

TRABALHO DE GRADUAÇÃO

PROJETO, INSTALAÇÃO E CONTROLE DE UM SISTEMA DE ELEVADORES USANDO TECNOLOGIAS DE AUTOMAÇÃO INDUSTRIAL

Por,
Gabriel Coelho do Amaral
Helton José de Almeida Costa

Brasília, 15 de Dezembro de 2015



**ENGENHARIA
MECATRÔNICA**
UNIVERSIDADE DE BRASÍLIA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia
Departamento de Engenharia Mecânica

TRABALHO DE GRADUAÇÃO

PROJETO, INSTALAÇÃO E CONTROLE DE UM SISTEMA DE ELEVADORES USANDO TECNOLOGIAS DE AUTOMAÇÃO INDUSTRIAL

POR,

Gabriel Coelho do Amaral
Helton José de Almeida Costa

Relatório submetido como requisito parcial para obtenção
do grau de Engenheiro de Controle e Automação.

Banca Examinadora

Prof. Dr. Guilherme Caribé de Carvalho, UnB/ ENM
(Orientador)

Prof. Gerson Henrique Pfitscher, UnB/ ENE
(Coorientador)

Prof. Luis Felipe Cruz Figueredo
(Examinador Interno)

Brasília, 15 de Dezembro de 2015

FICHA CATALOGRÁFICA

AMARAL, GABRIEL COELHO, COSTA, HELTON JOSÉ DE ALMEIDA.

Projeto, instalação e controle de um sistema de elevadores usando tecnologias de automação industrial.

[Distrito Federal] 2015.

Xi, 109p., 297 mm (FT/UnB, Engenheiro, Controle e Automação, 2015). Trabalho de Graduação – Universidade de Brasília. Faculdade de Tecnologia.

- | | | |
|------------------|--------------------------|-------------------------|
| 1. Introdução | 2. Revisão Bibliográfica | 3. Materiais Utilizados |
| 4. Procedimentos | 5. Conclusão | 6. Bibliografia |
| 7. Anexos | | |

I. Mecatrônica/FT/UnB

REFERÊNCIA BIBLIOGRÁFICA

AMARAL, GABRIEL C. , COSTA, HELTON J. A. (2015). Projeto, instalação e controle de um sistema de elevadores usando tecnologias de automação industrial. Trabalho de Graduação em Engenharia de Controle e Automação, Publicação FT TG-022/2015 Faculdade de Tecnologia, Universidade de Brasília, Brasília, DF, 109p.

CESSÃO DE DIREITOS

AUTORES: Gabriel Coelho do Amaral e Helton José de Almeida Costa

TÍTULO DO TRABALHO DE GRADUAÇÃO: Projeto, instalação e controle de um sistema de elevadores usando tecnologias de automação industrial.

GRAU: Engenheiro ANO: 2015

É concedida à Universidade de Brasília permissão para reproduzir cópias deste Trabalho de Graduação e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desse Trabalho de Graduação pode ser reproduzida sem autorização por escrito do autor.

Gabriel Coelho do Amaral

Helton José de Almeida Costa

RESUMO

Elevadores de passageiros fazem parte do dia-a-dia das pessoas. Cada vez mais comuns até em prédios de pequeno porte, o mercado disponibiliza diversas opções de soluções para a implementação destes. Neste trabalho é realizado um estudo sobre os diversos componentes e equipamentos utilizados para o funcionamento e controle eficiente de um elevador de tração para passageiros. Além disso, é realizada a montagem mecânica e elétrica do sistema, com seus respectivos diagramas. O controle do sistema é feito por meio de um controlador lógico programável. Sensores são utilizados para obter informações e auxiliar no sistema de controle. Os motores dos elevadores são acionados por meio de controladores de velocidade, que são parametrizados de acordo com as especificações daqueles. Todos os sensores, atuadores, controladores e interfaces homem-máquina do sistema são integrados por meio de uma rede industrial de comunicação. Uma estratégia de controle do sistema é utilizada para a implementação da programação dos controladores.

Palavras-Chave: Elevadores, Controlador Lógico Programável, Controle de Velocidade, Sensores, Atuadores, Estratégias de Controle.

