



**JOÃO PAULO MORAES OLIVEIRA**

**AUTOMAÇÃO DO PROCESSO DE APLICAÇÃO DE  
HOLDERS EM VIDROS AUTOMOTIVOS.**

**LAVRAS – MG**

**2022**

**JOÃO PAULO MORAES OLIVEIRA**

**AUTOMAÇÃO DO PROCESSO DE APLICAÇÃO DE HOLDERS EM VIDROS  
AUTOMOTIVOS.**

Relatório Técnico apresentado à  
Universidade Federal de Lavras,  
como parte das exigências do Curso  
de Engenharia de Controle e  
Automação, para a obtenção do  
título de Bacharel.

Prof. Dr. Leonardo Silveira Paiva  
Orientador

**LAVRAS – MG  
2022**

## **AGRADECIMENTOS**

A Deus, pela vida e pelas oportunidades cedidas a mim para que eu pudesse alcançar meus objetivos e sonhos, sempre estando ao meu lado.

A família, que sempre me incentivou na busca dos meus sonhos e por todo o amor e apoio durante os momentos difíceis, mesmo eu estando ausente em momentos importantes.

Ao corpo docente da Universidade Federal de Lavras, por todos esses anos de ensinamentos e orientações, que contribuíram para a formação profissional e pessoal.

Aos amigos, por todos esses anos de convivência e aprendizado, que contribuíram muito para meu desenvolvimento pessoal.

Aos colegas de trabalho, que me ajudaram nesse início de carreira com suas experiências e lições.

## RESUMO

O presente trabalho foi desenvolvido a partir das experiências obtidas na empresa Emax Serviços de Automação, localizada na cidade de São José dos Campos - SP. A Emax é uma empresa atuante no setor de automação industrial, visando o desenvolvimento de máquinas que auxiliam na produção de empresas clientes, tais como Pilkington, da NSG Group, Saint Gobain, Ts Trim Brasil, General Electric, entre outras. Neste trabalho, foi projetada uma máquina capaz de aplicar peças *holder* em vidros automotivos, visando agilizar tal processo que antes era feito de maneira braçal. Para isso, foi desenvolvida a programação do controlador da máquina, utilizando o diagrama *Ladder* para seu controle, além da interface gráfica homem-máquina, para a integração entre o operador e o processo a ser executado. Para melhor execução do processo, seguimos um procedimento padrão da empresa para melhor apresentar o projeto ao cliente. Primeiro é realizado um teste com os componentes de comunicação em rede Profinet, a fim de verificar sua funcionalidade e a troca de dados com o CLP, em seguida é feita a verificação dos elementos de segurança, como cortinas de luz, botões de emergência e sensores de portas. Então, quando a estrutura da máquina e seu painel elétrico estão prontos e montados, é feita a programação seguindo as diretrizes dadas pelo cliente para melhor execução do processo de aplicação de *holders*, como a sequência de acionamento de válvulas pneumáticas para a centralização do vidro e aplicação da peça. O projeto conta com modos de controle manual, onde os processos são realizados pelo operador através da interface homem máquina (IHM), e controle automático, que faz todo o procedimento citado com o mínimo de auxílio do operador, sendo limitado apenas ao posicionamento das peças e do vidro.

**Palavras-Chave:** Aplicação. CLP. *Holders*. Vidro.

## ABSTRACT

The current paper was developed from the experience obtained at Emax Serviços de Automação, located at São José dos Campos - SP. Emax is a company active in the industrial automation sector, building machines to other industries, like Pilkington, from NSG Group, Saint Gobain, Ts Trim Brasil, General Electric, among others. In this paper, a machine was built to apply holder pieces in car windows , changing from the manual labor that was being done. The task that was assigned was the programming of the PLC, using Ladder diagram, to control the machine and the HMI screen, to allow the control by the operator. To better execute this task, we follow the company's procedure to develop this machine and present it to the client: First, tests are made with the components that use the Profinet communication, to check the data trade between the components and its functionality, followed by the safety elements being checked, like the security light curtain, emergency buttons and door sensors. Then, when all the structure and the electric panel are assembled, the programming begins, following the instructions of the client to better execute the application of the holders, instructions like the sequence of valves that must be activated to center the window and apply the holder. The project uses manual control, where the operator can control the machine by the HMI buttons, and the automatic control, where the machine does the application with minimum help from the operator, who just positions the holder pieces and the car window at the machine.

**Keywords:** Application. Holders. PLC. Window.

**LISTA DE FIGURAS**

Figura 1 - Antiga Sede da Emax em São José dos Campos - SP	12
Figura 2 - Linha do tempo das Revoluções Industriais	15
Figura 3 - Ciclo de funcionamento do CLP	17
Figura 4 - Código implementado para acionar lâmpada	18
Figura 5 - Sensor Indutivo	20
Figura 6 - Fotoresistor (LDR)	21
Figura 7 - Chave Fim-de-Curso	22
Figura 8 - Sensor Capacitivo	23
Figura 9 - Topologia de comunicação de rede Profibus	25
Figura 10 - Topologia de comunicação de rede Profinet	27
Figura 11 - Telas HMI Basic Panel Siemens	28
Figura 12 - Tela Principal de uma IHM	29
Figura 13 - <i>Holder</i> de Vidro Dianteiro	30
Figura 14 - Configuração de Hardware	31
Figura 15 - Configuração dos endereços do CLP	32
Figura 16 - Sinais de entrada	32
Figura 17 - Sinais de saída do bloco de válvulas	33
Figura 18 - Programa teste para acionamento das válvulas	34
Figura 19 - Saídas da segurança	35
Figura 20 - <i>Data Type</i> “UDT_Cilindros”	36
Figura 21 - <i>Data Block</i> DbCilindros	37
Figura 22 - Lista de <i>Data Block</i> e DbModos	38
Figura 23 - Modo de Segurança e Modo Manual	39
Figura 24 - Bloco de controle de válvulas pneumáticas	40
Figura 25 - Network passo 110 do AutoReset	41
Figura 26 - Sequência AutoReset	42

Figura 27 - Comando de avanço e recuo automático	45
Figura 28 - Tela By Pass Sensores	46
Figura 29 - Temporizador de aplicação de <i>holder</i>	46
Figura 30 - Tela Parâmetros	47
Figura 31 - Telas de Produção	48
Figura 32 - Máquina de Aplicar <i>Holder</i> s	49

**LISTA DE TABELAS**

Tabela 1 - Configuração dos ajustes frontais e laterais	42
Tabela 2 - Ordem de avanço dos elementos de centralização por modelo	44
Tabela 3 - Tabela de Produção	50

**LISTA DE GRÁFICOS**

Gráfico 1 - Amostras de tempo de ciclo

51

## SUMÁRIO

<b>1. INTRODUÇÃO</b>	<b>11</b>
<b>2. OBJETIVOS</b>	<b>13</b>
2.1.    Objetivo Geral	13
2.2.    Objetivos Específicos	13
<b>3. REFERENCIAL TEÓRICO</b>	<b>13</b>
3.1.    Automação industrial	14
3.2.    Controladores lógicos programáveis (CLP)	15
3.3.    Instrumentação	19
3.4.    Comunicação de Rede	23
3.5.    Interface homem máquina (IHM)	27
<b>4. MATERIAIS E MÉTODOS</b>	<b>30</b>
4.1.    Materiais	30
4.2.    Configuração de hardware	31
4.3.    Teste do bloco de válvulas (Manifold)	33
4.4.    Configuração de segurança	34
4.5.    Comandos manuais	35
4.6.    Sequenciamento automático	40
4.7.    Parametrização e alarmes	45
4.8.    Registro de produção	47
<b>5. RESULTADOS</b>	<b>49</b>
<b>6. CONCLUSÃO</b>	<b>51</b>
<b>7. REFERÊNCIAS</b>	<b>53</b>

## 1. INTRODUÇÃO

Desde a Primeira Revolução Industrial, tecnologias vêm sendo desenvolvidas com o intuito de facilitar processos de fabricação, com desenvolvimento de instrumentos capazes de realizar as tarefas além da capacidade humana, como levantar cargas muito pesadas. Seguindo essa ideia, surge a Automação Industrial.

Desde o surgimento da máquina a vapor, os processos industriais são executados mais facilmente, já que os equipamentos vieram com o intuito de melhoria na produção e desde então, muitas empresas e universidades atuam com foco na inovação tecnológica. Assim, foram surgindo controladores lógicos, sensores, atuadores, robôs, entre outras tecnologias, visando a melhor produção, garantia de qualidade, precisão e otimização de tempo.

Em 1957, nos Estados Unidos, surgiu a internet, implementada pelos militares para evitar a perda de informações durante a Guerra Fria em um iminente ataque da União Soviética. Eles idealizavam um modelo de comunicação rápida e descentralizada, assim evitando a exposição de suas informações sigilosas (SACOMANO; e.t, 2018). Com o passar dos anos e as melhorias feitas na internet, passou a ser possível a integração dela com os processos de manufatura e operação remota, podendo controlar equipamentos de lugares distantes.

Em 2011, na Alemanha, houve a primeira menção da Indústria 4.0. Ela surge com a ideia de que os equipamentos industriais possam trocar informações entre si, com o intuito de otimizar a produção e atender as necessidades do cliente. A principal ideia é que todo o processo de manufatura seja feito remotamente, desde a confirmação do pedido do cliente até a entrega do produto final, verificando necessidade de materiais, disponibilidade deles no mercado, prazos de entrega, possíveis reconfigurações de linhas de produção, entre outros fatores que possam interferir na entrega. No desenvolvimento das indústrias 4.0, alguns elementos essenciais são tratados, tais como *Cyber-Physical System* (CPS), Internet of Things (IoT), *Internet of Services* (IoS), inteligência artificial, computação em nuvem, segurança cibernética, integração de sistemas, entre outros elementos. Esse conceito ainda está em formação e por isso ainda não vemos indústrias que são consideradas como indústria 4.0, apesar de diversas pesquisas estarem sendo realizadas nessa área.

A Emax Serviços de Automação é uma empresa de automação industrial e prestadora de serviços, situada na cidade de São José dos Campos - SP e tem como objetivo

desenvolvimento de máquinas industriais e linhas de produção que atendam as necessidades de seus clientes.

Desde 2007, ela atua na área de automação industrial e possui diversas máquinas produzidas para clientes na região sudeste do Brasil, como Pilkington, Saint-Gobain, Ts Trim Brasil, Takaimec, Lightburry, etc. sendo essas máquinas para vários tipos de aplicações, como produção de luvas de borracha, transporte de vidros, cravamento de botões, a aplicação que será descrita neste projeto, que é a aplicação de *holders* em vidros automotivos, sendo esses os elementos responsáveis por acoplar o vidro ao mecanismo de abertura da janela, podendo esse mecanismo ser mecânico ou elétrico.

Além do desenvolvimento de máquinas, a Emax também desenvolve projetos que utilizam células robotizadas, para realização de processos mais complexos e que necessitam de braços robóticos para realizá-los, como transportes e aplicação de produtos em vidros automotivos. A Figura 1 mostra a antiga sede da empresa, sendo ela subdividida nos setores de engenharia, onde são desenvolvidos os projetos; mecânica, responsável pela montagem estrutural da máquina; elétrica, que cuida de todo o cabeamento e sensoriamento, e o setor administrativo, responsável pelos recursos humanos, orçamentos, compras, entre outras tarefas.

Figura 1 - Antiga Sede da Emax em São José dos Campos - SP



Fonte: Site da empresa.

## 2. OBJETIVOS

O trabalho descrito neste documento teve os objetivos citados a seguir.

### 2.1. Objetivo Geral

O objetivo geral deste projeto foi utilizar os conhecimentos adquiridos durante o curso de engenharia de controle e automação.

O intuito é desenvolver a programação da máquina de aplicar *holders*, utilizando linguagem *Ladder* para integrar sinais de sensores e parâmetros estabelecidos na Interface Homem Máquina para acionar os atuadores pneumáticos, a fim de melhor executar o processo de aplicação.

### 2.2. Objetivos Específicos

Com o intuito de melhor desenvolver o projeto, foram listados os seguintes objetivos:

- Verificar o acionamento das válvulas pneumáticas através da comunicação de rede Profinet;
- Especificar sensores a serem utilizados no projeto;
- Desenvolver programação de segurança para CLP Safety;
- Desenvolver controles manuais na programação do CLP;
- Desenvolver modelo de interface gráfica para operar a máquina por controles manuais;
- Implementar, na interface gráfica, indicadores para auxiliar o operador;
- Implementar, na programação, indicadores de quantidade de peças produzidas e tempo de ciclo;
- Verificar, junto ao cliente, possíveis melhorias a serem feitas no processo de aplicação;

## 3. REFERENCIAL TEÓRICO

Para melhor desenvolver esse projeto, foram revisados alguns conceitos vistos em sala de aula, como a automação industrial, Interface Homem-Máquina (IHM), Controladores Lógicos Programáveis (CLP), protocolos de comunicação de rede e instrumentação.

### 3.1. Automação industrial

Segundo Rosário (2009), a automação industrial pode ser definida como a integração de conhecimentos substituindo a observação, os esforços e as decisões humanas por dispositivos (mecânicos, elétricos e eletrônicos, entre outros) e *softwares* concebidos por meio de especificações funcionais e tecnológicas, com uso de metodologias. Existem várias maneiras de se integrar todos esses elementos para se atingir esse objetivo e atualmente os mais utilizados são os controladores lógicos. Assim como mostra a Figura 2, houve uma série de evoluções de tecnologia que levou a esse momento.

Antigamente toda a produção era feita manualmente, sem a utilização de máquinas que faziam todo o processo. Assim, toda produção estava sujeita a erros humanos e tomavam um tempo muito longo para serem finalizadas. A partir dessa ideia, foram sendo desenvolvidas tecnologias capazes de diminuir possíveis erros e aumentar a produção por tempo, acarretando em uma série de eventos definidos como Revolução Industrial.

Ao todo, podemos definir 4 revoluções industriais através dos avanços tecnológicos apresentados. Na primeira, pesquisas foram feitas com o intuito de transformar carvão em uma fonte de energia, assim surgiram as máquinas a vapor e locomotivas (SAKURAI; ZUCHI, 2018), o que tornou a Inglaterra daquela época pioneira da indústria têxtil, uma vez que essa indústria foi uma das primeiras a utilizar a tecnologia da máquina a vapor em suas produções.

