória é alterado para **0** (S), ou para **1** (R), respectivamente.

Para testar o valor de uma determinada variável, o procedimento é o mesmo utilizado para testar o estado de uma entrada. Vamos associar à respectiva variável uma chave NA (ou NF). Caso o valor da variável seja **1**, o estado original da chave no nosso diagrama Ladder será alterado; se o valor for **0**, o estado original é mantido.

Após as devidas apresentações, vamos começar com a análise do nosso diagrama Ladder. Basicamente, temos uma entrada (CH), uma saída (Y) e quatro variáveis auxiliares (S0, S1, S2 e S3). Essas variáveis serão utilizadas para identificar qual é a etapa ativa do GRAFCET no momento. Um **1** significa que a respectiva etapa está ativa e um **0**, que ela está inativa. Desse modo, somente uma deve estar ativa por vez; caso contrário, teremos problemas na execução do processo.

Suponha, agora, que o programa inicie com todas as variáveis internas nulas. A primeira tarefa é garantir que a etapa inicial (S0) seja ativada. É justamente o que faz a **linha 1** (Figura 14, a seguir) do nosso programa. Caso o valor das variáveis S1, S2 e S3 seja **0**, a variável S0 recebe **1**, indicando que o sistema se encontra na etapa inicial **0**. Note que as variáveis auxiliares S1, S2 e S3 estão sendo representadas por chaves do tipo NF (linha 1) - ou seja, por chaves que se encontram originalmente na condição fechada. A atribuição de um **0** a essas variáveis, como já mencionado, faz com que a condição original da chave seja mantida. Então, na primeira linha, como S1, S2 e S3 estão em **0**, as chaves utilizadas no diagrama (linha 1) continuam fechadas (condição original). A mesma ideia é empregada nas representações com chaves NA. Observe que, na Figura 13, existem os dois tipos de representação. Na linha 4, por exemplo, a variável S2 (anteriormente representada por uma chave NF na linha 1) é agora representada por uma chave NA.

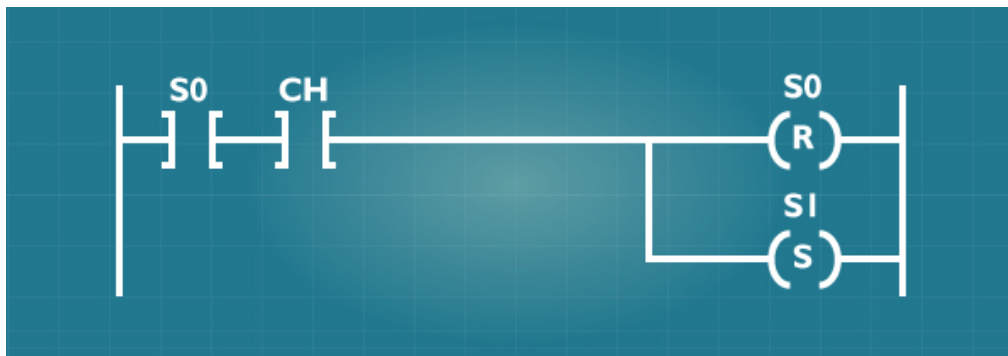
Figura 14 - Linha 1 do diagrama Ladder para o exemplo da lâmpada.



Fonte: Maitelli (2005).

A próxima linha a ser executada corresponde à **linha 2** (Figura 15), desde que a chave CH esteja fechada. Nesse caso, como a variável S0 (representada por uma chave NA) está em **1** (ou seja, na condição fechada), ela recebe **0** (R) e a variável S1 recebe **1** (S). Isso significa que o sistema sai da etapa **0** para a etapa **1**.

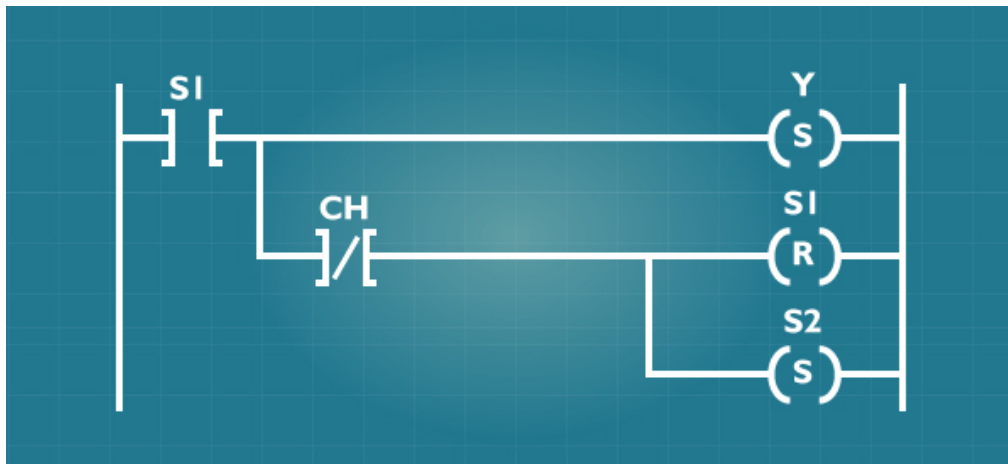
Figura 15 - Linha 2 do diagrama Ladder para o exemplo da lâmpada.



Fonte: Maitelli (2005).

A próxima linha a ser executada corresponde à **linha 3** (Figura 16, em seguida). Note que a chave NA, utilizada para representar a variável S1, está fechada, uma vez que S1 recebeu 1 na execução anterior. Sendo assim, automaticamente a saída Y recebe 1 (S). Apesar de satisfazer parte do objetivo principal (acender a lâmpada), essa linha precisa, ainda, garantir que a próxima etapa (etapa 2) seja ativada quando CH for aberta. A ramificação inferior da Figura 16 garante essa operação. Nesse caso, a variável S1 recebe 0 (R) e a variável S2 recebe 1 (S). Ou seja, o sistema sai da etapa 1 para a etapa 2.

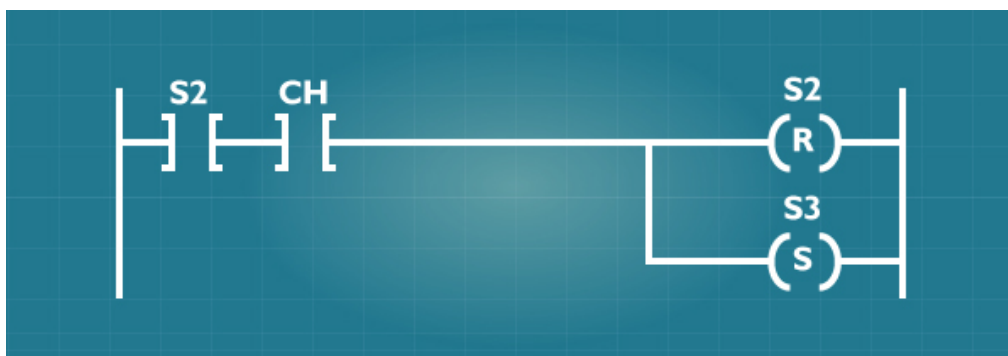
Figura 16 - Linha 3 do diagrama Ladder para o exemplo da lâmpada.



Fonte: Maitelli (2005).

