

**UNIVERSIDADE DE TAUBATÉ**  
**PEDRO HENRIQUE BIDOIA DE LIMA**

**AUTOMAÇÃO INDUSTRIAL DE UM SISTEMA DE**  
**ARMAZENAMENTO EM UM GALPÃO**

**TAUBATÉ**  
**2021**

**PEDRO HENRIQUE BIDOIA DE LIMA**

**AUTOMAÇÃO INDUSTRIAL DE UM SISTEMA DE  
ARMAZENAMENTO EM UM GALPÃO**

Trabalho apresentado para obtenção  
do certificado de especialização pelo  
curso de Engenharia de controle e  
Automação do Departamento de  
Engenharia Mecânica e Elétrica da  
Universidade de Taubaté,  
Área de concentração: Automação  
industrial  
Orientador: Marcelo Pinheiro Werneck

**TAUBATÉ – SP  
2021**

**Grupo Especial de Tratamento da Informação - GETI**  
**Sistema Integrado de Bibliotecas – SIBi**  
**Universidade de Taubaté - Unitau**

L732a	<p>Lima, Pedro Henrique Bidoia de Automação industrial de um sistema de armazenamento em um galpão / Pedro Henrique Bidoia de Lima. -- 2021. 28 f. : il.</p> <p>Monografia (especialização) – Universidade de Taubaté, Pró-reitoria de Pesquisa e Pós-graduação, Taubaté, 2021. Orientação: Profa. Me. Marcelo Pinheiro Werneck, Departamento de Engenharia Elétrica.</p> <p>1. CLP. 2. Programação em ladder. 3. CodeSys. 4. Factory IO. 5. Automação. I. Universidade de Taubaté. Departamento de Pesquisa e Pós- graduação. Especialização em Controle e Automação. II. Título.</p> <p style="text-align: right;">CDD – 629.8</p>
-------	--

Ficha catalográfica elaborada por Shirlei Righeti – CRB-8/6995

**PEDRO HENRIQUE BIDOIA DE LIMA**  
**AUTOMAÇÃO INDUSTRIAL DE UM SISTEMA DE ARMAZENAMENTO EM UM GALPÃO**

Trabalho apresentado para obtenção do certificado de especialização pelo curso de Engenharia de controle e Automação do Departamento de Engenharia Mecânica e Elétrica da Universidade de Taubaté,  
Área de concentração: Automação industrial  
Orientador: Marcelo Pinheiro Werneck

**Data:** \_\_\_\_\_

**Resultado:** \_\_\_\_\_

**BANCA EXAMINADORA**

PROF. DR. \_\_\_\_\_

ASSINATURA \_\_\_\_\_

PROF. DR. \_\_\_\_\_

ASSINATURA \_\_\_\_\_

PROF. DR. \_\_\_\_\_

ASSINATURA \_\_\_\_\_

## RESUMO

Automação industrial nada mais é do que a aplicação de um conjunto de técnicas cujo objetivo seria automatizar os diversos processos industriais através da substituição da mão-de-obra por equipamentos diversos. Com isso em mente, para a indústria nada é mais importante do que manter uma alta produtividade com alto nível de qualidade. A mão de obra, com o passar do tempo, começou a atingir seu limite de produção, e então as empresas tiveram que se modernizar. Essas inovações ocorreram através da utilização dos Controladores Lógicos Programáveis entre outras novidades. Graças a essa mudança e outras, a mão-de-obra vem sendo substituída pouco a pouco com o passar do tempo. Com isso em mente o presente trabalho visa utilizar os simuladores CODESYS e FACTORY IO, todo o conhecimento adquirido ao longo do curso e os resultados obtidos ao longo do desenvolvimento, para elaborar um programa em CLP, que irá paletizar caixas recebidas, separá-las por altura, armazená-las e por fim retirá-las do armazém dentro de uma empresa fictícia, substituindo a mão de obra lá existente.

Palavras-Chave: CLP. Programação em ladder. CodeSys. Factory IO. Automação.

## **ABSTRACT**

Industrial automation is nothing more than the application of a set of techniques whose objective would be to automate the various industrial processes by replacing labor with various equipment. With this in mind, for the industry nothing is more important than maintaining high productivity with a high level of quality. As time went by, labor began to reach its production limit, and so the companies had to modernize. These innovations occurred through the use of Programmable Logic Controllers, among other innovations. Thanks to this change and others, labor has been replaced little by little as time goes by. With this in mind the present work aims to use the CODESYS and FACTORY IO simulators, all the knowledge acquired throughout the course and the results obtained throughout the development, to elaborate a program in PLC, which will palletize incoming boxes, separate them by height, store them and finally remove them from the warehouse within a fictitious company, replacing the existing labor force there.

Keyword: CLP. Ladder programming. CodeSys. Factory IO. Automation

## SUMÁRIO

1. INTRODUÇÃO	7
2. Referencial Teórico	8
DESENVOLVIMENTO	
3. Metodologia	9
4. Simulação	10
5. Programação	12
6. Comunicação	16
7. Resultados	19
8. CONCLUSÃO	20
9. Referencias	21
10. Anexos	22

## 1. INTRODUÇÃO

O ser humano vem a muito tempo procurando formas de simplificar as suas obrigações, e com o passar do tempo a automação vem ajudando-o a atingir esse objetivo. Quando se fala em automação, se fala sobre aplicar técnicas em um processo para reduzir parcialmente ou totalmente a mão-de-obra no mesmo. Segundo Scielo (2011, p.120), a palavra automação vem do latim, *automatus* que significa “mover-se por si”. Desde muito tempo atrás o homem é visto tentando poupar esforços, para isso ele vem criando máquinas, desde as mais primitivas como a roda, até os mais complexos computadores atualmente. A automação industrial começou a se destacar a partir do século XVIII.

Foi aproximadamente nesse século que as primeiras máquinas de automação começaram a surgir, mas foi somente no próximo século que elas se tornaram inteiramente autônomas. Essas máquinas, por sua vez, surgiram pela necessidade de as empresas sempre querer produzir mais, e isso aumentou a necessidade de inovações tecnológicas na área. Já no século XX, componentes como controladores programáveis, servomecanismos e computadores foram introduzidos na automação industrial. Podemos, então, dizer que a inclusão dessas partes formou a base para a automação atualmente utilizada.

Atualmente a automação se divide em três categorias sendo essas a fixa, programável e flexível. No primeiro, toda operação é realizada acompanhando as configurações do equipamento. A principal característica desse sistema é ausência de configurabilidade, ou seja, não é possível alterar a forma como funciona, sendo assim difícil criar produtos diferenciados. Esse sistema é amplamente utilizado em linhas de montagem.

Já a automação programável, como o nome implica, possui a capacidade de reprogramação, permitindo assim variações na sua operação. Além de ser possível alterar a programação, é possível incluir novos programas, criando assim produtos novos.

A programação flexível por sua vez, é o aperfeiçoamento das duas anteriores. Com essa característica, é possível uma indústria fabricar diversos produtos ao mesmo tempo e sem comprometer a qualidade do produto, garantindo assim uma eficiência maior.



Sendo assim, como é possível notar, a utilização da automação industrial traz inúmeras vantagens, tais como um padrão de qualidade na produção, além de facilitar um grande aumento na produção em si. Com isso em mente, o presente trabalho tem como objetivo facilitar o armazenamento de caixas, após as mesmas serem paletizadas, dentro de um galpão fictício para melhorar o processo e substituir a mão-de-obra no local, utilizando o conhecimento adquirido ao longo do curso, juntamente com uma simulação do mesmo funcionando para melhor entendimento.

