

Programa o de CLPs

Aula 05 - Programa o de CLP usando o Automation Studio

Apresentação

Nesta aula, veremos o ambiente de programação em Ladder desta vez utilizando o Automation Studio versão 5.7. O Automation Studio é uma poderosa ferramenta que permite uma simulação virtual de projetos que vão desde uma planta industrial, até máquinas, contendo sistemas automatizados nas principais áreas de desenvolvimento na indústria: Hidráulica, pneumática, elétrica e mecânica.

Nesse contexto, o Automation Studio versão 5.7 permite que sejam criadas “interfaces de interação e operação”, ou seja, desenvolvimento de projetos que simulam uma condição real. Para isso o Automation dispõe de uma vasta biblioteca de componentes comumente usados tais como: Chaves, botoeiras, sinalizadores, medidores, etc.

Portanto o comportamento da interface é como se estivesse no mundo real. Considerando que a simulação é um fator muito importante no desenvolvimento de um projeto de automação, a programação em Ladder nos oportuniza observar como esse projeto se comportará, e é justamente esse o foco de nossa aula de hoje, vamos lá?

Objetivos

- Conhecer o Automation Studio;
- Programar em linguagem de programação Ladder utilizando o Automation Studio;
- Interpretar o funcionamento de um projeto de simulação virtual do Automation Studio;
- Observar como acontece a simulação em tempo real.

Introdução

Como foi visto nas aulas passadas, para o desenvolvimento de todo e qualquer projeto de automação é importante que se conheça como a planta ou o projeto irá funcionar, ou seja, quais as perspectivas de trabalho, pois só assim temos condições de iniciar a automação bem como a programação do CLP em Ladder, evidentemente.

Portanto, daremos ênfase a três projetos no Automation Studio:

- Um sistema de esteira rolante para transporte de objetos;
- Um sistema de carregamento de peças;
- Um sistema de encaixotamento de caixas.

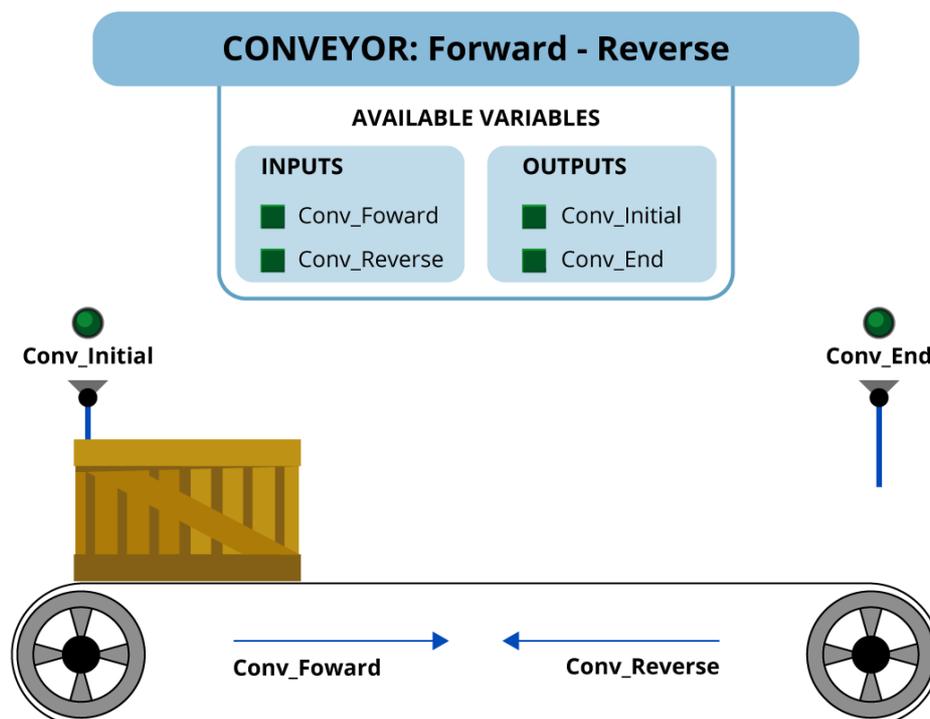
Em todos eles a interface está pronta para operação. É como se toda a estrutura mecânica, elétrica e pneumática estivesse pronta e montada e que, para entrar em operação, tenhamos que desenvolver a aplicação em Ladder e carregá-lo no CLP.

Relembramos que as aplicações mostradas aqui poderão ter mais de uma solução, portanto o Ladder desenvolvido aqui é uma sugestão de aplicação, ou seja, é normal que um programador desenvolva uma programação diferente. De certo modo isso é salutar, pois mostra que você está evoluindo no quesito “desenvolvimento da lógica Ladder”.

Sistema de Esteira Rolante para Transporte de Objetos

Vamos desenvolver uma aplicação em Ladder para elaboração de um projeto de acionamento de uma esteira, conforme a **Figura 1**, usada para transportar objetos.

Figura 01 - Sistema de transporte de objetos com esteira.



Fonte: Adaptado de Famic Technologies.

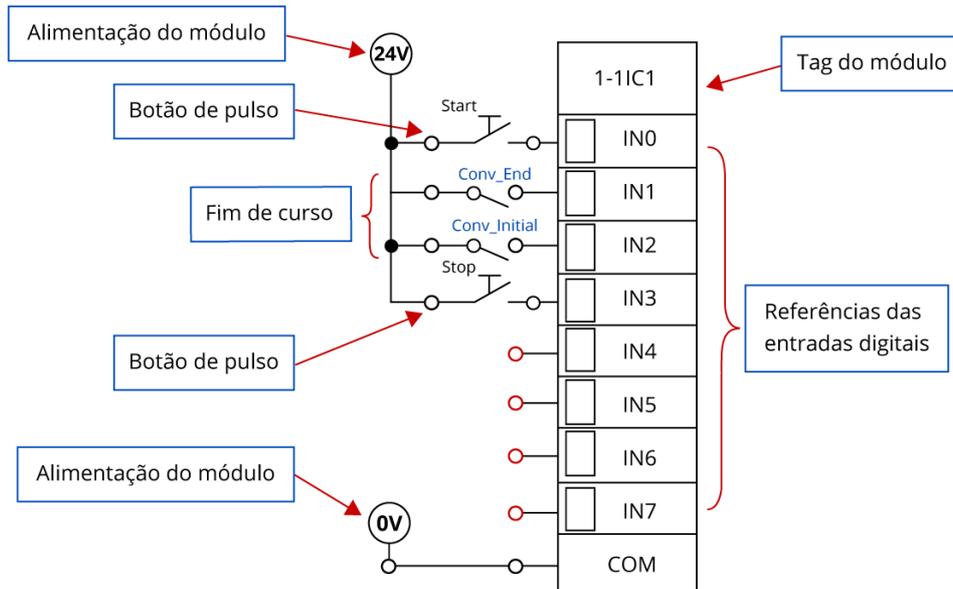
Componentes do projeto:

- 01 esteira para transporte;
- 01 Motor para movimentar a esteira nos sentidos direto (Forward) e reverso (Reverse);

O propósito deste projeto é que a caixa fique deslocando direta e reversamente de modo contínuo em um ciclo fechado. Para o desenvolvimento do projeto precisamos montar a estrutura de programação Ladder que é composta de:

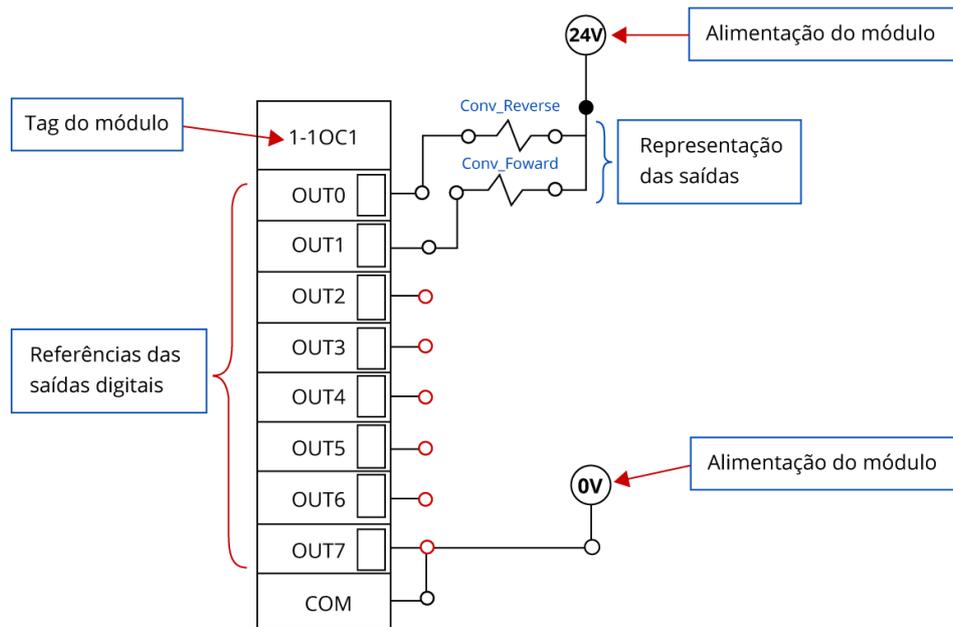
- Um módulo de entrada (Figura 2);
- Um módulo de saída (Figura 3);
- Ambiente de programação em Ladder (Rung) (Figura 4).