A próxima linha a ser executada corresponde à **linha 4** (Figura 17), que é bem similar à linha 2. A ideia é novamente passar para a próxima etapa, se a chave CH for fechada. Nesse caso, como a variável S2 (representada por uma chave NA) já está em 1 (ou seja, na condição fechada), S2 recebe 0 (R) e a variável S3 recebe 1 (S). O sistema sai da etapa 2 para a etapa 3

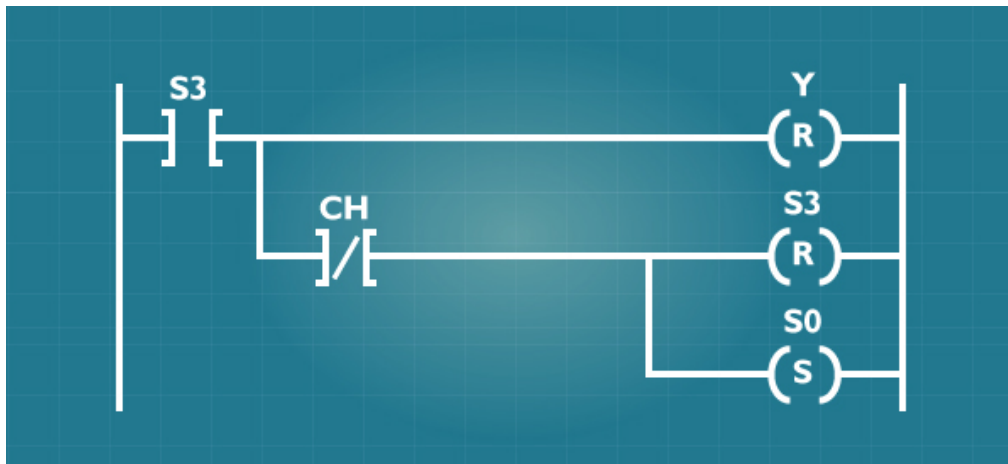
Figura 17 - Linha 4 do diagrama Ladder para o exemplo da lâmpada.



Fonte: Maitelli (2005).

A próxima linha a ser executada corresponde à **linha 5** (Figura 18), que é bem similar à linha 3. Como a chave utilizada para representar S3 está fechada (ver execução da linha anterior), a saída Y recebe 0, fazendo com que a lâmpada do nosso processo se apague. Além disso, caso a chave CH esteja aberta (receba 0), a variável S3 recebe 0 (R) e a variável S0 recebe 1 (S). O sistema sai da etapa 3 para a etapa 0, reiniciando o ciclo de trabalho.

Figura 18 - Linha 5 do diagrama Ladder para o exemplo da lâmpada.



Fonte: Maitelli (2005).

Atividade 06

1. Pesquise sobre o método do GRAFCET para programação do CLP. Apresente um exemplo de programação em Ladder que utilizou o método do GRAFCET. Exponha detalhes de funcionamento do sistema, bem como uma descrição completa do diagrama Ladder.

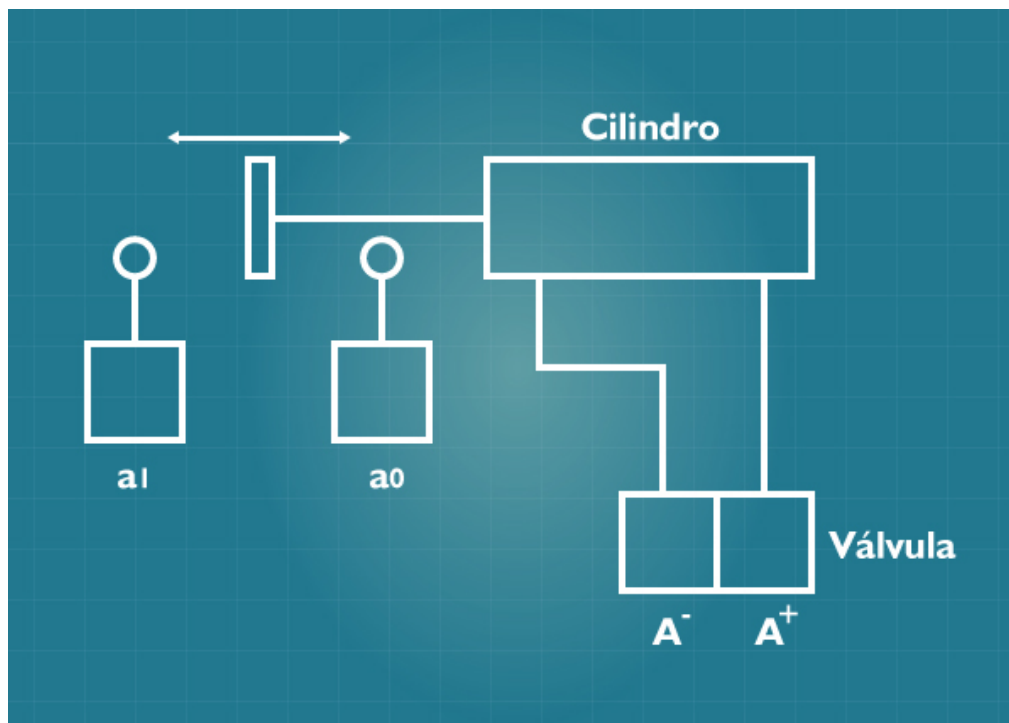
Aplicação - Avanço e recuo de um cilindro

Vamos começar a nossa aula com uma aplicação industrial bem simples: o avanço e recuo automático de um cilindro de dupla ação (Figura 19). Esse procedimento é o “carro chefe” de boa parte das aplicações industriais, e sem ele não poderíamos, por exemplo, empurrar caixas, amassar chapas, ou selecionar um dado material na linha de produção. Dessa forma, ele será o nosso ponto de partida para as demais aplicações.

A ideia é desenvolver um programa em Ladder que utilize três sinais de entrada e dois sinais de saída. São eles: um sinal de entrada referente à botoeira de início (S^1); dois sinais de entrada referentes aos sensores de fim de curso (a_0 e a_1); e dois sinais de saída referentes à válvula de comando, sendo um para cada solenoide (A^+

e A^-). O objetivo inicial é realizar o avanço e recuo automático do cilindro uma única vez através de S_1 . Considere que todas as chaves físicas (sensores de fim de curso) e botoeiras são do tipo NA.

Figura 19 - Esquema referente ao avanço e recuo automático de um cilindro de dupla ação.



Fonte: Elaborado pelo Autor (2014).

Para solucionar esse problema, utilizaremos o método do GRAFCET, dividindo o nosso processo em etapas. A primeira delas corresponde à etapa inicial, que será identificada por "0" e terá como finalidade apenas aguardar até que a botoeira S_1 seja pressionada (primeira condição de transição). Se isso acontecer, o sistema deve prosseguir para uma nova etapa (etapa "1"), em que o cilindro vai avançar até que o sensor de fim de curso (a_1) seja acionado (segunda condição de transição). Nesse caso, o sistema deve sair da etapa atual (etapa "1") e passar para a seguinte (etapa "2"), que será responsável por recuar o cilindro até que o sensor de fim de curso a_0 seja acionado (terceira condição de transição). Por fim, se a_0 for ativado, o sistema deve retornar para a etapa "0", na qual ficará aguardando até que a chave de início seja novamente pressionada. O Quadro 2 apresenta um resumo da referida sequência.

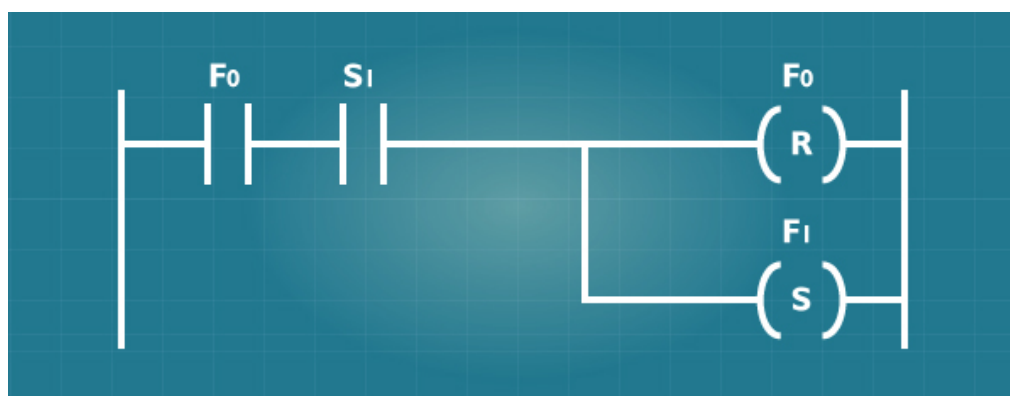
Etapa de origem	Etapa de destino	Condição de transição entre as etapas de origem e destino	Ação da etapa de origem
0	1	S_1	Aguardar
1	2	a_1	Avançar Cilindro
2	0	a_0	Recuar Cilindro

Quadro 2 - Resumo do GRAFCET para a aplicação de avanço e recuo automático de um cilindro de dupla ação.

