

Programa o de CLPs

Aula 04 - Programa o de CLP em Ladder usando Temporizadores e Contadores

Apresentação

Nesta aula continuaremos com exemplos de aplicações, em Ladder, envolvendo temporizadores e contadores. Neste tópico perceberemos que essas aplicações estão em um nível um pouco mais avançado do que foi visto, pois à medida que colocamos condições na operação do sistema mais linhas de comando serão necessárias, bem como a atenção do programador será redobrada. Para tanto, serão abordados sistemas que requerem uso tanto de temporizadores quanto de contadores, por meio da plataforma da Siemens e também da plataforma de programação do Automation Studio.

Objetivos

- Desenvolver aplicações em Ladder, no TIA Portal da Siemens;
- Utilizar temporizadores em aplicações Ladder;
- Utilizar contadores nas aplicações em Ladder;
- Entender o funcionamento de sistemas mais robustos.

Introdução

Tendo visto alguns exemplos de programação em ladder na aula passada, já estamos aptos a programar sistemas que requerem um pouco mais de raciocínio por parte do programador, e nesse contexto abordaremos circuitos que envolvem aplicações com temporizadores e contadores.

No que tange a temporizadores foi visto, também na aula passada, o comportamento de cada temporizador apresentado na plataforma Siemens, porém observe agora as aplicações que esses temporizadores acrescentam em um sistema a ser automatizado, pois se trata de uma variável a mais. E atente para o fato de que quanto mais condições há no programa mais robusto ele fica.

Em alguns momentos há a necessidade de fragmentar o programa, em outras palavras, organizá-lo em seções, de maneira que uma eventual falha na programação seja de fácil diagnóstico. Para isso utilizaremos mais de um ambiente de programação, que no caso da Siemens, seriam várias "network's". Essas network's já estão presentes quando você abre o ambiente de programação, em que no início era suficiente trabalharmos em apenas uma network. No entanto, desta vez vamos trabalhar com mais de uma no intuito de organizar a programação.

Fazendo a mesma analogia surgem os contadores e suas serventias em aplicações industriais, bem como suas aplicabilidades. Seu modo de funcionamento também é bem simples de entender, seja na contagem crescente ou decrescente. Veremos algumas aplicações com os mesmos.

Apresentando uma abordagem diferente nesta aula, utilizaremos um ambiente de programação que faz uso de um software de simulação virtual chamado "Automation Studio". Ele permite que após a montagem do programa, em ladder, se faça uma simulação virtual de como irá se comportar o sistema proposto. Nesse software há suporte para CLP's da Siemens, o que facilita o entendimento. Nesse ambiente montaremos as interfaces (entradas e saídas) e seus sinais de tensões, a seguir elaboraremos o programa e por fim simularemos.

Portanto, ao iniciarmos esta aula daremos continuidade aos circuitos de acionamentos utilizando temporizadores, em seguida contadores. Sempre com duas soluções: uma no TIA Portal e outra no software Automation Studio.

Circuito Pisca-Pisca

Vamos desenvolver uma aplicação em ladder para um sistema tipo pisca-pisca, conforme mostra a figura 1, com o intuito de entendermos um circuito com “ciclo fechado”, e também para perceber como funcionam sistemas que estão intertravados ou interligados, já que um depende do outro para funcionar.

Para começar devemos respeitar as seguintes considerações:

- a. Ao pressionar o botão “Start” o sistema pisca-pisca inicia seu funcionamento, ligando a “lâmpada 1”;
- b. Após 5s da *lâmpada* 1 ligada, a mesma apagará e ascenderá a *lâmpada* 2;
- c. Após 5s da *lâmpada* 2 ligada, ela apagará e reinicia o sistema ligando a lâmpada 1, fechando assim o ciclo.
- d. Sob nenhuma hipótese as duas lâmpadas deverão ficar acesas.
- e. O botão de “stop” desliga todo o sistema.

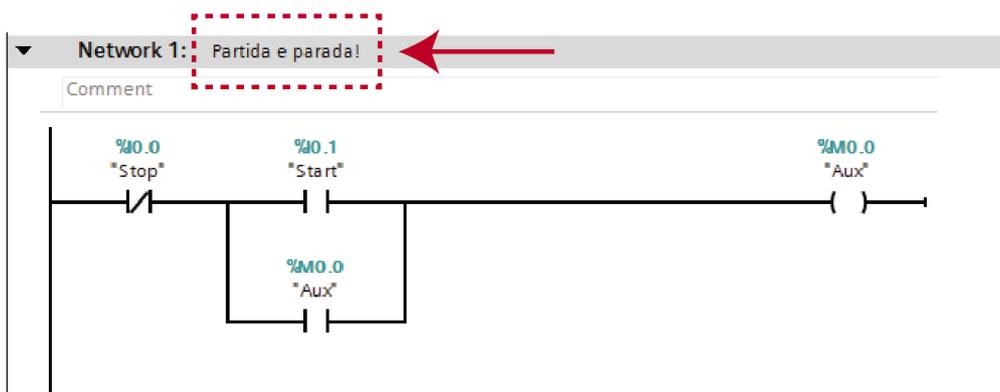
Figura 01 - Imagem de uma sugestão para pisca-pisca.



Fonte: Adaptado de <https://www.google.com.br/search?q=sinalizador+pisca+pisca&imgsrc=7qUpmBnVHcrCIM%3A> Acesso em: 10 jan. 2016.

Para o desenvolvimento do ladder vamos fazer um pouco diferente agora. Vamos inserir comentários, bem como fragmentar a programação para futuras organizações. Para isto o programa nos dá a possibilidade de criarmos vários ambientes de programação. Portanto, no espaço da "network 1" vamos fazer o sistema de "Start" e "Stop", fazendo o comentário de partida e parada, conforme mostra a figura 2.

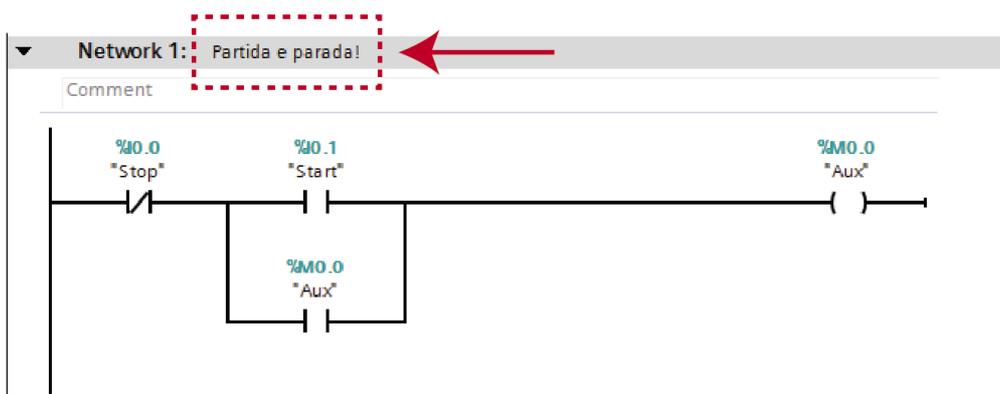
Figura 02 - Network 1 de partida e parada do sistema.



Note que estamos usando um relé auxiliar (com retenção) para ativar e desativar os temporizadores que farão a função pisca.

Na network 2 vamos montar o circuito com *temporizadores* tipo *TP* e montá-los como disposto na figura 3.

Figura 03 - Network 2 com sistema pisca-pisca, utilizando temporizadores.

