

## Capítulo II – Implementação de Redundância com CLPs para Sistemas de Controle Industrial

Danilo Faveri Massi <sup>4</sup>

Wesley Candido Silva <sup>5</sup>

Leonardo Yuji Ishizaki <sup>6</sup>

### RESUMO

Este artigo apresenta um projeto de redundância com Controladores Lógicos Programáveis (CLPs) para aplicações industriais, visando aumentar a confiabilidade e disponibilidade do processo. O estudo desenvolve um sistema redundante aplicado a um protótipo didático de uma esteira separadora de materiais, utilizando referências bibliográficas da área de automação e controle. A implementação prática, com equipamentos industriais em uma aplicação diferenciada, demonstrou que a redundância é uma solução eficaz e confiável para processos industriais.

**Palavras-chave:** Redundância; Controladores Lógicos Programáveis; Confiabilidade Industrial

### Implementation of Redundancy with PLCs for Industrial Control Systems

### ABSTRACT

This article presents a redundancy project using Programmable Logic Controllers (PLCs) for industrial applications, aiming to increase process reliability and availability. The study develops a redundant system applied to a didactic prototype of a material sorting conveyor, utilizing bibliographic references from the field of automation and control. The practical implementation, using industrial equipment in a different application, demonstrated that redundancy is an effective and reliable solution for industrial processes.

**Keywords:** Redundancy; Programmable Logic Controllers; Industrial Reliability.

---

<sup>4</sup> Engenheiro Eletricista. UniSenaiPR - Campus Londrina.

<sup>5</sup> Especialista. UniSenaiPR - Campus Londrina, [wesley.candido@sistemafiep.org.br](mailto:wesley.candido@sistemafiep.org.br)

<sup>6</sup> Mestre em Engenharia Elétrica. [yuji096@gmail.com](mailto:yuji096@gmail.com)

## 1 INTRODUÇÃO

A indústria vem evoluindo junto à sociedade tanto que estamos no processo da quarta revolução industrial também conhecido como indústria 4.0 a primeira vez que se tem registro sobre o termo indústria 4.0 segundo Kagermann et al., (2013) O termo Indústria 4.0 foi apresentado publicamente na Feira de Hannover na Alemanha em 2011. Depois, a palavra mencionada foi discutida até agora, e ainda haverá novidades.

A indústria 4.0 tem alguns pilares como, por exemplo, análise de dados, inteligência artificial, *machine learning*, *internet of Things*(IoT), *cloud computing*. Todas elas juntas têm o objetivo de agilizar o nível e a qualidade da produção e buscando, gerenciamento e confiabilidade no processo.

Diante de tantas tecnologias, a forma de analisar dados também precisou ser adaptada de forma que segundo TOTVS, (2021) A análise de abundantes dados coletados de sensores no chão de fábrica garante visibilidade em tempo real dos ativos de manufatura. Assim, a empresa tem total controle sobre o que acontece e como os processos são conduzidos na fábrica. O resultado é uma maior produtividade na indústria. Sendo assim a utilização destes dados melhora o processo de produção, aumenta a quantidade produzida e também com mais informações e sensores os itens produzidos têm maior qualidade garantido que erros e defeitos, desperdícios sejam diminuídos exponencialmente.

Em um cenário de indústria altamente tecnologia também se faz necessário gerenciamento, tanto como gerenciar processo de qualidade, produção e manutenção. Segundo TOTVS (2021) para minimizar desperdício de insumos, ocorrência de erros e principalmente de tempo são necessárias linhas de produção mais elaboradas e comunicação ativa e gerenciamento a todo momento.

No mesmo cenário onde se é necessário gerenciamento de tudo tem que se haver confiabilidade no processo e na segurança, onde tudo que está conectado à rede tem vulnerabilidade em qualquer peça pode se haver ataques ao sistema. Para a empresa se antecipar a problemas tem que haver confiabilidade no sistema e nas máquinas operacionais, sendo assim o primeiro passo e tornar a máquina confiável ao nível operacional para assim após conectá-la a uma rede e obter os dados.

O uso de redes industriais e protocolos de comunicação são pilares da indústria 4.0 trazendo comunicação e garantia de confiabilidade. O presente trabalho visa tratar redundância dos controladores lógicos programáveis como prioridade para termos um hardware de confiabilidade no presente processo que será simulado através de um protótipo, assim podendo visualizar que não haverá interrupções no processo, tornando o controle confiável e gerando um banco de dados para visualizar a produção, máquina ligada e desligada, e se houver erro em um dos controladores.

## 1. FUNDAMENTAÇÃO TEÓRICA

### 1.1. Automação e a Pirâmide da Automação

De acordo com SHERIDAN (1992), automação é o controle automático de operações ou sistemas realizados por dispositivo elétrico, eletrônico ou mecânico em que há substituição dos órgãos humano de visão, decisão ou esforço na operação.

Para MORAES (2010) A automação industrial exige a realização de muitas funções. A Figura 1 representa a chamada Pirâmide de Automação, com os diferentes níveis de automação encontrados em uma planta industrial.

Figura 1 – Pirâmide da Automação

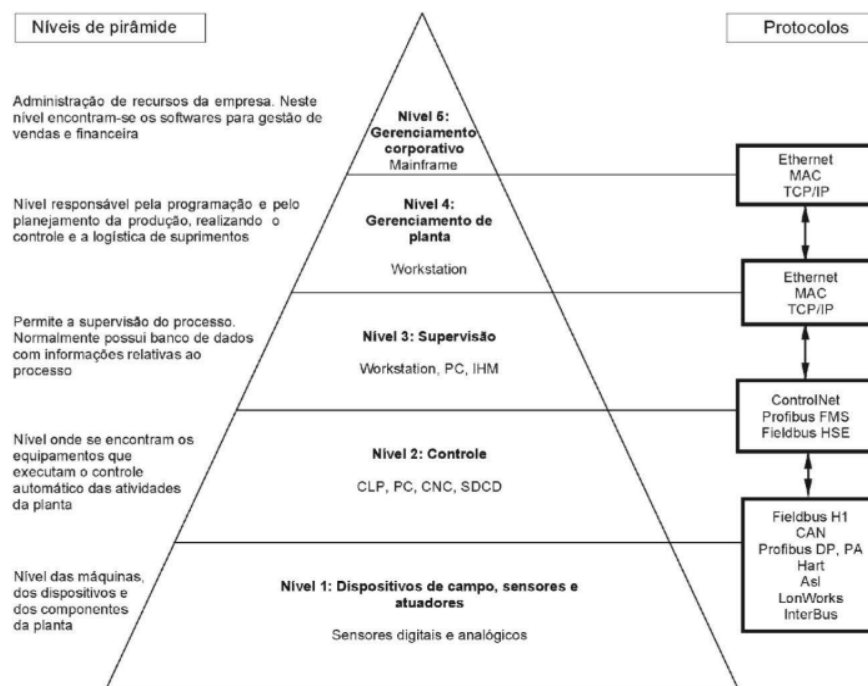


FIGURA 1.8 Pirâmide de automação.