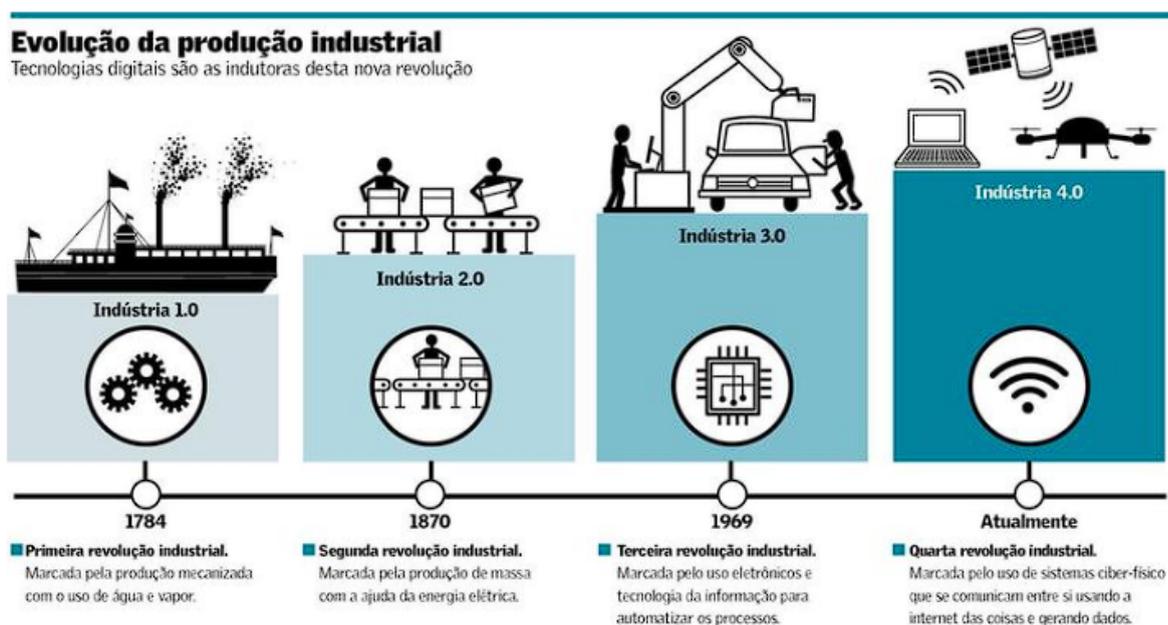
A aplicação de energia elétrica em máquinas industriais, transformação do ferro em aço e os avanços nas tecnologias de comunicação e meios de comunicação foram o marco da segunda revolução (SILVA; GASPARIN, 2011), tornando seus principais criadores pioneiros da tecnologia, como Estados Unidos e Japão. Esse novo método aumenta a produção na indústria, uma vez que a tecnologia permitia a produção em massa, o que barateia custos e aumenta os lucros das empresas que a aplicavam, apesar da necessidade de especialização de trabalho (SAKURAI; ZUCHI, 2018).

Já a terceira revolução ficou marcada pelo desenvolvimento da tecnologia de telecomunicações, robótica, informática, nanotecnologia, transportes e biotecnologia, conhecida também como Revolução Técnico-Científica e Informacional (BOETTCHER, 2015). A utilização de recursos da informática na produção predominou a indústria e otimizou as linhas de produção, uma vez que foram implementadas técnicas de controle de operação através de controladores lógicos, ou seja, diminuiu a interferência humana na produção. Porém, com o desenvolvimento dessas novas técnicas, houve a necessidade de maior

especialização dos trabalhadores, necessidade de um número menor de trabalhadores, o que gerou ampliação de leis trabalhistas (SAKURAI; ZUCHI, 2018).

Por fim, a quarta revolução industrial, ou Indústria 4.0 é definida por um sistema de produção inteligente, onde a tomada de decisão é ajustada de maneira automática, aumentando a eficiência de energia e diminuindo o desperdício. Essa produção baseia-se na aquisição instantânea de dados e tratamento imediato dos mesmos (INOUE et. al, 2019). Atualmente, esse modelo de indústria ainda está sendo implementado, mas já se fala nos impactos que ela causará, como novos modelos de negócios, mercados mais exigentes, a evolução contínua das tecnologias de segurança da informação e maior capacitação de profissionais (SAKURAI; ZUCHI, 2018).

Figura 2 - Linha do tempo das Revoluções Industriais.



Fonte: Sakurai; Zuchi, 2018

### 3.2. Controlador Lógico Programável (CLP)

Após a segunda Revolução Industrial, as máquinas a vapor foram sendo substituídas por equipamentos que utilizam eletricidade para seu funcionamento, e com isso, ocorreu o surgimento de novas indústrias, automação do trabalho e produção em massa. Nesta época, o controle de máquinas e processos era realizado com auxílio de relés eletromecânicos, que são dispositivos que utiliza a corrente elétrica que passa por uma bobina para movimentar um

mecanismo que, ao entrar em contato com um terminal condutor, permitirá a passagem de corrente elétrica, e assim acionar um dispositivo. Sendo assim um relé utiliza lógica combinacional para realizar o acionamento dos equipamentos desejados, o que viabiliza o desenvolvimento de controles de funções mais sofisticadas e complexas (FRANCHI, CAMARGO; 2008). Esse método de controle possui algumas desvantagens, sendo alguns deles a difícil manutenção e o desgaste dos relés devido aos contatos mecânicos. No entanto, na década de 60, a empresa General Motors apresentou um dispositivo para substituir os painéis de relés.

Atualmente, o Controlador Lógico Programável (CLP) é a tecnologia de controle mais utilizada na indústria, uma vez que muitas das desvantagens dos painéis de relé não se aplicam ao CLP, como por exemplo a dificuldade de reprogramação, a grande quantidade de cabos de energia que são utilizados e o desgaste mecânico (PETRUZELLA, 2014). Dentre as vantagens dos CLP's temos:

- Controle em alta velocidade
- Comunicação em rede
- Confiabilidade em testes.

### 3.2.1 Funcionamento

O CLP surgiu como uma alternativa aos controles por relés, sendo inicialmente utilizados para executar apenas as mesmas funções. Porém, com o tempo, seu desenvolvimento permitiu o acréscimo de outras funções além da lógica combinacional, como indicadores de diagnósticos e possibilidade de mudança de lógica sem necessidade de alterações físicas (FRANCHI, CAMARGO; 2008).

Segundo o professor Gladimir P. da Silva (2006), seu princípio de funcionamento atual depende de 3 elementos básicos, que são uma unidade de processamento, entradas e saídas. Ao se iniciar um CLP, primeiro são feitas as checagens dos módulos de entrada e saída ligados a eles, além da verificação de um programa existente já instalado nele. Ao ser determinado que os módulos estão aptos à operação e já existe um programa carregado, o CLP inicia seu ciclo de varredura, onde são feitas as leituras das entradas e saídas e armazenadas na memória, chamadas de imagens. Em seguida é feita uma comparação dessas imagens com o *software* implementado pelo programador e atualizados os estados das saídas.

Esse ciclo é ilustrado na Figura 3, e ele se repete pelo tempo que o CLP se mantém ligado, sendo refeito o processo de inicialização caso ele seja desligado e ligado novamente.

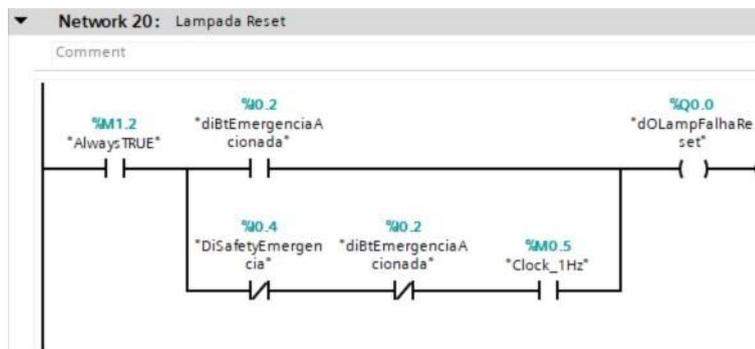
Figura 3 - Ciclo de funcionamento do CLP.



Fonte: Silva, 2006.

Na Figura 4 temos uma implementação de acionamento de uma lâmpada. O sinal “*Always TRUE*” é um sinal próprio do sistema, que é um sinal digital 1 que nunca será alterado. Os sinais “*diBtEmergenciaAccionada*” e “*diSafetyEmergencia*” são sinais de entrada provenientes do botão de emergência e do CLP de segurança, respectivamente, podendo essas entradas serem também provenientes de sensores, botoeiras e outros elementos. O CLP faz a leitura desses sinais e aciona a saída “*dOLampFalhaReset*”, que se trata de uma lâmpada led, de acordo com a combinação dos sinais de entrada. Nessa aplicação, a lâmpada ficará acesa constantemente caso o botão de emergência seja acionado, ou irá piscar caso ela não receba o sinal do botão de emergência e do CLP de segurança, mudando de estado a cada 500 ms. No caso de receber o sinal do botão e da segurança, a lâmpada ficará apagada.

Figura 4 - Código implementado para acionar lâmpada.



Fonte: Autor, 2022.

### 3.2.2 Aplicações

O CLP foi criado pela General Motors a fim de melhorar a produção de veículos da empresa, porém ele é utilizado em inúmeros tipos de aplicações na indústria. Por ser bastante versátil e capaz de atender várias áreas, é necessário que esse equipamento esteja em constante evolução, a fim de atender a demanda de aplicações (FRANCHI, CAMARGO; 2008).

Por ser um dispositivo simples de se trabalhar, facilmente programável e funcionamento cíclico assegurado, o CLP pode ser utilizado em aplicações industriais desde as mais simples até as mais complexas (GOMES, 2003). Alguns desses sistemas seriam uma máquina de usinagem de peças, onde apenas cilindros pneumáticos são controlados, ou até mesmo um sistema de liberação de operação de uma máquina de costura, onde é feita a análise de uma imagem por um sistema de visão e enviado o sinal de aprovação ao CLP, e com isso é feita a liberação da máquina para ser operada.

Em 2004, Marco Antonio B. de Souza, fez um estudo comparativo entre um sistema de controle dedicado a automação predial e a utilização de CLP's para exercer esse controle, e aponta vantagens na utilização deste para essa aplicação, pois os sistemas dedicados, apesar de serem mais simples de programar, não possuem suporte para comunicação entre equipamentos de fabricantes diferentes. Além disso, também é apontado que o programa desenvolvido em CLP não fica limitado à utilização de blocos prontos, podendo ser adicionadas exceções de acordo com a necessidade.

Em uma aplicação que foi implementada em um outro projeto da empresa Emax, o CLP foi utilizado para estabelecer comunicação com um dispositivo desprovido de

comunicação Profinet, sendo necessário utilizar um bloco de função para comunicar através do protocolo TCP/IP. Nessa aplicação, foi necessário um bloco de conexão, um bloco para envio de mensagem do CLP ao equipamento, um bloco para recebimento de mensagem vinda do dispositivo, e um bloco para desconectar as duas partes.

Também vemos dispositivos de controle de eixos elétricos, os chamados drivers, que são conectados ao CLP tanto por rede Profinet quanto por sinal digital nas entradas e saídas. Nessas aplicações, sinais são enviados ao driver e recebidos dele a fim de controlar o eixo elétrico de acordo com a programação implementada, podendo utilizar controles de posição, velocidade e torque do eixo.

### **3.3. Instrumentação**

Segundo o professor Marcelo Wendling (2010), sensores são instrumentos sensíveis a alguma forma de energia, que são utilizados na automação para dar valores às condições naturais do ambiente que queremos monitorar. Eles são os responsáveis por colocar valores às variáveis físicas do ambiente, como temperatura, pressão, vazão, velocidade, entre outros. Para projetos com CLP, as variáveis do ambiente são transformadas em sinais elétricos e enviados ao controlador, onde são analisados em uma lógica combinacional para acionarmos os atuadores.

Apesar de o controlador receber sempre um sinal elétrico, nem todos os sensores são do mesmo tipo. Muitos dos sensores precisam passar por um circuito eletrônico que irá definir o sinal a ser enviado para o controlador. Tendo isso em mente, é possível nomear alguns tipos de sensores que são utilizados nas indústrias, como sensores indutivos, fotoelétricos, a laser, mecânicos, capacitivos, entre outros.

#### **3.3.1 Sensores indutivos**

O sensor indutivo, que está ilustrado na Figura 5, é um sensor capaz de detectar a presença de metais que passam pelo seu campo magnético sem o contato direto. Quando metais que apresentam características magnéticas se aproximam do campo magnético gerado pela bobina do sensor, o mesmo detecta a presença do objeto e assim envia o sinal elétrico ao controlador, com nível lógico 1.

Estes sensores também podem ser utilizados na para determinar distâncias entre um material magnético e o sensor, onde simulam a utilização de sensores indutivos para

guiamento de veículos, estabelecendo uma relação entre a tensão de saída do sensor e a distância do material condutor (DIOGO, BICHO, FURTADO; 2005)

Figura 5 - Sensor Indutivo.



Fonte: Wendling (2010).

### 3.3.1.2 Sensores fotoelétricos

Os sensores fotoelétricos são sensores que utilizam luz para medir as variáveis do ambiente. Muito utilizados nas indústrias e com inúmeras aplicações, os sensores fotoelétricos, como o LDR (Figura 6), não desgastam com o tempo de uso e são muito mais rápidos, pois utilizam a luz para fazer suas medições (WENDLING, 2010), diferente de sensores mecânicos, que necessitam de elementos mecânicos comutando para acionar o sensor.

Dentre os tipos de sensores fotoelétricos, temos:

- Foto-resistor (LDR)
- Fotocélula
- Fototransistor
- Fotodiodo

Os sensores fotoelétricos podem ser utilizados para auxiliar a captação de energia solar, a fim de convertê-la para energia elétrica, sendo o sensor utilizado para captar a luz solar e direcionar as placas fotovoltaicas para que haja maior incidência de luz solar (CAMPOS, et al, 2014).

Figura 6 - Fotorresistor (LDR).



Fonte: Site da Siemens.