ABSTRACT

Passenger elevators are part of day-to-day lives. Increasingly common, even in small buildings, the market offers a choice of solutions to implement those. This work is a study on the various components and equipment used for the operation and efficient control of a traction elevator for passengers. In addition, the assembly of mechanical and electrical system is carried out along with the respective diagrams. The system control is implemented through a programmable logic controller. Sensors are used to get information and assist in the control system. The motors of the elevators are driven by variable frequency AC drives, which are parameterized according to the specifications of those. All sensors, actuators, controllers and man-machine interfaces of the system are integrated via an industrial communication network. A system control strategy is used to implement the programming of controllers.

Keywords: Elevators, Programmable Logic Controller, Speed Control, Sensors, Actuators, Control Strategies.

SUMÁRIO

1 INTRODUÇÃO	1
1.1 MOTIVAÇÃO	1
1.2 OBJETIVO	1
2 REVISÃO BIBLIOGRÁFICA	3
2.1 VISÃO GERAL	3
2.2 ESTRUTURA MECÂNICA	3
2.3 MAQUINAS DE TRAÇÃO	6
2.3.1 MOTORES HIDRÁULICOS	6
2.3.2 MOTORES ELÉTRICOS	6
2.3.2.1 MOTORES DE CORRENTE CONTÍNUA	7
2.3.2.1 MOTORES DE CORRENTE ALTERNADA	7
2.3.3 ACIONAMENTO DE MOTORES ELETRICOS	8
2.4 SENSORES	9
2.4.1 SENSOR INDUTIVO	10
2.4.2 SENSOR CAPACITIVO	10
2.4.3 SENSOR ULTRASSONICO	11
2.4.4 SENSOR FOTOELÉTRICO	13
2.4.5 SENSOR DE CONTATO COM ACIONAMENTO MECÂNICO	14
2.5 CONTROLADOR DE VELOCIDADE	15
2.6 CONTROLADOR LÓGICO PROGRAMÁVEL (CLP)	16
2.6.1 LINGUAGENS DE PROGRAMAÇÃO DE CLPS	17
2.7 REDES INDUSTRIAIS	20
2.7.1 MODBUS	21
2.7.2 PROFIBUS	22
2.7.3 DEVICENET	23
2.8 ALGORITMOS DE CONTROLE	24
2.8.1 ALGORITMOS HEURÍSTICOS	26
2.8.2 ALGORITMOS GENÉTICOS	27
2.8.3 LÓGICA NEBULOSA(FUZZY)	28
2.9 TECNICAS DE MODELAGEM	30
2.9.1 REDES DE PETRI	30
2.9.2 CSP	31
2.9.3 UML	32
3 MATERIAIS UTILIZADOS	34
3.1 ESTRUTURA MECÂNICA	34
3.2 MOTORES CA DE INDUÇÃO	35
3.2.1 ESPECIFICAÇÕES DO MOTOR	37
3.2.2 RESFRIAMENTO DO MOTOR	38
3.3 SENSORES	38
3.3.1 SENSOR ALLEN-BRADLEY	39
3.3.2 FAIXA DE TRABALHO	40
3.3.3 ALVO PADRAO	41
3.3.3 POSICIONAMENTO DOS SENSORES	41
3.4 POWERFLEX 70	42
3.4.1 CONTROLADOR DE VELOCIDADE	42
3.4.2 DEFINIÇÃO DE PARAMETROS E FUNÇÕES	43
3.5 CONTROLLOGIX 5555	44
3.6 PANELVIEW	44
3.7 REDE INDUSTRIAL	45
3.7.1 PROTOCOLO CAN	45
3.7.2 CAN E DEVICENET	47
3.7.3 DEVICENET	47
3.7.3.1 CAMADA FISICA	47
3.7.3.2 CAMADA DE ENLACE DE DADOS	48
3.7.3.5 ARQUIVO DE CONFIGURAÇÃO	49
3.7.3.6 MODOS DE COMUNICAÇÃO	49
3.7.3.7 CONJUNTO DE CONEXÕES PREDEFINIDAS MESTRE/ES CRAVO	50
4 PROCEDIMENTOS	50
4.1 ANÁLISE PRÉVIA	50
4.2 LIMPEZA E LUBRIFICAÇÃO	51
4.3 REPOSICIONAMENTO E FIXAÇÃO DOS SENSORES DE FINAL DE CURSO	51