## 2. REFERENCIAL TEÓRICO

Para um melhor entendimento do assunto é necessário primeiro falar sobre os conceitos que foram utilizados para elaborar esse projeto, sendo esses: controladores programáveis lógicos, programação ladder.

O controlador lógico programável é uma parte fundamental em qualquer processo de automação industrial, pois é ele quem controla tudo através da execução de funções específicas em um dado programa desenvolvido por um usuário.

Esse controle que o CLP exerce sobre um processo ocorre em ciclos de varredura ou scans. Segundo Moraes (2006, p. 83), o “CPU do controlador programável executa todas as instruções, começando pela primeira linha do programa e indo até a última instrução da última linha do programa”, ou seja, um ciclo de varredura ou scan nada mais é do que uma sequência de eventos que é executado seguindo uma lógica pré-programada. Na figura abaixo é possível ver uma imagem ilustrativa de um ciclo composto pelo início do processo, leitura das entradas, execução do programa, atualização das saídas.

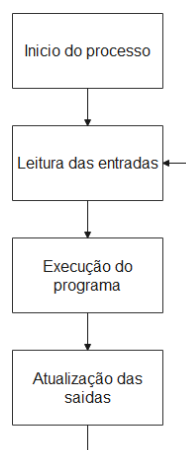


Figura 1 – Ciclo de varredura  
Fonte: Autor

Esse ciclo de varredura é perceptível na programação ladder que foi utilizado neste trabalho. Essa linguagem originalmente foi utilizada em circuitos a relés, que por sua vez, eram utilizados em linhas de produção. Segundo Ribeiro(2001, p.82), o nome ladder (escada em inglês) foi dado a esse tipo de linguagem de programação devido à forma como ele é montado, de cima para baixo e da esquerda para a direita, e que cada linha feita se chama de rung (degrau).

Um rung, por sua vez, pode possuir vários componentes dentro dele sendo esses, contatos abertos ou fechados, saídas, temporizadores, contadores, instruções como MOV, ADD e GT entre outros. Na figura a seguir é possível ver uma representação das mesmas dentro de uma rung.

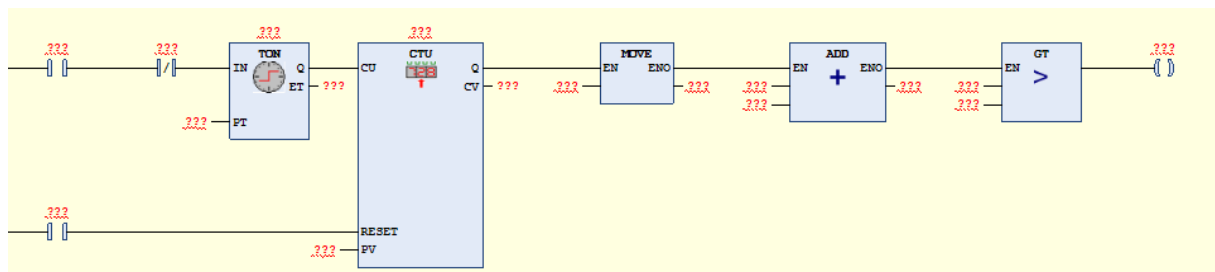


Figura 2 – Imagem ilustrativa de uma rung com todas as instruções utilizados neste projeto  
Fonte: Autor

### 3. METODOLOGIA

O presente trabalho foi feito baseado nos conselhos do orientador, no livro Engenharia de Automação Industrial de Moraes (2006), conhecimento adquirido dentro da sala de aula durante todo o curso e através da tentativa e erro. O primeiro passo na elaboração desse projeto foi escolher o cenário sobre o qual montar o programa, que nesse caso foi o de automatizar um sistema que organiza caixas por altura e as armazena em um galpão de uma empresa fictícia, de modo a substituir a mão-de-obra humana por máquinas.

Com o plano em mente, a segunda coisa a se fazer foi montar o galpão no programa Factory IO onde a simulação ocorreria. Após as máquinas estiverem no devido lugar dentro do galpão, será necessário desenvolver o programa em ladder, dentro do CODESYS, que irá controlar todo o processo de paletização, separação por altura, armazenamento e retirada do armazém.

Com o programa desenvolvido e as máquinas localizadas dentro do galpão, será necessário fazer com que o CODESYS e o FACTORY IO se comuniquem entre si, para ser possível rodar a simulação do sistema funcionando.

#### 4. SIMULAÇÃO

Utilizando o programa FACTORY IO foram escolhidos quatro cenários diferentes, sendo esses, uma máquina que separa peças por altura, uma máquina que irá pegar essas peças e colocá-las em pallets, uma máquina que irá armazenar as caixas junto com os pallets e por último a mesma máquina só que está fará o contrário, ou seja, irá retirar as caixas junto com o pallet. As figuras a seguir servirão para melhor entender cada item discutido acima.



Figura 3 – Máquina separadora por altura  
Fonte: Autor

Na figura 3 acima, temos a máquina separadora por altura, sua função é receber as caixas de tamanho grande ou pequena, verificar qual tipo de caixa está sobre o pallet e então direcioná-las para o local adequado, sendo esses, esquerda para caixas pequenas e direita para caixas grandes.

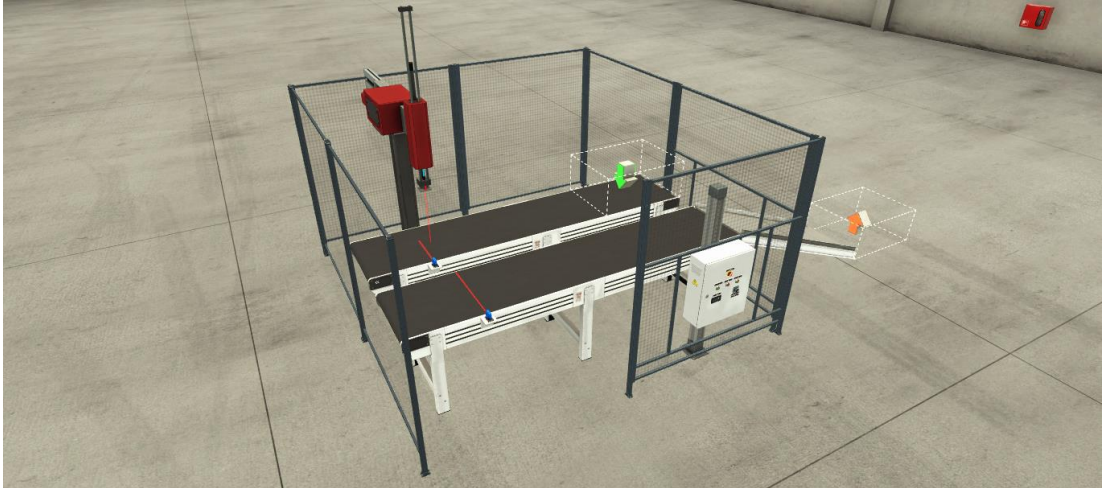


Figura 4 – Máquina paletizadora  
Fonte: Autor

Na figura quatro acima, temos a máquina paletizadora, cuja função é colocar as caixas, pequenas ou grandes, em cima de um pallet que irá surgir na outra esteira.