### 3.3.3.3 Sensores mecânicos

Sensores mecânicos, como o próprio nome já diz, são compostos apenas por componentes mecânicos, condutores de eletricidade, que emitem um sinal quando é detectada a presença, o movimento ou a posição desejada de algum objeto. Diferente dos outros tipos de sensores, o sensor mecânico possui um desgaste relacionado ao seu acionamento, já que é preciso que o sensor se movimente e entre em contato com outros objetos para que seja feita a detecção do objeto. Um exemplo de sensor mecânico é a chave de fim de curso, ilustrada na figura 6, muito utilizada para referenciamento de equipamentos que utilizam controles de posição, como por exemplo as mesas centralizadoras. Existe também a *Reed-Switch*, que é uma chave mecânica que fecha sob a presença de material magnético, podendo ser considerada também um sensor magnético (WENDLING, 2010). Diferente dos sensores indutivos, onde a presença de campo magnético gera corrente elétrica na bobina, o Reed Switch apenas movimenta a chave até que ela entre em contato com o terminal condutor.

Figura 7 - Chave Fim-de-Curso.



Fonte: Wendling (2010).

### 3.3.3.4 Sensores capacitivos

Sensores capacitivos são aqueles que seguem o princípio do capacitor elétrico, onde sua capacitância é determinada pela distância entre duas placas condutoras, o dielétrico aplicado às placas e o material que se encontra entre as placas (WENDLING, 2010). Com isso, a aproximação do objeto com o sensor, ilustrado na Figura 8, altera a capacitância, e essa alteração é detectada pelo sensor, assim a tensão de saída do mesmo é alterada. Esse sensor é mais utilizado para aplicações com sinal analógico, uma vez que a tensão de saída varia de acordo com a aproximação do objeto.

Esses tipos de sensores são mais utilizados para aplicações onde o objeto a ser lido não é composto por material metálico, e sim materiais não condutores ou semicondutores. Para aplicações onde o sinal é digital, a aproximação do objeto causa alteração na capacitância do sensor e, quando o mesmo atinge um valor determinado, o sensor envia um sinal ao controlador, indicando a presença de um objeto.

Este sensor pode também ser utilizado como sensor analógico, sendo analisada a variação de capacitância do sensor para avaliar a grandeza desejada, como por exemplo a umidade no solo (BORIM, PINTO; 2015).

Figura 8 - Sensor Capacitivo.



Fonte: Wendling (2010).

### 3.4. Comunicação de rede

A comunicação por rede é o que permite ao ser humano obter informações de maneira rápida. Isso se deve ao fato que esse tipo de comunicação permite a troca de informações entre os computadores e outros dispositivos (OLIVEIRA, 2016). A internet é o maior exemplo de comunicação por rede que temos, já que é o meio de comunicação mais utilizado da atualidade, pois permite obter informações ou enviá-las a qualquer hora em qualquer lugar, desde que haja em mãos algum dispositivo capaz de acessá-las. Para os ambientes industriais, é fundamental que haja comunicação de dados rápida e eficiente entre o CLP e os instrumentos a ele conectados. Os dados podem ser transmitidos a ele de duas formas:

- Cabeamento elétrico simples, onde os dados são transmitidos por meio de sinais elétricos analógicos (variação de tensão ou corrente), ou sinais digitais. No caso dessa transmissão, existe a limitação de que os sinais devem ser elétricos, ou então convertidos e ainda sim devem ser interpretados pelo CLP.
- Comunicação por rede, onde apenas um cabo é necessário, e podem ser enviados dados de todas as formas. Para a maioria dos projetos utilizando CLP, são utilizadas comunicação Profibus e comunicação Profinet.

Uma comunicação de dados entre dois dispositivos depende de 4 características fundamentais, que são a entrega de dados ao destino correto, a precisão na entrega, ou seja, a mensagem enviada deve ser a mesma recebida pelo destinatário, a sincronização do tempo de entrega e o chamado *jitter*, que é a variação de tempo na chegada dos pacotes da mensagem (FOROUZAN, 2009).

Segundo Forouzan (2009), existem 3 tipos de fluxos de dados, sendo eles o simplex, onde somente um dispositivo transmite os dados e outro os recebe, o half-duplex, onde ambos

dispositivos transmitem e recebem dados, porém em momentos diferentes, e o full-duplex, onde ambas as partes transmitem e recebem simultaneamente.

Rede é definida como um conjunto de dispositivos interligados, onde cada dispositivo executa uma tarefa, não sendo necessário um único capaz de executar todas elas. Além disso, é necessário que uma rede tenha bom desempenho, confiabilidade e segurança.

Para que haja a comunicação de dados entre dois dispositivos, é necessário que ambas sigam o mesmo conjunto de regras. Os protocolos de comunicação servem para que as mensagens enviadas possam ser compreendidas por quem as recebe, senão seriam apenas um conjunto de bits enviados sem entendimento. Logo, para estabelecer comunicação, ambas as partes devem utilizar o mesmo protocolo, sendo ele definido por elementos como sintaxe, que se refere ao formato dos dados, pela semântica, que é o significado de cada seção dos dados e pelo timing, que determina o momento e a velocidade do envio (FOROUZAN, 2009).

Dentre os principais protocolos utilizados atualmente, temos:

- IP (*Internet Protocol*)
- TCP/IP (*Transmission Control Protocol*)
- HTTP/HTTPS (*Hypertext Transfer Protocol*)
- FTP (*File Transfer Protocol*)
- SFTP (*Simple Transfer Protocol*)

Para CLP's, os dados transmitidos por comunicação de rede podem adotar os protocolos Profinet e Profibus, que são os mais comuns encontrados para comunicação com CLP Siemens, porém podem ser utilizados outros protocolos, como a aplicação citada no capítulo 3 seção 2.2 - Aplicações do CLP, onde foi estabelecida uma comunicação via protocolo TCP/IP.

### **3.4.1. Comunicação Profibus**

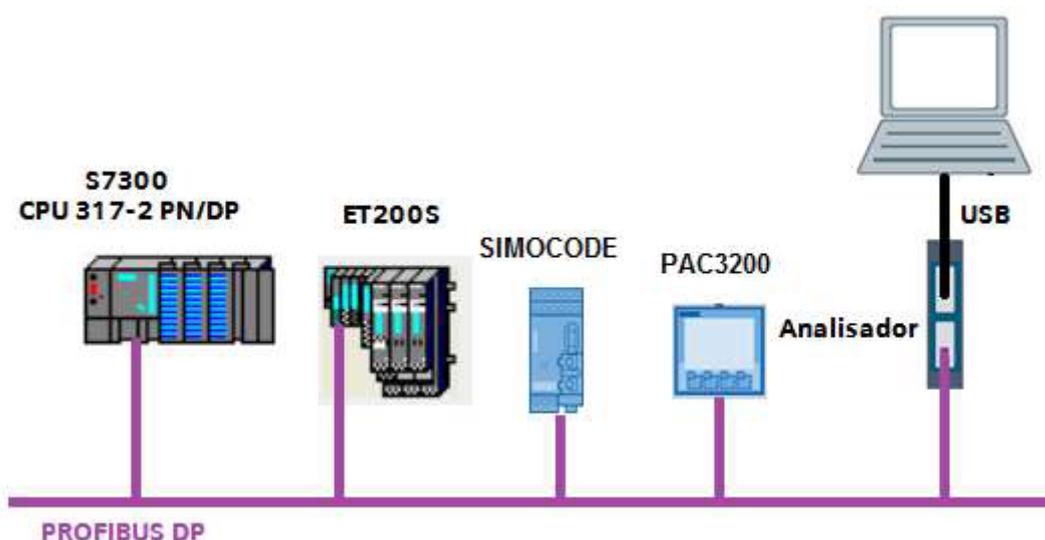
Redes Profibus é um dos tipos de comunicação utilizadas na indústria para a transmissão de dados entre componentes com o mesmo tipo de comunicação, como é ilustrada na Figura 9. O protocolo Profibus foi criado pela Siemens, com capacidade de conectar computadores, sensores, atuadores e CLP's, podendo ser utilizado por todo e qualquer tipo de equipamento que tenha suporte para seu protocolo, independente de fabricante (PERES, 2012). As conexões diretas entre sensores e atuadores com o CLP exigem uma quantidade considerável de cabos, uma vez que cada sensor e atuador exige cabos de alimentação e

dados. Já com a rede Profibus, fica possibilitada o uso de estações remotas para comunicação com o controlador central, assim é possível a redução na quantidade de cabos, sendo os sensores e atuadores conectados a estação remota próxima a eles, e a estação conectada a central pela rede Profibus (TORRES, 2013).

A comunicação Profibus utiliza o meio de comunicação serial, podendo ser RS-485, considerado o meio universal, que utiliza apenas um cabo para alimentação e transmissão de dados, sendo esse um cabo *shield*, e com o valor mais acessível; IEC 61158-2, desenvolvido para atender requisitos de segurança de indústrias químicas e petrolíferas; e Fibra Óptica, que é a utilizada apenas em ocasiões onde a rede pode sofrer grandes interferências eletromagnéticas, ou quando necessita de um comprimento elevado de rede (PERES, 2012). Alguns modelos de CLP's Siemens possuem comunicação serial como um de seus principais meios, porém na maioria dos casos, para que seja possível utilizar esse tipo de comunicação, é necessário um cartão de expansão para comunicação serial RS-485.

Esse tipo de comunicação é utilizada na estrutura mestre e escravo, podendo estar presente em inúmeras aplicações, como a comunicação entre dois semáforos, controlados por CLP S7 313 Siemens e comunicando se via rede Profibus (DE MORAES, DA SILVA, ARRUDA; 2017).

Figura 9 - Topologia de comunicação de rede Profibus.



Fonte: Dias et al (2014).

### 3.4.2. Comunicação Profinet

Redes Profinet são redes de comunicação de dados que operam na comunicação entre o controlador central, geralmente CLP, e dispositivos descentralizados, como drivers, *manifolds*, entre outros instrumentos que tenham suporte para esse tipo de comunicação, como o exemplificado na Figura 10. A troca de dados entre controlador e dispositivos ocorre em tempo real, diferente de dados de parametrização e configuração, que são transferidos por meios sem suporte de transmissão em tempo real (DIAS et al, 2014).

Existem dois tipos de redes Profinet: a Profinet CBA (*Component Based Automation*) possui uma comunicação em tempo real de elementos presentes em uma mesma planta. Já a Profinet IO (*Inputs/Outputs*) realiza a comunicação com os dispositivos de campo. A versão CBA atualmente não é encontrada nos dispositivos mais recentes, porém ainda há suporte para essa versão para os equipamentos que dispõem dela (TURCATO, 2015).

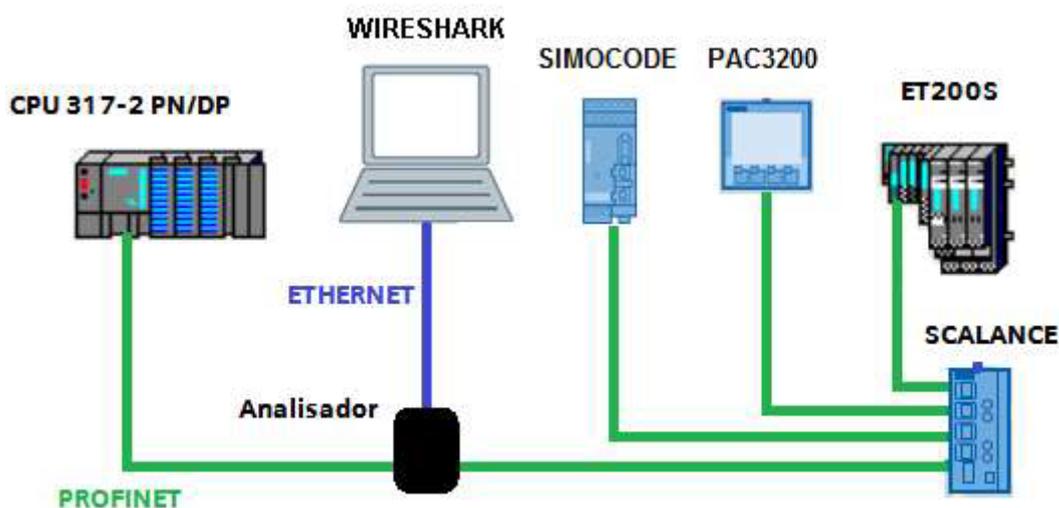
Assim como a rede Profibus, a rede Profinet é uma rede muito utilizada na indústria, podendo substituir a conexão direta entre sensores, atuadores e outros dispositivos, assim diminuindo o número de cabos ligados ao controlador, porém o meio de transmissão usa os mesmas funcionalidade da rede Ethernet, como o TCP/IP. Ela utiliza conexão RJ45 para sua transmissão de dados, podendo utilizar qualquer cabo Ethernet para sua comunicação, já que a estrutura de seu cabeamento é a mesma.

Atualmente a comunicação por rede é a mais utilizada não só na indústria, mas também no dia a dia do ser humano. Isso se deve ao fato dessa comunicação ser mais rápida que as outras, como a Profibus, e oferecer maior segurança contra perda de dados.

Assim como a rede Profibus, a rede Profinet possui inúmeras aplicações, como por exemplo uma máquina raspadora de pneus, onde é utilizada a rede Profinet para comunicar os inversores de frequência com o CLP, para controle dos motores, a IHM e a ET200, que são utilizadas para coletar os sinais de sensores e acionar atuadores (ARTACHO, 2015).

Alguns projetos utilizam comunicação de rede para enviar dados de equipamentos para o controlador, a fim de que ele realize seu procedimento da maneira mais eficiente possível. Este é o caso de eixos elétricos, que usam comunicação de rede Profinet para enviar dados como velocidade, posição, torque, bits de estado, como “drive em erro” ou “eixo em movimento”.

Figura 10 - Topologia de comunicação de rede Profinet.



Fonte: Dias et. al (2014).

### 3.5. Interface Homem-Máquina

Interfaces Homem-Máquina são a maneira utilizada para a interação humana com a máquina, sendo a máquina definida como qualquer equipamento mecânico ou eletrônico que transmite ou transforma energia para atuar ou assistir em tarefas humanas. Conforme o computador foi se desenvolvendo, novos dispositivos de IHM foram sendo criados, sendo atualmente distanciados das interfaces com *mouse* e teclado. O objetivo principal dessas interfaces é a melhoria da produção e qualidade, assim como redução de riscos de lesão, fadiga, erro e desconforto (CANNAN, HU; 2011).