Fonte: MORAES (2010)

Na base da pirâmide está frequentemente envolvido o Controlador Programável, atuando via inversores, conversores ou sistemas de partida suave sobre máquinas e motores e outros processos produtivos. No topo da pirâmide, a característica marcante é a informatização ligada ao setor corporativo da empresa.

A pirâmide tem seus níveis que representam dos dispositivos de campo até o gerenciamento da empresa segundo SANTOS (2021), o primeiro nível é composto por dispositivos de campo, ou seja, atuadores, sensores, transmissores e alguns outros componentes. Já o segundo nível tem equipamentos que realizam controle dos atuadores da planta, por exemplo, CLPs, SDCDs e relés. No terceiro nível são destinados à supervisão do processo executados por uma célula trabalhando na planta como um IHM, PC, workstation, dando suporte a um banco de dados com informações do processo.

Com o terceiro nível em execução, o quarto nível tem responsabilidade pela parte de programar e planejar a produção, auxilia no controle do processo quanto na logística de suprimentos, sendo aplicado o termo de gerenciamento da planta. Sendo executados todos os níveis a para completar com o quinto nível e responsável pela administração dos recursos empresariais para contemplar estas atividades são utilizados alguns softwares para gestão, venda, gestão financeira, BI (*business intelligence*) tudo para auxiliar na tomada de decisões para não afetar negativamente a empresa e o resto da pirâmide.

Para ajuda na tomada de decisões e planejamento da produção existem 2 sistemas, o ERP e o MÊS que segundo NARCISO (2021), o ERP e o MES são sistemas complementares porém diferentes em suas funções. O ERP (*Enterprise Resource Planning*) não consegue coletar dados diretamente da máquina, ele depende de algum outro sistema ou ser alimentado manualmente, metas são organizadas para uma análise mensal, anual e semanal, os KPIs mais importantes são Faturamento, Custos e Lucros além de atender ainda diversas áreas na empresa como financeiro, vendas, compras, RH e manufatura, e essa é uma das suas mais importantes funções, entender como toda a empresa está funcionando, permitindo tomadas de decisões gerenciais globais. Já o MES (*Manufacturing Execution System*), é um sistema mais especializado, voltando o olhar para a produção, seus KPIs mais importantes são coleta, tratamento e disponibilidade de dados de manufatura. É a

ferramenta apropriada para ajudar o gestor da produção a buscar competitividade para sua operação fabril analisando OEE, Produção e Qualidade é um cronômetro mais barato e mais apropriado para medir as coisas que acontecem na fábrica, num ritmo muito mais acelerado.

## 1.2. Controladores Programáveis

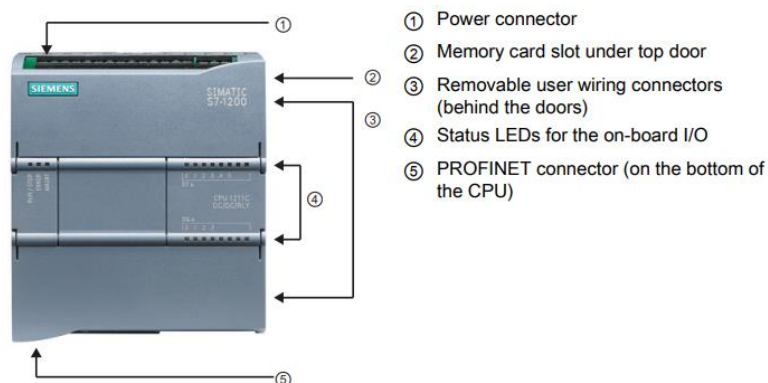
Um controlador pode ter alguns nomes variados, porém e o mesmo equipamento segundo FRANCHI (2008) Devido ao intuito inicial de substituírem os painéis de relés no controle discreto, foram chamados de PLC (*Programmable Logic Controllers*) traduzido para o português como CLP (Controladores Lógicos Programáveis). Porém, atualmente, os controladores são bem mais complexos, pois as plantas industriais normalmente precisam manipular não somente funções lógicas binárias, como, por exemplo, tipo E, OU, mas também controlar malhas analógicas, motivo pelo qual podem ser chamados atualmente apenas de PC (*Programmable Controllers*) ou CP (Controladores Programáveis), já que não são limitados a operações com condições lógicas. No entanto, o nome CLP fixou-se como sinônimo de produto.

A definição de CLP segundo FRANCHI (2008) controlador lógico programável pode ser visto como, um equipamento eletrônico de processamento que possui uma interface amigável com o usuário cuja função é executar controle de vários tipos e níveis de complexidade. Sendo assim pode se fazer uma programação no mesmo e executar algumas atividades em sua planta.

O CLP S71200 segundo SIEMENS (2012), é um controlador modular ilustrado na figura 2, onde o mesmo se encontra com entradas e saídas digitais e uma porta Ethernet integrada pode também ser adicionado mais módulos como o de entradas e saídas analógicas porta serial entre outros essa seria de controladores é um investimento garantido para várias aplicações além de ser seguro e confiável podendo ser expandido e atender uma série de demandas tecnológicas na indústria. Estes controladores atendem a norma IEC 61508 até a SIL 3 e ISO 13849-1 até a PL onde contém vários requisitos para poder estar no mercado e ser seguro e de alta confiabilidade, além disso o desenvolvimento dos seus programas ser no TIA Portal

que oferece diversas opções de comando operações e bloqueios para os programas relacionados à segurança.

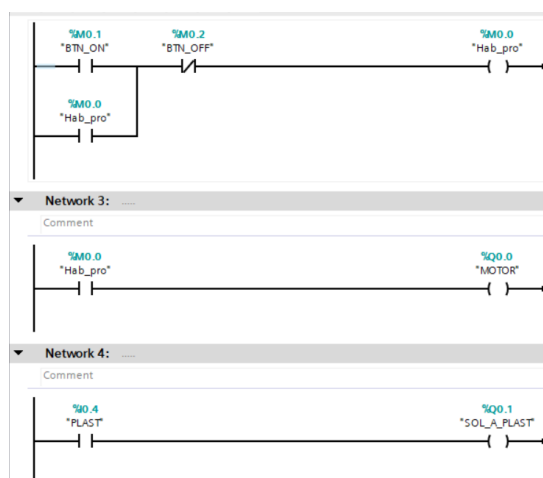
Figura 2 – Ilustração CLP



Fonte: SIEMENS (2012)

A programação ladder segundo FRANCHI, CLAITON MORO (2008) É uma linguagem gráfica baseada na lógica de relés e contatos elétricos para a realização de circuitos de comandos de acionamentos. Por ser a primeira linguagem utilizada pelos fabricantes, é a mais difundida e encontrada em quase todos os CLPs da atual geração. Esta linguagem também conta com alguns componentes e símbolos utilizados para programar e acionar saídas segundo FRANCHI, CLAITON MORO (2008) Bobinas e contatos são símbolos utilizados nessa linguagem. Os símbolos de contatos programados em uma linha representam as condições que serão avaliadas conforme a lógica. Como resultado determinam o controle de uma saída, sendo normalmente representado pelo símbolo de uma bobina como ilustrado na figura 3.