Figura 02 - Módulo de entrada com 8 entradas digitais;



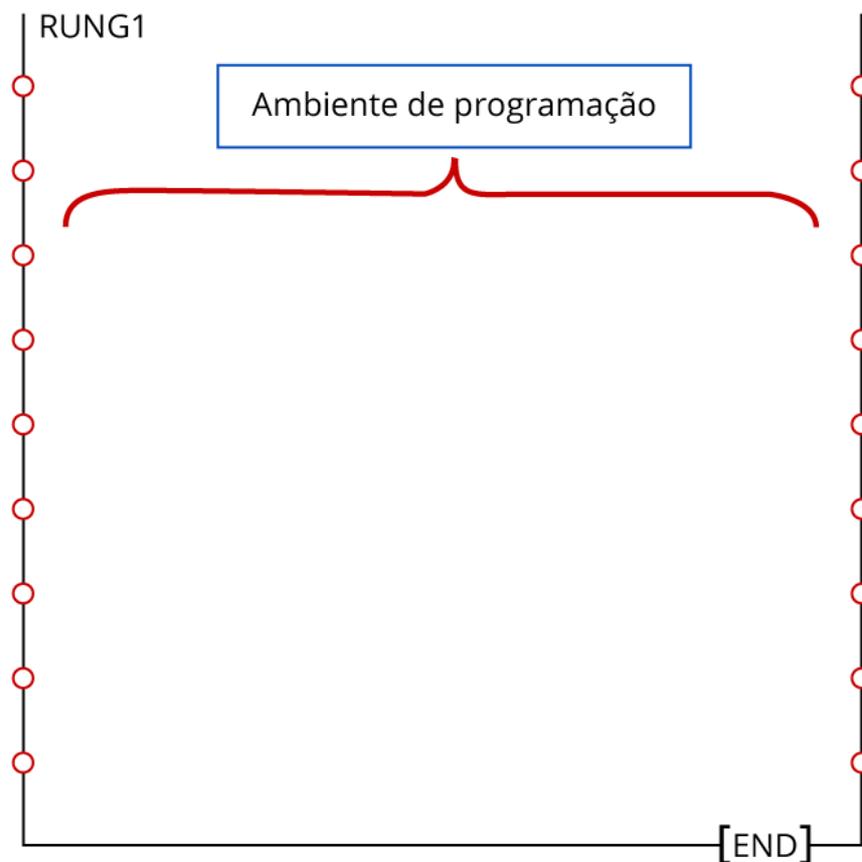
Fonte: Adaptado de Famic Technologies.

Figura 03 - Módulo de saída com 8 saídas digitais;



Fonte: Adaptado de Famic Technologies.

Figura 04 - Ambiente de programação Ladder (Rung)

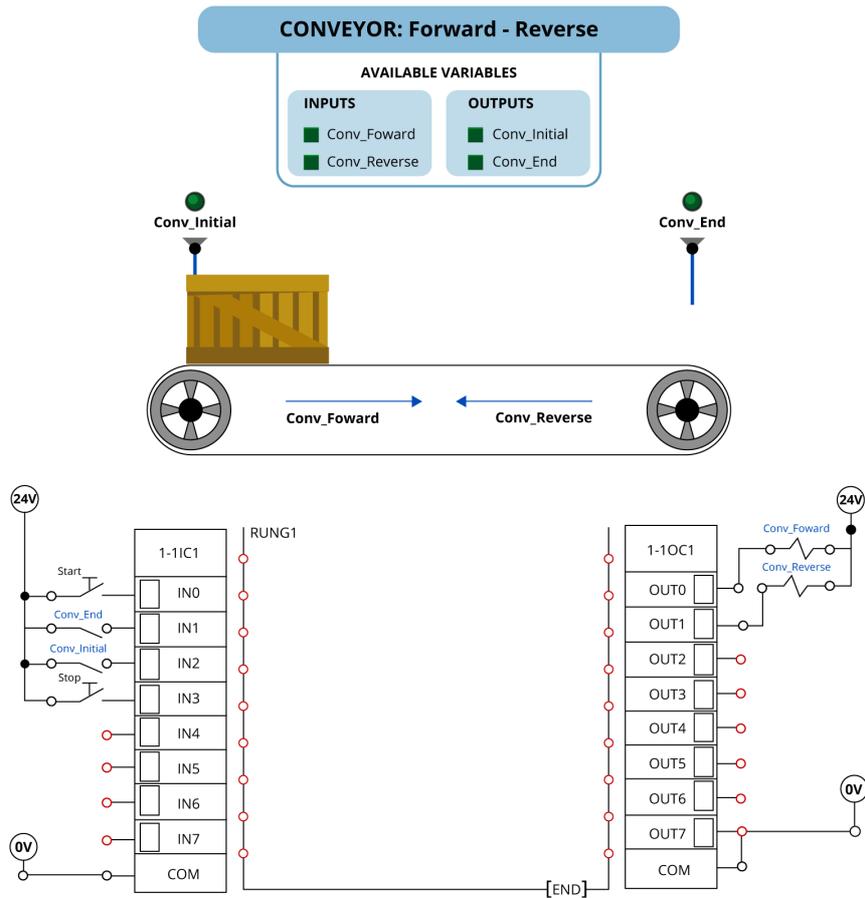


Fonte: Adaptado de Famic Technologies.

Assim, a montagem no Automation Studio do projeto da esteira e a programação em Ladder terá o aspecto mostrado na **Figura 5** e seu funcionamento ocorre da seguinte forma:

- Ao pressionar o botão "Start" a esteira é acionada deslocando a caixa no sentido direto até tocar no fim de curso "Conv_End", de acordo com a condição do fim de curso "Conv_Initial";
- Ao bater no fim de curso "Conv_End" a esteira inverte seu sentido de rotação, deslocando a caixa no sentido reverso até encostar no fim de curso "Conv_Initial";
- Ao bater no fim de curso "Conv_Initial" a esteira inverte novamente sua rotação deslocando a caixa no sentido direto e assim sucessivamente, até que seja pressionado o botão "Stop".

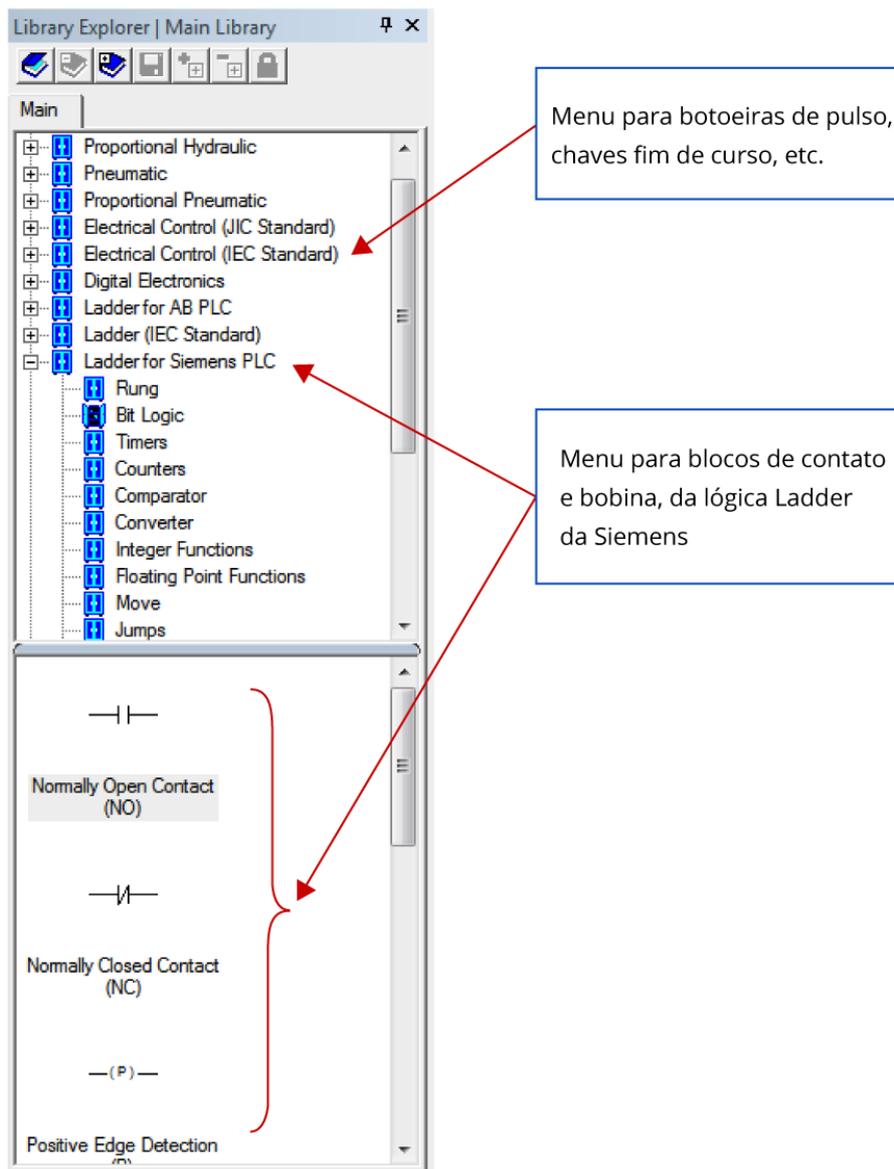
Figura 05 - Layout do projeto Esteira de transporte de objetos.