4.4 INSTALAÇÃO ELÉTRICA.....	53
4.5 ENTRADAS E SAIDAS DO CLP	60
4.6 CONFIGURAÇÃO DOS PARAMETROS DO INVERSOR	62
4.6.1 CURVAS DE ACELERAÇÃO	63
4.6 ESTRATÉGIA DE CONTROLE	65
4.7 MODELAGEM E PROGRAMAÇÃO	66
4.7.1 MODELAMENTO POR REDE DE PETRI	66
4.7.2 DIAGRAMA LADDER	69
4.8 PANELVIEW	71
5 CONCLUSÃO	73
5.1 TRABALHOS FUTUROS	75
6 BIBLIOGRAFIA	76
7 ANEXOS	78
ANEXO 1 – BITS E/S DO CLP	78
ANEXO 2 – PARAMETROS DO INVERSOR	81
ANEXO 3 – PROGRAMACAO LADDER	88

LISTA DE FIGURAS

2.1	Esquema de disposições usuais dos componentes do elevador	5
2.2	Conexões dos terminais do motor	8
2.3	Principais componentes de um sensor indutivo.....	10
2.4	Principais componentes de um sensor capacitivo	11
2.5	Princípio de funcionamento de sensor ultrassônico	12
2.6	Principais componentes de um sensor fotoelétrico	13
2.7	Posições possíveis de um sensor de contato	14
2.8	Equação da velocidade nominal de um motor trifásico.....	15
2.9	Painel de controle de elevador baseado em relés.....	16
2.10	Visão geral de controle utilizando CLPs	17
2.11	Exemplo de código IL	18
2.12	Exemplo de código ST	18
2.13	Exemplo de código LD	19
2.14	Exemplo de código FDB	19
2.15	Exemplo de código SFC	20
2.16	Tipos de protocolo MODBUS	22
2.17	Exemplo de rede DeviceNet.....	24
2.18	Fluxo de passageiros em um prédio.....	26
2.19	Lógica fuzzy e suas operações	29
2.20	Fuzzificação de variáveis.....	30
2.21	Elementos da Rede de Petri.....	32
3.1	Desenho da estrutura mecânica	35
3.2	Descrição dos componentes de um motor CA de indução do tipo gaiola	36
3.3	Indicação da posição do freio no motor	37
3.4	Placa identificadora do motor	37
3.5	Componentes responsáveis pelo resfriamento do motor	38
3.6	Sensor indutivo de proximidade da Allen-Bradley	39
3.7	Alvo padrão (d = 20mm)	41
3.8	Distâncias necessárias para correto posicionamento dos sensores	41
3.9	PowerFlex 70 Allen-Bradley	43
3.10	CLP ControlLogix 5555 Allen-Bradley	44
3.11	PanelView 600 Allen-Bradley	45
3.12	Estrutura dos bytes do protocolo CAN	46
3.13	Frame de dados do CAN	48
4.1	Sensores de final de curso reposicionados	52
4.2	Sensores nas laterais do elevador	53
4.3	Foto dos alvos fixados com sensores indutivos reposicionados	53
4.4	Topologia da rede DeviceNet	54
4.5	Conexões e montagem da rede de comunicação	55
4.6	Conexões e identificações no Flex I/O	56
4.7	Conexão danificada dos sensores indutivos	57
4.8	Conexão ajustada dos sensores indutivos.....	58
4.9	Conexões danificadas no Flex I/O.....	58
4.10	Conexões no Flex I/O após ajustes.....	59
4.11	Representação da rede de comunicação para o elevador A	59
4.12	Representação da rede de comunicação para o elevador B	60
4.13	Definindo o parâmetro Access Lvl.....	62
4.14	Partida soft-starter do controlador de velocidade.....	64
4.15	Partida utilizando S-Curve.....	64
4.16	Parâmetro de configuração 146 do PowerFlex 70.....	64
4.17	Rede de Petri para a movimentação completa do sistema.....	68
4.18	Sequência de passos para a movimentação entre-andares	68