A IHM é criada pelo programador para que o operador responsável pela máquina possa compreender os dados da máquina de maneira simples e também controlá-la de forma fácil e rápida. Portanto cabe ao programador definir o design da interface, tendo em mente que o operador precisa do acesso àquela informação, devendo então serem mostradas de forma organizada (PUPO, 2002).

Em uma pesquisa feita por James Cannan e Huosheng Hu, 2011, foram identificadas 5 categorias principais das interfaces homem-máquina, sendo elas a acústica, onde a interação é feita através do som, como comandos de voz; a óptica, onde a interação é dada através da luz, utilizando câmeras, lasers e leds; a biônica, que é a combinação de biologia, robótica e computação, utilizadas para monitorar qualquer aspecto biológico do corpo; o movimento,

que se baseia em tecnologias de detecção de movimento; e a tátil, que requer o toque do ser humano para realizar comandos, como teclados e *touchscreens*.

Uma vez que é possível utilizar o computador para tais aplicações, uma IHM pode ser um supervisor desenvolvido por meio de um *software* de implementação da interface, como *WinCC* e *Indusoft*, ou então um hardware específico para essa aplicação, como as telas IHM Siemens (Figura 11), sendo elas *TouchScreens* programáveis, que utilizam comunicação Profinet e/ou Profibus para comunicar com o CLP. As telas HMI Siemens são desenvolvidas através da mesma plataforma de programação dos CLP's, no caso o TIA Portal. Nela, podem ser utilizados botões digitais para realizar controles manuais da máquina, abertura de telas de parâmetros, configurações de sistemas, log de alarmes entre outras funções.

Figura 11 - Telas HMI Basic Panel Siemens.



Fonte: Site da Siemens.

As telas acima são hardwares da empresa Siemens, voltadas para a implementação de interfaces gráficas para o controle e monitoramento de variáveis da máquina. A Emax utiliza essas telas para gerar comandos e monitorar dados essenciais para a operação de seus projetos, podendo ser feitas várias operações.

O exemplo ilustrado na Figura 12 mostra uma tela implementada pelo autor. Nessa tela foram destacados alguns pontos onde são demonstradas informações importantes a respeito da aplicação e do controle da máquina:

a - Representa o cabeçalho, onde é demonstrado modos de controle da máquina, nome da tela acionada, data e hora e o logotipo da empresa.

b - Representa a barra de alarmes, onde é mostrado o último alarme acionado. Essa barra fica invisível enquanto não houverem alarmes acionados.

c - Representa uma chave manual/automática, que é um comando utilizado para alterar o modo de operação da máquina.

d - Indica o acionamento dos sensores de laser que captam a presença de *holder* nos berços.

e - Barra indicativa do passo a passo da máquina, sendo descrito todo o processo automático dela.

f - Representa os botões de troca de telas.

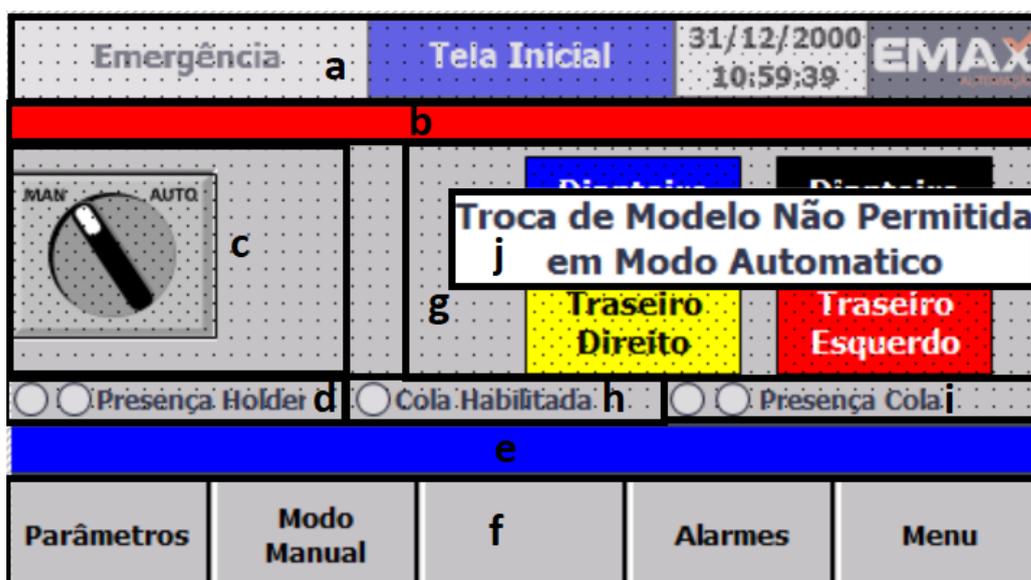
g - Representa os botões de troca de modelos, sendo cada um deles representado por uma cor distinta para evitar confusão.

h - Representa um parâmetro da aplicação, que pode ou não estar acionado, o que pode afetar o ciclo da máquina.

i - Tem a mesma função de (d), porém é monitorada outra variável da aplicação.

j - É uma mensagem que fica invisível enquanto ela não é necessária. Informa ao operador que a troca de vidro deve ser feita no modo de operação manual, para evitar erros de aplicação.

Figura 12 - Tela Principal de uma IHM.



Fonte: Autor (2022).

## 4. MATERIAIS E MÉTODOS

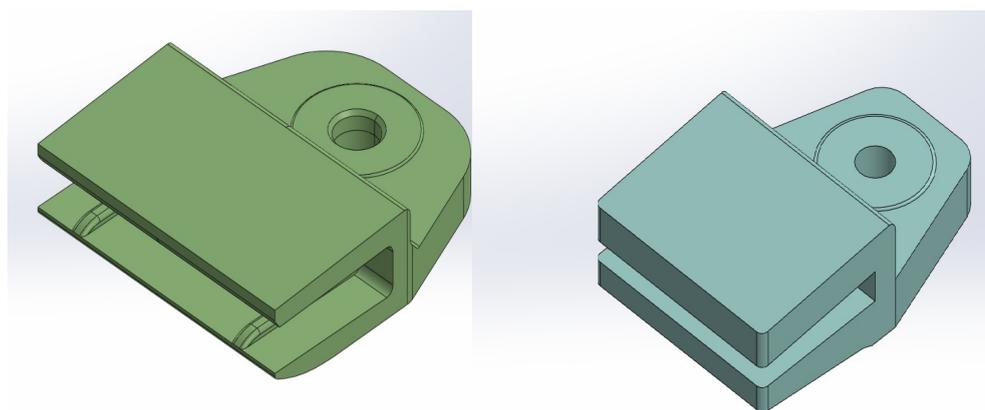
Neste capítulo são descritos os materiais utilizados para o desenvolvimento do projeto e a metodologia utilizada para chegar no melhor resultado possível. Para facilitar, a divisão será feita em materiais, configuração de *hardware*, teste de bloco de válvulas (*manifold*), configurações de segurança, comandos manuais, sequenciamento automático, registros de produção e parametrização.

### 4.1. Materiais

A atividade principal executada nesse projeto foi a programação do controlador, observando os sinais digitais de sensores e saídas para o acionamento de atuadores pneumáticos, para realizar o cravamento das peças *holders*, ilustradas na Figura 13 e para isso, foi necessário:

- CLP Siemens S7-1200
- CLP de Segurança Sirius 3SK2
- IHM Siemens *Basic Panel* KTP 4"
- Bloco de Válvulas Manifold SMC EX260
- Notebook Dell para desenvolvimento de *software*
- Sensores indutivos, a laser e capacitivo

Figura 13 - *Holder* de vidro dianteiro.



Fonte: Autor (2022).

### 4.2. Configuração de Hardware

Os *softwares* dos CLPs Siemens podem ser implementados no TIA Portal (*Totally Integrated Automation Portal*) ou no Step7, nesse projeto foi utilizado o TIA Portal. Nele, são

implementados lógicas combinacionais utilizando a linguagem *Ladder*, que é a linguagem mais utilizada na programação de CLPs, e realizadas também configurações de hardware, que é a inclusão de todos os elementos físicos ligados ao CLP pela rede Profinet. Uma vez que os hardwares são inseridos na configuração, eles devem ser nomeados e atribuídos a um endereço de IP, sem que haja repetição de endereço, para evitar conflitos. No caso desse projeto, os elementos da rede Profinet são o CLP, a IHM e o bloco de válvulas, como é ilustrado na Figura 14.

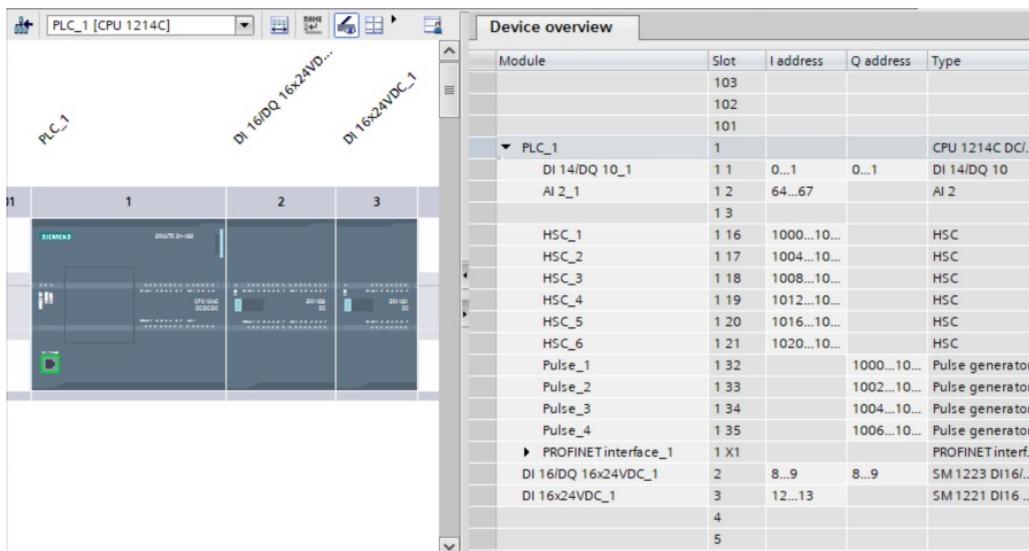
Figura 14 - Configuração de Hardware.



Fonte: Autor (2022).

Uma vez incluídos os dispositivos da rede, é preciso realizar a configuração dos endereços de entradas e saídas, pois isso determina a leitura e o acionamento correto dos elementos. A configuração de endereços é demonstrada na Figura 15, onde os elementos de entrada são definidos pela letra I, e os elementos de saída são definidos pela letra Q. O número indicado se refere ao byte ao qual o sensor ou atuador está alocado, sendo especificado pelo número do bit, representado por um número após o ponto.

Figura 15 - Configuração dos endereços do CLP.



Module	Slot	I address	Q address	Type
	103			
	102			
	101			
PLC_1	1			CPU 1214C DC/...
DI 14/DQ 10_1	1 1	0...1	0...1	DI 14/DQ 10
AI 2_1	1 2	64...67		AI 2
	1 3			
HSC_1	1 16	1000...10...		HSC
HSC_2	1 17	1004...10...		HSC
HSC_3	1 18	1008...10...		HSC
HSC_4	1 19	1012...10...		HSC
HSC_5	1 20	1016...10...		HSC
HSC_6	1 21	1020...10...		HSC
Pulse_1	1 32		1000...10...	Pulse generator...
Pulse_2	1 33		1002...10...	Pulse generator...
Pulse_3	1 34		1004...10...	Pulse generator...
Pulse_4	1 35		1006...10...	Pulse generator...
PROFINET interface_1	1 X1			PROFINET interf...
DI 16/DQ 16x24VDC_1	2	8...9	8...9	SM 1223 DI16/...
DI 16x24VDC_1	3	12...13		SM 1221 DI16...
	4			
	5			

Fonte: Autor.

Nesse projeto, por haver muitos sinais de entrada a serem interpretados, foi necessária a adição de um cartão de expansão de entradas, também especificado na configuração de hardware. O cartão permite a leitura de sinais digitais além dos limites da CPU, podendo ser adicionados mais de um cartão de acordo com a demanda do projeto. Para essa aplicação, foram necessários 1 cartão de expansão com 16 entradas e 16 saídas e 1 cartão de expansão com 16 entradas. A Figura 16 mostra alguns dos sinais que são lidos pela CPU e pelos cartões de expansão.

Figura 16 - Sinais de entrada.

Entradas								
	Name	a type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	diBTstart	DI	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Botão Start
2	diBTResetFalha	DI	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Botão Reset
3	diBTEmergenciaAcionada	DI	%I0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Botão Emergencia
4	diSafetyPorta	DI	%I0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Entrada Vinda do safety para porta ok
5	diSafetyEmergencia	DI	%I0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Entrada vinda do safety para emergencia ok
6	diSafetyCortina	DI	%I0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Entrada vinda do safety para cortina ok
7	diPressostato	DI	%I0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Sensor Pressostato
8	diPresencaVidro	DI	%I0.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Presença Vidro
9	diCilAplicadorColaAv	DI	%I1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Aplicador Cola Avançado
10	diCilAplicadorColaRec	DI	%I1.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Aplicador Cola Recuado
11	diCilAvancoHolderAv	DI	%I1.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Avanço Holder Avançado
12	diCilAvancoHolderRec	DI	%I1.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Avanço Holder Recuado
13	diCilAjusteFrontalEsqAv	DI	%I8.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Empurra Centralizador Esquerdo Avan...
14	diCilAjusteFrontalEsqRec	DI	%I8.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Empurra Centralizador Esquerdo Recu...
15	diCilAjusteLateralEsqAv	DI	%I8.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Trava Centralizador Esquerdo Avançad...
16	diCilAjusteLateralEsqRec	DI	%I8.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Trava Centralizador Esquerdo Recuado
17	diPresencaHolderLE	DI	%I8.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Presença Holder Esquerdo
18	diCilCentralizadorEsqRec	DI	%I8.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Centralizador Esquerdo Recuado
19	diCilTravaFrontalEsqAv	DI	%I8.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Trava Frontal Esquerda Avançado
20	diCilTravaFrontalEsqRec	DI	%I8.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Trava Frontal Esquerda Recuado
21	diCilEmpurradorEsqAv	DI	%I9.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cilindro Centralizador Frontal Esquerdo Avanc...