Figura 3 – Programação Ladder



Fonte: Do Autor (2022)

### 1.3. STL (Standard Template Library)

STL traduzindo para o português Biblioteca de Modelos Padrão que segundo EUGENIO (2009) STL é um conjunto de tipos abstratos de dados e funções projetados para manipularem diferentes tipos de dados transparentemente. As operações sobre os dados são realizadas por métodos que também aceitam tipos de dados genéricos. Como o próprio nome diz, STL recorre a templates, já definidos, para implementação de diversos algoritmos que manipulam dados eficientemente, como, por exemplo, vetores, dinâmicos, listas, pilhas, etc. No STL também foi adicionado um tipo *String*, que facilita as operações de manipulação de caracteres quando comparado a biblioteca *string.h* da linguagem C., além disso, também existem funções de busca, ordenação, dentre outros.

### 1.4. Redundância em CLPs

Pode se definir redundância segundo KOREN; KRISHNA (2010) A redundância de hardware é fornecida pela incorporação de hardware extra no design para detectar ou substituir os efeitos de um componente com falha.

A redundância no caso de CLPs pode ser tratada de forma estático ou dinâmico segundo KOREN; KRISHNA (2010) hardware estático tem o objetivo principal o mascaramento imediato de uma falha. Outra forma diferente de redundância de hardware é a redundância dinâmica, na qual os componentes sobressalentes são ativados após a falha de um componente atualmente ativo.

Para garantir o funcionamento de um processo como, por exemplo, óleo e gás, açúcar e etanol, usinas hidrelétricas, e agroindústria. Segundo ALTUS (2021), há vários motivos para utilizar sistemas de arquitetura redundante, como robustez, disponibilidade e segurança formam o tripé que sustenta a base deste tipo de metodologia. A mais utilizada na indústria é a *hot stand-by*, técnica em que um ou mais módulos ficando em espera enquanto o equipamento principal está operacional. Neste tipo de arquitetura, os módulos em espera funcionam em sincronia com o equipamento ativo e, caso uma falha seja detectada, o mesmo está pronto para se tornar operacional imediatamente.

## 1.5. Redes Industriais

As redes industriais vieram para ajudar no tráfego de informações e ligações no CLP segundo MORAES (2019) as vantagens trazidas por essas ferramentas tornam os sistemas de redes industriais muito atraentes, possuindo grande confiabilidade e modularidade, facilidade de compreensão e redução de custos em comparação aos sistemas centralizados anteriormente utilizados, com CLP.

Porém uma das maiores vantagens, segundo CASSIOLATO [s.d.] é fazer com os dados trafeguem desde o chão de fábrica, em máquinas, equipamentos, etc., que possa chegar até o nível de informação gerencial da empresa.

Redes industriais tem protocolos de comunicação como, por exemplo, profibus, que segundo CASSIOLATO [s.d.] O PROFIBUS é um padrão de rede de campo aberto e independente de fornecedores, onde a interface entre eles permite uma ampla aplicação em processos, manufatura e automação predial. Esse padrão é garantido segundo as normas EN 50170 e EN 50254. Desde janeiro de 2000, o PROFIBUS foi firmemente estabelecido com a IEC 61158

Também há o profinet que segundo CASSIOLATO [s.d.] O PROFINET é um conceito de automação compreensível que emergiu como resultado da tendência na tecnologia de automação para máquinas reusáveis e modulares em plantas com inteligência distribuída. Suas particularidades atendem postos-chaves das demandas da tecnologia de automação já a comunicação consistente entre os diversos níveis de gerenciamento desde o campo até os níveis corporativos usando Ethernet. Uma abundância de fabricantes em um protocolo e sistema aberto; utiliza padrões IT; integração em sistemas PROFIBUS sem mudanças dos mesmos.

## 1.6. SCC e SCD

O SCC (Sistemas de Controle Centralizado) é centralizado em apenas um processador, segundo MAHALIK, LEE, (2003) todo o processamento ocorre num processador (controlador) central onde todos os sensores e atuadores devem estar conectados. Este tipo de sistema requer um ponto de comunicação entre o processador central e cada sensor/atuador. Para sistemas grandes, o número de sensores/atuadores pode ser superior a mil, tornando árdua a instalação e a

manutenção desses sistemas, já que todos os pontos devem ser conectados a um único processador ou ponto de processamento.

Já os SCD (Sistemas de Controle Distribuído) são descentralizados cada sensor ou atuador tem seu processamento de acordo com MAHALIK, LEE (2003) cada sensor ou atuador possui um microprocessador responsável por todo o processamento local, isto é, cada sensor ou atuador é uma unidade básica de processamento ou nó do sistema. Os controladores nesses sistemas são encapsulados nos sensores e atuadores (nós). A funcionalidade do sistema é resultado do trabalho em conjunto dos nós que são unidades de processamento autônomas. Com esse método, divide-se uma tarefa grande em pequenas tarefas, cada uma residindo em um nó. O sistema de controle fica distribuído entre os vários nós que cooperam entre si, tornando remota a possibilidade de falha total, já que a falha em um nó não paralisa o funcionamento dos demais.

### **1.7. Node-RED**

O Node-RED é uma plataforma baseada em Javascript que segundo FERENCZ, DOMOKOS (2019) é um ambiente de programação virtual baseado em processos que cria “fluxos de dados” do sensor para a nuvem conectando hardware e softwares. É adequado para escrever processos de dados, facilitando assim em processamento.

Também tem aplicações de processamento que segundo FERENCZ, DOMOKOS (2019) Ele pode ser usado para compilar facilmente a lógica de processamento de dados e transferir os dados processados para sistemas de nível superior (servidor SQL, sistema, coletor central de dados, serviço baseado em nuvem) em minutos ou exibi-lo imediatamente. Em vez de programar uma página da Web exibindo dados de sensores diferentes, o Node-RED fornece uma interface do Dashboard que permite criar uma interface sem a necessidade de conhecimentos especiais de HTML, CSS.

### **1.8. Linguagem Fluxo (Node-RED)**

A ferramenta Node-RED tem sua linguagem de fluxo e programação segundo REIS (2021) Por meio de nodes (ou fluxos) é possível a execução de eventos HTTP,

TCP (Protocolo de Controle de Transmissão), *websocket* (tecnologia de Comunicação Bidirecional), MQTT (Transporte de telemetria de enfileiramento de mensagens, utilizado para troca de mensagens entre sensores) e outros. Sua edição é baseada no navegador, sem que haja restrições para se facilitar as conexões. O principal destaque desta aplicação se baseia no fato desenvolvimento ser possível em qualquer sistema operacional.