4.19	Lógica em ladder para chamadas externas	70
4.20	Movimentação do quinto ao primeiro andar	70
4.21	Retorno do segundo para o quinto andar.....	71
4.22	PanelView desenvolvido.....	72

LISTA DE TABELAS

3.1	Valores obtidos da placa do motor	39
3.2	Informações encontradas nos Bytes dos sensores	41
3.3	Fatores de redução para determinados materiais	41
3.4	Taxa de transmissão X tamanho da rede DeviceNet.....	48

LISTA DE SÍMBOLOS

Ω	Resistência Elétrica	
m	Metro - Unidade de Medida de Comprimento	
A	Ampères - Unidade de Medida de Corrente Elétrica	
V	Volt - Unidade de Medida de Tensão Elétrica	
Vdc	Tensão Contínua	[V]
CA	Corrente Alternada	[A]
CC	Corrente Contínua	[A]
kbps	Kilo bits por segundo	
W	Watts- Unidade de potência	

LISTA DE SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
UnB	Universidade de Brasília
CLP	Controlador Lógico Programável
GRACO	Grupo de Automação e Controle – UnB
NA	Normalmente aberto
NF	Normalmente fechado
FT	Faculdade de Tecnologia – UnB
CAN	Controller Area Network
CIP	Common Industrial Protocol
IL	Lista de instruções
ST	Texto estruturado
LD	Diagrama Ladder
FDB	Diagrama de blocos funcionais
SFB	Sequenciamento gráfico de funções
LED	<i>Light-emitting diode</i>
GM	<i>General Motors</i>
IEC	<i>International Electrotechnical Commission</i>
LRC	<i>Longitudinal Redundancy Check</i>
CRC	<i>Cyclic Redundancy Check</i>
TCP	<i>Transmission Control Protocol</i>
IP	<i>Internet Protocol</i>
I/O	<i>Input/Output</i>
E/S	<i>Entrada/Saída</i>
CSP	<i>Communicating Sequential Process</i>
UML	<i>Unified Modelling Language</i>
RPM	Rotações Por Minuto
HP	<i>Horse Power</i>
Hz	Hertz
VVVF	<i>Variable Voltage Variable Frequency</i>
IHM	Interface Homem Máquina
ISO	<i>International Organization for Standardization</i>
OSI	<i>Open System Interconnection</i>
CSMA	<i>Carrier Sense Multiple Access</i>
NBA	<i>Non-destructive Bitwise Arbitration</i>
EDS	<i>Electronic Data Systems</i>
SG	Serviço Geral

1 INTRODUÇÃO

1.1 MOTIVAÇÃO

O GRACO (Grupo de Automação e Controle) da UnB possui em seu laboratório um protótipo de um grupo de elevadores em escala reduzida. O grupo de elevadores é composto de dois carros montados em uma estrutura metálica. Suas cabines se movimentam pela estrutura metálica para chegar aos pavimentos desejados. Sensores para o controle do mesmo estão posicionados em cima das cabines e nas extremidades superior e inferior da estrutura metálica. Para o controle dos motores usa-se um variador de velocidade. Os sinais dos sensores e atuadores são controlados através de um Controlador Lógico Programável (CLP) através de uma rede DeviceNet.

Infelizmente este equipamento não está em funcionamento e, portanto, possibilita uma grande oportunidade para aplicar-se o conhecimento adquirido nas várias áreas da engenharia durante o curso de Engenharia Mecatrônica. Neste trabalho, almeja-se tornar funcional o grupo de elevadores, mediante a implementação de tecnologias de automação para sua integração e controle, entregando-o funcional e documentado.

1.2 OBJETIVO

O objetivo principal deste projeto é a utilização de tecnologias de automação industrial para um sistema de elevadores, permitindo o controle e acionamento do sistema como um todo de maneira robusta e eficaz.

Para alcançar este resultado devem-se seguir alguns passos, sendo o primeiro deles realizar uma análise prévia do estado do sistema, de modo a encontrar as falhas existentes e as possíveis soluções para as mesmas.