Fonte: Elaborado pelo Autor (2014).

Com base na tabela apresentada, podemos obter o GRAFCET da Figura 20. Note o sentido de evolução das etapas, as condições de transição e as ações associadas.

Figura 20 - GRAFCET referente ao avanço e recuo automático de um cilindro de dupla ação.



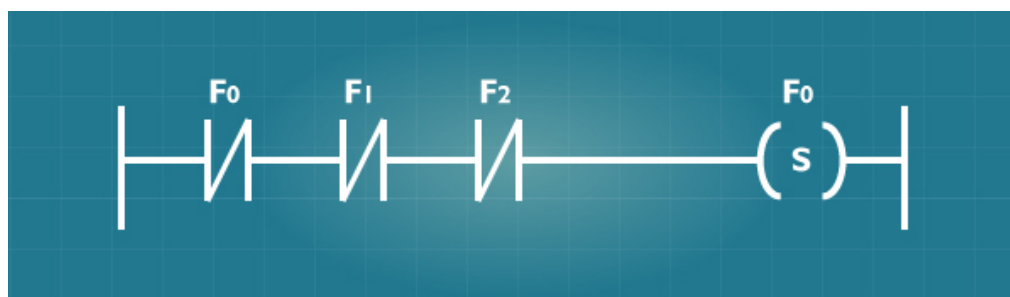
Fonte: Elaborado pelo Autor (2014).

O próximo passo é desenvolver o nosso diagrama Ladder. Assim como na aula passada, vamos utilizar chaves e solenoides para representar as variáveis auxiliares do sistema (F_0 , F_1 e F_2), chaves para representar os sinais de entrada (nesse caso, somente uma, S_1) e solenoides para representar as saídas (A^+ e A^-). Novamente, cada variável auxiliar está associada a uma etapa do GRAFCET (Figura 20).

A primeira situação que devemos considerar está relacionada ao “despertar” do CLP. Normalmente, quando o programa é iniciado, o valor de todas as variáveis auxiliares é “0”. Dessa forma, é preciso garantir que a etapa inicial (F_0) seja ativada e, assim, forçar o início do processo. É o que deve fazer a linha 1 (Figura 21) do nosso

programa. Caso o valor das variáveis F_0 , F_1 e F_2 seja "0" (nesse caso, nenhuma etapa está definida como ativa), a variável F_0 recebe "1", indicando que o sistema vai para a etapa inicial "0". Lembre-se que é possível representar as nossas variáveis (e entradas) por chaves do tipo NA ou NF, o que depende da lógica desejada. Quando são utilizadas chaves NF, por exemplo, caso a variável associada receba "0", a chave se mantém na sua condição original, ou seja, fechada. No caso da linha 1, a alimentação do solenoide F_0 está condicionada ao estado das chaves F_0 , F_1 e F_2 , que são do tipo NF. Em outras palavras, a variável F_0 só recebe "1" (solenoide F_0 alimentado), caso o valor das variáveis F_1 , F_2 e F_3 seja "0" (chaves F_1 , F_2 e F_3 fechadas).

Figura 21 - Linha 1 do diagrama Ladder para o avanço e recuo automático de um cilindro de dupla ação.



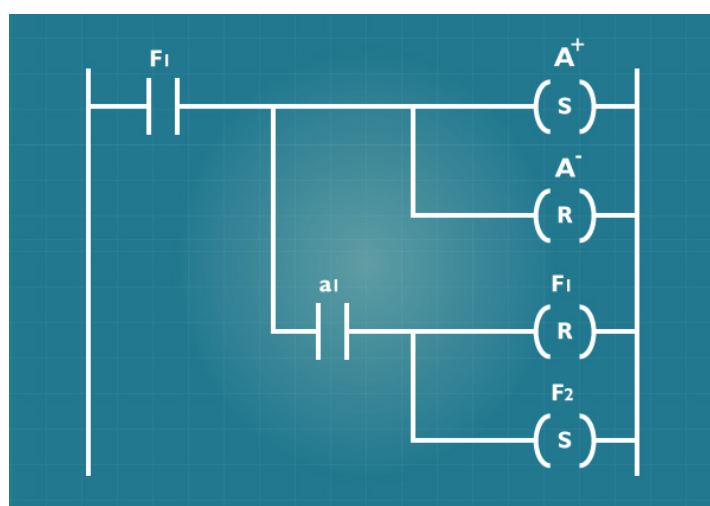
Fonte: Elaborado pelo Autor (2014).

A próxima linha do programa (linha 2) deve ser responsável pela etapa "0" do sistema, ou seja, pela implementação da sua ação associada e da respectiva condição de transição para a etapa seguinte. Então, o que devemos fazer? De acordo com o GRAFCET da Figura 20, o programa deve apenas aguardar até que um sinal externo (fechamento da chave S_1) seja identificado, gerando uma mudança de etapa logo em seguida. Note que não existe uma ação associada explícita (além de aguardar), apenas uma condição de transição. Em relação ao Ladder, podemos "traduzir" essa etapa em duas chaves do tipo NA em série, uma que representa o estado F_0 e outra que representa a botoeira externa S_1 , além de dois solenoides em paralelo, porém em série com as chaves. Um dos solenoides corresponde ao da variável F_0 , indicando que, caso seja alimentado, a variável F_0 recebe "0" (solenoide do tipo "R"). Em paralelo, o segundo solenoide corresponde ao da variável F_1 , indicando que, caso seja alimentado, a variável F_1 recebe "1" (solenoide do tipo "S"). Em suma, caso as chaves F_0 e S estejam na condição fechada, os solenoides F_0 e F_1 serão alimentados e, dessa forma, o sistema sai da etapa "0" para a etapa "1".

A próxima linha do programa (linha 3) deve ser responsável pela implementação da etapa “1” do sistema. Inicialmente, trabalharemos a sua ação associada que, nesse caso, corresponde ao avanço do cilindro (Figura 20) e, em seguida, a sua condição de transição. Sendo assim, precisamos, antes de tudo, saber se a etapa “1” está ativa. Em caso positivo, devemos acionar o solenoide de avanço do cilindro e desacionar o solenoide de recuo (para evitar a queima da válvula). Para tal, vamos utilizar uma chave NA (associada à variável F_1) em série com duas bobinas que estão em paralelo, sendo uma para cada ação do cilindro, avanço (A^+) e recuo (A^-). Quando energizadas (chave F_1 fechada), a saída A^+ recebe “1”, enquanto a saída A^- recebe “0”.

Já definimos como proceder com a ação associada da etapa “1”. Agora, vamos nos preocupar com a sua condição de transição. De acordo com o GRAFCET da Figura 20, essa condição está relacionada ao sensor de fim de curso a_1 . Quando acionado, o sistema deve sair da etapa “1” para a etapa “2”. No nosso programa, isso deve se traduzir em uma chave NA (associada à variável F_1) em série com outra chave NA (associada à variável a_1), que por sua vez, está em série com dois solenoides em paralelo, sendo um para cada variável auxiliar, F_1 e F_2 . Dessa forma, quando essas chaves estiverem fechadas, os dois solenoides serão energizados, fazendo com que a variável F_1 receba “0” (R) e a variável F_2 receba “1” (S). Note que a chave F_1 , mencionada no parágrafo anterior, foi novamente utilizada para verificar se a etapa “1” está ativa.