Os nodes são fluxos e cada um tem seu propósito de acordo com CELESTRINI (2022) No Node-RED, programas são fluxos compostos por uma coleção de nós conectados para trocar mensagens. Cada nó tem um propósito bem definido tecnicamente, um fluxo consiste em uma lista de objetos Javascript que descrevem os nós e suas configurações.

### **1.9. Disponibilidade de dados em nuvem**

A disponibilidade dos dados e de suma importância, pois são dados que precisam ser acessados a qualquer momento segundo FWC (2019) Para se ter alta disponibilidade incorporada a ideia de acesso a serviços, ferramentas e dados a qualquer hora e em qualquer lugar, e é a base da empresa moderna. A disponibilidade também está relacionada à confiabilidade: um serviço que funciona 24 horas, 7 dias por semana, mas sempre instável, não é considerado bom. Para alcançar uma alta disponibilidade real na nuvem, não apenas seu serviço deve estar sempre ativo, mas também deve estar 99,2% disponível. Vale a pena notar que nem todos os serviços em nuvem precisam estar disponíveis o tempo todo. À medida que os serviços se tornam mais críticos, a disponibilidade do sistema se torna mais importante.

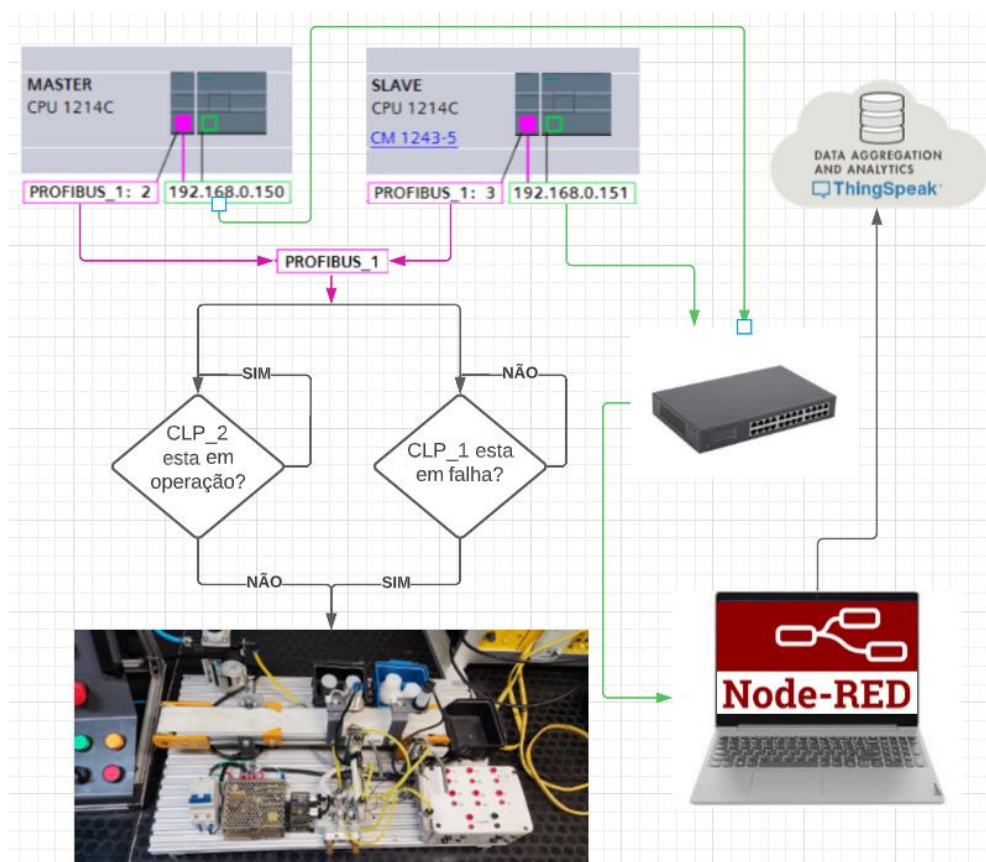
A confiabilidade está totalmente ligada a disponibilidade que há um recurso muito valioso segundo FWC (2019) O projeto para se ter alta disponibilidade em nuvem tem objetivo de proteger o recurso mais precioso: o tempo. Cada segundo do tempo de inatividade do seu sistema custa dinheiro para você e seus clientes. Como resultado, os administradores de sistema gastam mais custos de solução de problemas a cada segundo. As principais razões para o tempo de inatividade do sistema de TI da empresa estão relacionadas a incidentes de segurança e desastres naturais.

## 2. Metodologia

Para garantir que haja funcionalidade, disponibilidade e confiabilidade e o objetivo é que o sistema funcione 100% sem falhas no hardware de controle garantindo que não pare o processo, dessa maneira o sistema funcionara com dois controladores lógicos programáveis (CLP) siemens que possuem os mesmos dados e o mesmo programa assim sendo um mestre e um escravo sendo o programa rodando no master e o slave aguardando assim se houver falha o outro assumira o controle do processo dando confiabilidade para não haver perdas.

Para realização da redundância entre os CLPs será utilizado alguns itens para compor os projetos que serão: dois CLPs siemens S71200, modulo máster profibus, modulo slave profibus, Cabo profibus FC GP 2 vias blindado, esteira didática do SENAI Londrina, um switch e um computador com Node-RED instalado devidamente. Para ilustrar todos os itens listados e como foram instalados foi feito um fluxograma na figura 4.

Figura 4 – Fluxograma









Fonte: Do Autor (2022)

## 2.1. CLPs e programa

Para atuação em redundância será utilizado dois CLPs S71200 (siemens) programados devidamente conforme a utilização da esteira didática. Cada CLP haverá o mesmo programa compilado internamente com algumas diferenças de configuração e será utilizado módulos profibus sendo assim cada módulo tem sua configuração específica. O programa compilado foi utilizado uma estratégia de utilizar tags iguais como, por exemplo, a saída para ligar o motor da esteira for Q0.0 no CLP máster será no slave da mesma forma e assim segue conseqüentemente em todas entradas espelhando as tags entre os CLPs na figura 5 temos todas as tags utilizadas no programa onde compor com a inicial %Q são saídas digitais, e %I são entradas digitais.