Posteriormente devem-se realizar os ajustes tanto para as partes mecânicas e elétricas quanto de comunicação do sistema, garantindo que tudo esteja em perfeito funcionamento antes de tentar integrá-los uns aos outros. Neste momento, é de grande importância que se realize uma documentação de como ficou disposta toda a estrutura do sistema, para que projetos futuros sobre este mesmo sistema possam ser realizados rapidamente sem desperdício de tempo na identificação dos equipamentos e cabos.

A seguir, desenvolver-se-á a estratégia de movimentação e de controle do sistema de elevadores, que será implementada no sistema através do desenvolvimento da programação,

configuração dos parâmetros do variador de velocidade e configuração das entradas e saídas do CLP de acordo com a rede de comunicação.

Por fim, realizar-se-ão testes extensivos de modo a garantir que todo o sistema esteja em perfeito estado de funcionamento, respeitando todas as lógicas impostas pelo projeto e seguindo todo o fluxo de controle.

2 REVISÃO BIBLIOGRÁFICA

2.1 VISÃO GERAL

O elevador é o conjunto de todos os equipamentos necessários para a realização do transporte vertical, seguindo as normas técnicas. Este projeto foi desenvolvido com os equipamentos de automação e a estrutura de elevadores, com tamanho reduzido. Esses equipamentos são necessários para alcançar os objetivos propostos.

Basicamente, o elevador consiste em uma estrutura mecânica, em que o carro se movimenta entre andares. Motores indutivos são utilizados para realizar essa movimentação. Sensores são posicionados na estrutura mecânica, para que o controle dos mecanismos seja feito da maneira correta. Os motores são acionados por meio de variadores de velocidade baseados em circuitos inversores de frequência (*AC drives*). A lógica de comando para os equipamentos é feita por um CLP, que recebe as informações dos sensores. Os equipamentos trocam informações através de uma rede de comunicação utilizando protocolos industriais.

2.2 ESTRUTURA MECÂNICA

Um elevador convencional consiste de várias partes definidas pela norma NM 207 [1]. Algumas dessas definições que serão utilizadas ao longo deste trabalho são apresentadas abaixo e mostradas na figura 2.1.

- **Elevador de tração:** Elevador cujos cabos são acionados por atrito nas ranhuras da polia motriz da máquina.
- **Armação do carro:** Estrutura metálica sustentando a cabina, ligada aos meios de suspensão. Esta armação pode ser integrada com o fechamento da cabina.
- **Cabina:** A parte do elevador que transporta passageiros e objetos.
- **Cabo de comando:** Cabo elétrico flexível entre o carro e um ponto fixo.
- **Cadeia elétrica de segurança:** O total de dispositivos elétricos de segurança ligados em série.
- **Última altura:** Parte da caixa entre o pavimento extremo superior servido pelo carro do elevador e o teto da caixa.
- **Contrapeso:** Massa que assegura a tração.
- **Casa de máquinas:** Recinto no qual estão instaladas as máquinas e o equipamento relacionado com elas.

- **Guias:** Os componentes rígidos destinados a manter a direção do movimento do carro ou do contrapeso.
- **Caixa:** Espaço onde o carro e o contrapeso viajam. Este espaço é limitado pelo fundo do poço, as paredes e o teto.
- **Limitador de velocidade:** Dispositivo que, quando o elevador atinge uma velocidade predeterminada, causa a parada do elevador e, se necessário, aciona o freio de segurança.
- **Máquina:** A unidade que aciona e para o elevador.
- **Nivelamento:** Operação que proporciona precisão de parada nos pavimentos.
- **Freio de segurança:** Dispositivo mecânico para parar e manter travado nas guias o carro do elevador ou o contrapeso em caso de sobrevelocidade no sentido de descida ou de ruptura da suspensão.
- **Passageiro:** Qualquer pessoa transportada dentro da cabina do elevador.
- **Poço:** Parte da caixa situada abaixo do nível de parada mais baixo servido pelo elevador.
- **Usuário:** Pessoa que faz uso dos serviços de uma instalação elevadora.
- **Velocidade nominal:** Velocidade do carro para a qual o equipamento foi construído.