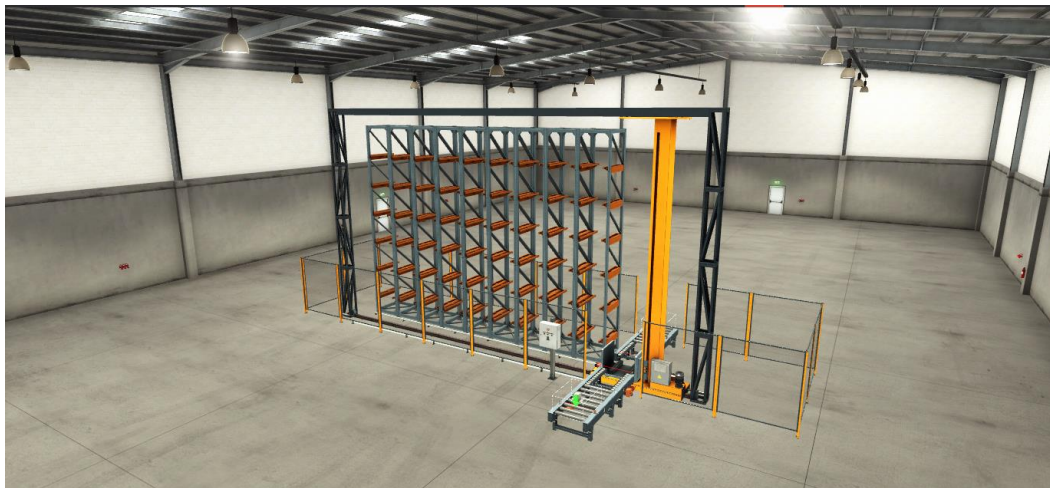


Figura 5 – Máquina de armazenamento  
Fonte: Autor

Na figura cinco acima, temos a máquina que irá armazenar as caixas, pequenas ou grandes dependendo de qual lado for, em seus devidos lugares, nesse caso é possível armazenar até no máximo 54 caixas. Lembrando que, mesmo não sendo possível ver na figura 5, essa mesma máquina será usada para desarmazenar as caixas, ou seja, retirá-las do seu lugar de armazenamento. Nesse caso haverá uma máquina do lado oposto da que armazena fazendo a função inversa.

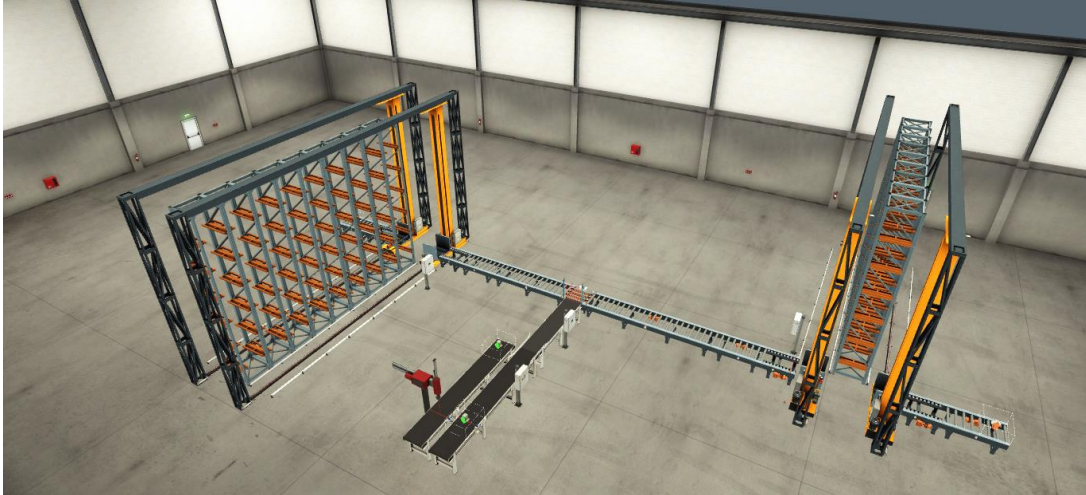


Figura 6 – Máquina finalizada  
Fonte: Autor

Já na figura seis, é possível ver a união entre os equipamentos das figuras três, quatro e cinco acima, foi possível montar uma máquina maior capaz de realizar o processo todo automatizado, sendo assim possível substituir a mão-de-obra fictícia dessa empresa.

## 5. PROGRAMAÇÃO

Com o galpão completamente simulado, agora seria necessário desenvolver o programa em ladder que irá controlar todo o maquinário. A programação em ladder será anexada a esse documento, devido ao fato de o mesmo ter ficado muito grande, sendo assim, para entender melhor a desenvolvimento desse programa, o mesmo será explicado utilizando os fluxogramas abaixo.



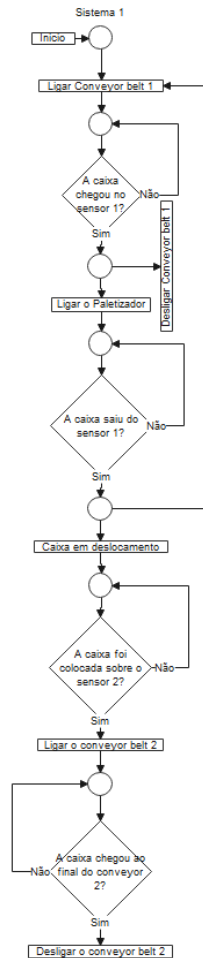


Figura 7 - Sistema 1  
Fonte: Autor

O sistema 1, representado pela figura 7, é equivalente à figura 4 visto anteriormente. Nele é possível ver o fluxograma da programação que controla o paletizador. Ele é um sistema bem simples onde, ao ser ligado a máquina o conveyor belt 1 é ligado, levando assim uma caixa para o primeiro sensor. Quando essa caixa chegar no sensor, o conveyor é desligado para que o paletizador possa pegá-lo, iniciando assim o movimento do conveyor 1 para o 2.

No mesmo momento em que a caixa é erguida e inicia seu deslocamento, ao sair do sensor 1, o conveyor volta a funcionar, podendo assim carregar a próxima caixa quando a mesma surgir. Quando a caixa concluir sua transferência e é colocada sobre o sensor 2, o próximo conveyor belt será ligado, levando a caixa para o sistema 2. No momento em que a caixa sair do conveyor 2 e chegar no 3, o conveyor 2 será desligado para que o próximo pallet possa receber a próxima caixa.

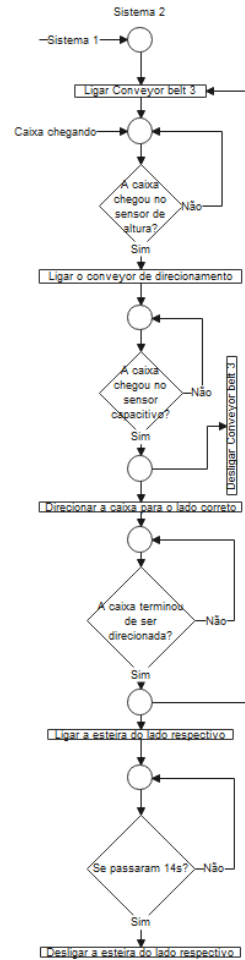


Figura 8 - Sistema 2  
Fonte: Autor

O sistema 2 por sua vez, representado pela figura 8 é a máquina separadora por altura, cuja função como o nome implica, é separar as caixas que chegarem nele. Assim que a máquina é ligada, o conveyor belt 3 é ligado, e quando a caixa do sistema 1 passar pelo conveyor belt 3 e atingir os sensores de altura, o próximo conveyor de direcionamento será ligado e o conveyor 3 será desligado para que não entrem várias caixas na próxima parte da máquina enquanto tiver uma já no seu interior. Esse conveyor de direcionamento irá puxar a caixa até ele atingir um sensor capacitivo. Quando o sensor for acionado, a caixa iniciará a ser deslocada para o lado respectivo dependendo do tamanho da caixa. E por fim após se passarem 14 segundos essa esteira será desligada e a caixa atingirá o sistema 3.