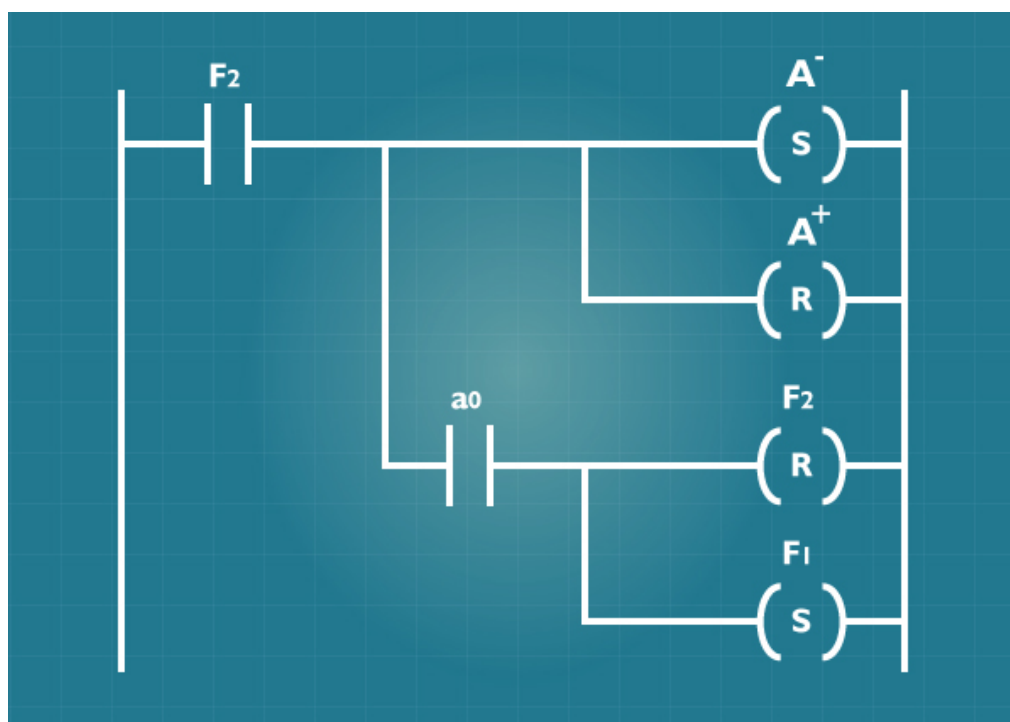
Figura 23 - Linha 3 do diagrama Ladder para o avanço e recuo automático de um cilindro de dupla ação.



Fonte: Elaborado pelo Autor (2014).

Por fim, a próxima linha do programa (linha 4) deve ser responsável pela implementação da etapa “2” do sistema. Essa etapa é muito similar à etapa anterior, exceto pelo fato de que o cilindro deve recuar (ação associada), em vez de avançar, até que o sensor de fim de curso seja acionado (condição de transição). Porém, a estrutura de programação é a mesma. Devemos apenas tomar cuidado em substituir as chaves e os solenoides anteriores pelos atuais. A Figura 24 apresenta essa implementação.

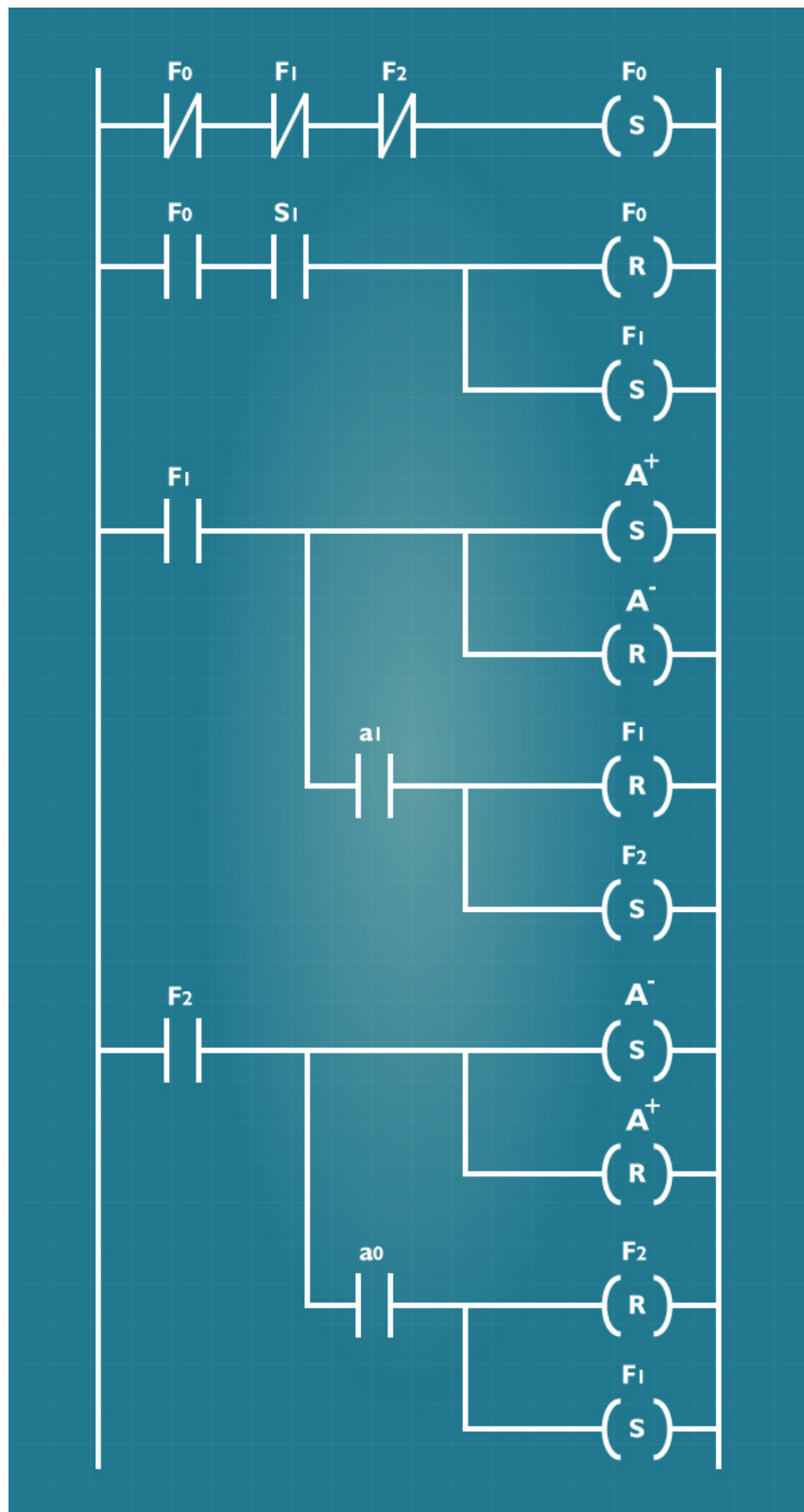
Figura 24 - Linha 4 do diagrama Ladder para o avanço e recuo automático de um cilindro de dupla ação.



Fonte: Elaborado pelo Autor (2014).

O diagrama Ladder completo é apresentado na Figura 25. Note que ele nada mais é do que a união de todas as linhas apresentadas anteriormente nas Figuras 21, 22, 23 e 24.

Figura 25 - Diagrama Ladder completo para o avanço e recuo automático de um cilindro de dupla ação.