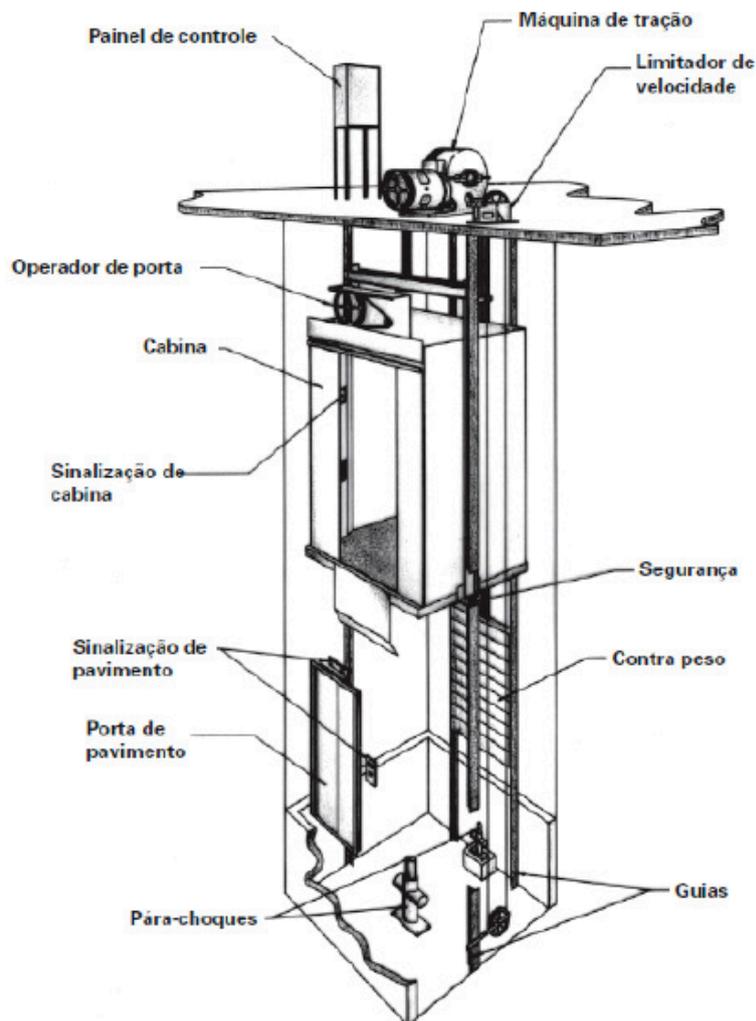


Figura 2.1: Esquema de disposições usuais dos componentes do elevador.[2]

O esquema básico de funcionamento de um elevador consiste em uma cabine montada sobre uma plataforma, em uma armação de aço, ligada aos meios de suspensão. Esse conjunto constitui a armação do carro. O contrapeso consiste em uma armação onde são fixados blocos de concreto, com massas iguais à massa total do carro, acrescido de 40 a 50% da capacidade licenciada.

Tanto a cabine como o contrapeso deslizam pelas guias, através de corredeiras. As guias são fixadas em suportes metálicos, chumbados nas vigas na caixa do elevador.

O carro e o contrapeso são suspensos por cabos de aço que passam por polias instaladas na casa de máquinas ou na parte superior da caixa.

O movimento do carro e do contrapeso é proporcionado pela máquina de tração, geralmente hidráulica ou elétrica. Esta máquina proporcionará rotação necessária à polia para que o carro tenha a velocidade correta. A parada é feita por um freio instalado na máquina de tração. O elevador possui freios de emergência no caso do freio da máquina não operar

corretamente.

O freio de segurança é instalado no carro e tem atuação mecânica. Quando acionado prende-se nas guias.[2]

2.3 MAQUINAS DE TRAÇÃO

A máquina de tração é responsável pelo movimento do elevador. Esta é ligada ao carro através dos cabos de aço. As máquinas de tração são usualmente hidráulicas ou elétricas. Um cabo conectado ao carro do elevador será tracionado através do motor, gerando assim movimento. Na hora da escolha do motor, leva-se em conta a compatibilidade com o trabalho que este realizará, a possibilidade de instalação do sistema, o custo, entre outros fatores.