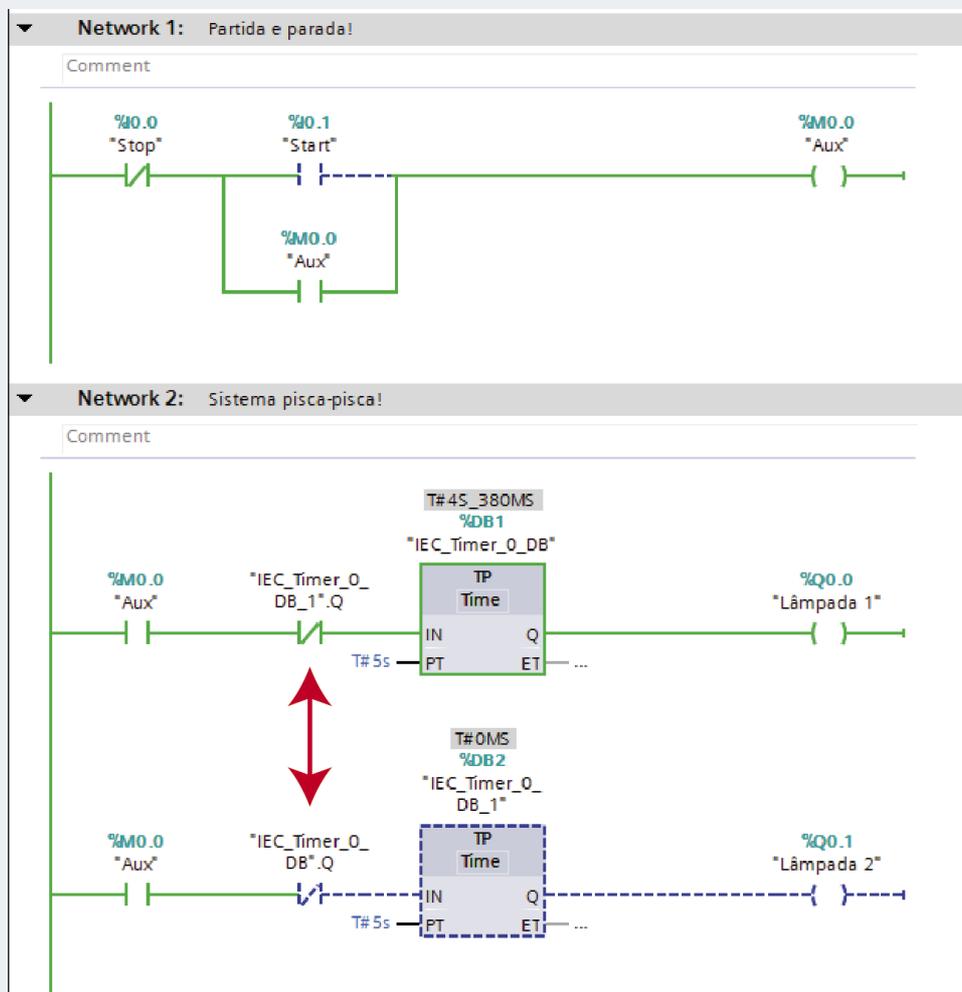


Depois de montado o sistema, feito o download para o CLP e colocado o CLP no modo "online", o sistema funcionará da seguinte maneira:

“Ao pressionar o botão start, o mesmo ativa a bobina do relé auxiliar (Aux), em seguida fecha sua retenção (contato aberto de Aux em paralelo com start). É importante observar o intertravamento entre os temporizadores, serão eles que impedirão que os dois entrem simultaneamente.

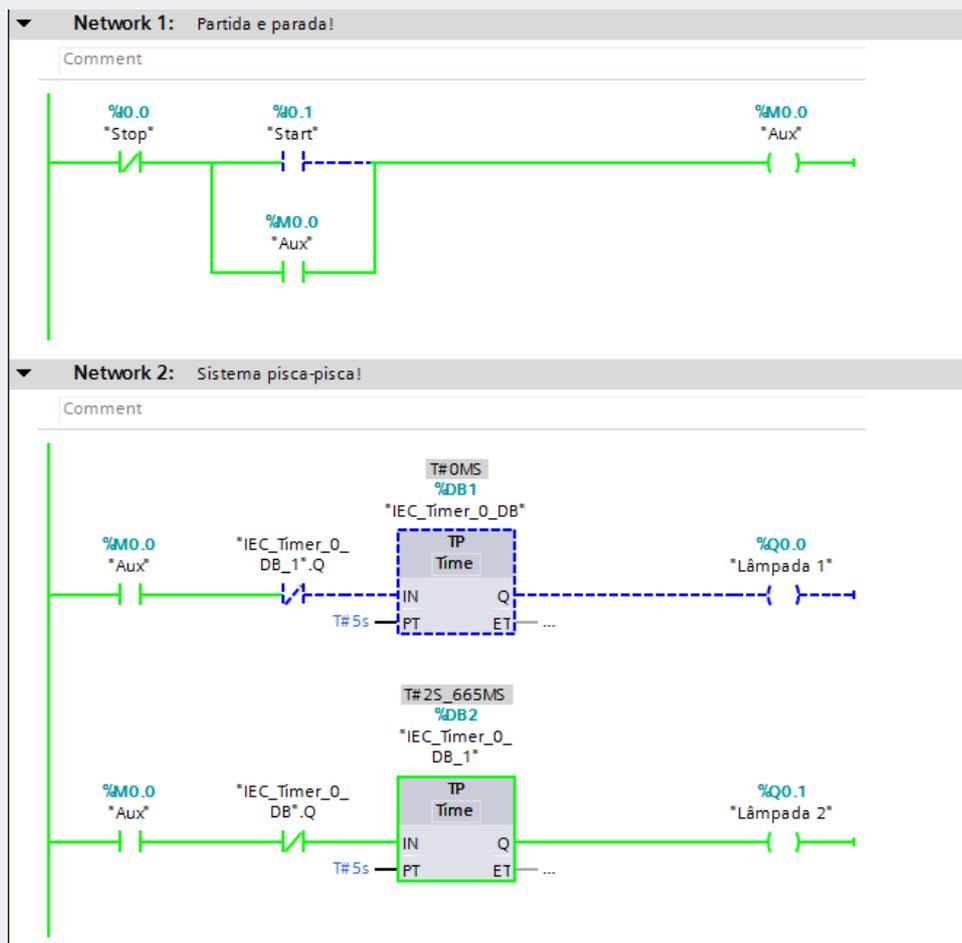
Quando o contato aberto do relé auxiliar fechar, o temporizador %DB1 ativará a lâmpada 1 (conforme mostra a figura 4). Neste mesmo instante o contato fechado do temporizador %DB1 será aberto garantindo que o sistema inicie sem que o temporizador %DB2 seja ativado.

Figura 04 - Detalhe do modo online no instante que inicia o sistema pisca-pisca.



Após a contagem do tempo de %DB1 a lâmpada 1 será desativada e o contato fechado (outrora aberto) do intertravamento é quem ativará %DB2 ativando a lâmpada 2 e abrindo o intertravamento em %DB1, conforme mostra a figura 5.

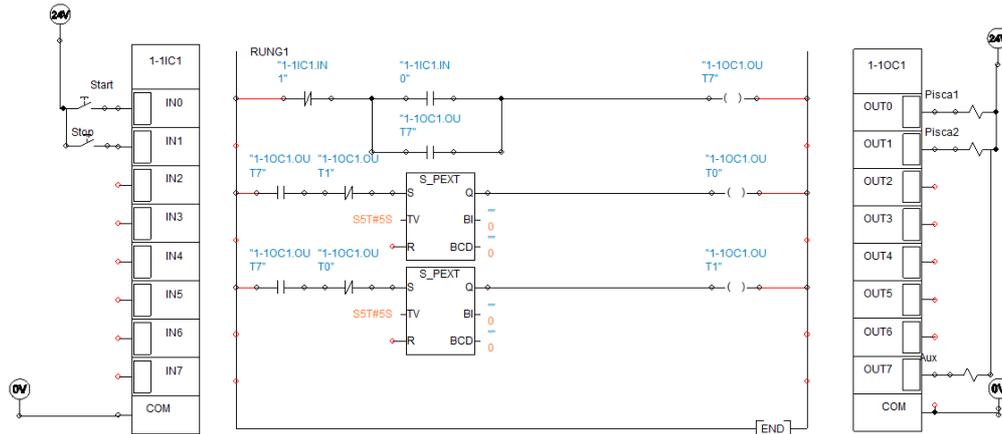
Figura 05 - Detalhe do modo online no momento que a lâmpada 2 é ativada.



E por último, após o tempo da lâmpada 2 ativada, ela apagará e iniciará um novo ciclo dando um pulso no temporizador %DB1 e assim consecutivamente até que o botão de stop seja acionado.”

Agora vejamos a solução do sistema pisca-pisca no ambiente de programação do Automation Studio, conforme a figura 6. Neste caso não foi necessário “seccionar” a programação:

Figura 06 - Sistema pisca-pisca no Automation Studio.



Observação

Para realizar uma programação no Automation Studio temos que “montar” um sistema básico formado por um bloco de entradas conforme a figura 7, e um bloco de saídas conforme a figura 8, com suas devidas alimentações de sinais. Em seguida, esse software permite simular o programa desenvolvido.

Veja também que foi utilizado o mesmo modelo de temporizador utilizado no TIA Portal (tipo TP).

Figura 07 - Bloco de entradas com 8 entradas digitais.