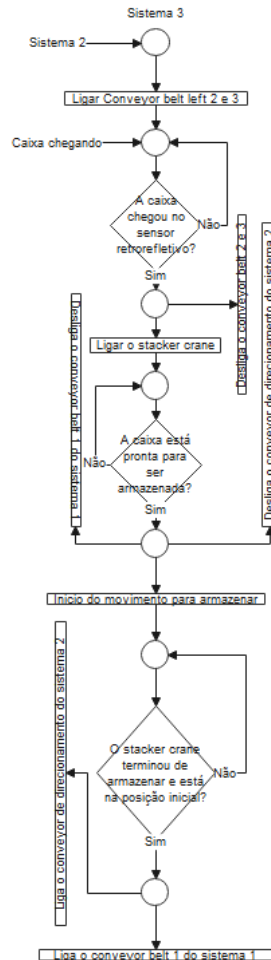


Figura 9 - Sistema 3  
Fonte: Autor

O sistema 3, visível na figura 9 e simulado na figura 5 é responsável por receber a caixa do sistema 2 e armazená-la. Esse processo tem início quando a máquina é ligada, fazendo com que o conveyor belt 2 e 3 do lado respectivo sejam ligados. Quando a caixa proveniente do sistema 2 passar por eles e chegar no sensor retro refletivo duas coisas irão acontecer. A primeira é que os conveyor 2 e 3 que trouxeram a caixa até o sensor serão desligados e o stacker crane, máquina que armazena a caixa, será ligado.

Quando o stacker crane pegar a caixa e colocá-la no seu interior, o conveyor belt 1 do sistema 1 será desligado, assim como o conveyor de direcionamento do sistema dois para que não ocorra um acúmulo de caixas nos sistemas com um todo devido ao tempo que a caixa demora para ser armazenada. Depois de a caixa ser armazenada o sistema 4 a seguir entra em ação, o stacker crane voltar para a posição inicial, o conveyor belt 1 do sistema 1 e o conveyor de direcionamento do sistema 2 voltam a funcionar, sendo assim possível receber novas caixas.



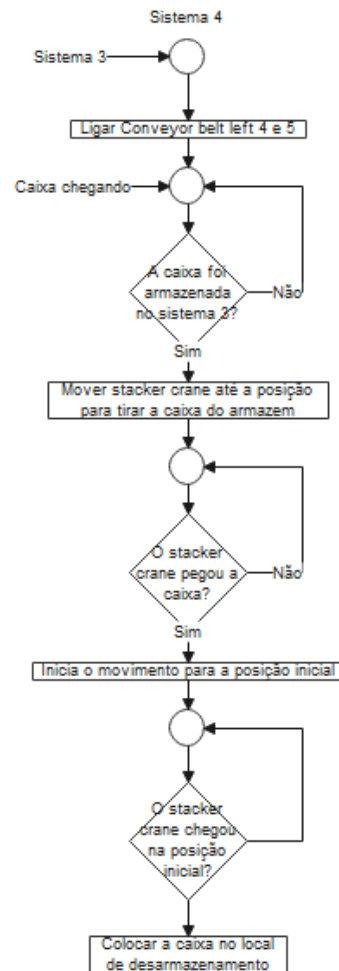


Figura 10 - Sistema 4  
Fonte: Autor

O sistema 4, visível na figura 10 e simulado na figura 5, juntamente com o sistema 3, é responsável pelo desarmazenamento das caixas após as mesmas serem guardadas. Esse processo tem início após a caixa ser armazenada e o stacker crane de armazenamento iniciar o seu retorno à sua posição inicial. Nesse momento o stacker crane de desarmazenamento vai até o local onde a caixa está armazenada, remove a caixa do local e inicia o retorno para a posição inicial. Após chegar na posição inicial ele coloca a caixa sobre a esteira e conclui o processo.

## 6. COMUNICAÇÃO

Com a programação e a simulação concluído será necessário fazer com que o FactoryIO e o CodeSys se comuniquem. O primeiro passo para que isso ocorra é escolher o driver apropriado para o FactoryIO, que nesse caso é o OPC Client DA/UA.

Em seguida, na hora da criação do projeto no CODESYS, o device do mesmo tem que ser o CodeSys Control Win V3 x64 e o plc\_prg tem que ser o Ladder Logic Program, caso contrário ocorrerá problemas na hora da comunicação.

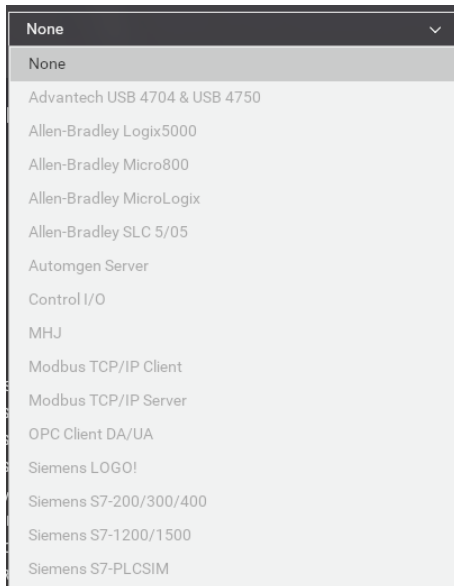


Figura 11 – Driver  
Fonte: Autor

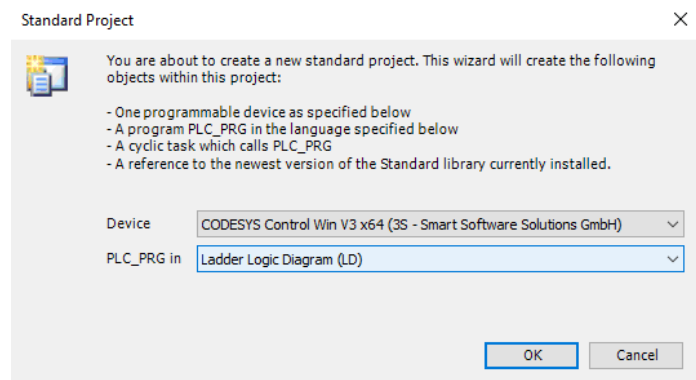


Figura 12 – Criação do programa  
Fonte: Autor

Com isso feito, agora seria a hora de desenvolver o programa e as variáveis, que no caso desse projeto estará em destaque nos anexos. O próximo passo será compartilhar essas variáveis para que os dois programas os utilizem, para isso será necessário clicar com o botão direito no application, e em seguida clicar em symbol configuration. Com isso aberto, será necessário clicar em build para ele buscar todas as variáveis que foram criadas dentro do programa, e em seguida selecionar as que serão compartilhadas com o FactoryIO.

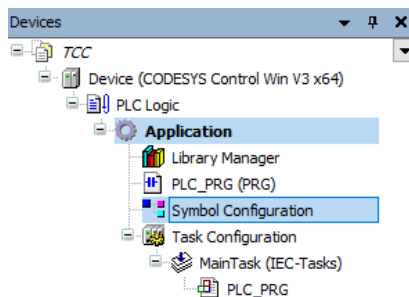


Figura 13 – Symbol configuration  
Fonte: Autor

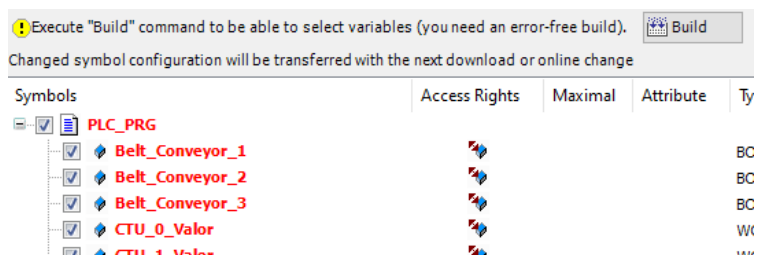


Figura 14 – Escolha das variáveis  
Fonte: Autor

Com as variáveis prontas para serem compartilhadas, agora será necessário abrir o programa chamado CODESYSControlSysTray que irá controlar a troca de informação entre os dois programas. Com o programa anterior aberto, será necessário fazer o login dentro do CodeSys.