2.3.1 MOTORES HIDRÁULICOS

Os motores hidráulicos de engrenagem podem ser utilizados como uma máquina de tração. Esse tipo de motor hidráulico é formado por duas engrenagens conectadas em uma caixa retangular, com tubos em lados opostos (entrada e saída). O fluido sob pressão que entra no motor faz com que as engrenagens girem em direções opostas, engrenando-se num ponto a meio caminho entre os pórticos de entrada e saída. O torque se desenvolve como resultado à resistência das engrenagens à vazão, obtendo a rotação contínua.

Para a instalação deste sistema é necessário um reservatório para o fluido que fará o acionamento do motor. A velocidade do motor muda de acordo com a vazão das válvulas e pressão do fluido. Assim, o controle de um sistema deste é feito por meio de redutores helicoidais ou planetários. Este sistema é muito robusto e pode trabalhar em ambientes com as condições mais adversas. A carga que estes equipamentos suportam é muito maior que soluções acionadas por tensão. Porém, o controle preciso de motores hidráulicos é bem mais difícil de ser implementado, além de ser uma solução mais cara e necessitar de constantes manutenções. Por conta desses fatores, motores hidráulicos não são recomendados como uma solução prática para realizar a movimentação de elevadores de tração.

2.3.2 MOTORES ELÉTRICOS

O motor elétrico é um dispositivo que transforma energia elétrica em mecânica. É um dos tipos de motores mais usados, pois combina as vantagens de utilizar energia elétrica com sua construção simples, custo reduzido, grande versatilidade de adaptação às cargas dos mais

diversos tipos. Os motores elétricos variam de acordo com o tipo de corrente que deve ser aplicada para o seu funcionamento.

2.3.2.1 MOTORES DE CORRENTE CONTÍNUA

Esses motores precisam de uma fonte de corrente contínua para o seu funcionamento. As concessionárias de energia elétrica fornecem, em sua rede, corrente alternada. Neste caso, pode ser necessário utilizar um circuito retificador para converter a corrente alternada para corrente contínua.

Eles funcionam com velocidades ajustáveis entre amplos limites e possuem controles de grande flexibilidade e precisão. Por isso ele é usado em casos especiais, desde que estas exigências compensem o custo muito mais alto da instalação da rede com corrente contínua. No caso de motores que não necessitam de uma potencia elétrica alta, podem-se utilizar fontes de corrente contínua, como pilhas e baterias.[3]

2.3.2.1 MOTORES DE CORRENTE ALTERNADA

Os motores de corrente alternada são amplamente utilizados, pois a distribuição de energia elétrica é feita normalmente em corrente alternada. Seu princípio de funcionamento é baseado no campo girante, que surge quando um sistema de correntes alternadas trifásico é aplicada em polos defasados fisicamente de 120° . Então, como as correntes são, em cada instante, um par de polos possui o campo de maior intensidade, cuja associação vetorial possui o mesmo efeito de um campo girante que se desloca ao longo do perímetro do estator. Os principais tipos de motores AC são:

- Motor síncrono: funciona com velocidade constante; utiliza-se de um campo induzido constante pré-definido, aumentando a resposta ao processo de arraste criado pelo campo girante. Utilizado, em geral, quando se necessita de velocidades estáveis sob a ação de cargas variáveis.
- Motor de indução: funciona normalmente com velocidade estável, podendo variar ligeiramente com a carga mecânica aplicada ao eixo. Devido a sua grande simplicidade, robustez e baixo custo, é o motor elétrico mais utilizado de todos, podendo ser usado para diversas aplicações. É possível controlar a velocidade deste tipo de motor com o auxílio de variadores de velocidade baseados em circuitos inversores de frequência.[3]

2.3.3 ACIONAMENTO DE MOTORES ELETRICOS

Os motores elétricos de tração dos elevadores podem ser acionados através de corrente alternada (CA) ou de corrente contínua (CC), alimentados pela energia elétrica fornecida pela rede do edifício. Nem todos os tipos de acionamento podem ser usados em elevadores. O uso de Soft-starters, por exemplo, não é recomendado pois elevadores necessitam de torque máximo a velocidade zero no instante da partida (por quê?) [ref?].

Existem quatro tipos de configurações encontradas em elevadores de passageiros.