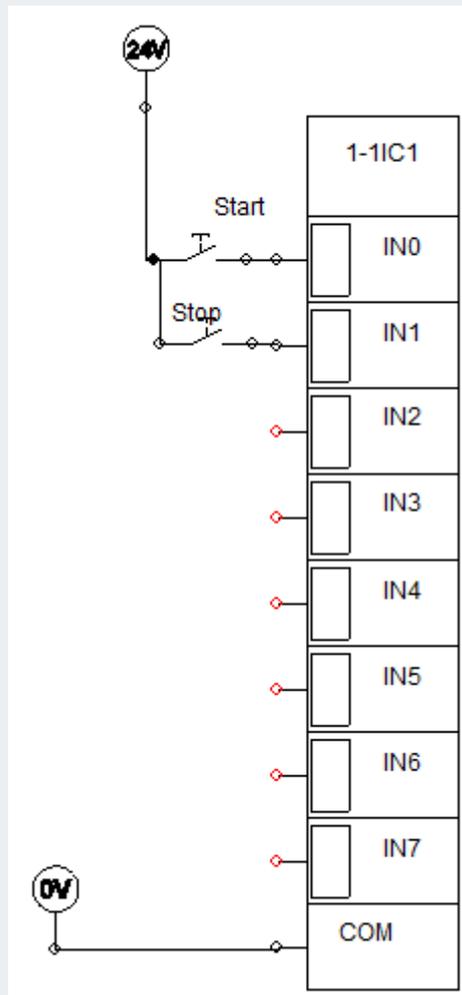
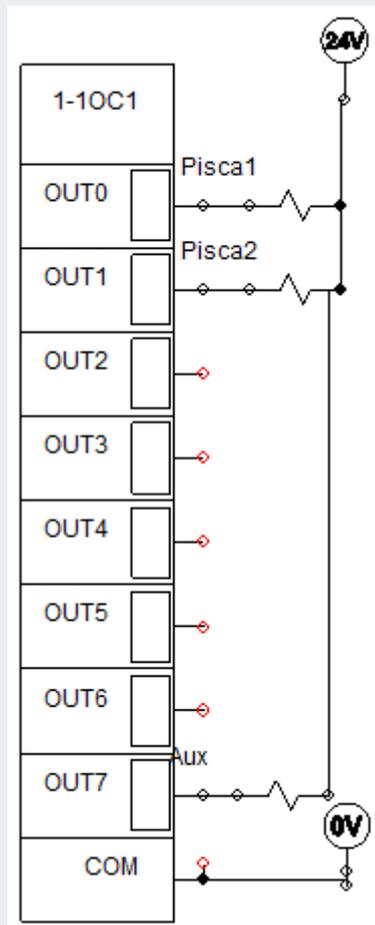


Figura 08 - Bloco de saídas com 8 saídas digitais.



Circuito de um Semáforo Simples

Agora que já sabemos como trabalhar em sistemas com ciclos fechados, vamos desenvolver uma aplicação em ladder para acionar um semáforo simples, conforme mostra a Figura 9 abaixo.

Figura 09 - Imagem ilustrativa de um semáforo simples.



Fonte: Adaptado de <https://www.google.com.br/search?q=desenho+semaforo+png&imgsrc=llQGEufUDrI6-M%3A> Acesso em: 12 jan. 2016.

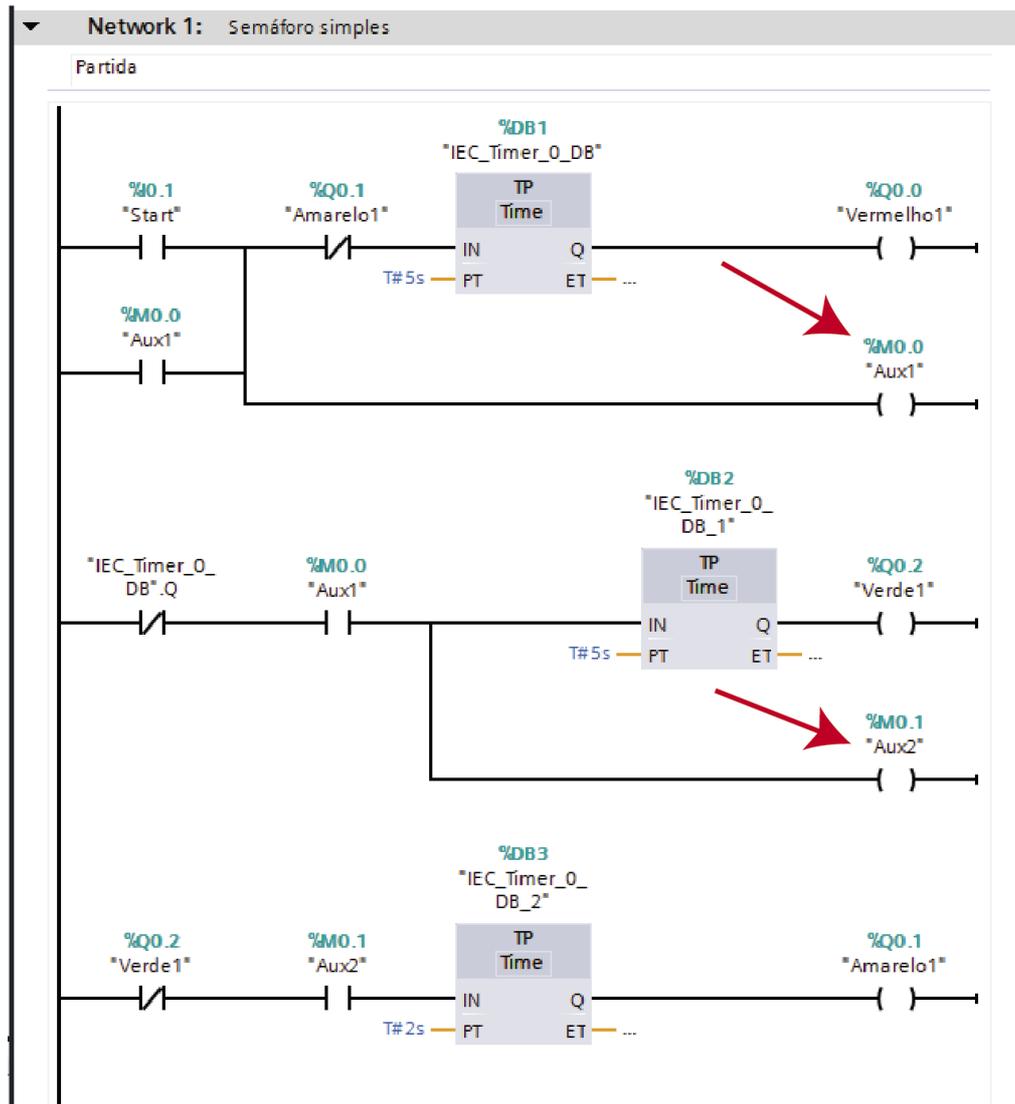
Bom, esse está mais fácil, tendo em vista que todos nós sabemos como é o funcionamento de um semáforo, pois então vamos lá:

- a. Ao pressionar "Start" inicia-se o processo ligando a lâmpada vermelha por 5s;
- b. Passado o tempo, desliga-se a lâmpada vermelha e liga a lâmpada verde também por 5s;
- c. Após os 5s da lâmpada verde, a mesma apaga, e liga a lâmpada amarela por 2s;
- d. Passado os 2s da lâmpada amarela um novo ciclo é iniciado (Vermelho, verde e amarelo...) até que o botão de stop seja acionado.

Vamos fazer a programação inserindo os comentários e fragmentando o programa, ou seja, desmembrando em duas partes, a partida e a parada. Só que na parada vamos utilizar bobinas reset's como uma opção a mais na programação.

Para o semáforo iremos utilizar três temporizadores do tipo TP (%DB1 para o vermelho, %DB2 para o verde e %DB3 para o amarelo), duas botoeiras (start e stop) e dois relés auxiliares, contudo montemos o circuito conforme mostra a figura 10.

Figura 10 - Detalhe da montagem dos contatos e temporizadores do semáforo simples.