Figura 15 – CODESYSControlSysTray  
Fonte: Autor

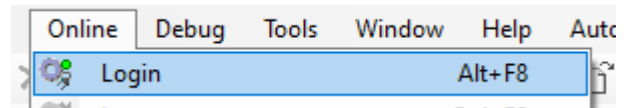


Figura 16 – Login  
Fonte: Autor

Agora que tudo dentro do CodeSys está pronto, falta apenas concluir tudo no FactoryIO. Com isso em mente será necessário clicar em configuration dentro dos drivers, buscar o servidor sendo usado entre os dois programas e escolhê-lo e buscar pelos itens que existem nesse servidor, no caso as variáveis que foram escolhidas anteriormente.



Figura 17 – Factory i/o Configuration  
Fonte: Autor

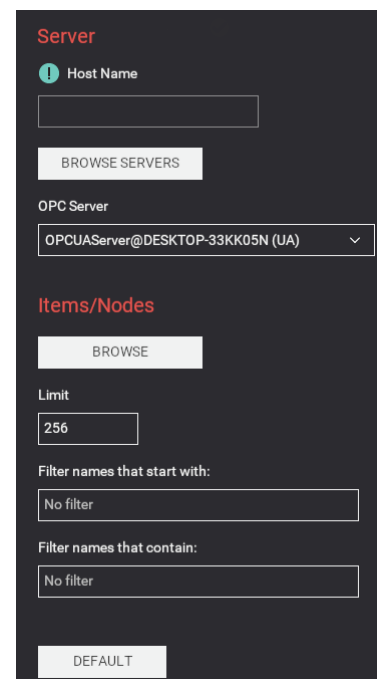


Figura 18 – Factory i/o escolhendo o servidor  
Fonte: Autor

Agora é simplesmente fazer a comunicação entre os sensores e os atuadores dentro dos drivers tudo estará pronto. A única coisa que resta é clicar play em ambos os programas para dar início à simulação.



Figura 19 – Factory i/o fazendo a comunicação com o CodeSys  
Fonte: Autor

Como é possível ver na figura 19 temos duas fileiras em branco, a da esquerda, responsável pelos sensores, e a da direita responsável pelos atuadores. Com os dois programas conectados, é possível ligar o lado esquerdo com o lado direito, fazendo com que o CodeSys possa receber e emitir informações para o simulador.

## 7. RESULTADOS

Com a simulação e o programa pronto, basta testar os dois juntos e ver o resultado, é importante ressaltar que, as máquinas simuladas nesse projeto foram programadas separadas, e depois de confirmar sua funcionalidade é que foram unidas.

O resultado inicial da simulação vista na figura 6, deu vários problemas como velocidades distintas entre o roller conveyor e o belt conveyor, a diferença entre o tempo das partes moveis para concluir uma ação, a desigualdade entre o tempo que o sistema 3 leva para guardar as caixas em posições diferentes e o desalinhamento das caixas durante seu percurso.

Para resolver esses problemas acima a forma mais fácil foi de encontrar algo em comum entre os três primeiros problemas, que nesse caso foi o tempo. Utilizando temporizadores, após várias tentativas, ao longo da programação foi possível achar um equilíbrio entre as partes do sistema um e dois para que não ocorresse um acúmulo de caixas. A utilização dos temporizadores foi eficaz nos locais onde o tempo

não se altera. No caso de sistema três onde as caixas são armazenadas, como o tempo de ida e volta se altera conforme o número de caixas no armazém aumenta, foi necessário utilizar um selo que impede que o sistema um e dois executem suas funções durante o processo de armazenagem de caixa. Dessa forma evitando que ocorra o acúmulo de caixas.

Já o problema de alinhamento foi resolvido utilizando alinhadores nas partes mais propícias a ocorrer desalinhamento, sendo essas a área onde a caixa é colocada sobre o palete, a junção entre o paletizador e o separador por altura e por último no fim do separador por altura para que a caixa chegue no sistema três pronto para ser armazenado.

Com tudo isso implementado foi possível elaborar um programa que é pouco provável de encontrar um problema, ele pode ficar ligado por várias horas e não ocorrer nada além do que o programado.

## **8. CONCLUSÃO**

Uma empresa sem automação se encontrará limitada com o tempo, em relação à produção, assim como a qualidade dos produtos que é fabricado por eles. Por isso é de extrema importância que toda empresa busque melhorar os seus processos de produção através da automação. Com o intuito de substituir a mão de obra existente em uma empresa de armazenagem de caixas, a mesma tornou sua atenção para o campo da automação.

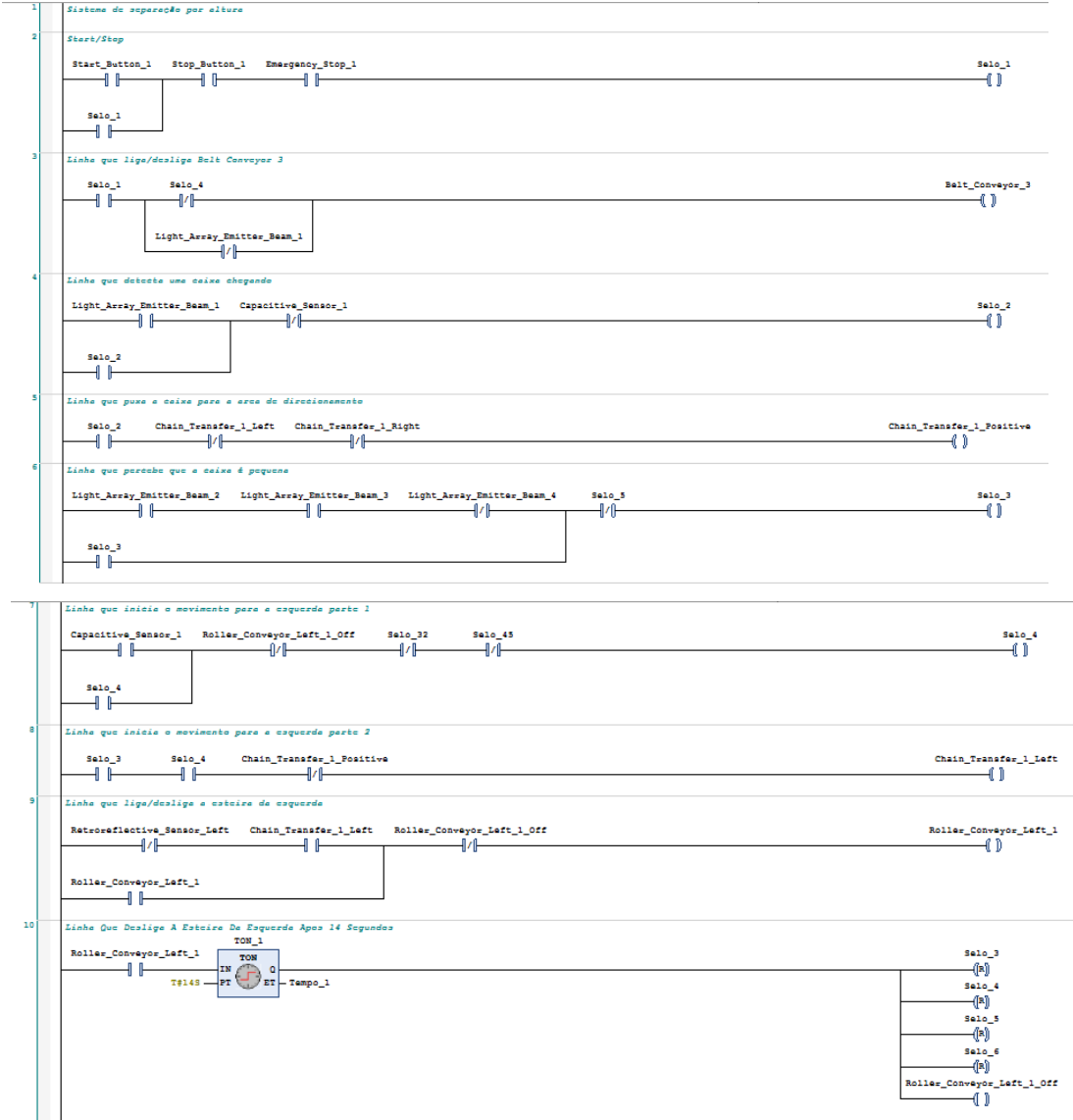
Para elaborar esse projeto foi necessário elaborar todos os sistemas utilizados, juntar eles em uma máquina só, testar e corrigir possíveis erros. Com o programa desenvolvido e os erros encontrados nos resultados corrigidos, foi possível automatizar o processo de armazenagem de caixas, possibilitando a substituição da mão de obra por uma máquina automatizada. Graças a isso a empresa conseguiu melhorar o seu processo de armazenagem e substituir a mão de obra, atingindo assim a meta deste trabalho.