Fonte: Autor (2022).

Para evitar conflitos, os bytes referentes ao bloco de válvulas é representado por um valor distante dos bytes da CPU.

### 4.3. Teste do bloco de válvulas (Manifold)

Uma vez definidas e carregadas, no CLP, as configurações de hardware, é iniciada a fase de teste do bloco de válvulas. Por ser um elemento da rede Profinet, o bloco não pode ser acionado pelas saídas comuns do CLP, sendo necessário o acionamento dos bits referentes aos bytes configurados anteriormente. Por ser apenas uma etapa de testes, a implementação do código é apenas provisória, a fim de ver se o bloco realmente está sendo acionado conforme o especificado. Os endereços das saídas do bloco são mostrados na Figura 17.

As válvulas ligadas ao bloco podem ser duplo acionamento, ou seja, um sinal para avanço e um sinal para recuo, ou acionamento simples, sendo da escolha do programador se é avanço ou recuo.

Figura 17 - Sinais de saída do bloco de válvulas.

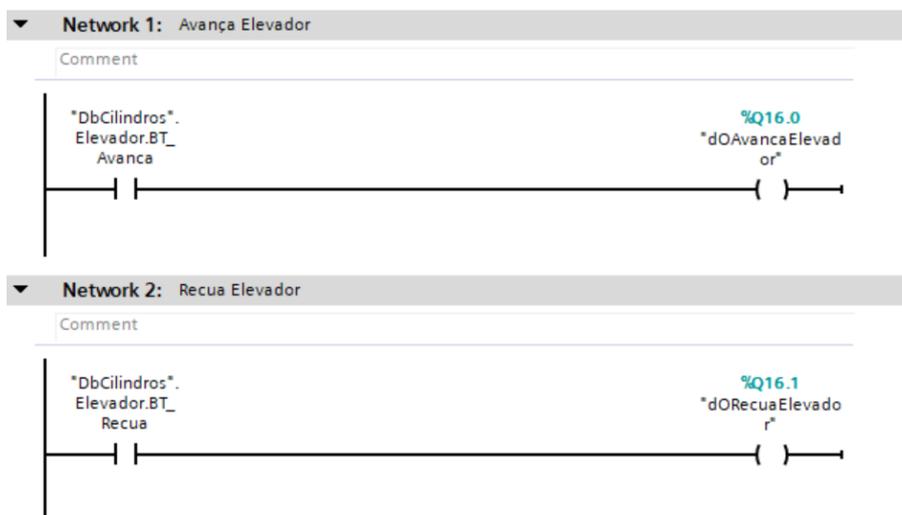
Saídas							
	Name	Data type	Address ▲	Retain	Acces...	Writa...	Visibl...
9	dOAvancaElevador	Bool	%Q16.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	dORecuaElevador	Bool	%Q16.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	dOAvancaVedador	Bool	%Q16.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	dORecuaVedador	Bool	%Q16.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	dOAvancaVedador2	Bool	%Q16.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	dORecuaVedador2	Bool	%Q16.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	dOAvancaTravaHolder	Bool	%Q16.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
16	dORecuaTravaHolder	Bool	%Q16.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
17	dOAvancaHolder	Bool	%Q17.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
18	dORecuaHolder	Bool	%Q17.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
19	dOAvancaTravaFrontalEsq	Bool	%Q17.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
20	dOAvancaTravaFrontalDir	Bool	%Q17.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
21	dOAjusteFrontalEsq	Bool	%Q17.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
22	dOAjusteFrontalDir	Bool	%Q17.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
23	dOAvancaCentralizadorEsq	Bool	%Q17.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
24	dOAvancaCentralizadorDir	Bool	%Q17.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
25	dOAvancaEmpurradorEsq	Bool	%Q18.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
26	dOAvancaEmpurradorDir	Bool	%Q18.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
27	dOAvancaAjusteLateralEsq	Bool	%Q18.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
28	dOAvancaAjusteLateralDir	Bool	%Q18.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
29	dOSobeAplicadorCola	Bool	%Q18.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Fonte: Autor (2022).

Por se tratar de uma fase de testes, toda a programação implementada é provisória, uma vez que o objetivo é apenas verificar se o endereçamento está correto e o acionamento está condizente com o especificado. Para que seja feito o teste, foi implementado o código da Figura 18, onde foi utilizado um bloco de dados (DB), que é um bloco onde podem ser especificadas variáveis auxiliares, para facilitar na programação. Uma vez alterada as

variáveis auxiliares para nível lógico 1, é feito o acionamento das válvulas, observando o comportamento do bloco, que acende um LED indicando o acionamento e liberando a saída de ar.

Figura 18 - Programa teste para acionamento das válvulas.



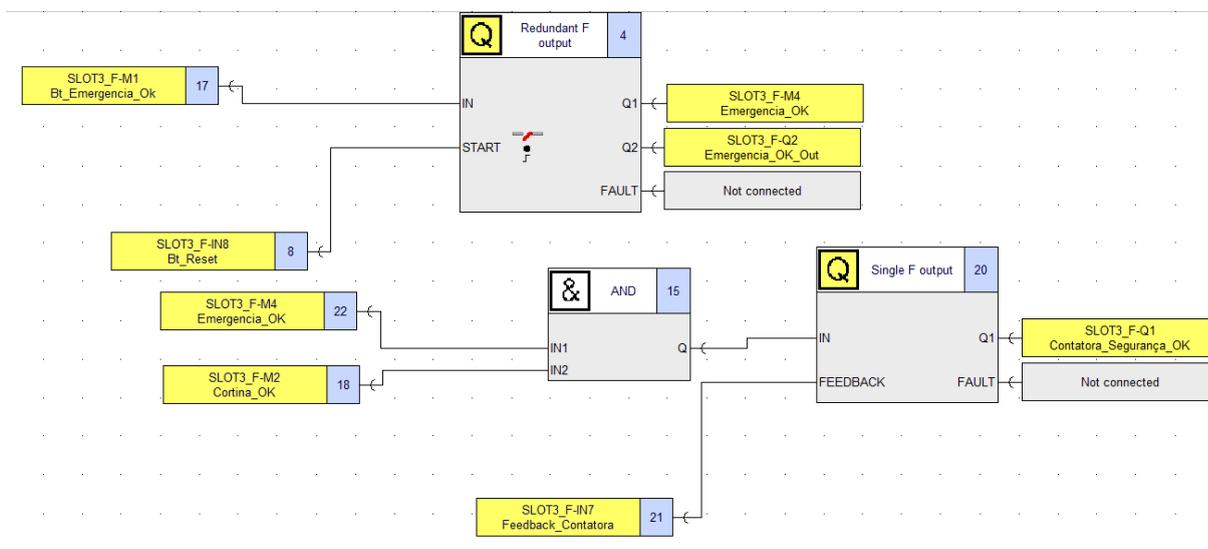
Fonte: Autor (2022).

#### 4.4. Configurações de Segurança

O projeto utiliza atuadores pneumáticos para executar sua função, e isso pode representar perigo ao operador da máquina, portanto são utilizados elementos de segurança para evitar possíveis acidentes. Dentre eles, temos um botão de segurança e uma cortina de luz, ambos elementos tem a função de cortar toda a alimentação de ar comprimido da máquina, a fim de evitar o avanço ou recuo dos atuadores, diminuindo os riscos de acidentes. O botão de emergência corta a alimentação de ar independente de qualquer ação ou etapa do processo, já a cortina de luz possui etapas onde a alimentação não é cortada, pois é necessário que algum atuador esteja avançado.

Para realizar as configurações de segurança, foi utilizado o *software* Safety ES, da Siemens, onde são definidas as entradas e saídas do CLP de segurança. Cada elemento de segurança já possui um arquivo ilustrativo nesse *software*, onde nele são definidas apenas quais são as entradas referentes a ele, e as saídas que devem acionar em suas condições seguras. A Figura 19 mostra a configuração da saída referente a liberação da alimentação de ar comprimido do sistema.

Figura 19 - Saídas da segurança.



Fonte: Autor (2022).

Nessa configuração, é necessário um sinal de cortina liberada, do botão de emergência e das contadoras de segurança para que haja a alimentação de ar. O botão, ao ser pressionado, precisa ser liberado, e depois pressionado um botão reset para retornar ao funcionamento normal dele, já a cortina de luz não precisa de nenhum tipo de reset.

Alguns dos sinais de emergência são utilizados também no CLP, como o sinal de cortina não invadida e botão de emergência não pressionado. Esses sinais são utilizados em alarmes, no sequenciamento automático e também no controle de modos de operação, que será descrito nos capítulos seguintes.

#### 4.5. Comandos manuais

Nesta etapa do projeto, os elementos mecânicos, pneumáticos e painel elétrico estão montados na estrutura da máquina, o que permite que o cliente possa estar fazendo o teste da máquina e passar o *feedback* com pendências e possíveis melhorias. Sendo assim, resta apenas o acionamento das válvulas para executar o processo da máquina. Por se tratar apenas de uma fase de testes e checagem de progresso, o controle manual foi suficiente para o controle.

Primeiro foi criado no TIA Portal uma *Data Type*, que é atribuída às variáveis de um *Data Block* para criar as condições e comandos dos cilindros. Nesse *Data Type* foram inseridas variáveis booleanas, como mostradas na Figura 20. O intuito dessa *Data Type* é

padronizar os comandos e informações dos cilindros do projeto, assim, ao criar o *Data Block* dos cilindros e nomeá-los, basta declarar cada cilindro sendo um *Data Type* criado, assim todos os elementos do *Data Block* possuirão a mesma atribuição de dados.

Figura 20 - Data Type “UDT\_Cilindros”.



UDT_Cilindros		
	Name	Data type
1	BT_Avanca	Bool
2	BT_Recua	Bool
3	Avancado	Bool
4	Recuado	Bool
5	Alarme_Ava	Bool
6	Alarme_Rec	Bool

Fonte: Autor (2022).

Criada a “UDT\_Cilindros”, facilita a criação do *Data Block* dos cilindros. Cada cilindro do sistema precisa de uma nomeação de acordo com sua função no sistema, a fim de facilitar a identificação de problemas. Alguns dos cilindros nomeados na Figura 21 foram retirados posteriormente, pois não houve a necessidade deles na decorrência do projeto. A cada um deles é atribuído o *Data Type* “UDT\_Cilindros”, e assim cada uma das variáveis do *Data Block* possui as mesmas variáveis booleanas.

Figura 21 - Data Block DbCilindros.

DbCilindros								
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	Elevador	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	BT_Avanca	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	BT_Recua	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	Avancado	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	Recuado	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	Alarme_Ava	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	Alarme_Rec	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	AvancaVedador	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	Vedador	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	TravaHolder	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	AvancaHolder	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
13	TravaFrontalEsq	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
14	TravaFrontalDir	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
15	AjusteFrontalEsq	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
16	AjusteFrontalDir	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
17	CentralizadorEsq	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
18	CentralizadorDir	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
19	EmpurradorEsq	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
20	EmpurradorDir	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
21	AjusteLateralEsq	*UDT_Cilindros*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Fonte: Autor (2022).

A partir da criação da “DbCilindros”, já temos a base para controle manual, que irá utilizar botões na IHM, atribuídos às variáveis “BT\_Avanca” e “BT\_Recua” para realizar o acionamento das válvulas pneumáticas. Porém, antes disso é preciso definir os modos de operação da máquina, para isso foi criada a “DbModos”, demonstrada na Figura 20. Nela são utilizadas apenas variáveis booleanas, que farão o controle dos modos de operação. No código implementado, apenas um desses modos é acionado por vez, sendo atribuídos um valor inteiro para cada modo e armazenado na variável “Cod\_Modo”, que então é indicada no cabeçalho da IHM. Ainda na Figura 22 temos também a listagem de todos os *Data Blocks* criados para auxiliar no desenvolvimento do projeto.

Figura 22 - Lista de *Data Block* e DbModos

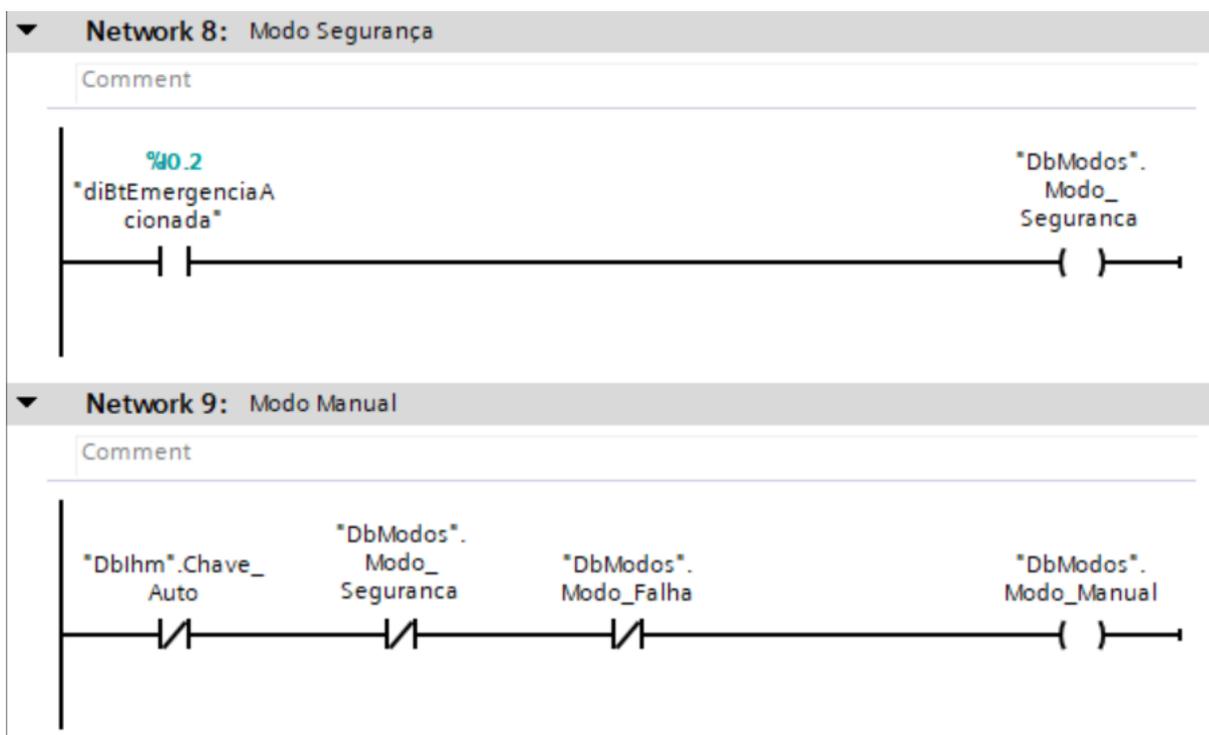
The screenshot shows the Siemens SIMATIC Manager interface. On the left, the 'Project tree' displays a list of 'DB's' (Data Blocks) under the 'Devices' section. The 'DbModos [DB110]' is highlighted. On the right, the 'DbModos (snapshot created: 9/17/2020 4:05:31)' table is displayed, showing a list of modes and their data types.