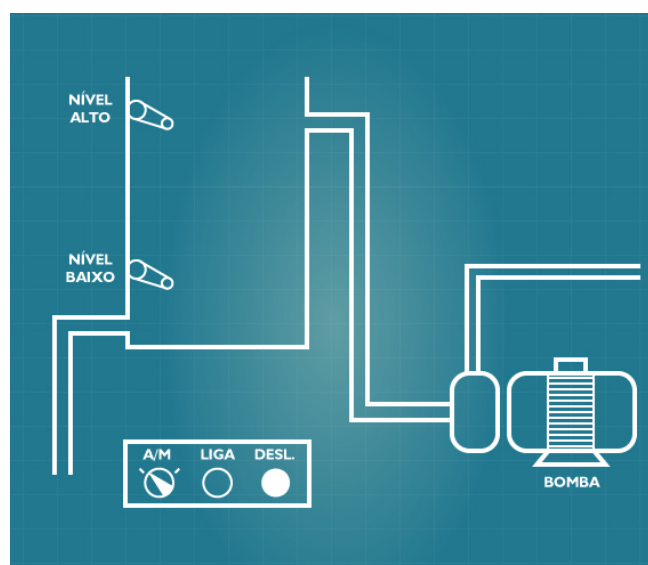
Fonte: Elaborado pelo Autor (2014).

Controle de nível em um reservatório

Em diversas aplicações, necessitamos de um controle de nível máximo e mínimo em um reservatório. Por exemplo, num condomínio de apartamentos, nunca podemos deixar o reservatório secar, nem tampouco transbordar, independentemente do consumo atual do prédio. No primeiro caso, os moradores correm o risco de ficar sem água e no segundo, podem pagar taxas exorbitantes de consumo (conta da CAERN). A segunda aplicação da nossa aula tem como objetivo resolver esse problema. A ideia é desenvolver um diagrama Ladder capaz de operar em modo manual, o qual o operador pode ligar e desligar a bomba quando quiser, e em modo automático, em que a bomba é ligada ou desligada automaticamente pelo CLP. Adicionalmente, no modo **automático**, o processo deve ligar e desligar a bomba 10 segundos após a identificação de nível baixo ou alto pelos sensores.

O sistema é composto fisicamente por um reservatório, uma bomba, dois sensores de nível (um para nível baixo e outro para nível alto), uma chave NA para selecionar o modo de operação (manual e automático) e duas botoeiras NA, uma para ligar a bomba e outra para desligar. Quando a chave de seleção (manual ou automático) estiver fechada, o modo automático está ativo, caso contrário, é o modo manual que está ativo.

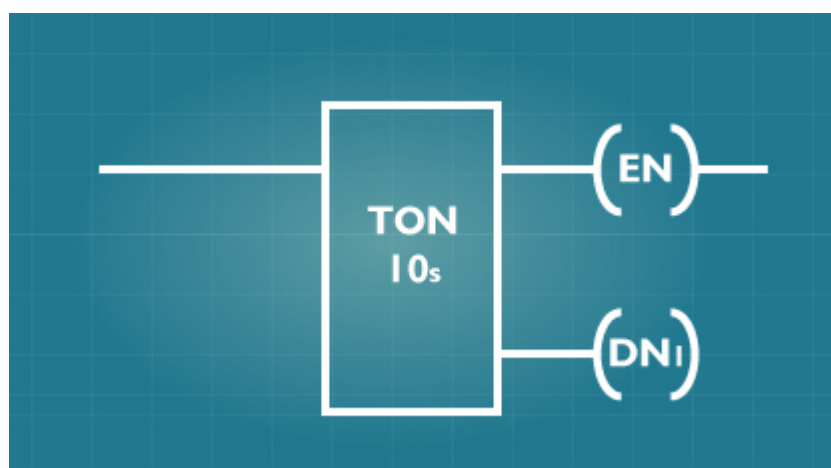
Figura 26 - Sistema de controle de nível alto e baixo em um reservatório.



Fonte: Adaptado de Maitelli (2005).

Nesse exemplo, em vez de trabalharmos com o método do GRAFCET, vamos implementar a lógica de trabalho de forma mais direta. Para tal, precisamos de chaves e solenoides para representar as variáveis auxiliares (DN_1 e DN_2), chaves para representar os sinais de entrada (sensor de nível alto N_A , sensor de nível baixo N_B , chave seletora de modo M, botão de ligar L_1 e botão de desligar L_2) e um solenoide para representar a saída do sistema (bomba B). Além disso, vamos precisar de um novo elemento no diagrama Ladder chamado de temporizador TON. Podemos “encarar” esse elemento como um solenoide que, ao ser energizado - o que depende das condições associadas (chaves da respectiva linha), inicia um processo de contagem de zero até um valor máximo predeterminado. Ao término desse processo, uma variável do contador chamada “DN” recebe “1”, informando o fim da temporização. Caso a alimentação seja interrompida, o processo de contagem é paralisado, “zerando” o temporizador e a sua respectiva variável “DN”. Após a contagem total, caso a alimentação seja interrompida a qualquer momento, a variável “DN” também recebe “0”.

Figura 27 - Simbologia em Ladder do temporizador TON.

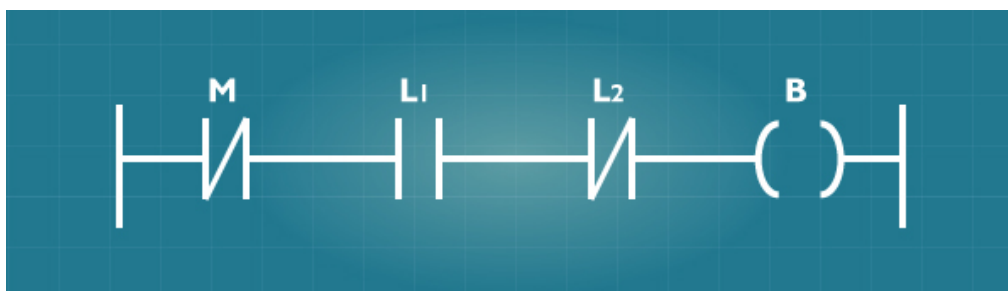


Fonte: Elaborado pelo Autor (2014)

Para desenvolver o nosso diagrama Ladder, vamos dividir essa aplicação em duas partes. A primeira delas é referente ao funcionamento do sistema no modo manual, e outra referente ao modo automático. A **linha 1** do nosso programa será responsável por implementar o modo **manual** do sistema. Sendo assim, o primeiro elemento que deve fazer parte dessa linha corresponde a uma chave NF associada ao modo de trabalho. O principal objetivo é verificar se o modo manual está ativo ou não, antes de tomar qualquer ação referente a esse modo. Em caso positivo, essa chave estará fechada no nosso diagrama, enquanto os demais elementos da linha (chaves e solenoides) podem então ser considerados. Em série com essa chave,

devemos inserir uma chave NA associada ao botão de ligar a bomba (L_1), uma chave NF associada ao botão de desligar (L_2) e um solenoide convencional (ou seja, sem memória) associado à saída do sistema (B). Esse esquema inicial é apresentado na Figura 28.

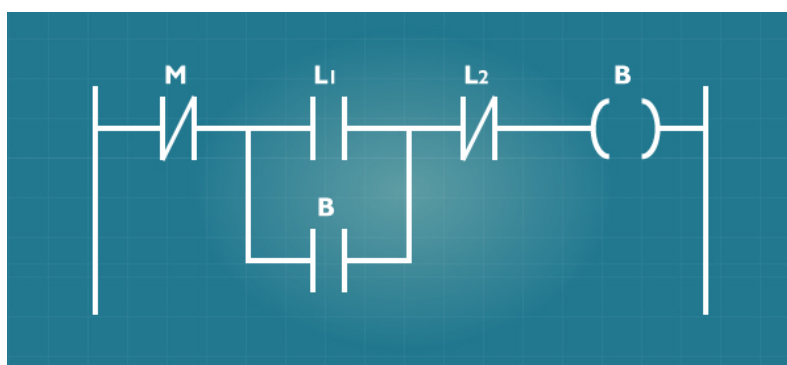
Figura 28 - Esboço inicial da linha 1 do diagrama Ladder para o controle de nível em um reservatório.