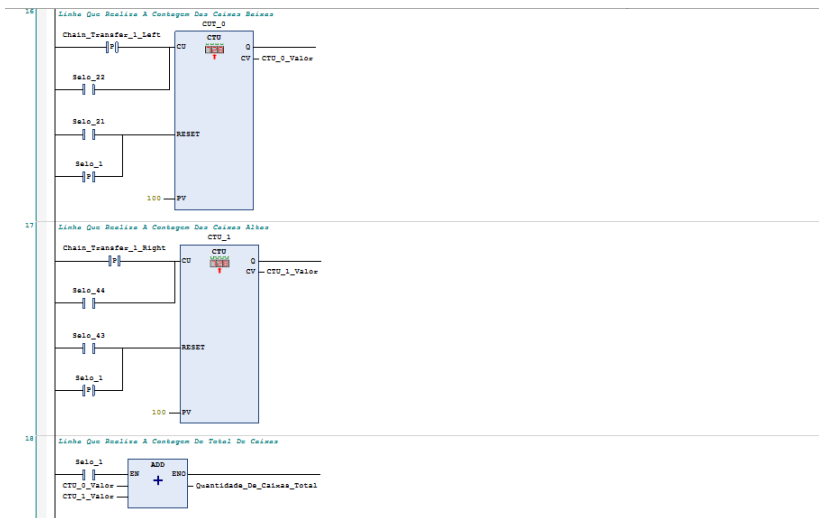
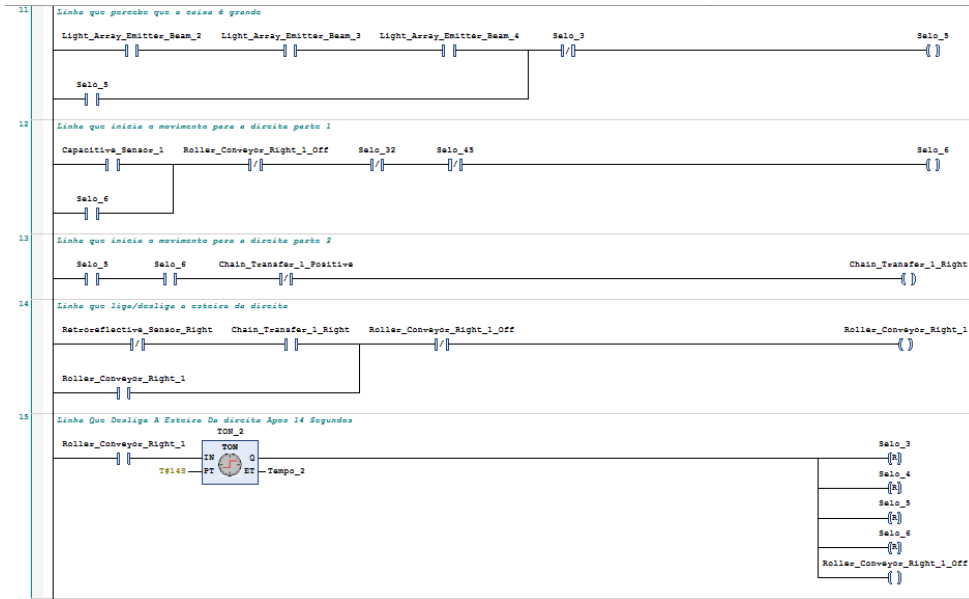
## 9. REFERÊNCIAS

- Uma breve história da automação industrial. **Smartec automação**, 2018.  
Disponível em: <https://www.smartec-automacao.com.br/blog/15532-2/>. Acesso em: 17 de Jan. de 2021
- Conheça os tipos de automação industrial, **Avozdaindustria**, 2019. Disponível em <https://avozdaindustria.com.br/inova-o/conhe-os-tipos-de-automa-o-industrial>. Acesso em: 17 de Jan. de 2021
- CLP – O que é e como funciona o Controlador Lógico Programável, **Alfacompbrasil**, 2019. Disponível em <https://alfacompbrasil.com/2019/02/11/clp-o-que-e-e-como-funciona/>. Acesso em: 16 de Jan. de 2021
- O que é CLP e qual a sua aplicação, **mundodaeletrica**, [s.d.]. Disponível em <https://www.mundodaeletrica.com.br/o-que-e-clp-para-que-serve/>. Acesso em: 16 de Jan. de 2021.
- Controlador Lógico Programável CLP, **mundodaeletrica**, [s.d.]. Disponível em <https://www.mundodaeletrica.com.br/controlador-logico-programavel-clp/>. Acesso em: 16 de Jan. de 2021.
- Conceito de automação na medicina laboratorial: revisão de literatura, **Scielo**, 2011. Disponível em <http://www.scielo.br/pdf/jbpml/v47n2/v47n2a05.pdf>. Acesso em 18 de Jan. de 2021.
- Moraes, Cícero Couto. Engenharia de Automação Industrial, 2ª edição, São Paulo: LTC Livros Técnicos e Científicos S. A., 2001.
- Ribeiro, Marco Antônio. Automação Industrial, 4ª edição, [s.i]: [s.n], 2001