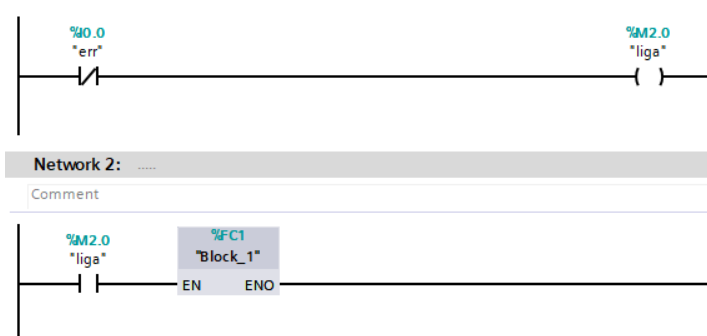
Figura 5 – Lista de tags

	Sensor Polimero	Default tag table	Bool	%I0.4
	Sensor Metal	Default tag table	Bool	%I0.5
	Motor Esteira	Default tag table	Bool	%Q0.0
	Solenóide Polimero	Default tag table	Bool	%Q0.1
	Solenóide Metal	Default tag table	Bool	%Q0.2
	Sinal	Default tag table	Bool	%Q0.3

Fonte: Do Autor (2022)

No programa do CLP slave há uma diferença é utilizado bloco um OB (*Organization Block*) Main que é executado de forma cíclica, e também um FC (*Function*) Block\_1 que precisa ser chamado dentro do OB Main. Como ilustrado na figura 6 que temos um contato para ativar a FC Block\_1.

Figura 6 – Contato do FC Block\_1



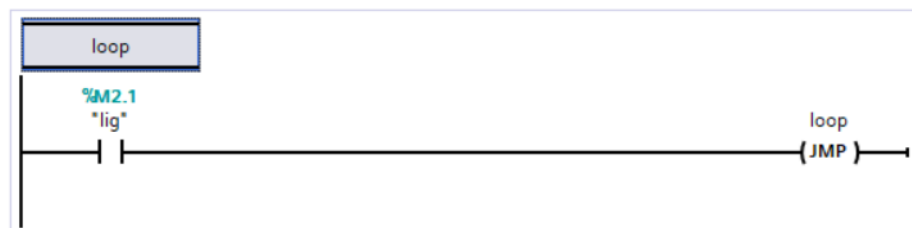
Fonte: Do Autor (2022)

O contato "err" é o sinal de erro vindo do máster dizendo que o mesmo está com problemas e está inativo e pedindo para ativar o bloco FC Block\_1 contendo o mesmo programa que já estava em operação assim assumindo o comando e dando

sequência no trabalho e operação dando confiabilidade no sistema, pois o programa estava parado esperando o erro se caso estivesse rodando em paralelo poderia causar algum conflito de sinal ou algum *delay* sendo assim foi optado por deixar o programa em espera.

Para simular o erro foi forçado com uma estratégia de programação dentro do CLP master utilizando uma estratégia de *cycle time* com um limite de 6 segundos para resposta se ultrapassar o CLP entra em erro e fica em estado de *stop*. Para esta configuração foi utilizado um OB *cyclic interrupt* que executado de forma cíclica, porem gera interrupção por tempo e fase pode também chamar FC que no caso chama o bloco FC Block\_1 contendo dentro uma bobina JMP que interromper a execução linear do programa e retoma em outra rede porem é retomada na mesma assim causando o erro igual mostrado na figura 7.

Figura 7 – Loop para erro

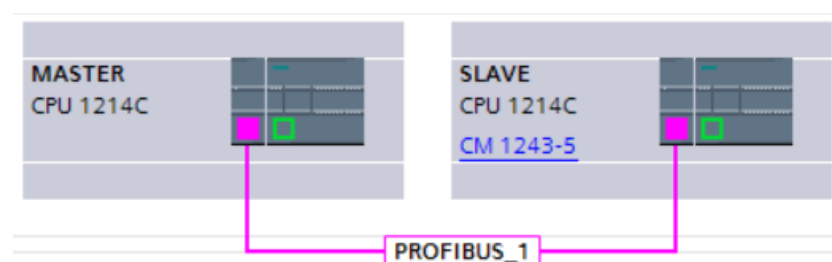


Fonte: Do Autor (2022)

## 2.2. Comunicação profibus

A comunicação entre os CLPs será utilizado um protocolo de comunicação seguro e confiável entre os CLPs onde consiga suprir a demanda e faça o papel de transportar informações, este protocolo a ser utilizado é o profibus que é muito utilizado em equipamentos similares e iguais aos que serão utilizados nesta aplicação conforme na figura 8 onde representa a comunicação profibus.

Figura 8 – comunicação entre CLPs



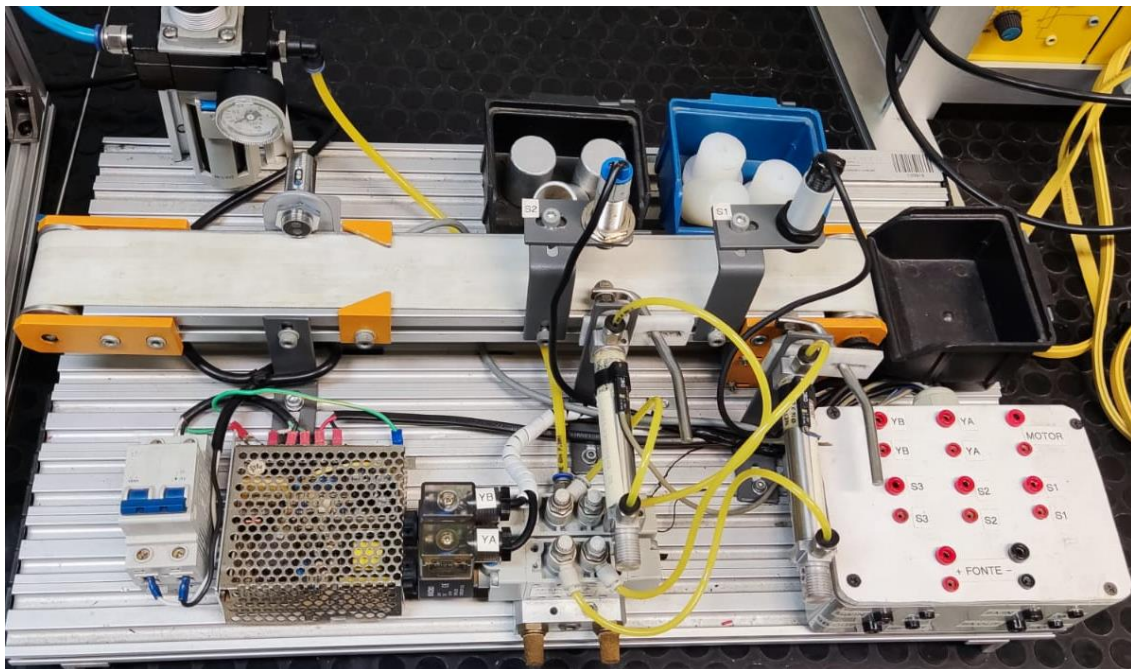
Fonte: Do Autor (2022)

Os módulos conversam entre si quando ocorre o erro no CLP máster o modulo entra em erro assim entrando em erro o modulo slave porem o CLP slave não para de funcionar só fica mostrando erro que é do próprio modulo profibus.