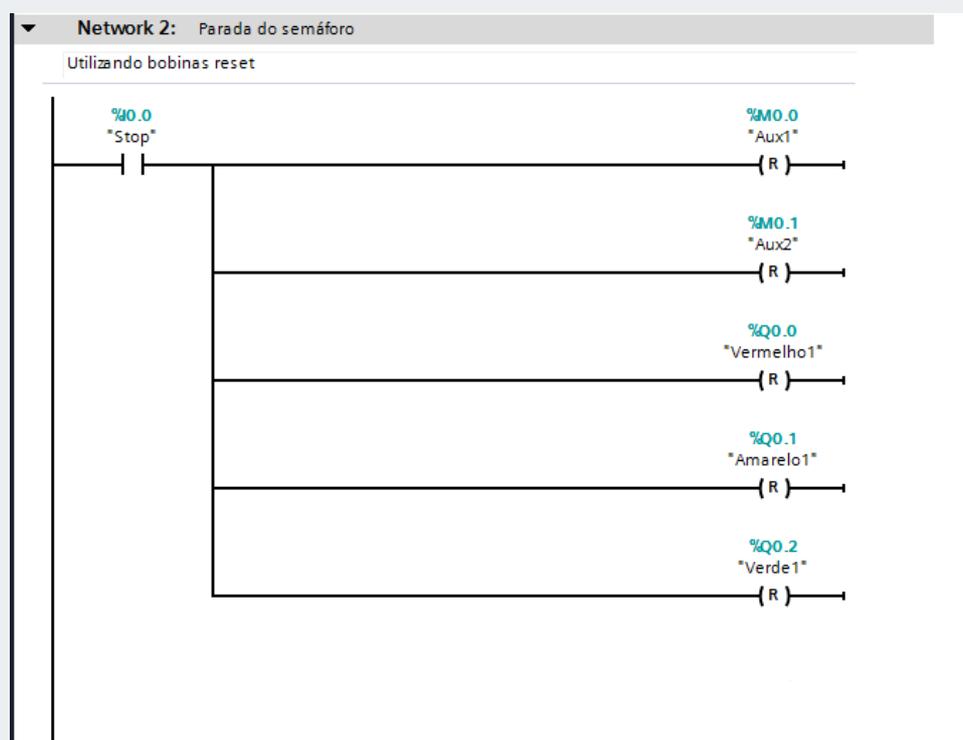
Observe que desta vez estamos utilizando duas retenções (*Aux1* e *Aux2*) para manter os sinais ativos. Portanto:

“Ao pressionar start o semáforo é ativado iniciando pelo temporizador %DB1 que liga a lâmpada vermelha e ativa a bobina de Aux1, bem como seus intertravamentos. Após 5s o contato fechado do intertravamento ativa o temporizador %DB2, desligando o vermelho e ligando a lâmpada verde.

Por sua vez, o verde ativa o segundo relé auxiliar (Aux2) para intertravar o temporizador %DB3. Com isso, logo após a contagem do tempo a lâmpada verde apagará e ativará o temporizador %DB3 da lâmpada amarela. É a lâmpada amarela que fechará o próximo ciclo, através de seu contato fechado em série com a entrada do temporizador %DB1.

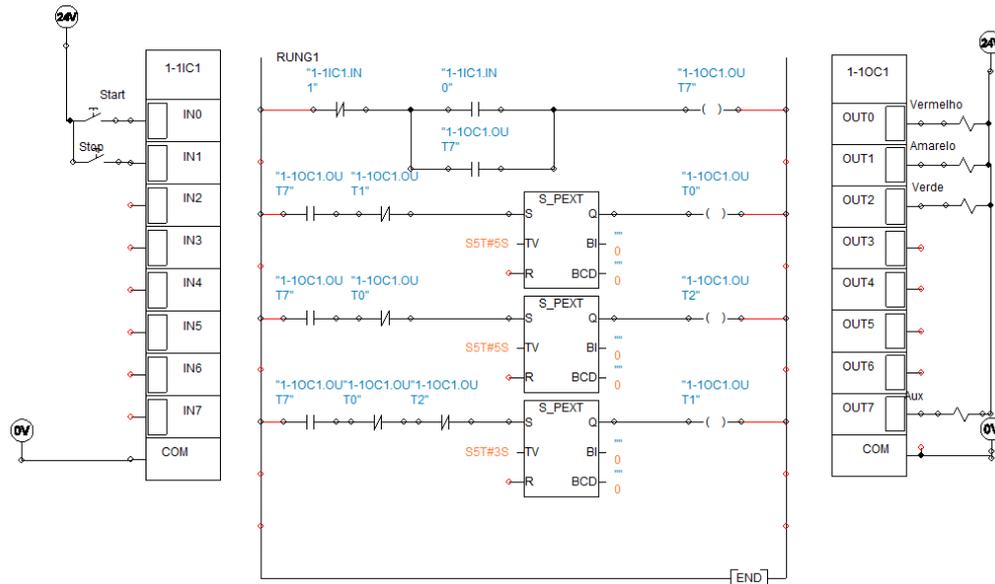
Para desligar o sistema utilizamos bobinas configuradas como reset, conforme mostra a montagem da network 2 na figura 11. Observe que nessa network está a programação para apenas desativar o sistema, ou seja, ele desativará todas as bobinas.”

Figura 11 - Detalhes da configuração dos reset's para desligar o sistema.



Vejamos agora como fica a solução do sistema do semáforo simples na ferramenta do Automation Studio, conforme a Figura 12:

Figura 12 - Sistema do semáforo simples implementado no Automation Studio.



Obsevação

Os blocos de entradas e saídas são os mesmos, porém desta vez foram usadas 4 saídas.

Contadores

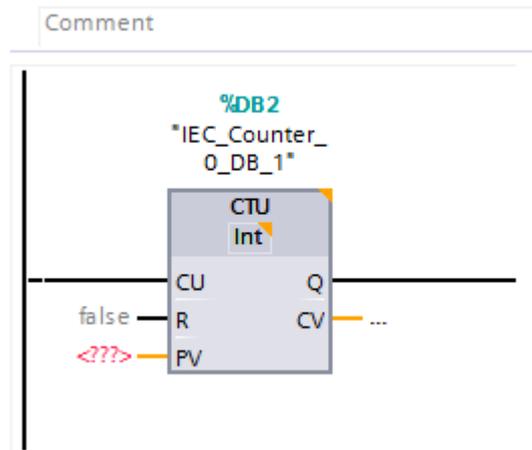
Os contadores usados nos CLP's comparam valores ajustados, de acordo com a operação e valores preestabelecidos, para controlar eventos de um modo geral. Pode ser tanto de contagem positiva (incremento "+1") quanto de contagem negativa (decremento "-1").

Na plataforma Siemens são três os tipos de contadores encontrados.

Contador Crescente (CTU)

Neste tipo de contador (conforme a figura 13) os impulsos originados na entrada do contato "CU" provoca um incremento do registro do contador.

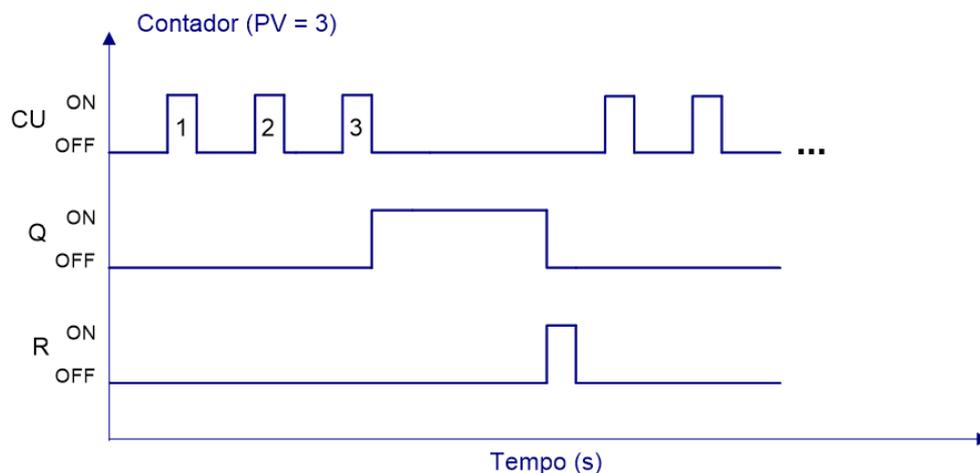
Figura 13 - Imagem do bloco contador CTU da Siemens.



A saída “Q” fica desativada (OFF) enquanto a contagem não atinge o valor estabelecido em “PV”. Quando a contagem chega ao valor de PV a saída Q é ativada e permanece assim até que seja dado um pulso no contato “R”, conforme mostra o gráfico da figura 14.