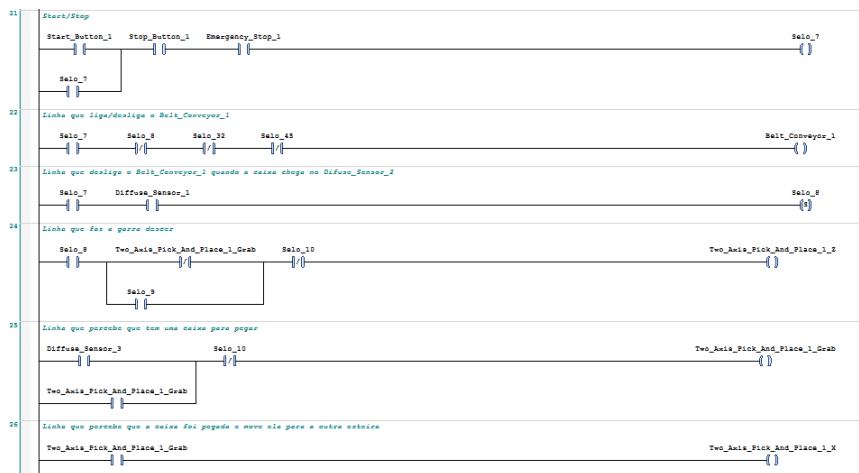
# 10. ANEXOS

## Sistema de Separação por altura

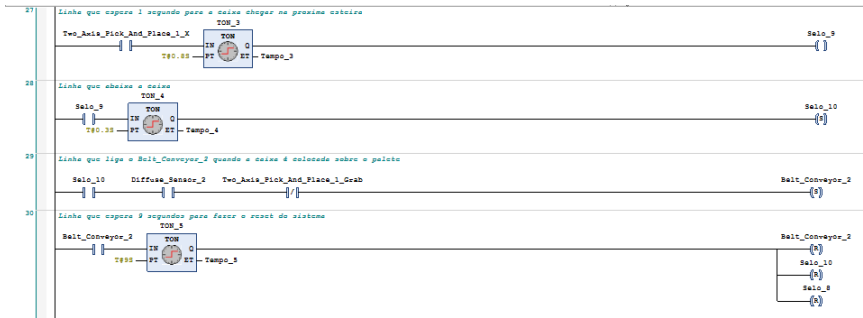




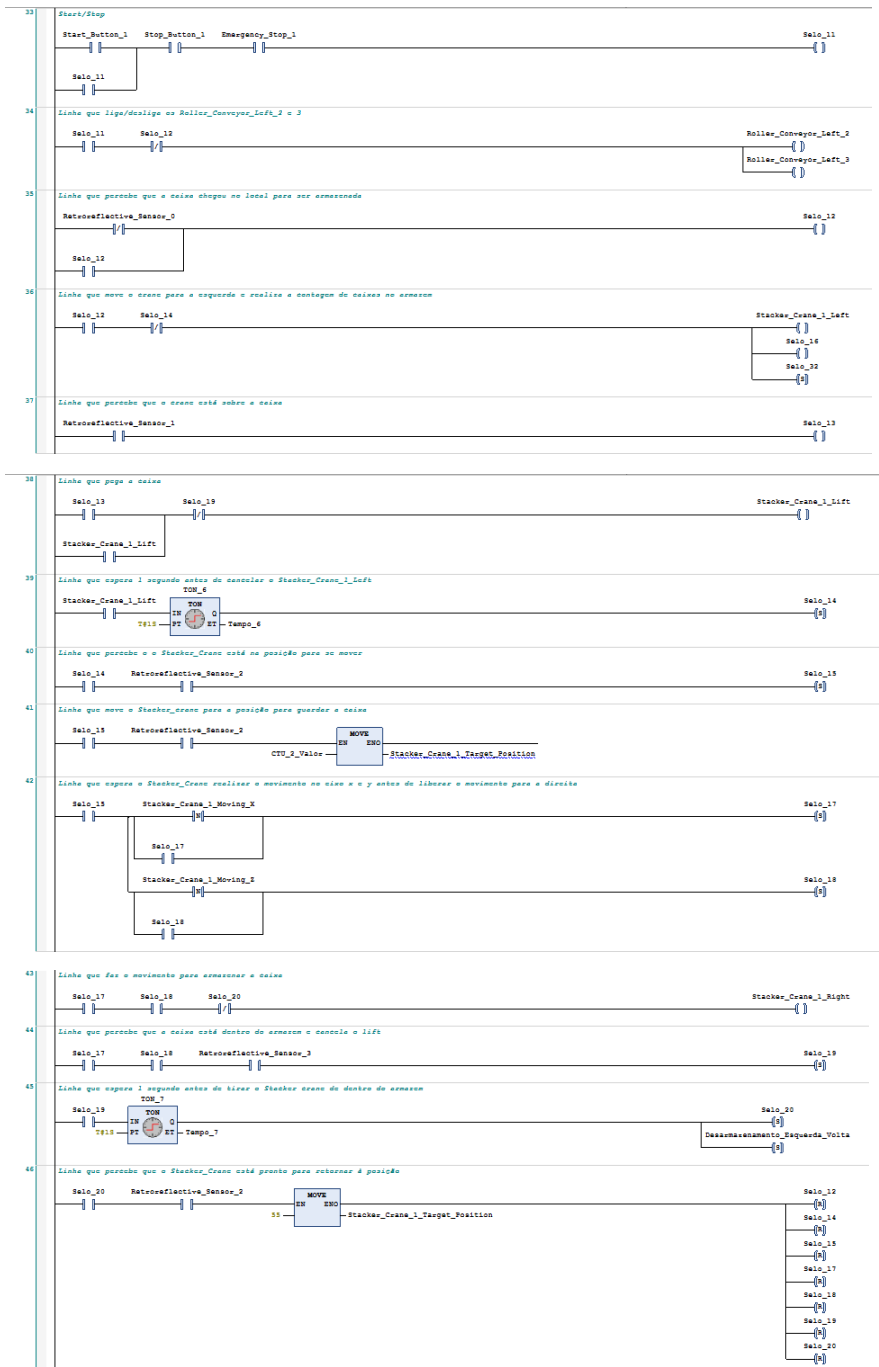
### Sistema de Empilhamento





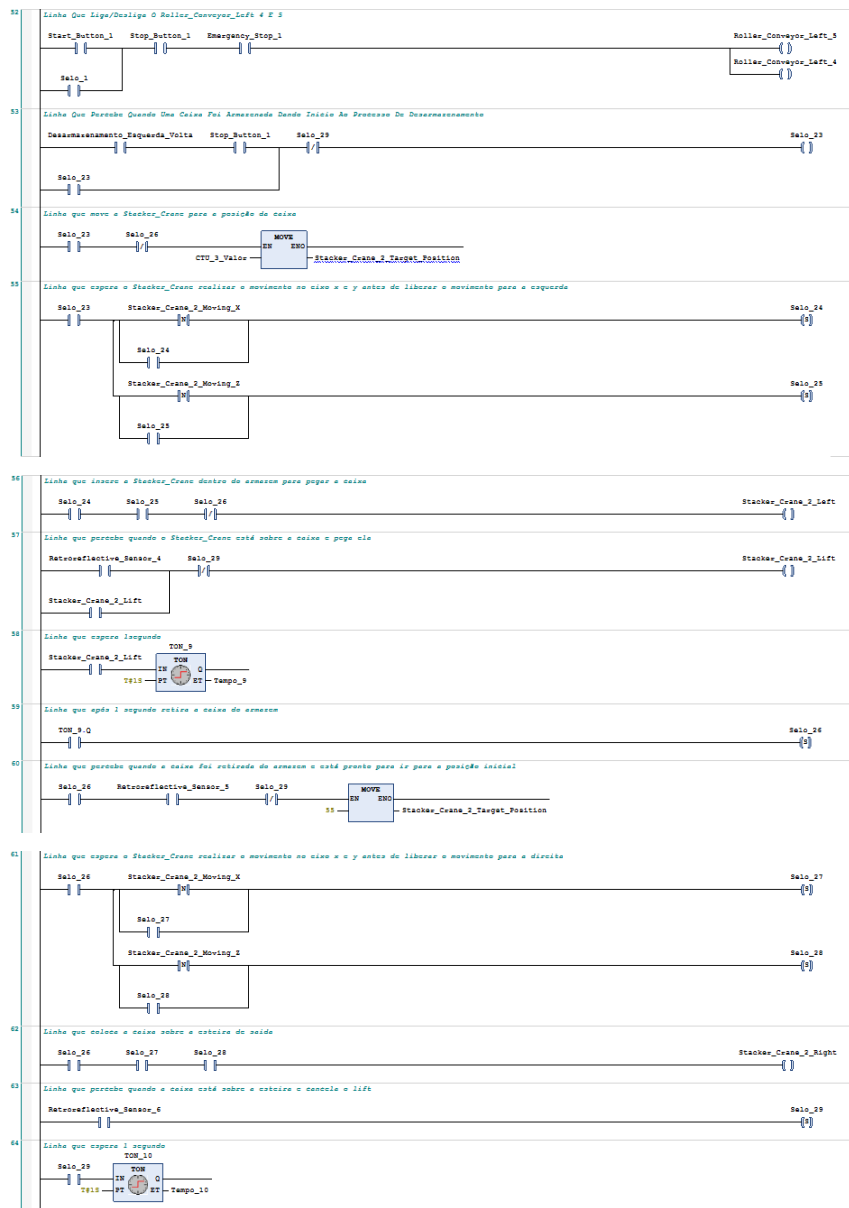


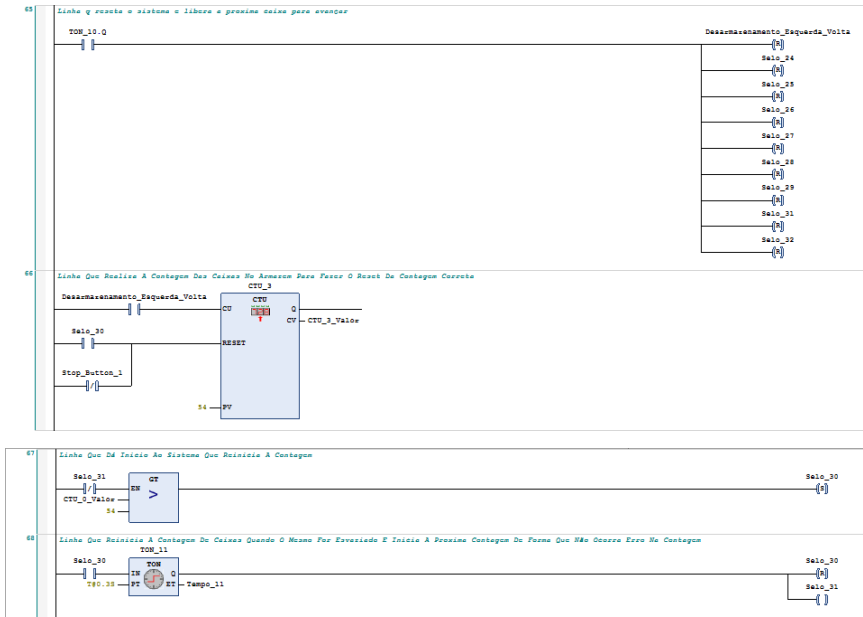