### 2.3. Esteira Didática

Para aplicar estas ferramentas será utilizado a esteira didática festo do Senai Londrina com a finalidade de separar materiais metálicos e não metálicos que estão sendo encaminhados por meio de uma esteira e poderão ser visualizados pelo sensor capacitivo e indutivo que farão o reconhecimento do material e apontando para qual atuador deve ser acionado para destinar corretamente o material para outra finalidade na figura 9 é apresentado a imagem da esteira.

Figura 9 – Esteira Separadora



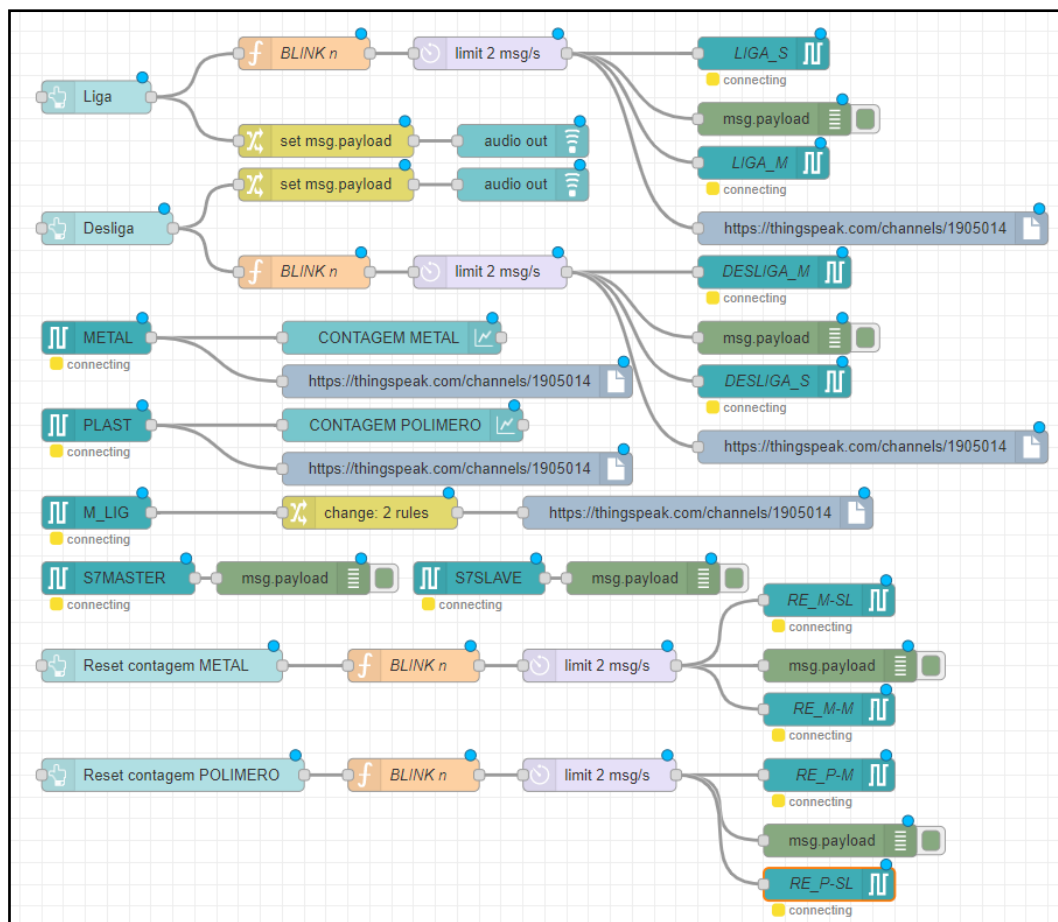
Fonte: Do Autor (2022)

Na esteira temos os bornes onde será conectado os terminais de entrada e saída dos CLPs. Para poder reconhecer os materiais os sensores terão de ser ligados nas entradas digitais dos CLPs, o sensor S1 capacitivo para os poliméricos e o sensor S2 indutivo para os metálicos, já para fazer a separação será ligado aos terminais de saída dos CLPs a solenoide YA para os poliméricos e YB para os metálicos e o MOTOR para ligar a esteira. Lembrando que o negativo da fonte da esteira tem que estar equalizado com as fontes dos CLPs.

## 2.4. Node-RED

Para este projeto foi utilizado a linguagem de blocos que o software online oferece para solucionar o caso de acionamento do processo e verificação da quantidade de material separado pela esteira e também ter a aplicação de coleta e envio de dados para nuvem. Para este processo foi feita a programação que pode ser identificada na figura 10.

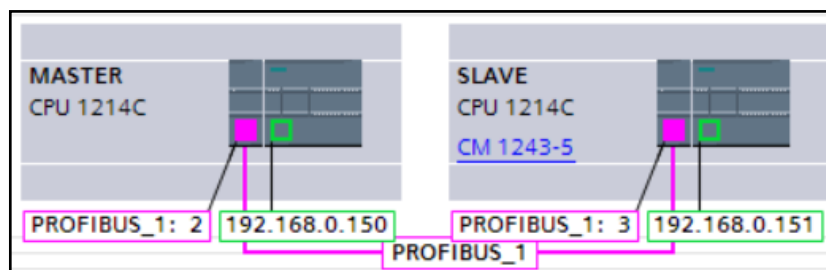
Figura 10 – Programação Blocos Node-RED



Fonte: Do Autor (2022)

Para comunicação entre os CLPs e o computador onde o software está instalado foi utilizado um switch para interligar as comunicações, sendo cada CLP tendo um IP para o Node-red poder encontrar os dados dos CLPs. Na figura 11 temos em verde cada IP dos CLPs.

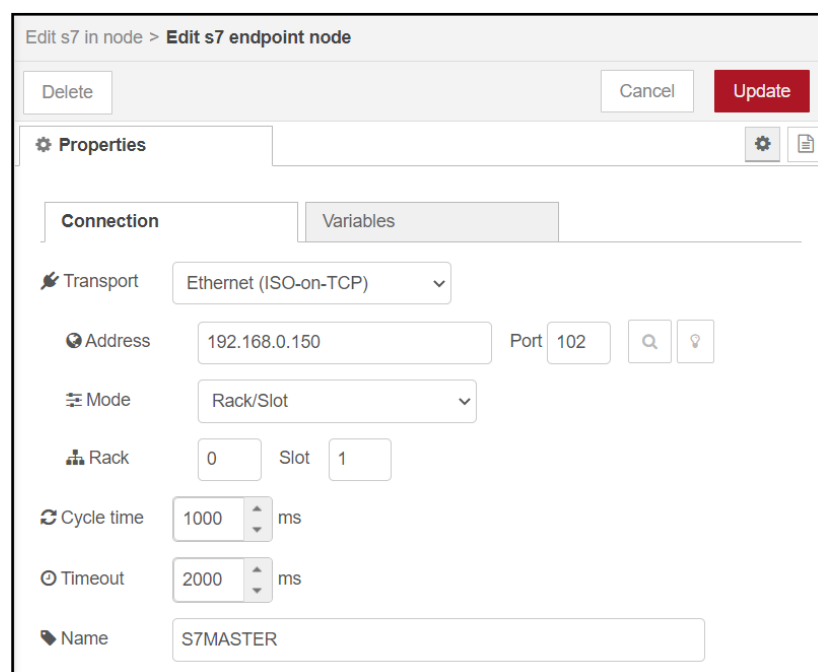
Figura 11 – IP dos CLPs



Fonte: Do Autor (2022)

Foi utilizado uma biblioteca ou chamado pelo software de *manage palette* onde foi feito a instalação da *node-red-contrib-s7* onde é feito a integração entre o CLP S71200 e o Node-RED onde tem que ser preenchido os dados conforme na figura 12.