Fonte: Adaptado de Famic Technologies.

Para montagem dos blocos de contatos e bobinas temos o menu do Automation Studio, conforme a **Figura 6**. É esse menu que contém os blocos de contatos que vamos utilizar para este projeto.

Figura 06 - Menu de contatos para montagem dos blocos.



Fonte: Adaptado de Famic Technologies.

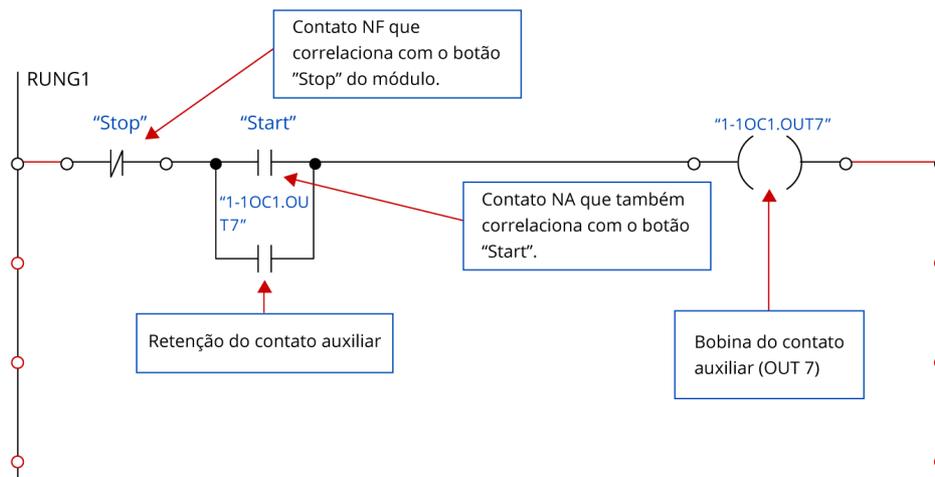
De posse dessas informações, agora vamos programar a lógica Ladder no ambiente "Rung":

1º Passo:

Perceba pelo Layout que existe um fim de curso (Conv_Initial) que já está ativado e, neste caso, para não correr o risco de o sistema iniciar com a esteira ligada, vamos utilizar uma saída auxiliar que não esteja sendo utilizada. Assim, montaremos a

primeira linha de comando conforme a **Figura 7**:

Figura 07 - Lógica que ativa o sistema de transporte da esteira.



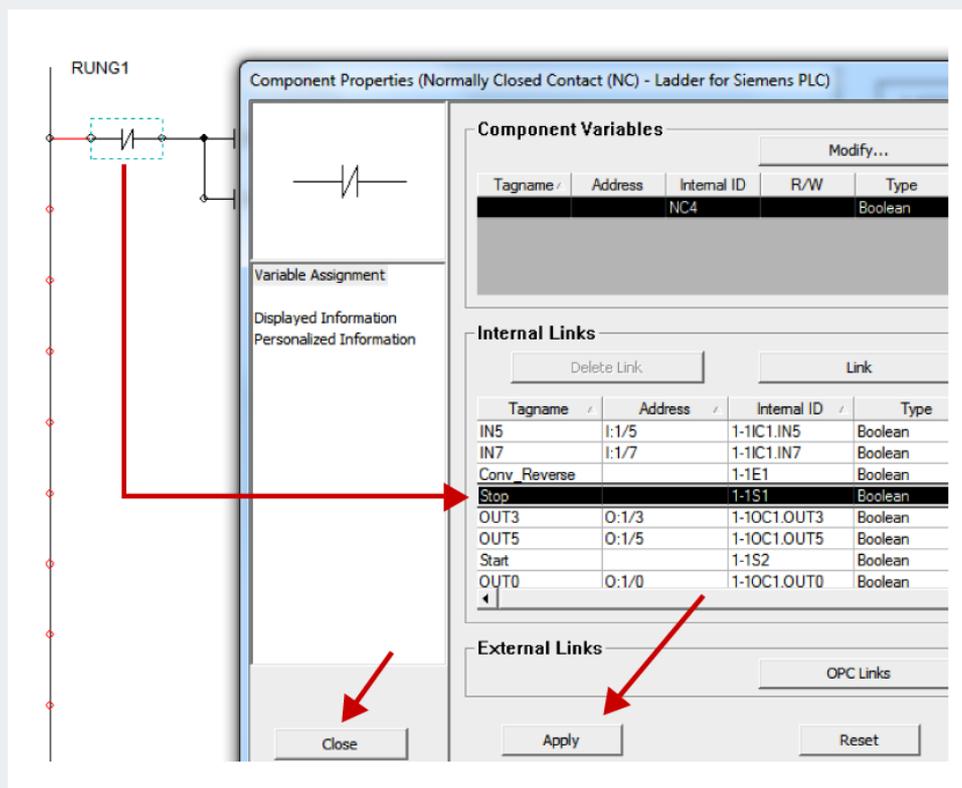
Fonte: Adaptado de Famic Technologies.

Observação:

Como foi visto em outras aulas, para colocar um bloco basta selecioná-lo de contato clicar e arrastar para dentro do "Rung".

Quando colocamos um bloco no "Rung" verifique que ele vem sem a "Tag". Portanto para colocar uma "Tag", basta dar um duplo clique no bloco e selecionar a Tag correspondente à sua finalidade e clicar em "Apply" (aplicar) e "close", conforme mostra a **Figura 8**.

Figura 08 - Tela para seleção da "Tag" do bloco de contato.

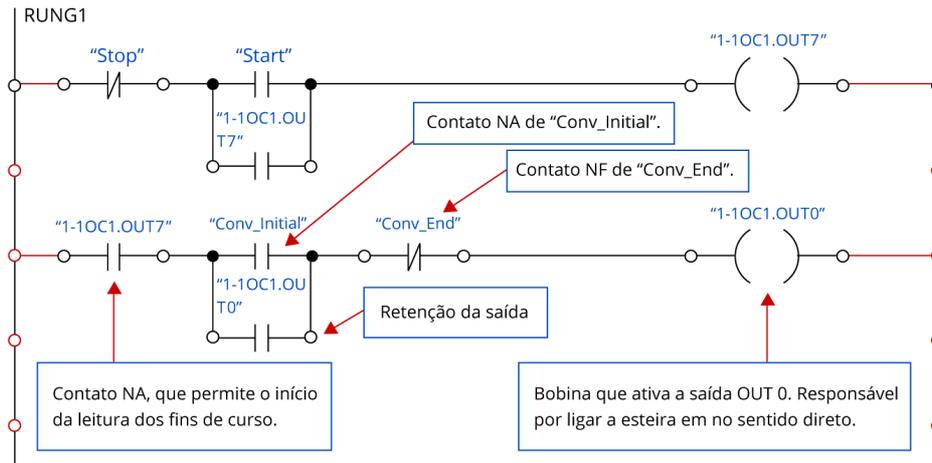


Fonte: Adaptado de Famic Technologies.

2º Passo:

Vamos acrescentar ao "Rung" a segunda linha do programa. Ela será responsável por colocar a esteira em funcionamento desde que a caixa esteja na posição inicial, ou seja, com o fim de curso "Conv_Initial" ativado. Portanto, vamos montar a segunda linha conforme a **Figura 9**, com todas suas "Tag's".

Figura 09 - Linha de comando que ativa o funcionamento da esteira.