	Name	Data type
1	Static	
2	Modo_Seguranca	Bool
3	Modo_Manual	Bool
4	Modo_AutoReset	Bool
5	Modo_Auto	Bool
6	Modo_Parada	Bool
7	Modo_Falha	Bool
8	Cod_Modo	Int
9	DryRun	Bool

Fonte: Autor (2022).

Com os *Data Blocks*, entradas e saídas definidos, foi implementado o controle manual do projeto. Primeiro foi desenvolvido uma função para controlar os modos de operação da máquina, definida por “100 - Controle de Modos”. Nela são implementadas as *networks* que fazem o acionamento dos bits correspondentes aos modos de operação, tendo cada um deles a condição própria. Para o controle inicial, foram definidos apenas o modo de segurança e o modo manual, que são as *networks* ilustradas na Figura 23. O modo segurança impede qualquer ação da máquina, uma vez que o que define esse modo é o sinal proveniente do CLP de segurança, responsável pela alimentação de ar do sistema. No modo manual, é permitido o controle manual do operador por meio de botões na IHM, desde que as condições de movimento do cilindro sejam atendidas. Para que ele seja acionado, a máquina não pode estar em modo segurança, ou em modo falha, e a chave manual/automática ilustrada na tela principal da IHM, deve estar virada para a posição manual. Nessa função não é implementado nenhum tipo de comando de controle, apenas o meio que a máquina será operada.

Figura 23 - Modo de Segurança e Modo Manual.



Fonte: Autor (2022).

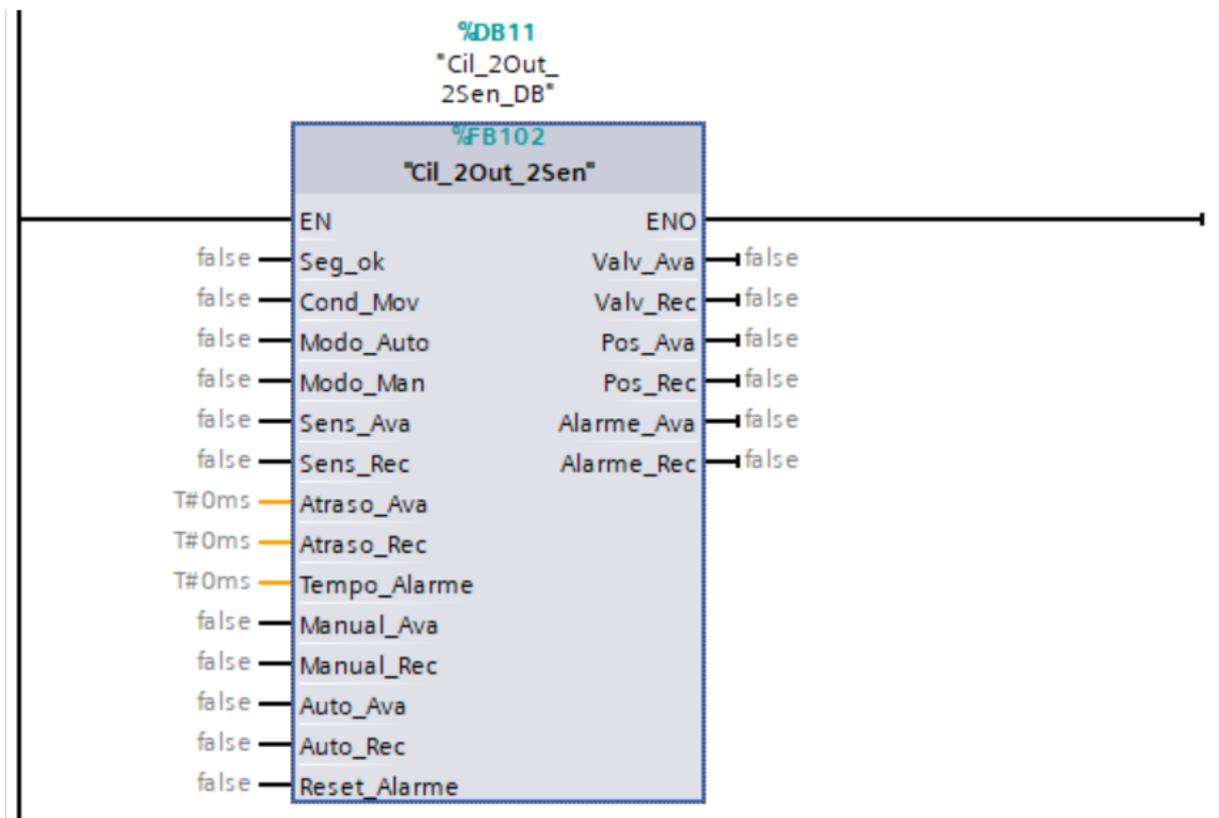
Implementado o modo de operação manual, podem ser implementados os comandos manuais para acionar as saídas das válvulas pneumáticas. Para isso, foi utilizado um *Function Block*, que armazena os valores permanentemente em *Data Blocks*, mesmo após sua execução. Essa *Function Block* foi desenvolvida pelo engenheiro chefe da empresa, e serve como bloco de função padrão para todas as aplicações da empresa em que são utilizados atuadores pneumáticos, onde também é necessário saber se o atuador tem condição de avançado ou recuado. A Figura 24 ilustra o bloco de função citado.

Nesse bloco é preciso de sinais de entrada para segurança, condições de movimento, modo manual ou automático, comandos para executar movimento, delays entre outras entradas. Já nas saídas, foram atribuídas as saídas do sistema, status de avançado e recuado, e alarmes.

Quando o modo manual é acionado, apenas as entradas "Manual\_Ava" e "Manual\_Rec" serão consideradas para executar movimento, portanto independente das entradas "Auto\_Ava" e "Auto\_Rec", o cilindro só movimentará de acordo com o acionamento dos botões manuais da IHM.

Para o acionamento dos status de avançado e recuado, o bloco utiliza os sinais dos sensores descritos por “Sens\_Ava” e “Sens\_Rec”, junto com os atrasos, que servem como um filtro para o sinal dos sensores. Por fim, os alarmes são acionados se os comandos de avanço ou recuo são enviados e os sinais de entrada referentes aos sensores não são acionados no período de tempo definido em “Tempo\_Alarme”. Sendo assim, ao passar o período de tempo, os alarmes são acionados, e assim a máquina passa a operar em modo falha.

Figura 24 - Bloco de controle de válvulas pneumáticas.



Fonte: Autor (2022).

Cada cilindro da máquina possui um bloco de controle associado a eles, assim como uma variável na “DbCilindros” e sensores responsáveis por dizer qual a condição em que ele está. Assim, esse conjunto de dados executa os comandos do atuador.

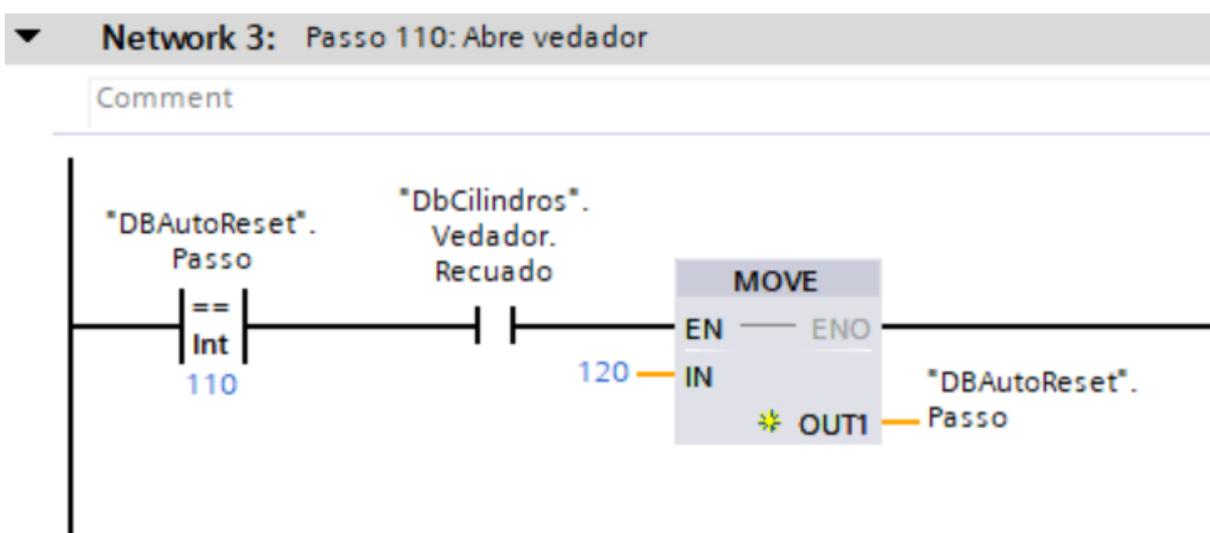
#### 4.6. Sequenciamento Automático

A etapa de comandos manuais tinha o objetivo de demonstrar a aplicação da máquina ao cliente, visando o feedback do mesmo para que fosse feita a implementação dos comandos em modo automático. Após a visita, foi definido a sequência na qual os cilindros devem

avançar e recuar, para que a aplicação fosse feita com qualidade e nas especificações exigidas pelo cliente. A fim de melhor executar essa sequência, primeiro é feito o retorno dos cilindros a seu estado inicial, que é nomeado AutoReset. Em seguida, é acionado o modo Automático da máquina.

Nas sequências de AutoReset e Automático, foram aplicadas técnicas de programação conhecidas como diagrama de fluxo sequencial, onde a sequência é definida em um passo a passo. Cada passo tem uma função associada a ele e a leitura de uma condição, que dará continuidade a sequência. A Figura 25 é um exemplo de como é feita essa aplicação. O diagrama de fluxo sequencial nesse projeto é implementado em linguagem *Ladder*, o que facilita o entendimento dos programadores da empresa, caso seja solicitada alguma alteração.

Figura 25 - Network passo 110 do AutoReset.



Fonte: Autor (2022).

#### 4.6.1. AutoReset

O AutoReset, como foi citado anteriormente, tem o objetivo de posicionar os atuadores na configuração inicial para cada modelo de vidro que será produzido na máquina. São ao todo 5 passos para retornar os cilindros a posição inicial por modelo, como é ilustrado na Figura 26, sendo cada movimento feito de acordo com a prioridade e o risco de quebra de vidros.

Figura 26 - Sequência AutoReset.

▶	<b>Network 1:</b> Reset
▶	<b>Network 2:</b> Start
▶	<b>Network 3:</b> Passo 110: Abre vedador
▶	<b>Network 4:</b> Passo 120: Recua Avanço vedador e avanço holder
▶	<b>Network 5:</b> Passo 130: Recua Empurrador, Centralizador e Trava Frontal
▶	<b>Network 6:</b> Passo 140: Posição inicial de acordo com modelo
▶	<b>Network 7:</b> Passo 200: Auto Reset Ready

Fonte: Autor (2022).

Nesse caso, os vedadores e cravadores de *holders* são os mais prováveis de causar danos tanto à máquina quanto ao vidro, portanto eles são os primeiros a retornar, sendo os vedadores abertos antes para facilitar o retorno. Em seguida, ocorre o retorno das travas frontais e empurradores, responsáveis por segurar os vidros e impedir seu movimento durante a aplicação. Por fim, temos os ajustes frontais e laterais, que se movimentam de acordo com o modelo de vidro que está sendo produzido, como indicado na Tabela 1.

Tabela 1 - Configuração dos ajustes frontais e laterais.

	Ajuste Frontal Direito	Ajuste Frontal Esquerdo	Ajuste Lateral Direito	Ajuste Lateral Esquerdo
Dianteiro Direito	Recuado	Recuado	Recuado	Avançado
Dianteiro Esquerdo	Recuado	Recuado	Avançado	Recuado
Traseiro Direito	Recuado	Avançado	Recuado	Avançado
Traseiro Esquerdo	Avançado	Recuado	Avançado	Recuado

Fonte: Autor (2022).

#### 4.6.2. Automático

O modo automático é o responsável por executar todo o processo de cravamento dos *holders* nos vidros, com a mínima intervenção humana possível. No programa implementado, a atividade humana necessária neste ciclo é apenas o posicionamento dos *holders* nos berços e o posicionamento do vidro nos apoios.

Para garantir a qualidade e o controle do processo, a máquina conta com alguns *Poka Yokes*, que são instrumentos para assegurar que os elementos essenciais de produção estejam corretos, como por exemplo o sensor a laser de cola, que capta se a quantidade mínima de cola foi aplicada aos *holders*, além dos sensores a laser que fazem a leitura da presença de *holders* nos berços.

A sequência inicia-se após o fim do AutoReset. Primeiro é checada a posição inicial de acordo com o modelo de vidro que está sendo produzido e também a presença de vidro nos apoios, através de um sensor capacitivo. Em seguida ocorre um avanço dos berços para facilitar o posicionamento dos *holders*, e o movimento de retorno deles ocorre após a liberação da barreira de luz. Ao liberá-la é feita a verificação da presença dos *holders*, retornando ao passo de avanço dos berços caso algum dos sensores não capte a presença do *holder*. Com a presença confirmada, fecham-se os vedadores e ocorre o levantamento das pistolas de cola, que são posicionadas à frente dos *holders* para liberar cola, sendo ela liberada por um tempo determinado e em seguida retornadas as pistolas.