Figura 12 – configurações Node-RED



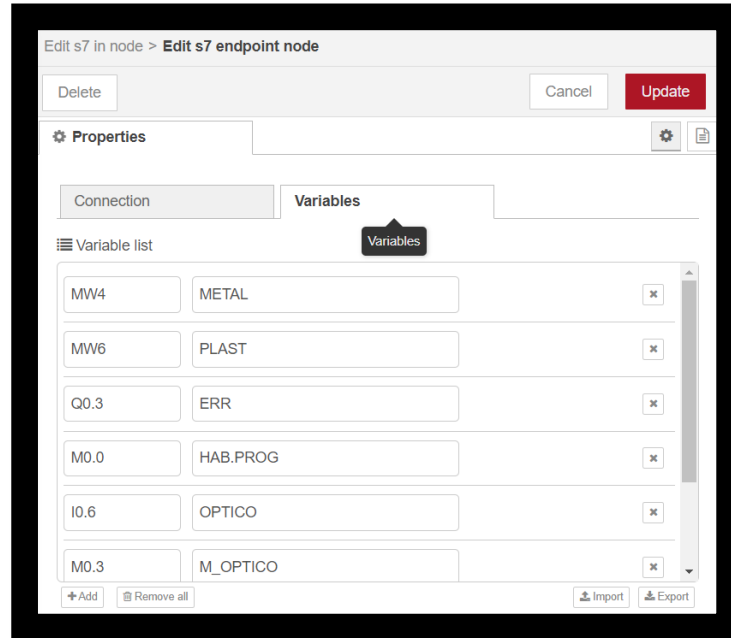
Fonte: Do Autor (2022)

Para poder verificar as variáveis e poder converter em ações como a de ligar e desligar o processo e mandar os dados para o aplicativo da nuvem tem que ser listado as variáveis que são as mesmas usadas no programa do CLP. O preenchimento tem que ser conforme a figura 13.

Para poder fazer acionamento do processo foi utilizado um *manage palette* chamado *node-red-dashboard* onde foi criado dois botões apenas para ligar e desligar

o processo podendo ligar e desligar o processo tanto enquanto está operando normalmente em master ou quando está em erro e o slave operando

Figura 13 – configurações Node-RED



Fonte: Do Autor (2022)

No dashboard também foi colocado para verificar a produção na tela um gráfico por hora trabalhada conforme a figura 14 do dashboard montado.

Figura 14-configuraçõesNode-RED



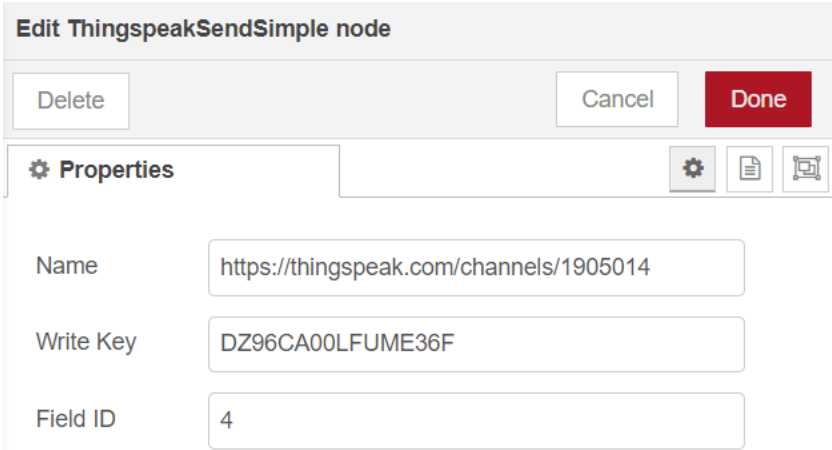
Fonte: Do Autor (2022)

Para o caso do banco de dados e registrar os dados foi utilizado um *manage palette* chamado *node-red-contrib-thingspeak* que comunica com a plataforma online do site ThingSpeak formando um banco de dados.

## 2.5. Dados em nuvem

Os dados em nuvem são registrados em forma de gráfico pela plataforma online do ThingSpeak que fornece uma integração com o Node-RED. Para a configuração tem que ser preenchida os dados conforme fornecidos pelo provedor da plataforma do banco de dados para registro dentro mesmo em seu canal conforme na figura 15.

Figura 15 – configurações Node-RED



Fonte: Do Autor (2022)

O banco de dados registrado pelo ThingSpeak é de forma online e registra em nuvem para acesso de dispositivo remotos só necessitam do link fornecido e a permissão do autor dentro tornando público ou no caso privado para acesso teria que ter uma conta no site e permitido acesso cadastrando e-mail para acesso ao banco.

## 3. Resultados

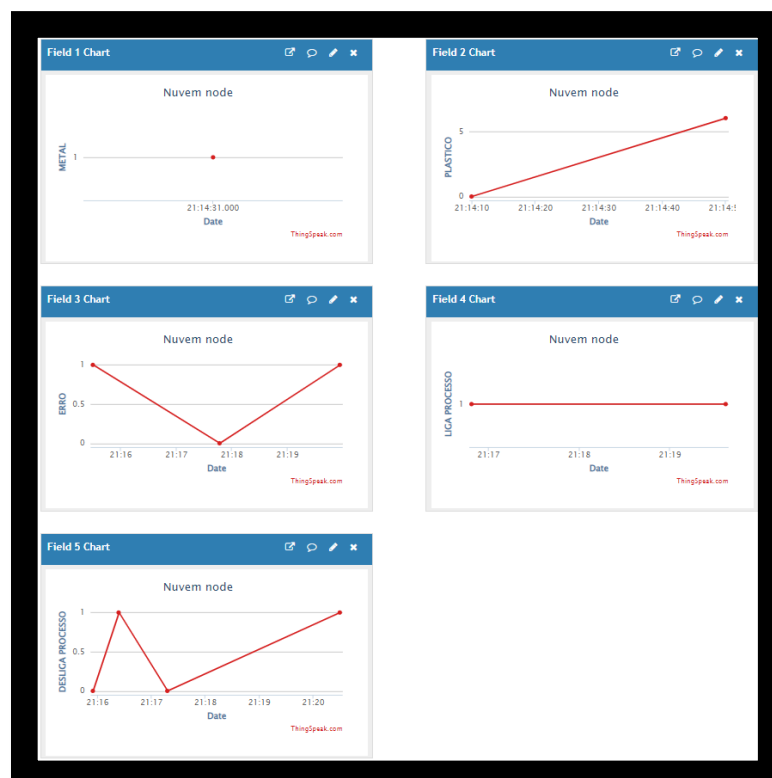
Os controladores lógicos programáveis (CLP) da Siemens foram devidamente aplicados e apresentaram resultados satisfatórios na redundância e na aplicação do Node-RED onde com a simulação do erro feito através do *cycle time* foi um sucesso, também foram feitos testes desenergizando o CLP master e retirando o cabo de comunicação Profibus onde ocorreram erros porém para não acabar danificando o CLP a alternativa mais amistosa aplicada foi dando erro de *cycle time*, pois é apenas