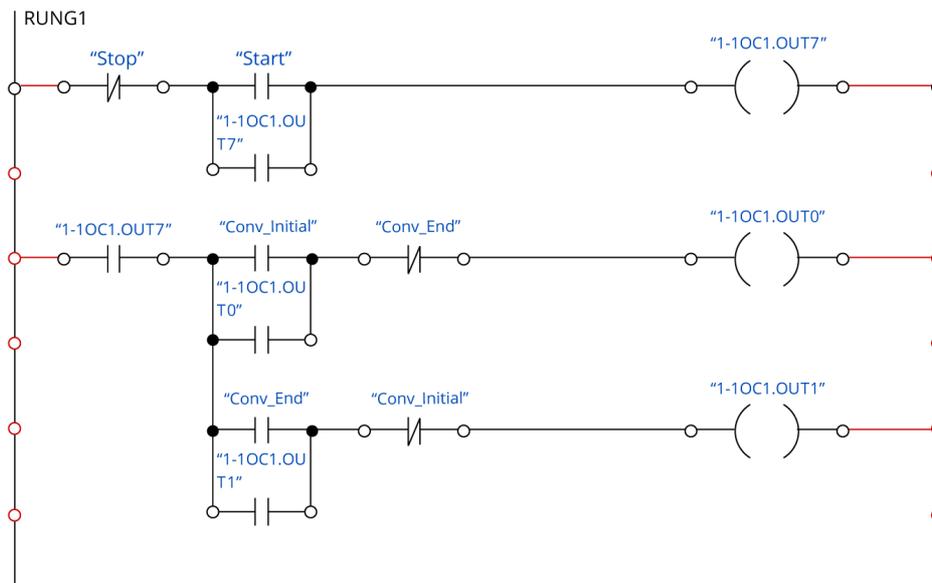


Fonte: Adaptado de Famic Technologies.

3º Passo:

Vamos também acrescentar a terceira e última linha. Ela será a responsável pela inversão do sentido de rotação quando a caixa tocar no fim de curso "Conv_End". Veja que a terceira linha, assim como a segunda, também depende do contato NA da bobina auxiliar (OUT 7). Contudo, vamos montar a terceira linha conforme mostra a **Figura 10**.

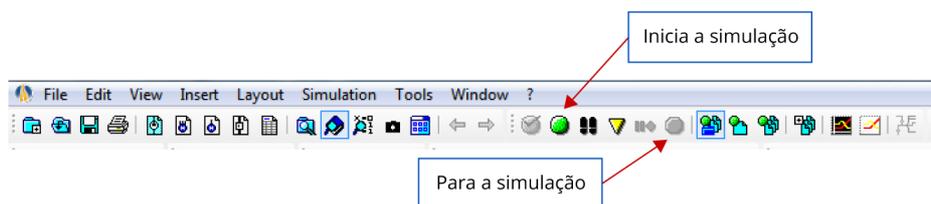
Figura 10 - Linha de comando da terceira linha, responsável pela inversão de rotação.



Fonte: Adaptado de Famic Technologies.

Pronto, finalizamos a aplicação. Para simulação, vá ao menu e clique no botão “Normal Simulation”, conforme mostra a **Figura 11**.

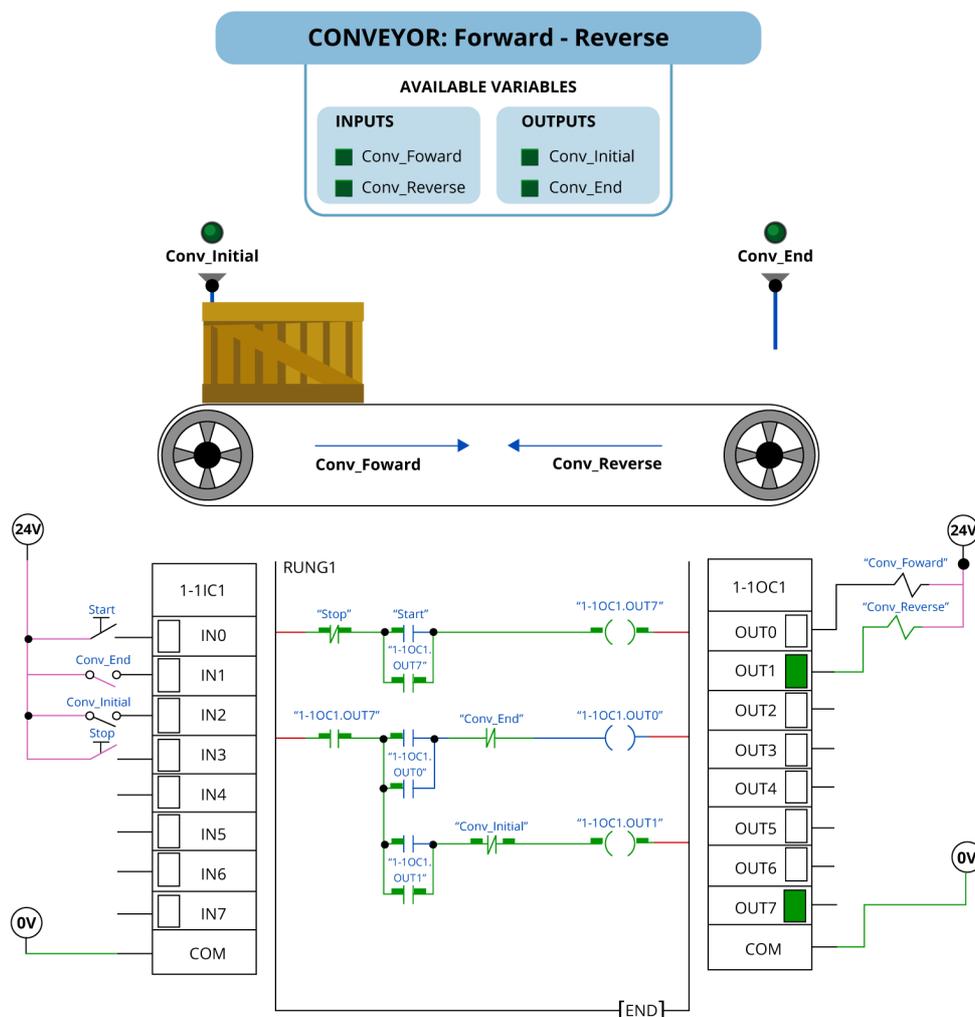
Figura 11 - Menu de simulação do Automation Studio.



Fonte: Adaptado de Famic Technologies.

Após esses procedimentos, o projeto da esteira de transporte de objetos no modo simulação terá o aspecto da **Figura 12**.

Figura 12 - Tela de simulação do projeto da esteira de transporte de objetos.

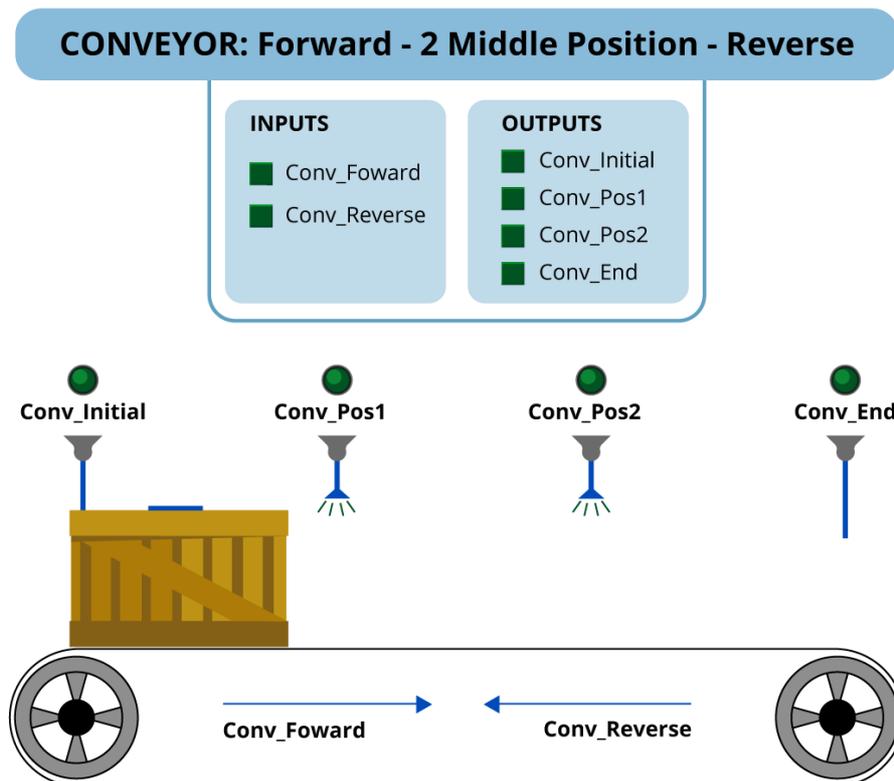


Fonte: Adaptado de Famic Technologies.

Sistema de Carregamento de Peças

Vamos fazer uma aplicação em Ladder para automatizar um sistema de carregamento de peças cujo projeto é mostrado na **Figura 13**.

Figura 13 - Projeto de uma esteira de enchimento de peças.



Fonte: Adaptado de Famic Technologies.