Com isso, inicia-se o posicionamento do vidro nos apoios, que é identificado pelo sensor capacitivo. Com isso, ao liberar a barreira de luz, tem início o processo de centralização e travamento do vidro. Nessa etapa, cada modelo tem sua combinação de cilindros que será atuado para garantir a centralização e o travamento dele. A Tabela 2 mostra os cilindros e a ordem em que serão atuados para cada modelo de vidro.

Tabela 2 - Ordem de avanço dos elementos de centralização por modelo.

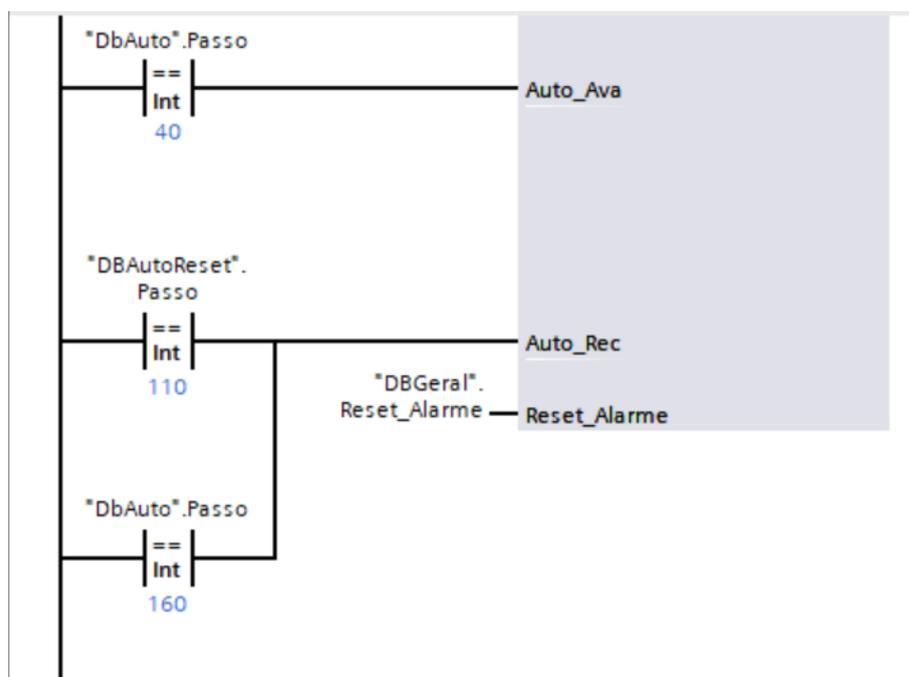
	Trava Frontal Direita	Trava Frontal Esquerda	Empurrador Direito	Empurrador Esquerdo	Centralizado r Direito	Centralizado r Esquerdo
Dianteiro Direito	1º	-----	-----	-----	-----	2º
Dianteiro Esquerdo	-----	1º	-----	-----	2º	-----
Traseiro Direito	-----	-----	1º	-----	-----	2º
Traseiro Esquerdo	-----	-----	-----	1º	2º	-----

Fonte: Autor (2022).

Com o vidro centralizado e travado, os vedadores e berços avançam com uma pequena diferença de tempo, com os vedadores avançando primeiro para garantir uma aplicação mais limpa. Ambos ficam avançados por um período de tempo para garantir que a cola possa segurar os *holders* e então é feita a verificação da quantidade de cola que fica exposta fora do *holder*. Em seguida o vedador, ainda fechado, faz o retorno e abre após um tempo determinado, para evitar sujeiras, retornando então os berços e liberando o vidro para ser retirado. O ciclo se encerra quando a cortina de luz é liberada, e então é feito o registro de contagem de peças e tempo de ciclo de aplicação.

A execução desse ciclo é feita utilizando a técnica de diagrama de fluxo sequencial, onde a cada passo do ciclo um atuador é acionado e/ou é feita a leitura de seu posicionamento para dar continuidade ao ciclo. Ao usar um bloco de comparação para comparar a variável inteira “DbAuto.Passo” com um número, ela passa um sinal binário para o bloco demonstrado na Figura 27, podendo ser enviado a entrada “Auto\_Ava” ou “Auto\_Rec”, dependendo do movimento que deverá ser feito, como é mostrado na Figura 25.

Figura 27 - Comando de avanço e recuo automático.



Fonte: Autor (2022).

Ao fim dessa etapa, pode ser realizado o segundo teste após ajustes e retrabalhos requisitados pelo cliente. O sequenciamento automático também é alterado após o teste conforme vontade do cliente e então é dada sequência às atividades previstas para esse projeto. No quesito programação, ainda é preciso definir os registros de produção e parametrização do processo.

#### 4.7. Parametrização e Alarmes

Como foi citado anteriormente, em algumas etapas do ciclo, é preciso fazer leituras de presença ou tempos devem ser aguardados para podermos dar sequência. Nesse caso, é possível que essas informações possam ser alteradas ou ignoradas.

Para os sensores que detectam a cola ou então a presença de *holders*, existe o chamado *By Pass*, que funciona como um caminho alternativo à leitura desses elementos. Além deles, ainda temos o *By Pass* da aplicação de cola, onde os passos de aplicação de cola serão ignorados e passará diretamente ao passo de depósito de vidro, o *By Pass* da cortina, o que permite a operação da máquina sem o corte de alimentação de ar comprimido pela cortina e o *By Pass* de Avanço de Aplicação, que é um recurso não utilizado pelo cliente, em que a

aplicação de cola é feita com os berços avançados. A Figura 28 mostra a tela de parâmetros *By Pass* que foi desenvolvida para esse projeto.

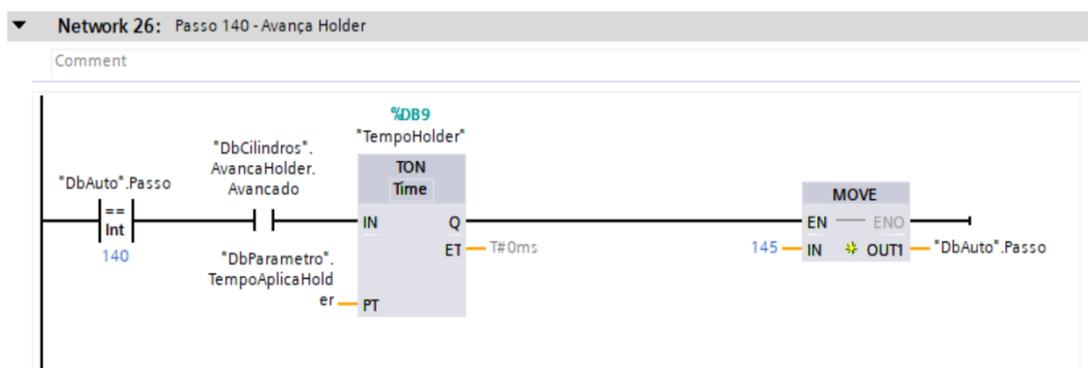
Figura 28 - Tela By Pass Sensores.



Fonte: Autor (2022).

Como citado, em algumas partes do projeto foi preciso utilizar temporizadores, como os ilustrados na Figura 29, para que a aplicação pudesse ser feita de maneira efetiva e garantindo a qualidade.

Figura 29 - Temporizador de aplicação de *holder*.



Fonte: Autor (2022).

Esses temporizadores necessitam que o programador diga o tempo que a entrada do bloco deverá estar acionada para que a saída possa ser acionada também. No caso desse projeto, alguns desses tempos podem ser alterados pelo cliente, sem que haja necessidade do programador alterar a programação já feita.

Para permitir essa alteração, uma tela na IHM foi criada, com campos input/output vinculados as tags dos parâmetros, como por exemplo a “DbParametro.TempoAplicaHolder”, que permite que o cliente possa alterar o tempo de acordo com sua necessidade. Apesar de ser possível que o operador acesse a tela da Figura 30, os parâmetros listados na tela não podem ser alterados por qualquer pessoa. Neles foram aplicadas autorizações no nível de engenharia, a fim de impedir as indevidas alterações dos parâmetros da máquina, sendo as autorizações válidas também para os *By Pass*.

Figura 30 - Tela Parâmetros.

Parâmetro	Valor
Tempo Aplicação Cola:	0,000 s
Tempo Espera Após Cola:	0,000 s
Delay Vedador Retorno/Abrir:	0,000 s
Tempo Aplicação Holder:	0,000 s
Aplica Cola:	Não

Fonte: Autor (2022).

Por fim é feito o registro dos alarmes da máquina. Os alarmes de avanço e recuo dos cilindros foram definidos durante o desenvolvimento do controle manual, portanto restavam apenas os alarmes referentes à segurança, falta de ar, ausência de *holders* nos berços e cola insuficiente após a aplicação. Cada um desses alarmes é uma variável booleana acionada quando as condições citadas são satisfeitas. Existem duas maneiras de identificar o alarme, sendo elas a barra vermelha que é descrita na Figura 12 pela letra b, ou então pela tela de alarmes, acessada por meio do botão Alarmes.

#### 4.8. Registros de produção

Para finalizar a programação, restam apenas os registros de informação da produção. A pedido do cliente, apenas duas informações necessitam ser registradas na IHM para o controle de produção, que eram a contagem de peças e tempo de ciclo da aplicação.

Para atender a demanda do cliente, foi implementado um código capaz de registrar os 10 últimos tempos de ciclo executados pela máquina, e uma contagem de peças que registra a contagem total de hora em hora, sendo contabilizados também o total por turno e o total geral. A Figura 31 demonstra ambas as telas de produção, sendo a contagem de peças apenas do modelo frontal esquerdo.

Figura 31 - Tela Parâmetros.

Cortina Acionada		Conta Peças F.Esquerda		19-02-2012 17:24:25		EMAX		Cortina Acionada		Tempo Ciclo		19-02-2012 17:25:53		EMAX		
TURNO 1			TURNO 2			TURNO 3					Atual		0,0			
08:00-09:00	0	16:00-17:00	0	00:00-01:00	0			1º	0,0	6º	0,0					
09:00-10:00	0	17:00-18:00	0	01:00-02:00	0			2º	0,0	7º	0,0					
10:00-11:00	0	18:00-19:00	0	02:00-03:00	0			3º	0,0	8º	0,0					
11:00-12:00	0	19:00-20:00	0	03:00-04:00	0			4º	0,0	9º	0,0					
12:00-13:00	8	20:00-21:00	0	04:00-05:00	0	ZERAR		5º	0,0	10º	0,0					
13:00-14:00	0	21:00-22:00	0	05:00-06:00	0											
14:00-15:00	0	22:00-23:00	0	06:00-07:00	0											
15:00-16:00	0	23:00-00:00	0	07:00-08:00	0											
TOTAL	8	TOTAL	0	TOTAL	0	Principal		Principal						Menu		

Fonte: Autor (2022).

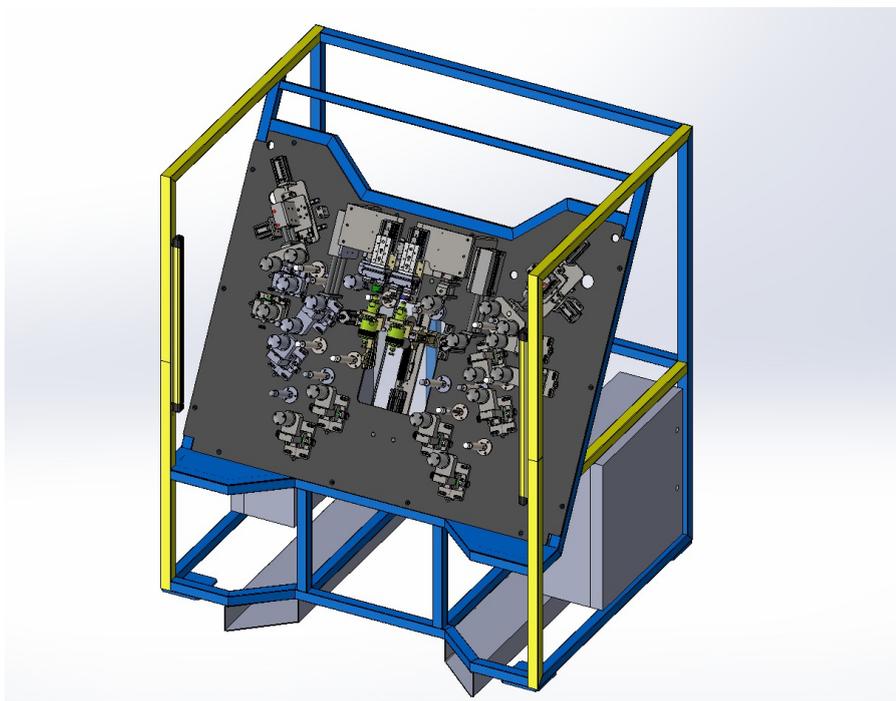
Para criar a função de contagem de peças, foi utilizado um bloco de função onde o horário é contabilizado pelo próprio CLP e armazenado em um *Data Block* com a variável do tipo “DTL”, que faz a divisão do horário em ano, mês, dia, hora, minuto, segundo, milissegundo e nanosegundo. Para a devida aplicação, foram criadas duas variáveis do tipo “DTL”, onde é descrito a data e hora registrada atualmente pelo CLP e uma data e hora para atualizar caso estejam incorretos, além de uma variável booleana para executar essa atualização. Para os registros por hora, são realizadas as leituras da hora atual e comparados com os possíveis intervalos de tempo, e então registrados a contagem ocorrida naquele intervalo de tempo.

Já o tempo de ciclo foi implementado usando um array de variáveis de tempo, com armazenamento para 11 tempos (0-10). Uma variável booleana “Start” dá início a contagem de tempo, sendo interrompida ao fim do ciclo por uma variável booleana “Stop”. Ao acioná-lo, cada elemento do array é direcionado para o endereço acima, ou seja, o tempo registrado em array[9] passa a ser registrado em array[10], para que ele receba o registro de array[8]. Com isso, mantemos os 10 registros requisitados e damos espaço para verificação do novo registro.

## 5. RESULTADOS

Finalizadas todas as pendências de projeto, a máquina entra em operação na Pilkington, unidade de Caçapava - SP. As primeiras produções foram feitas para análise de qualidade, com o acompanhamento da Emax para segurança do cliente, onde foram obtidos alguns dados de produção. A Figura 32 é uma imagem retirada do SolidWorks da máquina totalmente montada.