Fonte: Elaborado pelo Autor (2014).

Com base na configuração da Figura 28, supondo que a chave M esteja fechada, ao fechar a chave L_1 (pressionando o botão de ligar), o solenoide B será energizado. Porém, isso só ocorre enquanto a chave L_1 estiver fechada (botão pressionado). Ao abrir essa chave (botão liberado), a alimentação do solenoide é cortada, sendo necessário um contato de selo para manter a alimentação de B ativa. Para tal, podemos utilizar a própria saída do sistema (na condição de uma variável interna) representada por uma chave NA e que será posicionada em paralelo com a chave L_1 . Uma vez que vamos representar a variável B por uma chave NA, quando o solenoide for alimentado, a chave associada a ela será fechada. Caso se deseje desligar a bomba, é só abrir a chave L_2 (botão de desligar pressionado). A Figura 29 apresenta a linha 1 do nosso programa com a implementação do contato de selo mencionado.

Figura 29 - Linha 1 do diagrama Ladder para o controle de nível em um reservatório.

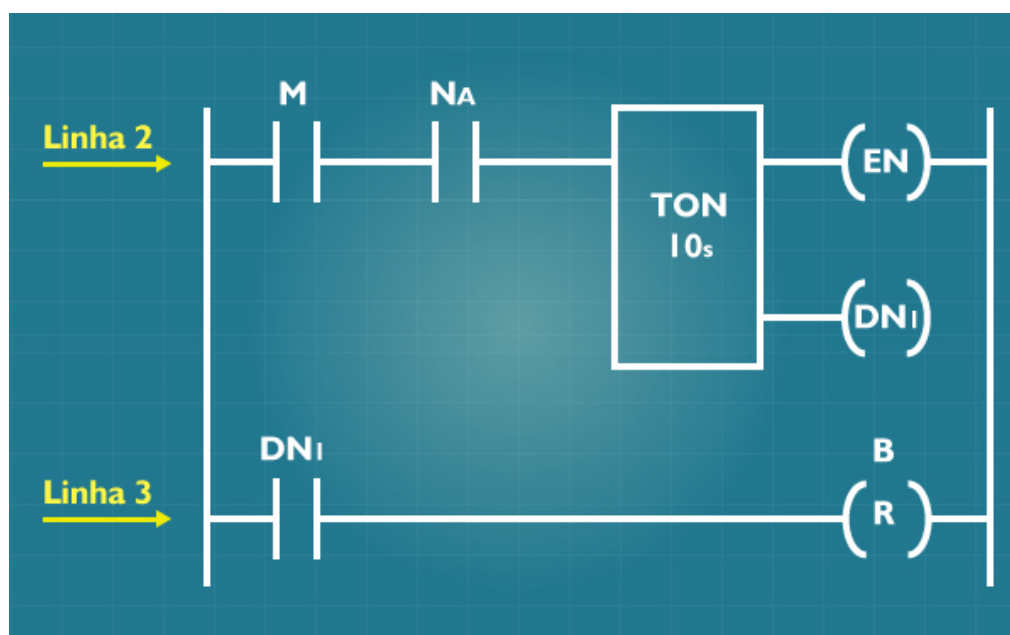


Fonte: Elaborado pelo Autor (2014).

Não se esqueça de como funciona a rotina de um CLP: as entradas físicas são lidas pelo dispositivo e armazenadas na memória (através de variáveis internas como M, por exemplo), em seguida, as ações são executadas com base no programa e os resultados salvos na memória (através de variáveis internas como B, por exemplo), e por fim, as saídas físicas são atualizadas.

Agora, vamos lidar com o modo **automático** do sistema, desenvolvendo as **linhas 2 e 3** (Figura 30) do programa. A linha 2 será responsável por acionar o nosso primeiro temporizador, caso o modo atual do sistema seja o **automático**, e o sensor de nível alto esteja acionado. Dessa forma, precisamos utilizar duas chaves NA em série (chaves M e N_A), sendo uma para cada condição mencionada. Caso essas chaves estejam fechadas, o temporizador TON será energizado, iniciando o processo de contagem de 0 a 10 segundos. Ao final da temporização, a sua variável DN_1 recebe "1", podendo ser utilizada na próxima linha do programa (linha 3) para energizar o solenoide B, responsável por desligar a bomba (note o "R" no interior do solenoide).

Figura 30 - Linhas 2 e 3 do diagrama Ladder para o controle de nível em um reservatório.

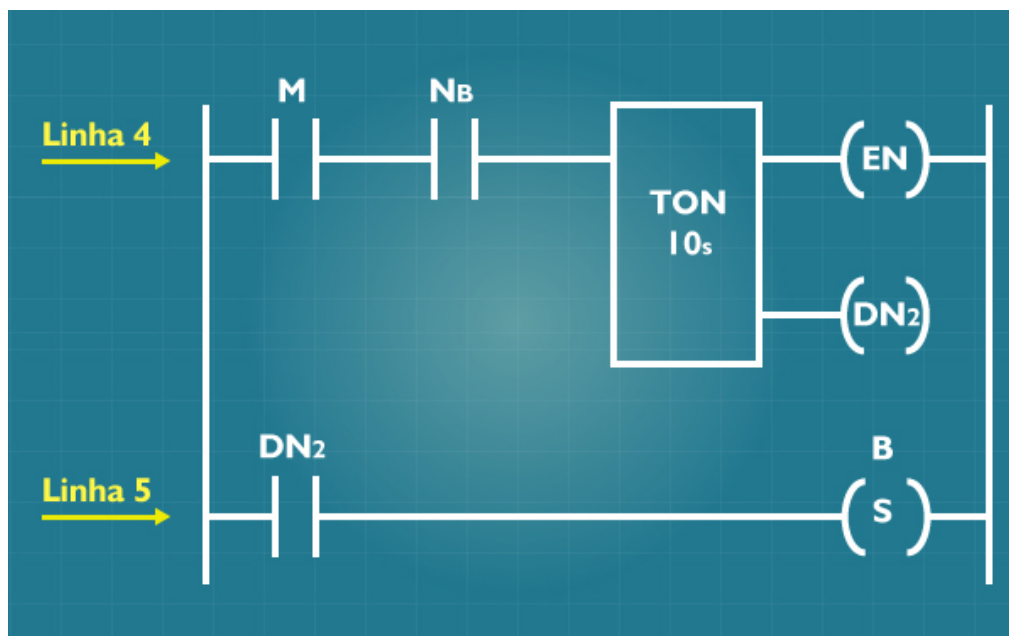


Fonte: Elaborado pelo Autor (2014).

Por fim, vamos novamente lidar com o modo **automático** do sistema, desenvolvendo as **linhas 4 e 5** (Figura 31) do programa. A linha 4 será responsável por acionar o nosso **segundo** temporizador, caso o modo atual do sistema seja o **automático**, e o sensor de nível **baixo** esteja acionado. Dessa forma, precisamos utilizar duas chaves NA em série (chaves M e N_B), sendo uma para cada condição

mencionada. Caso essas chaves estejam fechadas, o temporizador TON será energizado, iniciando o processo de contagem de 0 a 10 segundos. Ao final da temporização, a sua variável DN_2 recebe "1", podendo ser utilizada na próxima linha do programa (linha 5) para energizar o solenoide B, responsável por ligar a bomba (note o "S" no interior do solenoide).