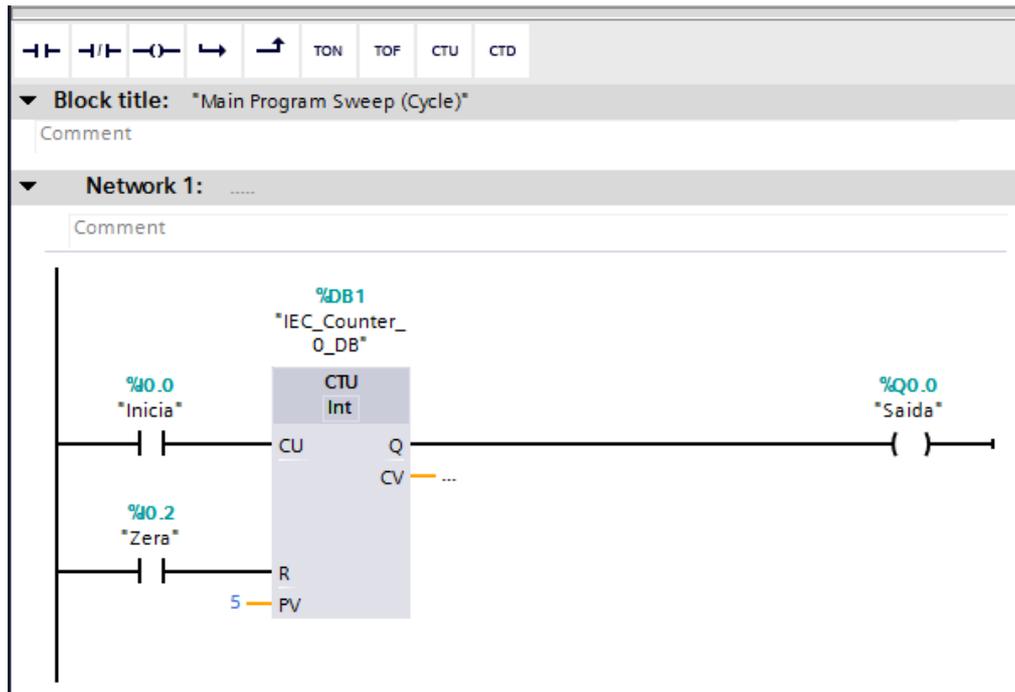
Caso o contador continue a receber os impulsos para contagem, ele continuará contando, mas sua saída “Q” não irá alterar seu estado, ou seja, ficará ativado.

Figura 14 - Gráfico de operação do contador tipo CTU da Siemens.



Como exemplo observe a figura 15 que mostra quando uma saída é ativada depois de executar uma contagem de 5 impulsos.

Figura 15 - Exemplo de aplicação para uma contagem utilizando o contador tipo CTU.



Portanto:

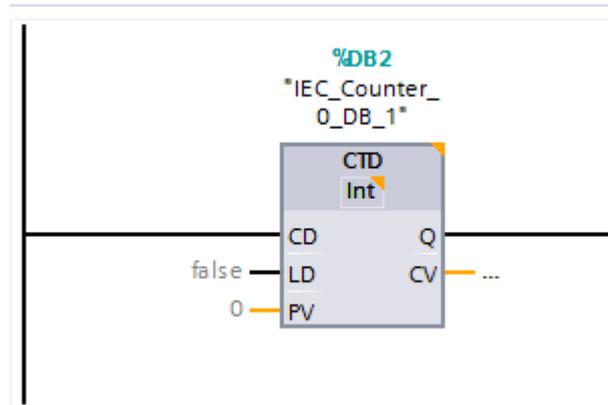
“Ao pressionar a botoeira “inicia” o contador começa a contagem com o 1º evento e por aí vai. No início do 5º evento ele libera sinal para a saída Q”.

“Para iniciar uma nova contagem basta dar um pulso no botão “zera” e ele ficará apto a iniciar uma nova contagem”.

Contador Decrescente (CTD)

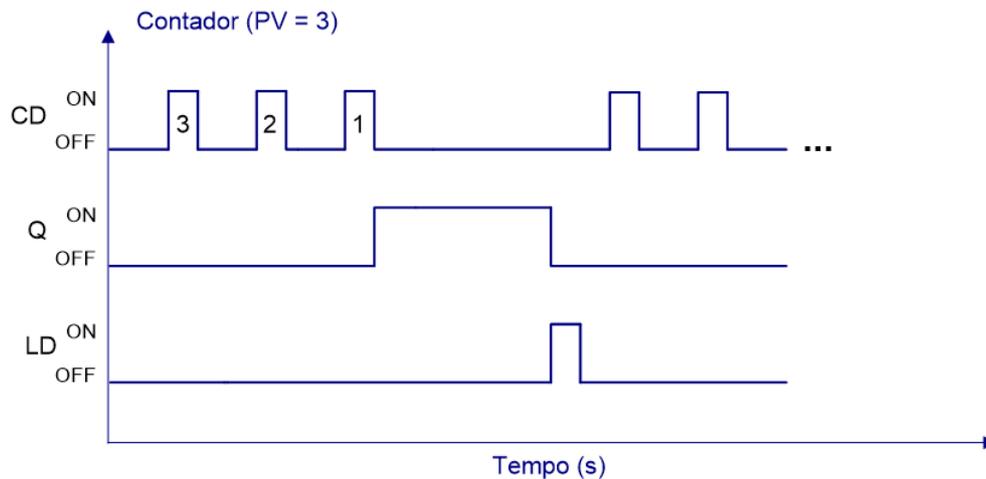
Neste tipo de contador (conforme a figura 16) os impulsos originados na entrada do contato “CD” provocam um decremento do registro do contador.

Figura 16 - Imagem do bloco contador CTD da Siemens.



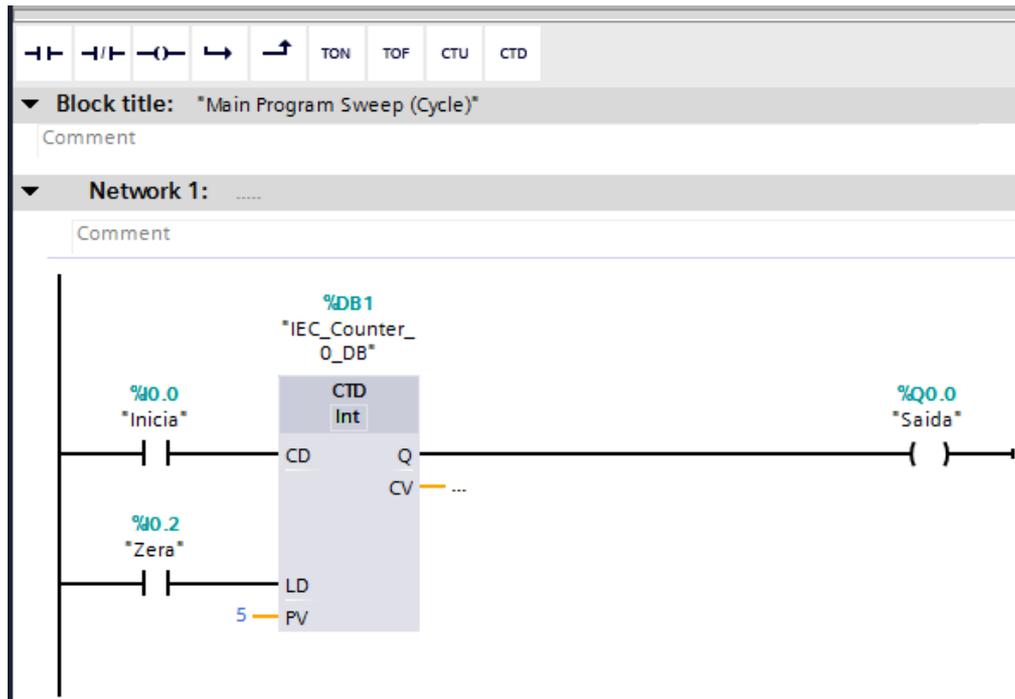
Esse modelo de contador já inicia a contagem de acordo com o valor pré-ajustado em “PV” e sua saída “Q” somente será ativada (ON) após a contagem decrescente atingir o valor “zero”, permanecendo assim até que seja dado um pulso no contato “LD” conforme mostra o gráfico da Figura 17.

Figura 17 - Gráfico de operação do contador tipo CTD da Siemens.



Como exemplo, conforme a Figura 18, veja a mesma aplicação, ou seja, deseje-se ativar uma saída utilizando desta vez um contador do tipo decrescente (CTD).

Figura 18 - Exemplo de aplicação para uma contagem utilizando o contador tipo CTD.