Figura 32 - Máquina de Aplicar *Holder*s.



Fonte: Autor (2022).

O objetivo era uma produção de aproximadamente 127 peças por hora com a máquina operando em modo automático, o que é o suficiente para atender a demanda do cliente e ainda para formação de estoque. A Tabela 3 é uma amostra da produção de peças executada pelo primeiro turno, onde temos seu horário de produção, divididos de hora em hora, e a quantidade de peças do modelo, especificado na primeira linha da tabela, produzida na hora especificada na primeira coluna.

Tabela 3 - Tabela de Produção

	Dianteiro Direito	Dianteiro Esquerdo	Traseiro Direito	Traseiro Esquerdo	Total por Hora
8:00 - 9:00	112	0	0	0	112
9:00 - 10:00	68	58	0	0	126
10:00 - 11:00	0	122	3	0	125
11:00 - 12:00	0	0	124	0	124
12:00 - 13:00	0	0	0	0	0
13:00 - 14:00	0	0	53	70	123
14:00 - 15:00	0	0	0	110	110
15:00 - 16:00	0	0	0	0	0
Total Produzido	180	180	180	180	

Fonte: Autor (2022).

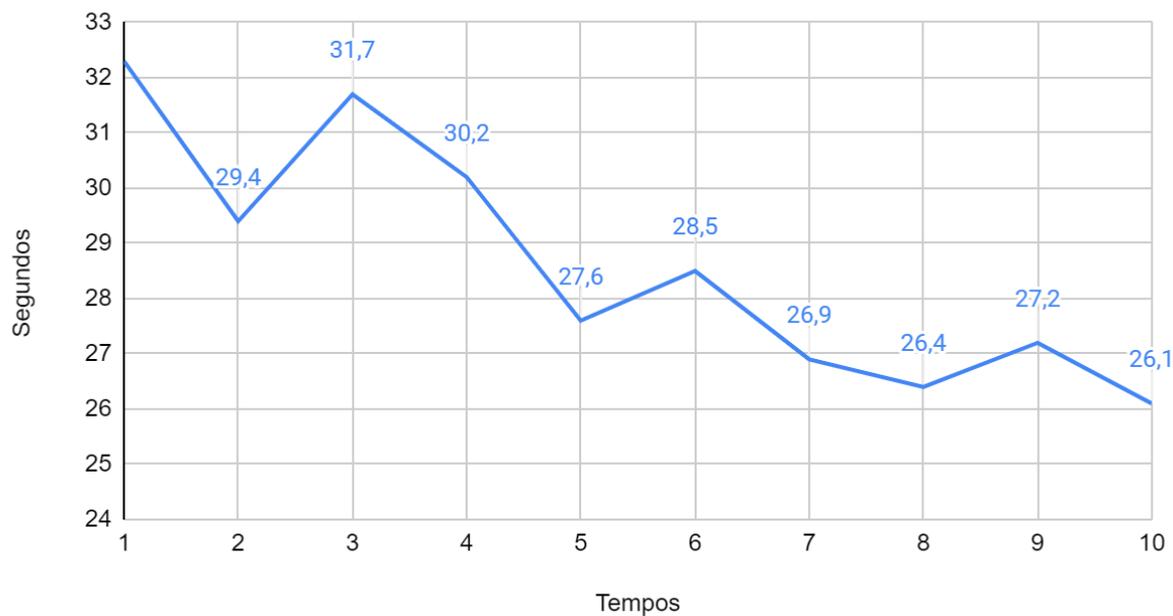
O objetivo era a produção de 3 *racks* de peças, cada uma comportando 60 peças. Durante a primeira hora foram feitos ajustes com o operador em relação ao método de produção, onde foi analisada a maneira mais rápida de retirar o vidro, produzir as peças e posicioná-lo no *rack* para retirada. Uma vez definido qual a melhor maneira de executar a tarefa, a produção pode alcançar a quantidade próxima ao desejado, o que foi satisfatório aos clientes, já que foram considerados os atrasos de adaptação ao ciclo de produção do operador, reposição de *holders*, trocas de *racks*, troca de *setups*, entre outros fatores. Atualmente a máquina está sendo operada com produção média de 1010 peças por dia.

Com relação ao tempo de ciclo, foram reduzidos todos os intervalos de tempo presentes na aplicação que não fazem parte dos parâmetros citados no capítulo 4.7 o que permitiu a redução do tempo de ciclo da máquina a fim de obter a produção desejada.

O Gráfico 1 trata-se de uma amostragem dos tempos de ciclo obtidos durante uma das produções da empresa. O eixo Y do gráfico mostra uma escala de tempo em segundos, sendo os pontos de intersecção com o eixo X o tempo de ciclo para os 10 ciclos amostrados, sendo o primeiro ciclo o mais longo, com 32,3 segundos, e encerrando as amostras no décimo ciclo, com 26,1 segundos.

Gráfico 1 - Amostras de tempo de ciclo

## Tempos de Ciclo



Fonte: Autor (2022).

O tempo de ciclo também se mostrou satisfatório para o cliente, considerando os possíveis contratempos citados. Uma vez que o operador se adapta ao ciclo de produção, vemos que o tempo tende a diminuir, chegando a uma média de tempo de 28,63 segundos por vidro produzido.

## 6. CONCLUSÃO

O objetivo do projeto apresentado era o desenvolvimento de um equipamento capaz de aplicar *holders* em vidros automotivos, dentro do tempo de ciclo estabelecido pelo cliente e com a qualidade desejada. Tais objetivos foram atingidos nesse projeto.

Esse projeto facilitou a produção em larga escala dos 4 modelos de vidros automotivos especificados pelo cliente, dando a ele também controle sobre alguns fatores importantes da aplicação, como quantidade de aplicação de cola, verificações de sensores, tempo de aplicação de *holder*, tempo de abertura de vedadores, entre outros.

Com o projeto já em funcionamento na fábrica, a necessidade de instrumentos de aplicação manual passou a ser obsoleta, uma vez que é preciso apenas de um operador treinado e capacitado para operar a máquina para obter uma produção de peças satisfatória. Para esse projeto, não há nenhuma dificuldade de treinamento para operador, uma vez definido o melhor método de produção, a adaptação a produção se torna simples.

O projeto também mostrou eficiência em relação a segurança, sendo essa acionada instantaneamente após invasão de barreira durante ciclo e ao pressionar botão start, cortando a alimentação de ar e garantindo a segurança do operador.

O sistema de segurança também realiza o monitoramento dos dois Contatores de Segurança, sendo gerenciado pelo CLP de segurança Sirius da Siemen, colocando o sistema de segurança na categoria 4.

Além disso, um relatório de segurança gerado no momento em que a configuração é carregada para o CLP, e anexado a documentação do projeto, a fim de evitar configurações indevidas. Isso garante a confiabilidade do sistema e impede que haja alterações de pessoas não autorizadas, sendo os responsáveis do projeto os únicos capazes de realizar alterações na segurança.

Também é possível analisar o ritmo de produção de peças analisando os dados salvos de produção de peças e tempo de ciclo de aplicação, visando encontrar maneiras mais eficientes de se produzir as peças desejadas.

Algumas dificuldades foram encontradas durante o desenvolvimento desse projeto, como o estabelecimento da comunicação entre o *manifold* e o CLP, uma vez que era o primeiro contato com essa tecnologia, então foram necessárias muitas pesquisas do seu funcionamento para então conseguir estabelecer tal conexão via rede Profinet.

Outro empecilho foi a definição da melhor sequência de atuação dos cilindros, uma vez que o cliente se via em dúvida da melhor opção de atuação, já que a ordem de avanço e

recuo dos cilindros interferia diretamente na centralização do vidro na máquina, o que interferia na posição dos *holders*, podendo estes estarem fora dos limites de tolerância de posição, exigido pelas montadoras.

Além disso, foram relatados problemas com a vedação da máquina, uma vez que ao aplicar o *holder* na máquina, a vedação se mostrava ineficiente na contenção da cola, sendo projetadas novas peças de vedação e alterados o ciclo de aplicação, como citado acima. Também foi alterado a pressão aplicada ao cilindro, através da válvula reguladora de pressão, para que haja maior força dos vedadores no momento em que são fechados, a fim de pressionar as laterais dos *holders* e impedir que a cola depositada neles venha a vazar e sujar os vidros.

No fim, o cliente se mostrou satisfeito com o resultado apresentado no projeto, atendendo todas as suas exigências, tendo o suporte técnico necessário para possíveis falhas e melhorias que possam ser pensadas futuramente para esse projeto.

A máquina atualmente está em operação na planta de Caçapava. A máquina foi desenvolvida para auxiliar o cliente a produzir as peças para abastecer a linha de produção de veículos da Honda, modelo não especificado. A produção atual da máquina é mais que o suficiente para atender a demanda da montadora, e ainda manter peças em estoque.

## 7. REFERÊNCIAS

ARTACHO, B. M. Automação em raspadora de pneus utilizando controle numérico computadorizado com controlador lógico programável integrado. 2015. 62 f. Trabalho de conclusão de curso (Bacharelado - Engenharia Elétrica) - Universidade Estadual Paulista, Faculdade de Engenharia de Guaratinguetá, 2015.

BORIM, A.C.A; PINTO, C.A.R. Medição de Umidade no Solo Através de Sensores Capacitivos, 2015. Disponível em: <<https://revista.pgsskroton.com/index.php/rcext/article/view/2395>> Acesso em 18/09/2022.

CAMPOS, G. P; JUNIOR, J. M. L. F; CARDOSO, D. L; MELO, C. L. S. Sistema Automatizado para Aumento da Eficiência da Captação Solar das Placas Fotovoltaicas na Geração de Energia Solar, 2014. Disponível em: <<http://www.abenge.org.br/cobenge/legado/arquivos/5/Artigos/129198.pdf>>. Acesso em 21/09/2022

CANNAN, J; HU, H; Human Machine Interaction (HMI) - A Survey, 2011. Disponível em: <<https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.186.1644&rep=rep1&type=pdf>>. Acesso em 12/09/2022.

DE MORAES, D; DA SILVA, M.T.S; ARRUDA, R.A. Desenvolvimento de um Sistema de Comunicação em Rede Profibus Entre CLP's Siemens S7 300 e Implementação de Sistema Supervisório. 2017. 109 f. Trabalho de Conclusão de Curso (Bacharelado Engenharia de Controle e Automação e Engenharia Elétrica) – Departamento Acadêmico de Eletrotécnica, Universidade Tecnológica Federal do Paraná. Curitiba. 2017.

DE SOUSA, M.A.B. Estudo Comparativo entre a Aplicação de Sistemas Dedicados e a Utilização de Controladores Lógicos Programáveis na Automação de Sistemas Prediais. 2004. 143 f. Tese (Mestrado em Engenharia de Construção Civil). Escola Politécnica, Universidade de São Paulo, São Paulo. 2004.

DIAS, A.L; SESTITO, G.S; MOSSIN, E.A; FERNANDES, R; BRANDÃO, D. Análise Comparativa de Desempenho de Redes Profibus DP e Profinet, 2014. Disponível em:

<<http://www.swge.inf.br/cba2014/anais/PDF/1569910651.pdf>>

Acesso em 13/06/2022.

DIOGO, J; BICHO, R; FURTADO, E. Sensor indutivo para guiamento de veículo, 2005. Disponível em:

<[https://repositorio.ipl.pt/bitstream/10400.21/8794/1/Sensor\\_RBicho\\_ADEETC.pdf](https://repositorio.ipl.pt/bitstream/10400.21/8794/1/Sensor_RBicho_ADEETC.pdf)>.

Acesso em 14/09/2022.

FOROUZAN, B.A - Comunicação de Dados e Redes de Computadores. AMGH Editora, 2009.

FRANCHI, C.M; CAMARGO, V.L.A - Controladores Lógicos Programáveis: Sistemas Discretos. 3ª Edição. Editora Saraiva Educação SA, 2008.

GOMES, A.F.N - WebGraf Aplicação Web para execução de GRAFCETs e redes de Petri em Controladores Lógicos Programáveis. 2003. 161 f. Tese (Mestrado em Automação, Instrumentação e Controle). Faculdade de Engenharia, Universidade do Porto, Porto. 2003.

INOUE, J.S.P, et al. Indústria 4.0 - Impactos da tecnologia da informação na nova indústria, 2019. Disponível em:

<<https://revistas.brazcubas.br/index.php/pesquisa/article/view/651/709>>

Acesso em 28/06/2022.

PERES, R.P. Redes de Comunicação Profibus, 2012. Disponível em:

<<http://lyceumonline.usf.edu.br/salavirtual/documentos/2352.pdf>>

Acesso em 13/06/2022.

PETRUZELLA, F. D. Controladores Lógicos Programáveis. 4ª edição. AMGH Editora, 2014

PUPO, M.S - Interface homem-máquina para supervisão de um CLP em controle de processos através da WWW. 2002. 113 f. Tese (Mestrado em Engenharia Elétrica). Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos. 2002.

ROSARIO, J. M. Automação Industrial. Editora Baraúna. 2009

SACOMANO, J.B. et al. Indústria 4.0. Editora Blucher, 2018.

SAKURAI, R; ZUCHI, J.D. As Revoluções Industriais até a Indústria 4.0. Disponível em:

<<https://revista.fatectq.edu.br/interfacetecnologica/article/download/386/335>>

Acesso em 28/06/2022.

SILVA, G.P - PLC Controladores Lógicos Programáveis, 2006. Disponível em:

<[https://www.academia.edu/download/54978742/Apostila\\_de\\_PLC\\_Gladimir.pdf](https://www.academia.edu/download/54978742/Apostila_de_PLC_Gladimir.pdf)>

Acesso em 06/09/2022.

TORRES, R.V. Simulador de redes profibus. 2013. 121 f. Tese (Mestrado em Sistemas Dinâmicos). Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos. 2013.

TURCATO, A.C - Desenvolvimento de uma metodologia baseada em redes neurais artificiais para a identificação de anomalias em redes de comunicação *Profinet*. 2015. 218 f. Tese (Mestrado em Sistemas Dinâmicos). Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos. 2015.

WENDLING, M. Sensores, 2010. 2ª Versão. Disponível em:

<<https://www.feg.unesp.br/Home/PaginasPessoais/ProfMarceloWendling/4---sensores-v2.0.pdf>>

Acesso em 12/03/2022.