### Sistema de Armazenamento da esquerda



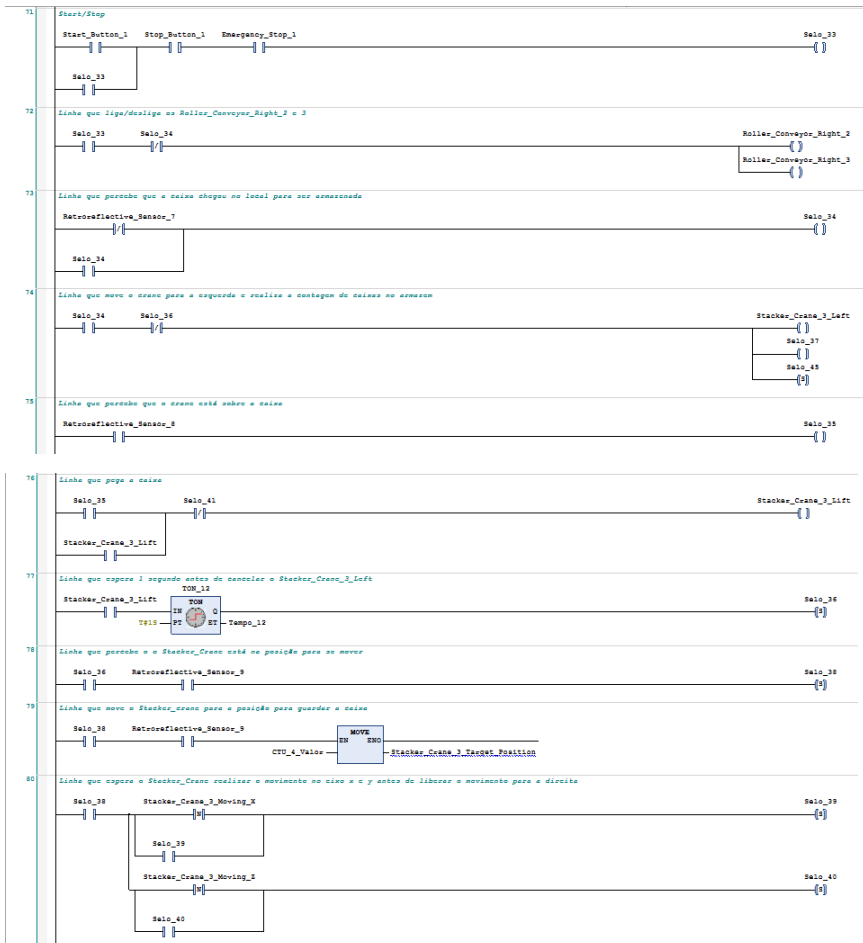


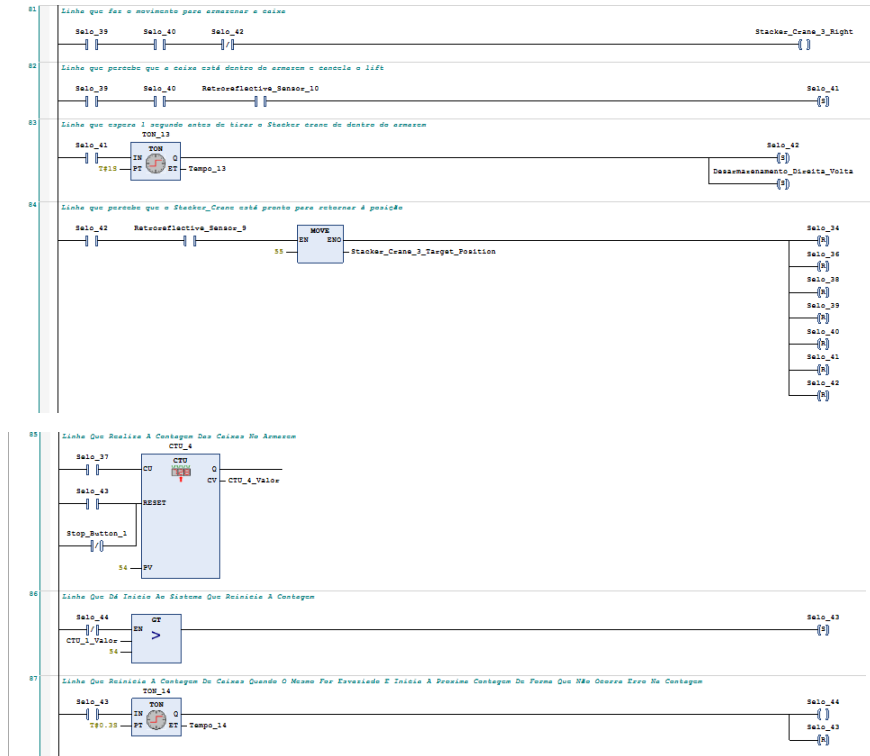
### Sistema de desarmazenamento da esquerda





### Sistema de armazenamento da direita





### Sistema de desarmazenamento da direita