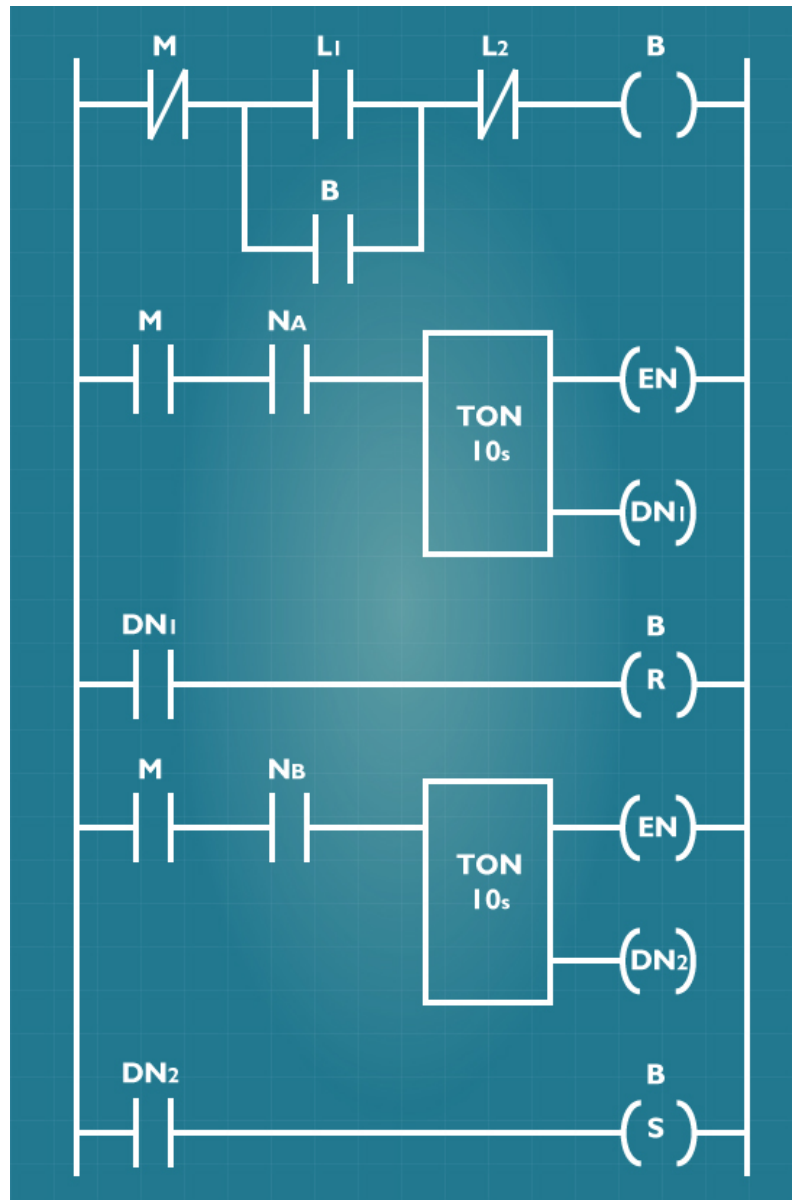
Figura 31 - Linhas 4 e 5 do diagrama Ladder para o controle de nível em um reservatório.



Fonte: Elaborado pelo Autor (2014).

O diagrama Ladder completo é apresentado na Figura 32. Note que ele nada mais é do que a união de todas as linhas apresentadas anteriormente nas Figuras 29, 30 e 31.

Figura 32 - Diagrama Ladder completo para o controle de nível em um reservatório.



Fonte: Elaborado pelo Autor (2014).

Atividade 08

1. Para o exemplo da Figura 27 (controle de nível em um reservatório), desenvolva um novo diagrama Ladder com base no método do GRAFCET.

Resumo

Nesta aula, fizemos uma pequena introdução ao Controlador Lógico Programável (CLP). A partir do seu antecessor, o relé eletromecânico, apresentamos as vantagens e as desvantagens dessa nova tecnologia, assim como as suas possíveis aplicações na indústria. Vimos ainda que o CLP pode ser classificado de acordo com a sua capacidade de expansão. Por fim, falamos um pouco sobre o hardware e o software desse dispositivo. Aprendemos a lidar com os circuitos elétricos de comando e com a linguagem de programação Ladder. Inicialmente, utilizamos os circuitos elétricos de comando para obter os circuitos em Ladder de uma forma bem simples. Apesar de mais direta, o circuito em Ladder pode ser mais trabalhoso, principalmente em processos industriais complexos. Uma alternativa é a utilização do método do GRAFCET, como forma de sistematizar essa atividade, sem a dependência prévia de um circuito de comando. Trabalhamos com duas aplicações bem interessantes: o avanço e recuo de um cilindro de dupla ação, e o controle de nível em um reservatório. Na primeira, desenvolvemos um diagrama Ladder com base no método do GRAFCET e, na segunda, um diagrama sem o apoio desse método, nos preocupando apenas com a implementação da lógica de comando. O objetivo foi apresentar diferentes abordagens para o acionamento dos solenoides, além de um novo elemento, o temporizador TON.

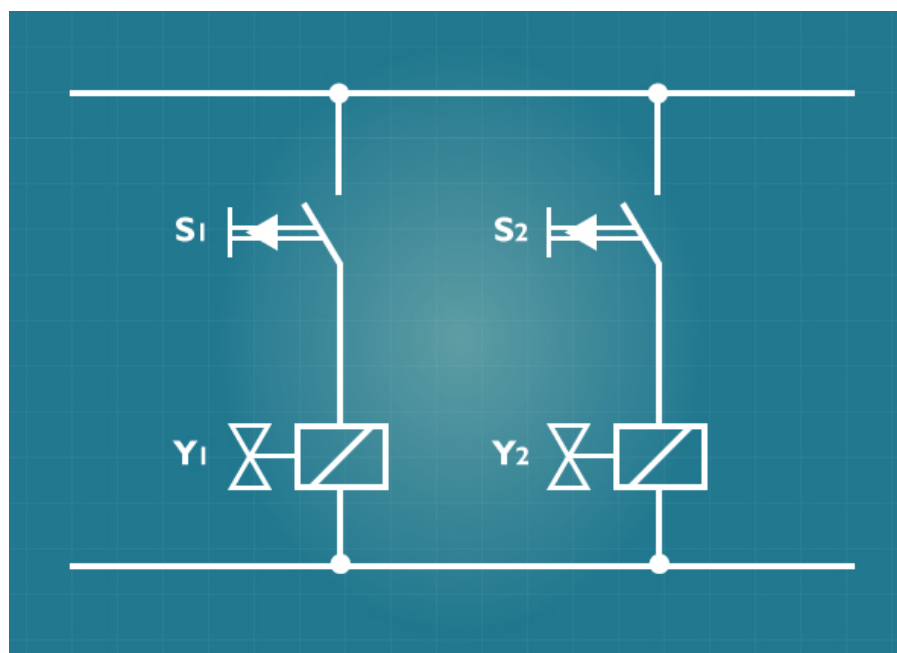
Autoavaliação

1. Descreva o que é um CLP.
2. Quais são as vantagens e desvantagens do CLP?
3. Apresente algumas aplicações do CLP na indústria.
4. Considere um sistema de controle de nível de um reservatório. Suponha ainda que a cada 50 cm do reservatório, existe um sensor que indica a presença ou não de líquido, num total de 4. Deseja-se utilizar um CLP para realizar o controle de nível deste reservatório, abrindo ou fechando totalmente duas válvulas diferentes: uma de admissão e outra de consumo. Liste quais são as entradas e saídas do CLP, além da classificação dos módulos E/S a serem utilizados. Desenhe um diagrama

de blocos do sistema, apresentando o reservatório, sensores, atuadores e o próprio CLP.

5. Suponha que um cilindro de dupla ação é acionado por duas botoeiras distintas: uma (S1) para fazer o cilindro avançar; outra (S2), para recuar. O avanço e retorno do cilindro é realizado por um pulso em S1 e S2, respectivamente. Caso as duas botoeiras sejam pressionadas ao mesmo tempo, a válvula mantém a última posição de avanço ou recuo, evitando, assim, a queima do dispositivo. A Figura 33, a seguir, apresenta o circuito de comando para uma válvula 5/2, com dois solenoides (Y1 e Y2). Construa um diagrama Ladder a partir do circuito apresentado.