Portanto:

“Ao pressionar a botoeira “inicia” o contador começa a contagem com o 5º evento, no segundo pulso segue o 4º evento, no terceiro pulso segue o 3º evento e por aí vai. No início do último evento ele libera sinal para a saída Q”.

“Para iniciar uma nova contagem basta dar um pulso no botão “zera” e ele ficará apto a iniciar uma nova contagem”.

Sistema de Empacotamento

Vamos desenvolver uma aplicação em ladder para automatizar o processo de um sistema de empacotar. Para isso faça de conta que uma empresa contratou você para fazer a automação.

A máquina é uma encaixotadora de peças (conforme a figura 19) e sua programação funciona da seguinte forma:

a. Modo Manual e modo automático;

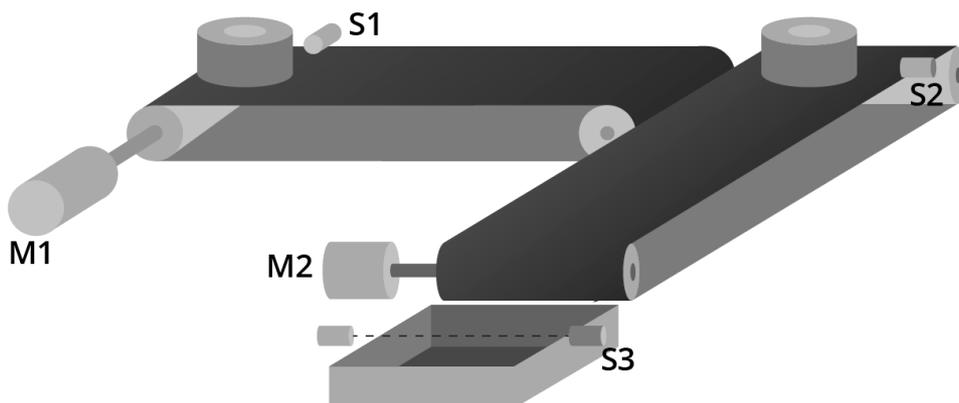
i. Modo manual:

1. As esteiras funcionam de forma independente.

ii. Modo Automático:

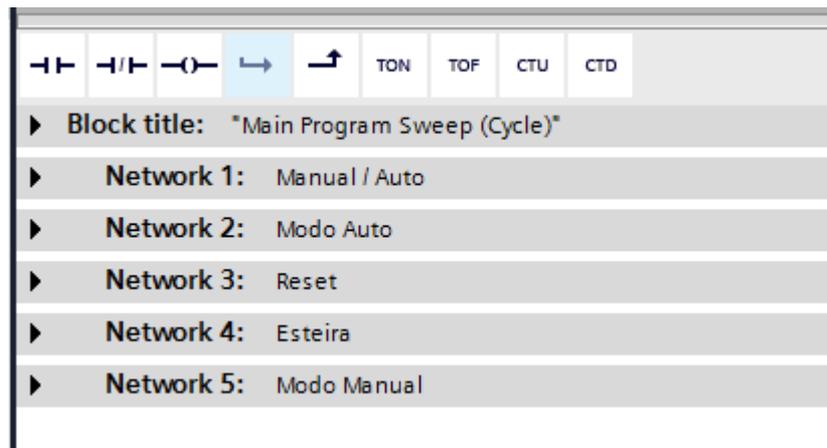
1. Ao pressionar o botão "start" a esteira M_1 aguarda a existência de peça no sensor S_1 e então liga;
2. Quando a peça chegar ao sensor S_2 a esteira M_1 desliga e liga então a esteira M_2 , até que a peça passe pelo sensor S_3 ;
3. Quando a peça passar pelo sensor S_3 desliga a esteira M_2 e conta uma peça na caixa;
4. Ao final de 5 peças o processo para. Um operador retira a caixa cheia, posiciona uma nova caixa para novamente encher de peças e pressiona o botão "reiniciar" para uma nova contagem e, assim, fechando o ciclo.

Figura 19 - Sistema de empacotamento com esteiras.



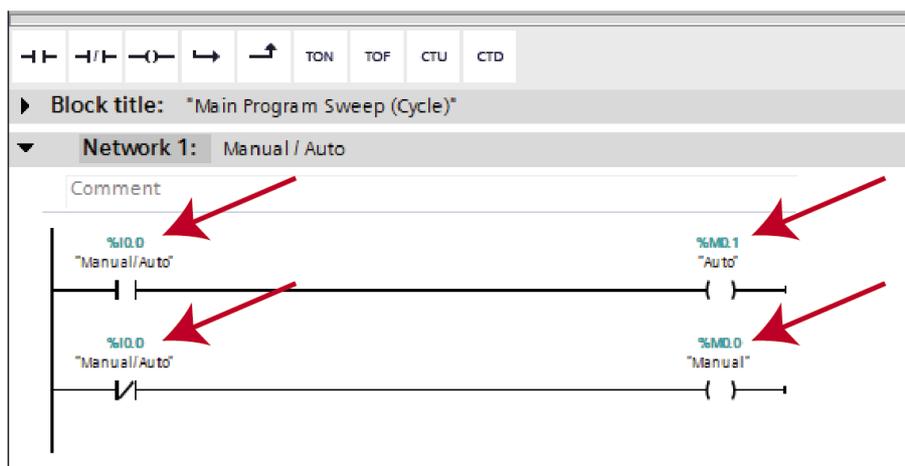
No desenvolvimento da programação Ladder desta aplicação, desmembramos o programa para deixá-lo mais organizado, de maneira que criamos 5 ambientes (5 networks) devidamente comentados, conforme mostra a figura 20.

Figura 20 - Detalhe da programação do sistema de empacotamento com os ambientes (network's) criados.



Na network 1, conforme mostra a figura 21, colocamos o modo de operação em manual ou automático para o operador escolher.

Figura 21 - Detalhe do ambiente de programação no modo de operação manual ou automático.