De acordo com o layout, o projeto é composto por:

- Um motor, que aciona a esteira nos sentidos direto e reverso;
- 2 fins de curso "Conv_Initial e Conv_End";
- 2 Sensores de detecção de caixa, para descarregamento de peças.

No entanto, o sistema funciona da seguinte maneira:

- O sistema deve prever um botão de "Start" para iniciar o ciclo;

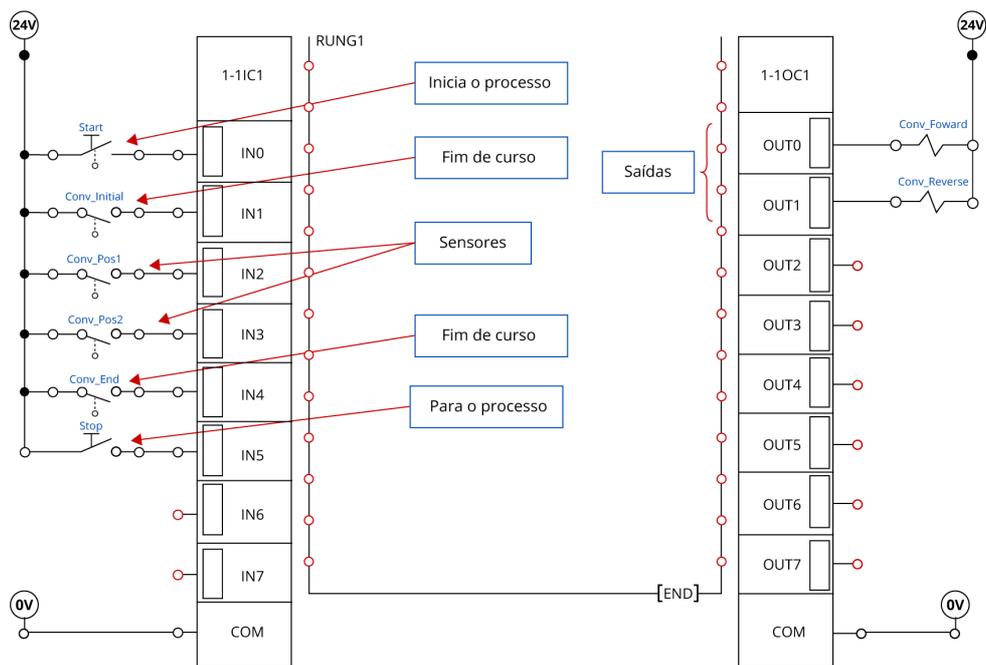
- Ao pressionar o botão “Start”, a caixa se desloca no sentido direto, de acordo com a condição do fim de curso “Conv_Initial” até que o sensor “Conv_Pos1” detecte a presença da caixa;
- Quando o sensor “Conv_Pos1” detecta a presença da caixa a esteira para e aguarda o carregamento por 5s, em seguida a caixa segue em frente até que o sensor “Conv_Pos2” também detecte a presença da caixa;
- Quando o sensor “Conv_Pos2” detecta a caixa a esteira para novamente e aguarda o carregamento da caixa por 5s.
- Após o carregamento da caixa no sensor “Conv_Pos2” a caixa segue em frente até o destino final, que é o fim de curso “Conv_End” o qual faz a caixa retornar à sua origem;
- Ao retornar, a caixa bate no fim de curso “Conv_Initial” e recomeça um novo ciclo.
- Um botão de “Stop” deverá parar o sistema a qualquer momento.

Bom, agora que sabemos como o sistema funcionará, vamos programar o Ladder que dará vida ao projeto. São cinco passos, vamos a eles?

1º Passo:

Vamos montar a estrutura de programação Ladder, com os blocos de entrada e saídas e o ambiente “Rung”, conforme mostra a **Figura 14**.

Figura 14 - Ambiente de configuração de programação do Ladder da esteira de carregamento de peças.

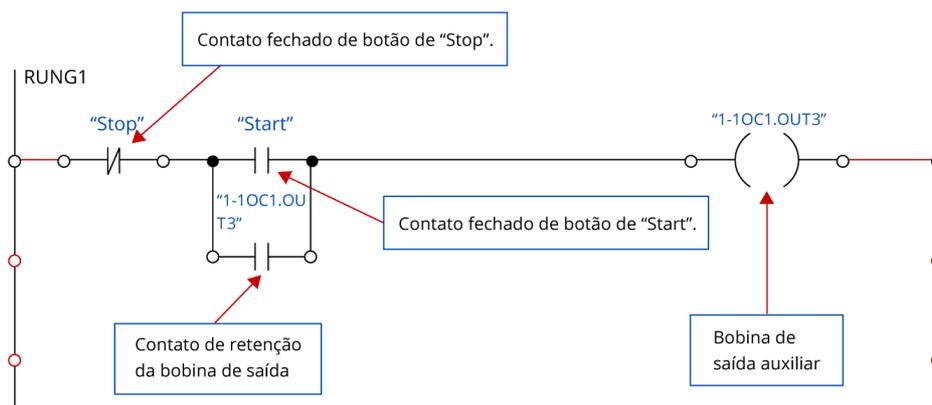


Fonte: Adaptado de Famic Technologies.

2º Passo:

Vamos iniciar o Ladder no ambiente "Rung" conforme mostra **Figura 15**, com a condição de partida e parada do sistema. Para isso, utilizaremos uma saída como relé auxiliar (uma saída que não esteja sendo usada), neste caso OUT 3.

Figura 15 - Linha inicial do Ladder para o projeto de carregamento.

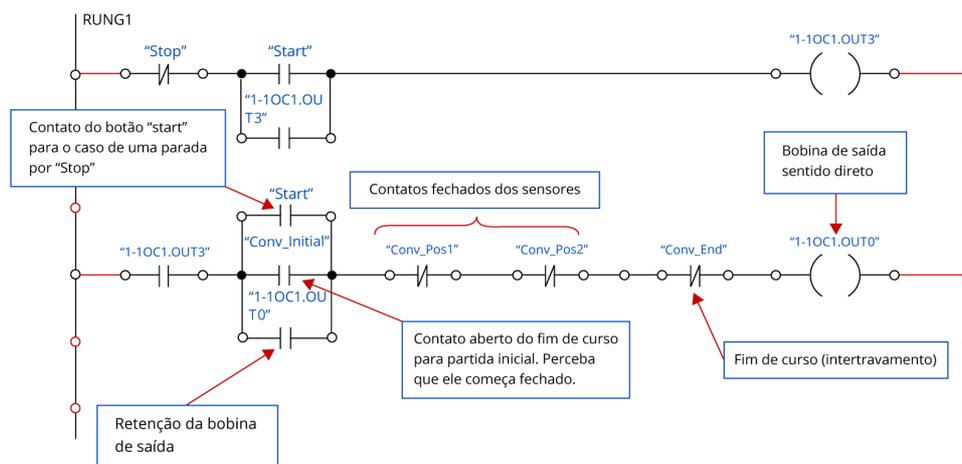


Fonte: Adaptado de Famic Technologies.

3º Passo:

A segunda linha será responsável pela partida inicial da esteira no sentido direto. Conforme mostra a **Figura 16**, atente para os contatos fechados dos sensores, pois serão eles que irão parar a esteira para descarregamento. Outro detalhe são os contatos abertos do botão de "Start" do fim de curso e da retenção, serão estes que manterão a esteira ligada no sentido direto.

Figura 16 - 2ª linha do programa Ladder responsável pela partida inicial da esteira.

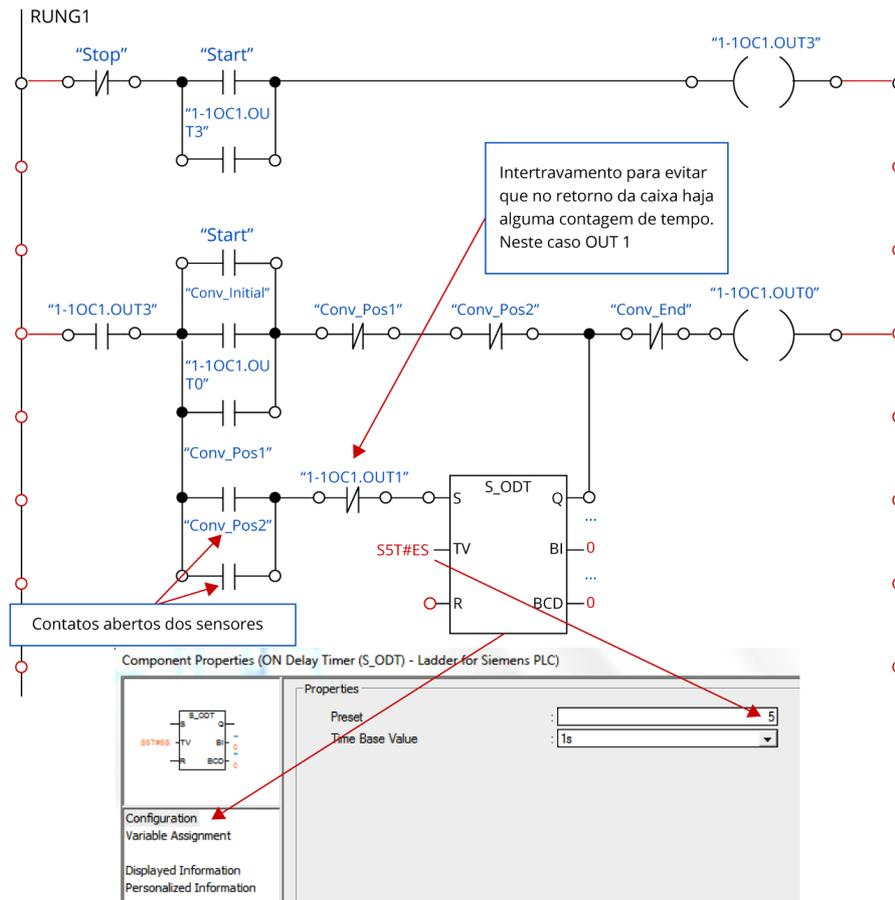


Fonte: Adaptado de Famic Technologies.