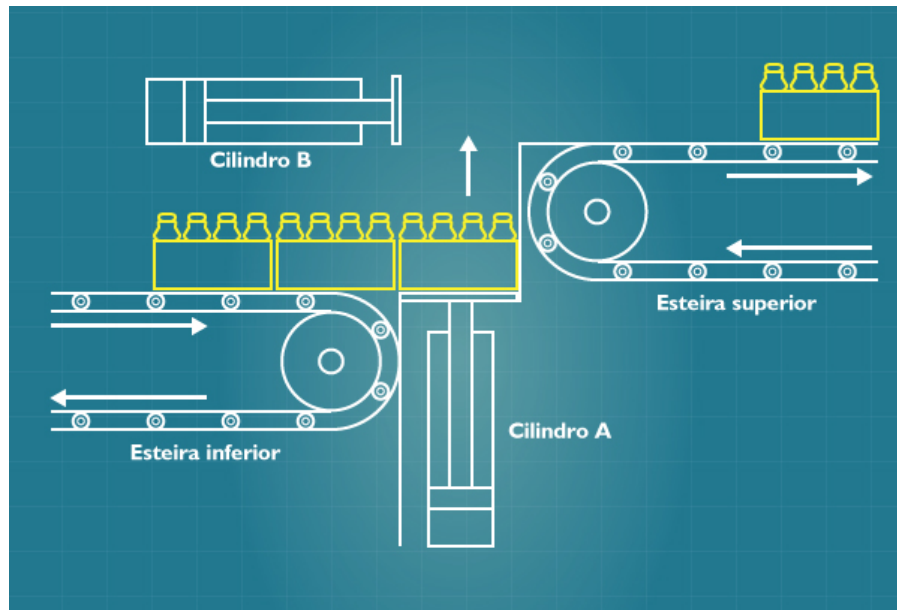
Figura 33 - Circuito de comando para um cilindro de dupla ação e válvula 5/2, com dois solenoides.



Fonte: PARKER TRAINING (2005).

6. Considere novamente o exemplo da Figura 33. Apresente um novo circuito de comando, caso seja utilizada uma válvula 5/2 com um único solenoide. Construa um diagrama Ladder a partir desse circuito.
7. Com base no método do GRAFCET, desenvolva um diagrama Ladder para a aplicação abaixo (ciclo de trabalho: $A^+ \rightarrow B^+ \rightarrow A^- \rightarrow B^-$). Compare o diagrama desenvolvido aqui com o apresentado na Figura 10, mostrado anteriormente.

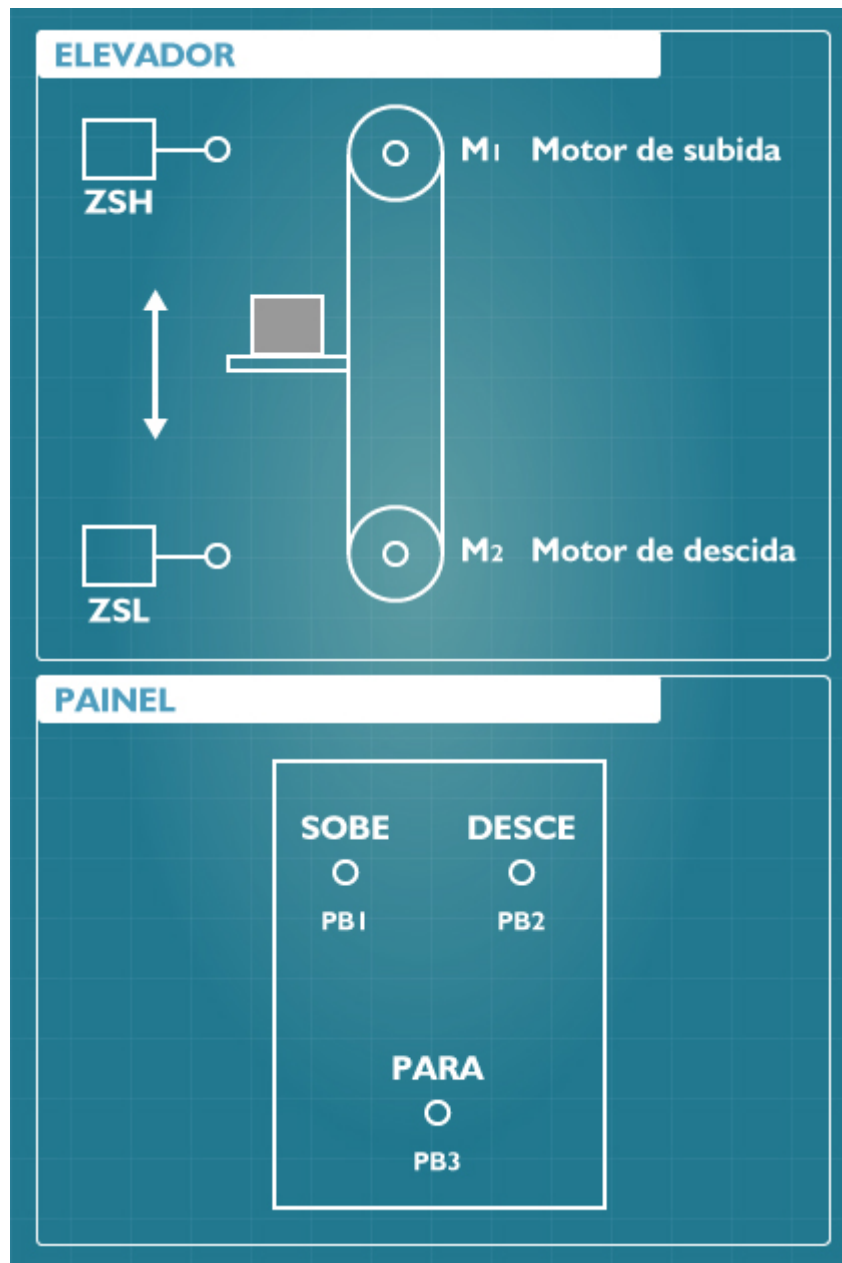
Figura 34 - Aplicação do uso de cilindros na indústria.



Fonte: Maitelli (2005).

8. Desenvolva um diagrama Ladder (utilizando o método do GRAFCET) para automatizar um elevador de carga (figura abaixo) composto por dois motores (M1 e M2), dois sensores de fim de curso (ZSH e ZSL) e três botoeiras (PB1, PB2, PB3). O princípio de funcionamento é o seguinte: a) quando a botoeira PB1 for acionada, a plataforma de carga deve subir, caso ela não esteja descendo, ou não esteja no seu limite superior (ZSH acionado); b) quando a botoeira PB2 for acionada, a plataforma de carga deve descer, caso ela não esteja subindo, ou não esteja no seu limite inferior (ZSL acionado); c) quando a botoeira PB3 for acionada, a plataforma deve parar na posição em que ela estiver.

Figura 35 - Aplicação do uso de cilindros na indústria.



Fonte: Maitelli (2005).

9. Desenvolva um novo diagrama Ladder para automatizar o elevador de carga da Figura 31, sem utilizar o método do GRAFCET.

Referências

BOLTON, W. **Programmable logic controllers**. 5. ed. USA: Elsevier, 2009.

CORRADI JÚNIOR, R. **Comandos Elétricos**. [S. l.]: COTUCA/UNICAMP [20??].

MAITELLI, A. L. **Notas de aula da disciplina controladores lógicos programáveis**. Natal: DCA/UFRN, 2005.

MORAES, C. C.; CASTRUCCI, P. **Engenharia de automação industrial**. 2. ed. São Paulo: LTC, 2007.

PARKER TRAINING. **Tecnologia eletropneumática industrial**. 1. ed. São Paulo: Parker Training Brasil, 2005

ROSÁRIO, J. M. **Princípios de mecatrônica**. 1. ed. São Paulo: Prentice Hall, 2005