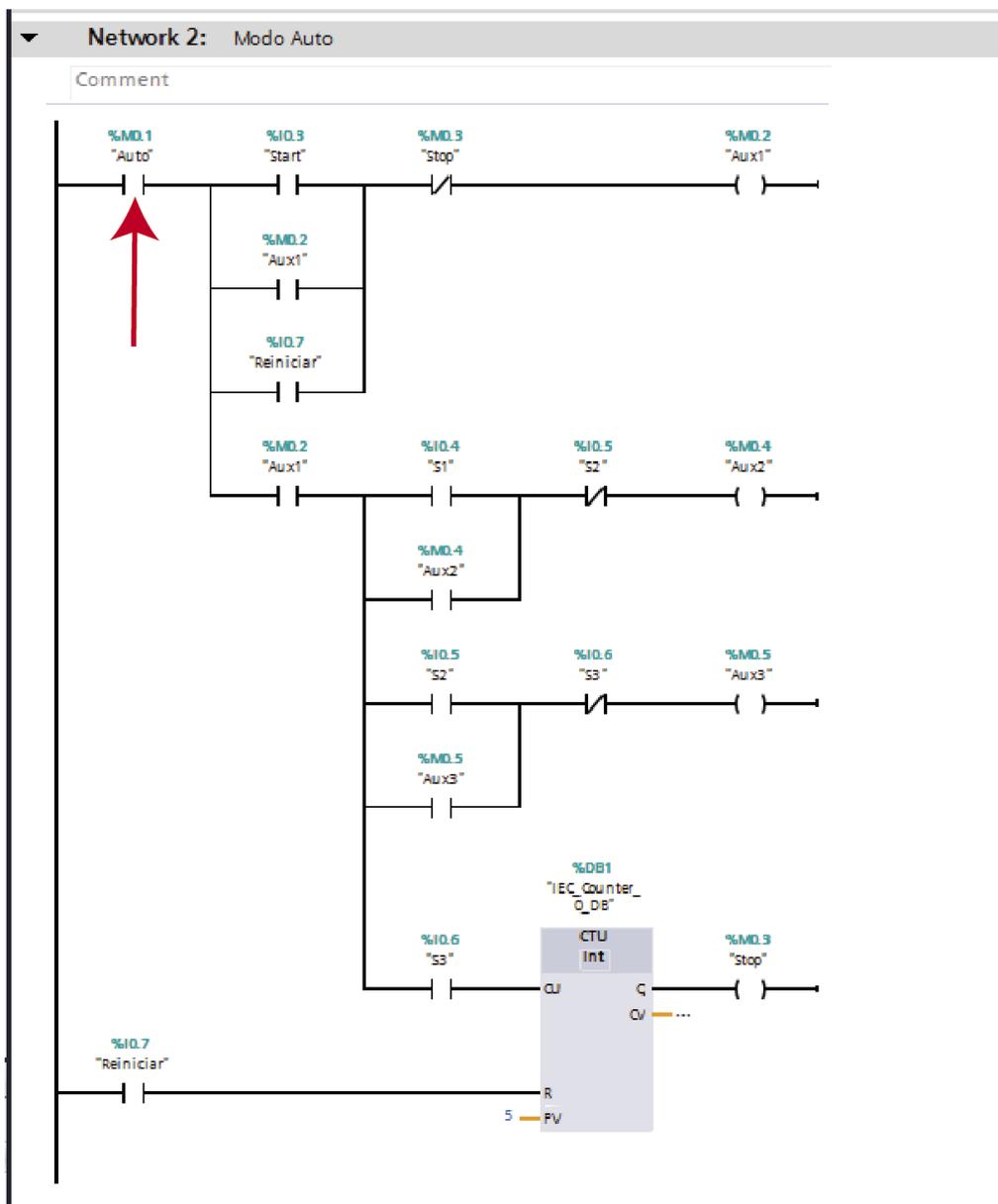
Observação

Veja que utilizamos a mesma entrada () para selecionar o modo manual ou automático. Isto é importante para o desenvolvimento do programa.

Como modo de seleção utilizamos dois relés auxiliares (e) que representam Manual e Automático, respectivamente.

Na network 2, conforme mostra figura 22, colocamos na programação as condições de operação no modo automático.

Figura 22 - Detalhe do ambiente de programação no modo de operação automático.



Observação

Detalhe do ambiente de programação no modo de operação automático.

Portanto, lê-se:

“Ao selecionar o modo Auto, o bloco de contato aberto de fecha dando permissão para que o operador dê início ao processo. Ao pressionar o botão “start”, um relé auxiliar, , fecha uma retenção que mantém o contato fechado e com isso fica aguardando o sinal do sensor S_1 para ligar a esteira 1.

Quando o sensor S_1 detecta uma caixa, ele dá um pulso de sinal para ativar o relé auxiliar o qual envia o sinal para network 4 (mais abaixo) e por sua vez ligará a esteira 1.

Quando o sensor S_2 detecta a presença da caixa, ele desativará o relé auxiliar $M_{0,5}$, que também envia o sinal para network 4, o qual ligará a esteira 2.

Com a esteira 2 ligada percebe-se que a peça irá passar pelo sensor S_3 antes de cair dentro da caixa. Desta maneira, como o sensor S_3 está ligado ao contador, ele ficará encarregado de contar as 5 peças. Fechando assim o primeiro ciclo.

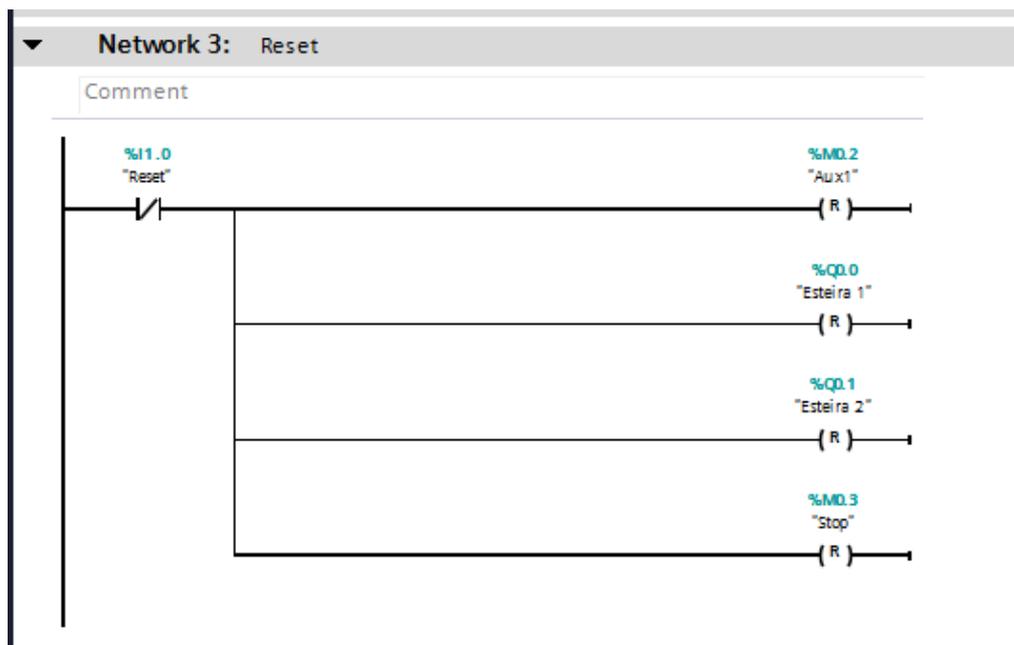
Depois virá a 2ª peça, 3ª peça e assim por diante.

Ao final da 5ª peça a saída “stop” () do contador é ativada, e seu contato fechado em série com a bobina do relé auxiliar desligará o processo.

Para recomençar outro empacotamento novamente, o operador pressiona o botão “reinicia”, que ao mesmo tempo: dará um reset no contador e ativará a bobina do relé auxiliar .”

Na network 3, conforme mostra a Figura 23, colocamos na programação os reset's das bobinas de saída da programação Ladder.

Figura 23 - Detalhe do ambiente de programação para resetar as bobinas de saídas.

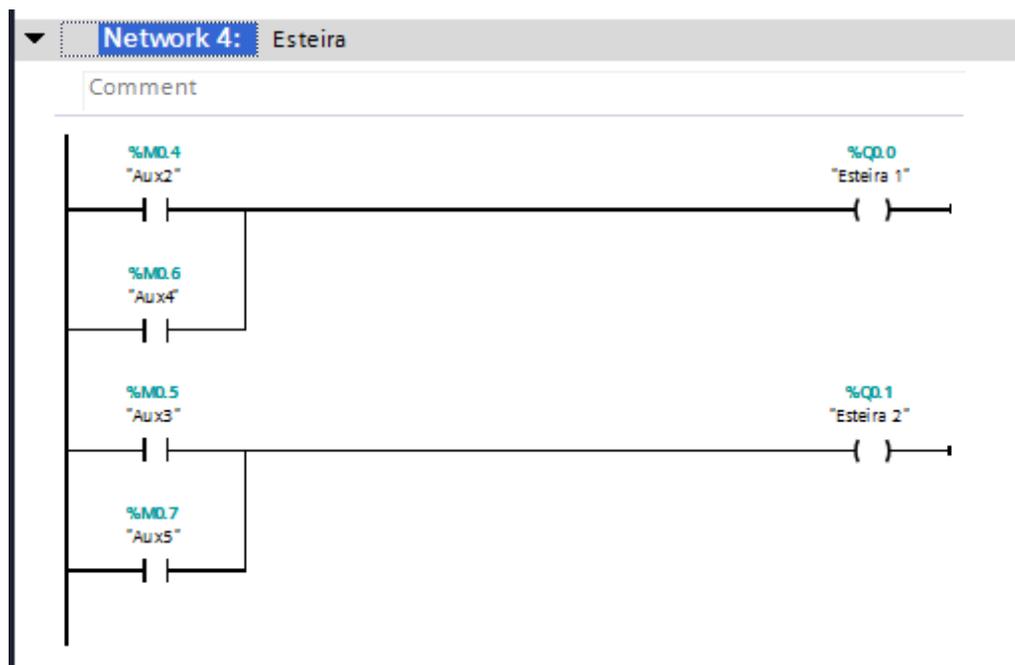


Observação

O reset evita que o CLP, no caso de um desligamento, quando venha a religá-lo esteja com alguma saída ativa. Para isso o operador pressiona o botão "reset" () para desativá-las.

Na network 4, conforme mostra Figura 24, colocamos na programação os acionamentos das esteiras 1 e 2.

Figura 24 - Detalhe do ambiente de programação de acionamento das esteiras.