4º Passo:

Veja que os sensores irão desligar a esteira quando detectarem a caixa. Portanto, no momento da parada precisamos iniciar uma contagem de tempo. Para isso, conforme mostra a **Figura 17**, vamos utilizar um temporizador ON-Delay Timer (S_ODT) para contar o tempo de 5s. Como os tempos de paradas são iguais, basta um temporizador para as duas paradas. Veja que quem vai iniciar a contagem de tempo será os contatos abertos dos sensores, cada um ao seu tempo.

Figura 17 - Configuração da 3ª linha com a inserção do temporizador e sua configuração.

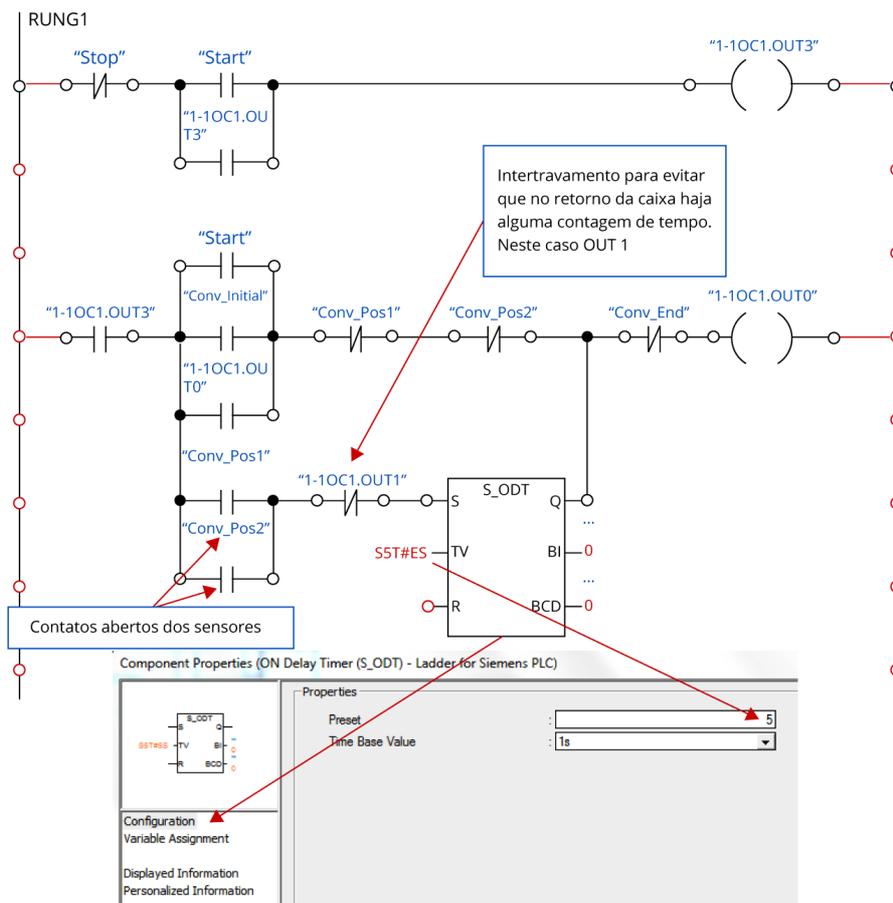


Fonte: Adaptado de Famic Technologies.

5º Passo:

Neste momento a caixa irá parar nos sensores que, por sua vez, contarão o tempo de descarregamento de cada um, depois a caixa seguirá em frente até o fim de curso final "Conv_End". Considerando que este 5º passo objetiva fazer a caixa voltar ao bater no fim de curso ("Conv_End"), vamos colocar uma terceira linha para contemplar essa ação, conforme mostra a **Figura 18**.

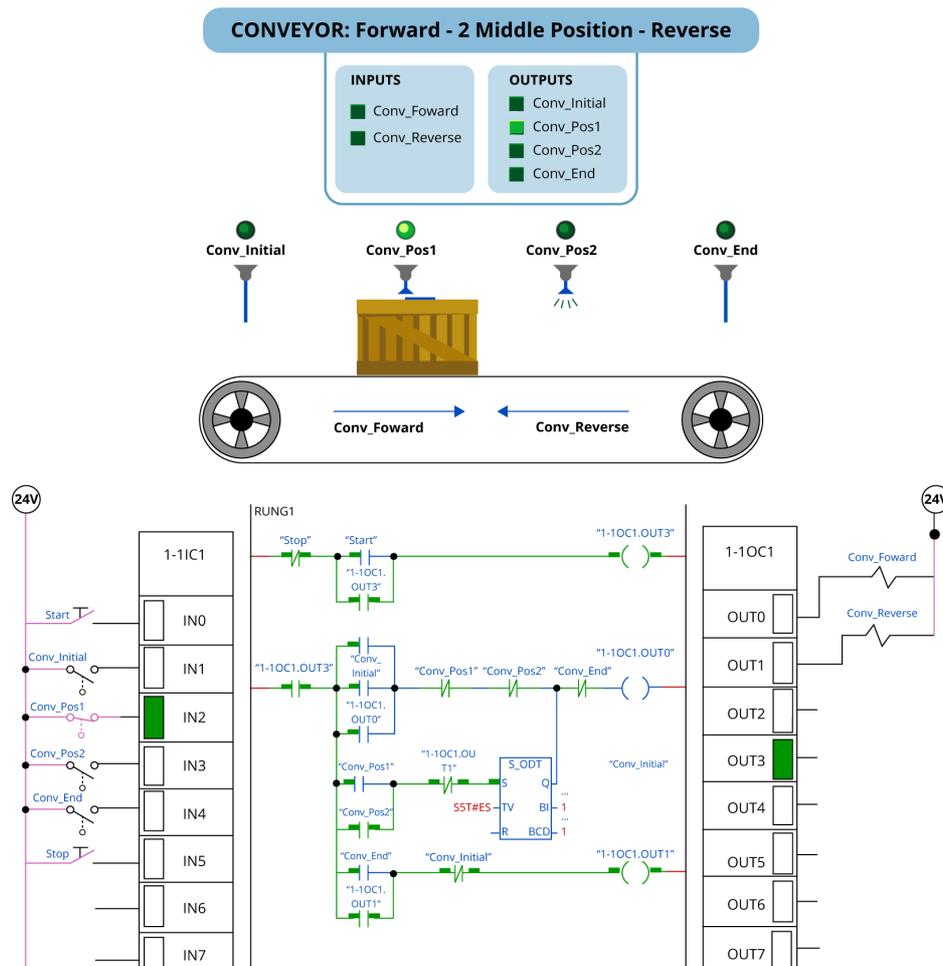
Figura 18 - 4ª linha da programação Ladder que é responsável pelo retorno da caixa após carregamento.



Fonte: Adaptado de Famic Technologies.

Feito o passo a passo, basta simular o projeto e observar seu funcionamento, conforme a **Figura 19**.

Figura 19 - Simulação do projeto final do sistema de carregamento de peças.