um erro de programação que não traria nenhum dano ao CLP e nem a comunicação com o Node-RED e nuvem de dados.

O Node-RED foi um sucesso conseguindo controlar a função de ligar, desligar o processo tanto em master quanto em slave, assim sendo aplicado como uma forma de supervisor, também registrando a separação de materiais metálicos e poliméricos e zerando a contagem, além disso, fazendo a ponte entre controlador e nuvem.

Para os resultados do banco de dados em nuvem foi feito os registros porem os registros não conseguiram ser registrados em tempo real obtendo um delay entre a mensagem enviada do controlador e o registrado pelo Thingspeak, pois há uma latência de rede onde não foi utilizado uma banda 5G, pois é uma tecnologia nova que está surgindo e também o Thingspeak é um software gratuito e o que é oferecido gratuitamente por ele não é uma tecnologia de ponta assim o banco de dados obteve registros porem com atraso e alguns se perderam e não foram registrados. Os registros obtidos através de alguns minutos estão registrados na figura 16.

Figura 16 – Registros do Thingspeak



Fonte: Do Autor (2022)

#### 4. Conclusão

Quando foi obtida a ideia do estudo sobre redundância a ideia inicial que obtida foi inteiramente um sucesso e cumprindo toda a expectativa porém infelizmente o registro de dados em nuvem não foi um sucesso absoluto, pois obteve alguns problemas com latência de rede e também por ser uma plataforma gratuita não foi obtido uma experiência das melhores porém o objetivo foi cumprido obteve registros de alguns dados e se for substituído o servidor para dados em nuvem e colocado uma rede de dados 5G pode haver um registro de dados em tempo real de melhor qualidade assim concluo que a pesquisa e testes que foram aplicados e demonstraram que a utilização e aplicação de redundância foram satisfatórias e confiáveis e podem ser aplicados em processos industriais apenas feitos alterações para obter dados mais preciso e mais rápidos por parte do banco de dados.

#### REFERÊNCIAS

- KAGERMANN, H et al. *Recommendations for implementing the strategic initiative industrie*. 2013.
- Indústria 4.0: o que é, impactos, benefícios e tecnologias. TOTVS, 2021. Disponível em: <https://www.totvs.com/blog/gestao-industrial/industria-4-0/>. Acesso em: 14 mai.2022.
- SHERIDAN, Thomas B. *Telerobotics, automation, and human supervisory control*. Primeira. [S.l.: s.n.], 1992.
- MORAES, Cícero Couto de; CASTRUCCI, Plínio de Lauro. *Engenharia de Automação Industrial*. 2ª ed. Rio de Janeiro: Ed. LTC, 2010.
- FRANCHI, CLAITON MORO, Claiton Moro Franchi, Valter Luís Arlindo de Camargo. *Controladores Lógicos Programáveis - Sistemas Discretos*. 1. ed. - São Paulo: Érica, 2008.
- KOREN, I.; KRISHNA, C. M. *Fault-tolerant systems*. [S.l.]: Morgan Kaufmann, 2010.
- CASSIOLATO, César. *Redes Industriais*. Smar technology company [s.d]. Disponível em: <https://www.smar.com/pt/artigo-tecnico/redes-industriais>. Acesso em: 20 mai.2022.

FERENCZ, DOMOKOS, Katalin Ferencz, József Domokos. *Using Node-RED platform in an industrial environment* - Jubileumi Kandó Konferencia 2019.

MORAES, Alexandre Fernandes de. *Redes industriais para automação industrial : AS-I, profibus e profinet* / Alexandre Baratella Luigli e Max Mauro Dias Santos - 1. ed. - São Paulo: Érica, 2019.

REIS, Elias da Cunha Lima. *Automação residencial com a utilização de raspberry pi*. Orientador: Me. Gustavo Siqueira Vinhal. 2021. 43 f. Trabalho de conclusão de curso (Bacharelado em Engenharia da Computação) - Pontifícia Universidade Católica de Goiás, [S.l.], 2021. Disponível em: <https://repositorio.pucgoias.edu.br/jspui/handle/123456789/3070>. Acesso em: 22 mai. 2022.

CELESTRINI, Jordano R. *Desenvolvimento de aplicações integrando serviços web, fontes de dados e dispositivos IoT com o uso do Node-RED*. [S. l.], [2022]. Disponível em: <http://inf.ufes.br/~zegonc/material/Redes%20de%20Sensores%20sem%20Fio/Minicurso%20Node-RED.pdf>. Acesso em: 22 mai. 2022.

MAHALIK, N. G. P. CC; LEE, S. K. *Design and development of system level software tool for DCS simulation*. *Advances in Engineering Software*, v. 34, p.451-465, 2003.

EUGENIO, George Eugenio. *Introdução a Programação Orientada à Objetos na Linguagem C* - Universidade Federal de Santa Maria, 2009.

Alta disponibilidade em nuvem: o que as empresas de sucesso não contam para você! FWC 2019. Disponível em: <https://www.fwc.com.br/blog/cloud/alta-disponibilidade-em-nuvem-o-que-as-empresas-de-sucesso-nao-contam-para-voce/>. Acesso em: 28 mai.2022.

SANTOS, Guilherme. *A pirâmide da Automação Industrial*. *Automação Industrial*, 2021. Disponível em: <https://www.automacaoindustrial.info/a-piramide-da-automacao-industrial/>. Acesso em: 27 ago. 2022.

NARCISO, Paulo Filho. *Qual a diferença entre MES e o ERP?* 2021. Disponível em: <https://www.livemes.com/qual-a-diferenca-entre-mes-e-o-erp/>. Acesso em: 28 ago. 2022.

SIEMENS. *S7-1200 Programmable controller System Manual*. Siemens AG Industry Sector Postfach 48 48 90026 NÜRNBERG GERMANY: Siemens AG, 2012, total de páginas 699.