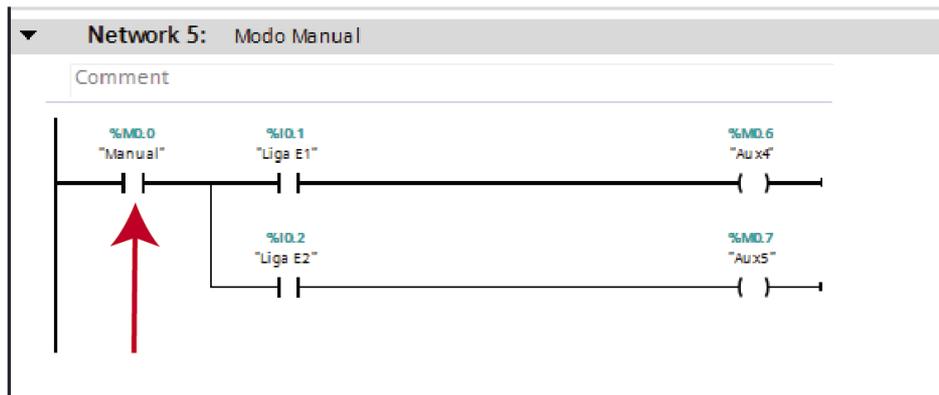
Observação

Perceba que para acionar as esteiras bastam que os respectivos contatos das bobinas auxiliares sejam ativados. Lembrando que e são contatos cujos relés estão no ambiente da network 2.

Já os contatos e pertencem aos relés auxiliares contidos no ambiente da network 5.

Já os contatos %M0,6 e %M0,7 pertencem aos relés auxiliares contidos no ambiente da network 5.

Figura 26 - Detalhe do ambiente de programação no modo de operação manual.



Observação

Da mesma maneira que foi visto no modo automático, atente para o detalhe do contato aberto do relé auxiliar, lá da network 1. É esse contato que dará permissão para que o operador em uma eventual falha no processo possa acionar as esteiras de modo independente, com o intuito de "limpar" o sistema, ou seja, em uma falha ele recolhe as peças que porventura estejam em alguma esteira para, assim, reiniciar o processo novamente.

Resumo

Ao observar nesta aula, na medida em que amadurecemos na programação Ladder, mais aplicações exigem raciocínio do programador e isso é natural, tendo em vista que, na indústria, a maioria das aplicações envolvem várias condições de operação em uma mesma máquina, como foi o caso do último exemplo.

Perceba, também, que foi imprescindível o conhecimento prévio da lógica de programação, tais como: retenção de sinal, função "E", função "OU", e assim por diante.

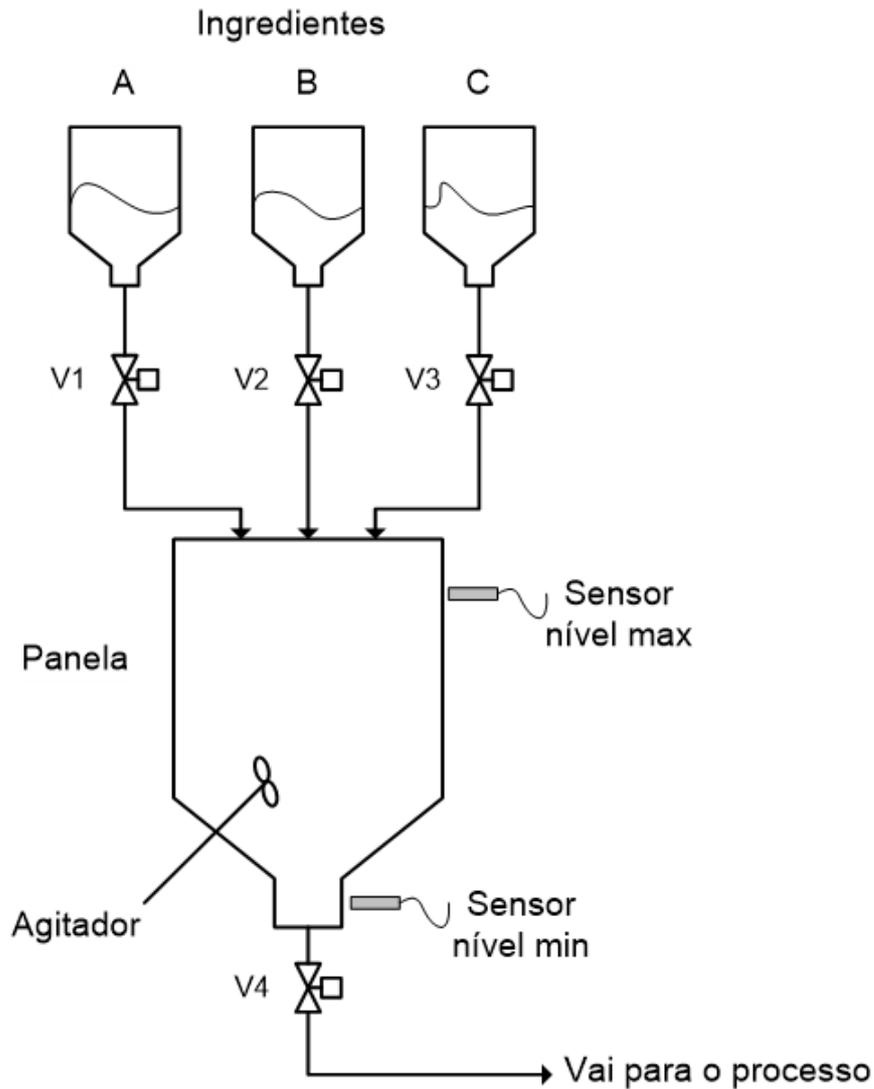
Na prática existem dois caminhos: o desenvolvimento de um projeto novo, ou seja, máquina nova. E o concerto de uma máquina já existente cujo CLP queimou. Para este último o programador faz uma verdadeira entrevista com o operador da máquina para saber quais as condições de operação da mesma, semelhante aos enunciados vistos ao longo da aula.

Mas atenção, em ambos os casos o programador deve ter bom conhecimento da linguagem Ladder.

Autoavaliação

1. Desenvolva uma aplicação em ladder para automatizar um determinado processo de uma empresa de laticínios. O sistema é um misturador. É formado por três silos que contém os ingredientes A, B e C, uma panela principal, quatro válvulas solenoides (V_1 , V_2 , V_3 e V_4), dois sensores de nível e um misturador (conforme mostra a Figura 26).

Figura 26 - Layout de uma planta para controle de processo de um misturador.



O misturador deve operar obedecendo às seguintes regras:

1. Deve ter uma opção de operação em manual ou automático;
2. No modo manual cada válvula é acionada individualmente, a qualquer instante e independente do nível;
3. No modo automático o operador aperta o botão "start" para iniciar o processo;
4. No modo automático, ao pressionar "start" o sistema só inicia se a panela estiver vazia, ou seja, no sensor de nível mínimo;
5. Existe uma receita em que a seguinte ordem de mistura dos ingredientes deve ser obedecida:

- a. 1° ingrediente A;
 - b. 2° ingrediente B;
 - c. 3° ingrediente C;
6. Cada ingrediente descarrega na panela por 5s;
 7. Quando iniciar a descarga do último ingrediente é necessário ligar o agitador por 10s;
 8. Após a mistura dos três ingredientes e a parada do agitador a descarga é acionada através da válvula 4 até chegar no sensor de nível mínimo;
 9. Para que a próxima mistura aconteça a panela deve estar vazia;
 10. Se a panela encher (sensor de nível máximo) um alarme sonoro deverá ser ativado;
 11. Coloque um contador para saber quantas tachadas foram feitas, para saber o consumo de ingredientes.

Referências

CARVALHO NETO, João T. **Controladores lógicos programáveis**. Disponível em: <<http://www.dca.ufrn.br/~joaoteixeira/ApostilaCLPFinal.pdf>>. Acesso em: 28 jan. 2016

MAITELLI, André L. **Controladores lógicos programáveis**. Disponível em: <<http://www.dca.ufrn.br/~maitelli/index.php?corpo=academica.php>>. Acesso em: 20 jan. 2016.

SIEMENS. **Sistemas de automação industrial**. Controlador SIMATIC S7 (PLC). Disponível em: <<http://w3.siemens.com.br/automation/br/pt/automacao-econtrole/automacao-industrial/simatic-plc/s7-cm/Pages/Default.aspx>>. Acesso em: 20 jan. 2016.

SILVA Filho, Bernardo S. **Acurso de controladores lógicos programáveis**. Rio de Janeiro: Faculdade de Engenharia, [20??]. Disponível em: <<http://www.lee.eng.uerj.br/downloads/cursos/clp/clp.pdf>>. Acesso em: 15 jan. 2016

WEG, CLP's e controle de processos. Disponível em:
<<http://www.weg.net/br/Produtos-e-Servicos/Drives/CLPs-e-Controle-de-Processos>>.
Acesso em: 22 jan. 2016.