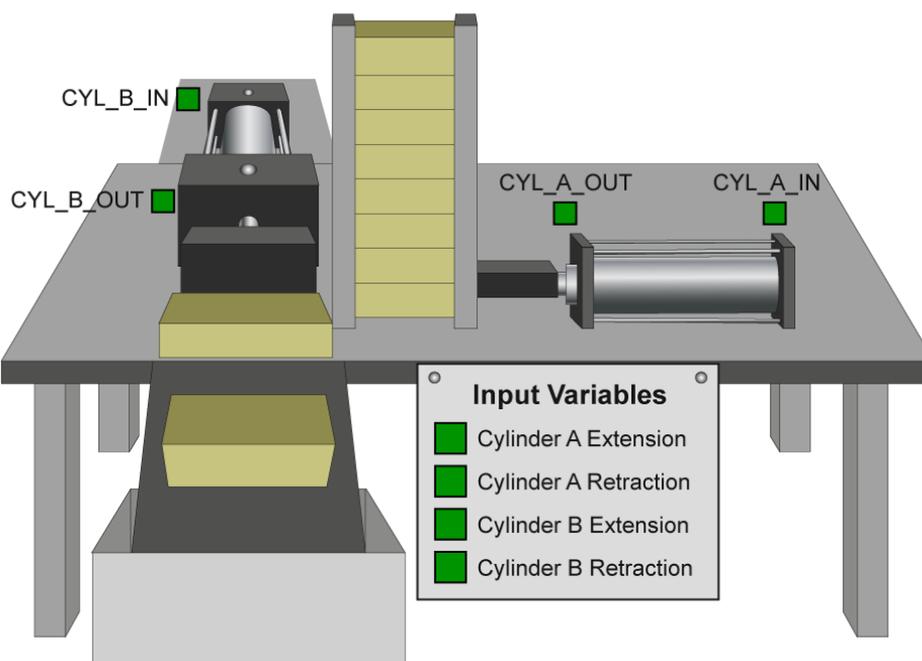
Fonte: Adaptado de Famic Technologies.

Sistema de Encaixotamento de Caixas

Agora, vamos desenvolver uma aplicação em Ladder para automatizar um sistema de encaixotamento de caixas conforme mostra a **Figura 20**.

O objetivo principal é fazer com que os cilindros A e B “encaixotem as caixas”, ou seja, uma vez que já existem caixas empilhadas o cilindro A desloca a caixa para que em seguida o cilindro B empurre-a para dentro do recipiente, e repita esse ciclo sempre que houver caixas ou, caso contrário, quando for pressionado o botão “Stop”.

Figura 20 - Layout de um sistema de encaixotamento de caixas.



Fonte: Adaptado de Famic Technologies.

Como podemos ver, o sistema é composto por:

- 2 cilindros pneumáticos A e B;
- 4 sensores indutivos: CYL_A_OUT, CYL_A_IN, CYL_B_OUT e CYL_B_IN. (dois para cada cilindro).

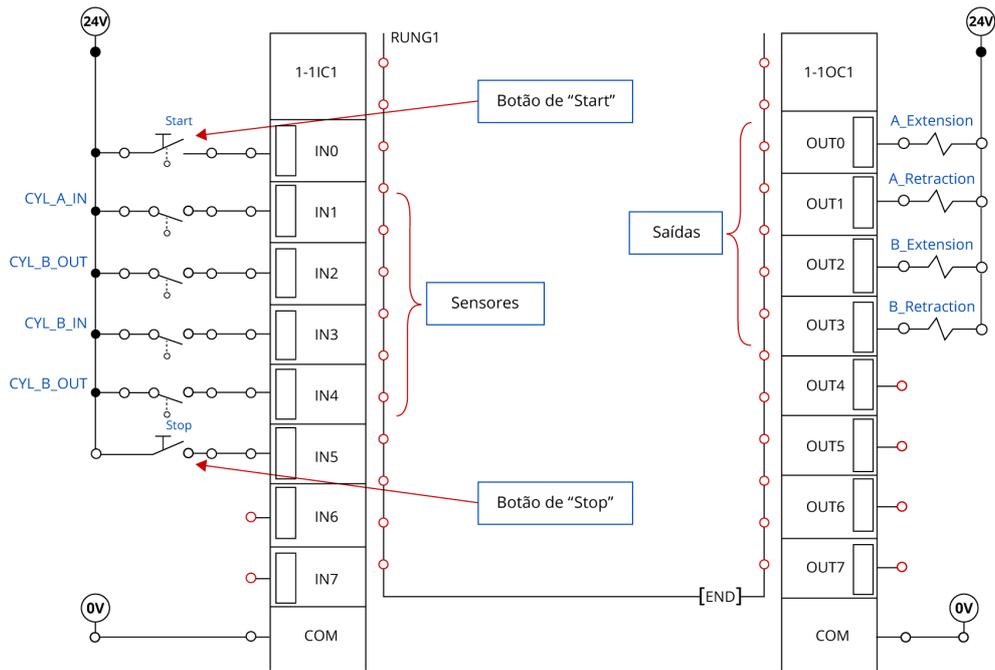
Então, vamos ao passo a passo, de cinco, para programar o Ladder que fará o projeto funcionar?

Passo a Passo

1º Passo:

Vamos montar a estrutura dos blocos conforme mostra a **Figura 21**.

Figura 21 - Montagem e configuração do ambiente de programação Ladder.

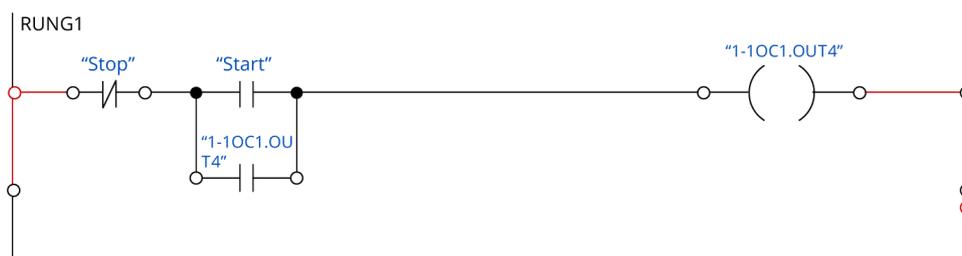


Fonte: Adaptado de Famic Technologies.

2º Passo:

Vamos utilizar a mesma ideia desenvolvida nas outras aplicações. Uma bobina auxiliar para partida e parada do sistema, portanto vamos montar no "Rung" a linha de programação conforme mostra a **Figura 22**.

Figura 22 - Programação da 1ª linha com sistema de partida e parada do projeto.

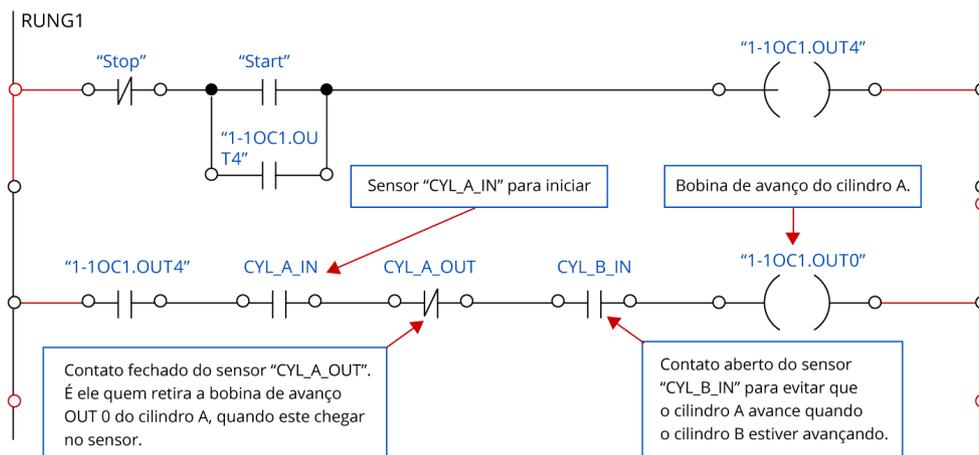


Fonte: Adaptado de Famic Technologies.

3º Passo:

Com a programação da partida e parada do projeto vamos fazer a primeira ação acontecer, que é o cilindro A avançar. Para isso, vamos montar a segunda linha da programação em Ladder, conforme mostra a **Figura 23**. Fique atento aos contatos de intertravamentos.

Figura 23 - Linha de programação em Ladder para condição inicial de avanço do cilindro A.

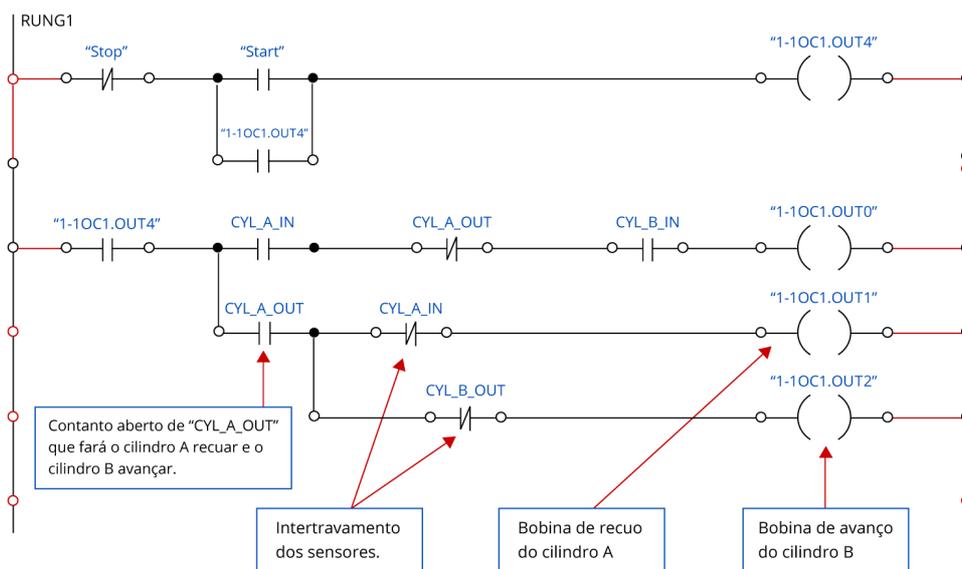


Fonte: Adaptado de Famic Technologies.

4º Passo:

Uma vez que o cilindro A avançou, ele levou uma caixa para que o cilindro B empurre-a para o recipiente. Portanto, uma vez que o cilindro A Chegou no sensor “CIL_A_OUT”, precisamos avançar o cilindro B e retornar o cilindro A. Então vamos programar a 3ª e a 4ª linha do Ladder para fazer tais ações, conforme mostra a **Figura 24**.

Figura 24 - Programação da 3ª e 4ª linha do Ladder que fará o cilindro A retornar e o cilindro B avançar.

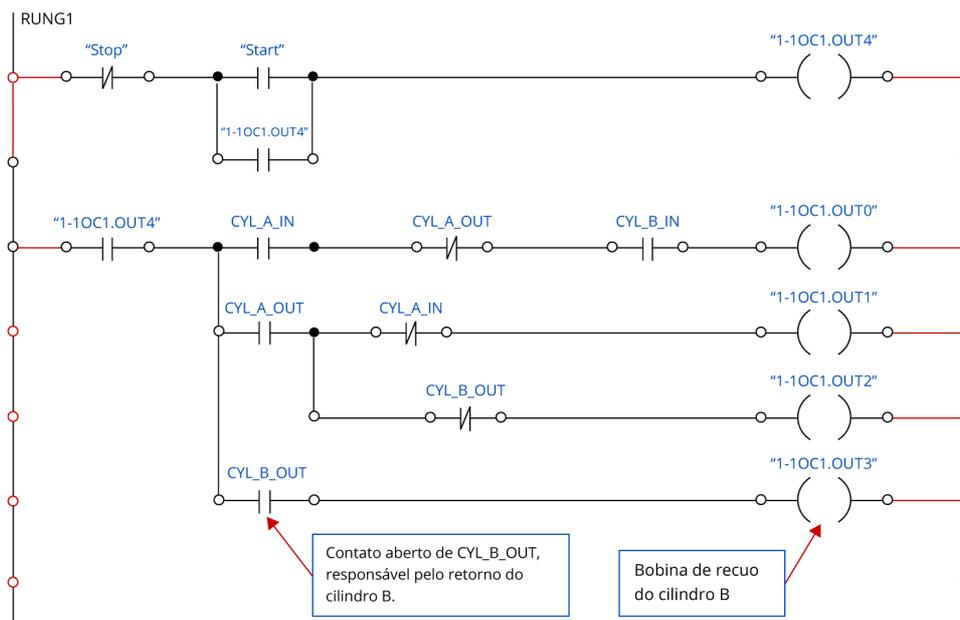


Fonte: Adaptado de Famic Technologies.

5º Passo:

Até aqui o cilindro A avançou juntamente com a caixa, o cilindro B empurrou-a para o recipiente, o cilindro A, por sua vez, recuou, falta agora o cilindro B também recuar para que dê início a um novo ciclo. Para isso, vamos programar a 5ª e última linha do programa Ladder, conforme mostra a **Figura 25**. Essa linha será responsável tanto pelo recuo do cilindro B quanto pelo início de um novo ciclo.

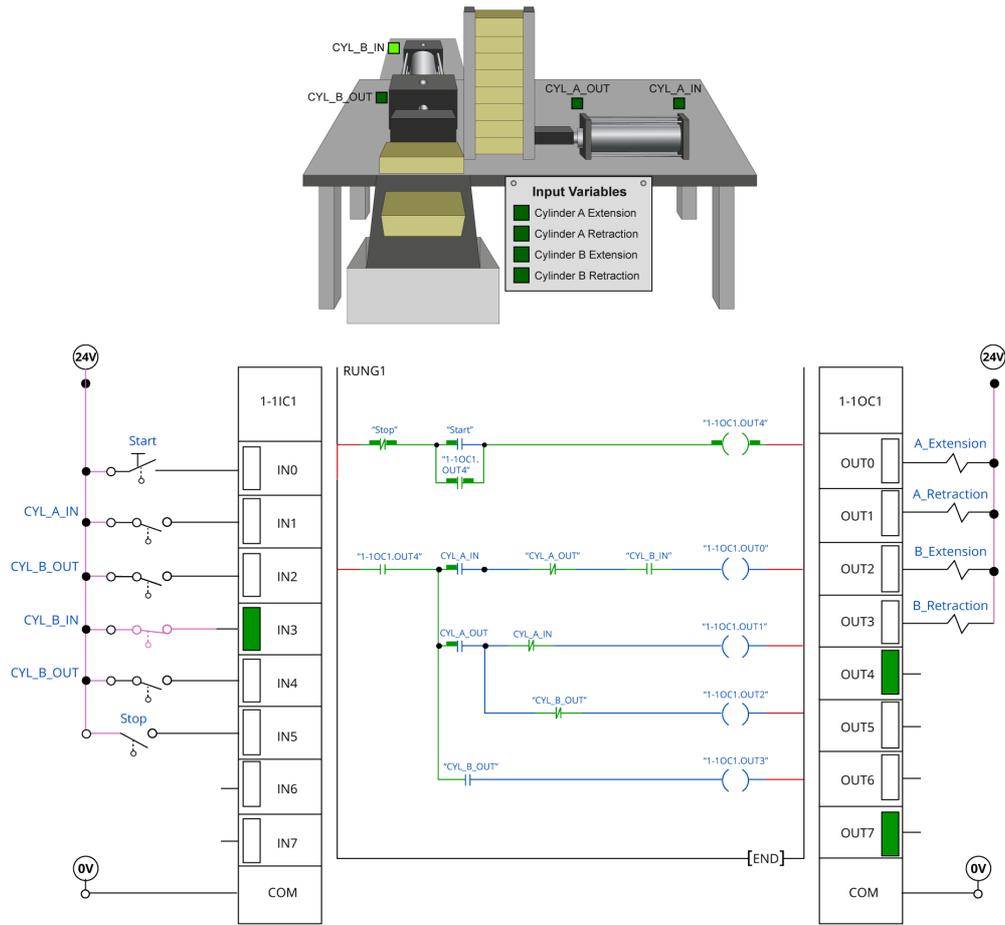
Figura 25 - Linha do programa que retornará o cilindro B e iniciará um novo ciclo.



Fonte: Adaptado de Famic Technologies.

Pronto! Agora é só juntarmos tudo e fazermos a simulação do sistema, conforme mostra a **Figura 26**.

Figura 26 - Simulação do projeto final de um sistema de encaixotamento de caixas.



Fonte: Adaptado de Famic Technologies.

Leitura Complementar

Apostila do Senai, do ano de 2012, sobre o tema estudado na aula de hoje:

https://daniellnunes1.files.wordpress.com/2012/11/u2_linguageladder1.pdf

Resumo

Nesta aula, abordamos o software de simulação de projetos virtuais Automation Studio, o qual permite a criação de interfaces de simulação como se fosse na “vida real”. Do ponto de vista de quem trabalha com projetos de automação é um software extremamente importante, pois permite que o programador tenha uma visão geral do comportamento do projeto em execução. Não tenha dúvida de que grandes projetos de automação passam por um processo de simulação virtual até que se tenha uma posição final de como a máquina estará pronta.

Autoavaliação

1. De posse do projeto de encaixotamento, visto no item 4, acrescente um contador crescente para contar 5 caixas no recipiente com o objetivo de parar o sistema. E após isso, será necessário um novo “Start” para começar a encaixotar em outro recipiente.

Referências

FAMIC Technologies. Automation Studio: P6: professional edition. Disponível em: <<http://www.famictech.com/flipbook/as-p6-english/#p=1>>. Acesso em: 30 mar. 2016.

FAMIC Technologies. Automation studio: user's guide. 2000 Inc. Disponível em: <http://www.tekniikka.oamk.fi/~penttihu/hydraulijarjestelmat/automationstudio/AS4_GUI_EF01_008.pdf>. Acesso em: 29 mar. 2016.

WEG, CLP's e controle de processos. Clic02. Disponível em: <<http://www.weg.net/br/Produtos-e-Servicos/Drives/CLPs-e-Controle-de-Processos>>. Acesso em: 30 mar. 2016.