- Motor de corrente alternada – Acionamento direto;
- Motor de corrente alternada – Duas velocidades;
- Motor de corrente alternada – Controle contínuo de velocidades;
- Motor de corrente contínua

O motor de corrente alternada com acionamento direto permite ao elevador partir da velocidade zero diretamente para a sua velocidade nominal, invertendo o processo na frenagem. Este motor foi muito utilizado no passado em elevadores de passageiros. Por não proporcionar parâmetro de conforto ou economia de energia, seu uso passou a ser descartado, se restringindo a equipamentos de transporte vertical de cargas.

Da mesma maneira que o motor CA com acionamento direto, o motor de corrente alternada de duas velocidades vai do repouso diretamente para sua velocidade nominal. Sua diferença está na parada. Antes da frenagem final, sua velocidade é reduzida em $\frac{1}{4}$ da velocidade nominal. Sua aplicação se restringe a prédios de pequeno porte com tráfego de pessoas reduzidas.

As diferentes velocidades são selecionadas pela conexão externa dos terminais de enrolamento do estator do motor a um dispositivo de partida de múltiplas velocidades, como pode-se ver na Figura 2.2.

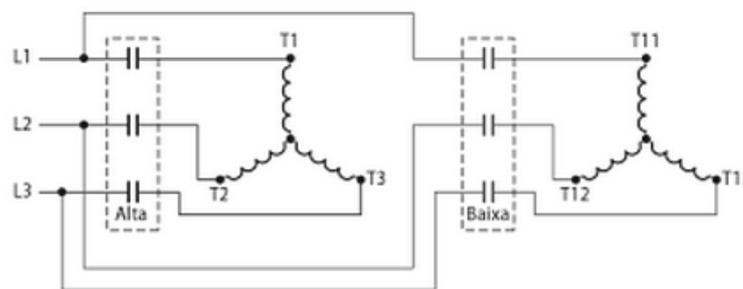


Figura 2.2: Conexões dos terminais do motor.[3]

No motor de corrente alternada com controle de velocidade, o acionamento é feito por tensões e frequências variáveis. A velocidade é controlada de acordo com o padrão desejado.

Isso permite a obtenção de acelerações e desacelerações suaves do carro. Isso evita mudanças bruscas de velocidade. Essa solução pode ser integrada aos mais modernos recursos de comando e controle de velocidade, podendo alcançar velocidades bem maiores que outros tipos de acionamento. Esta solução é a mais viável para o tipo de implementação desejada.

Os motores de corrente contínua possuem características de acionamento e frenagem similares ao motor CA com controle de velocidade. Isso é possível utilizando conversores estáticos ou motogeradores que fornecem tensão variável ao motor de tração. Este tipo de motor é uma ótima solução para elevadores, desde que a rede de energia forneça corrente contínua de média tensão.[3]

2.4 SENSORES

Os sensores são parte fundamental de diversos processos. Na indústria são utilizados na integração de equipamentos com controladores, que são utilizados para o comando e supervisão dos equipamentos e processos.

Inúmeros tipos de sensores são utilizados em equipamentos eletrônicos. Podendo ser simples chaves ou dispositivos de acionamento momentâneo do tipo mecânico ou até sensores ultrassônicos que detectam objetos através de ondas sonoras. Esses sensores servem para informar um circuito eletrônico a respeito de um evento que ocorra externamente, sobre o qual ele deva atuar, ou a partir do qual ele deva comandar uma determinada ação. Apesar de parecer um procedimento simples, muitas vezes um projeto de automação pode sucumbir devido à escolha errônea de um sensor. [4]

Sensores são utilizados em outras atividades do dia-a-dia, como em postes que ascendem automaticamente ao anoitecer, em veículos, nos dispositivos eletrônicos. Não poderia ser diferente para o controle moderno de elevadores, que os utiliza amplamente para diversos tipos de funções, como na identificação de andares, no local exato de parada, na porta, para evitar esmagamentos, na detecção de presença dentro das cabines e na detecção de sobrecarga no carro..

A escolha dos sensores utilizados nos elevadores tem que levar em conta sua função, o ambiente em que ele estará, a frequência de uso e a confiabilidade requerida. Por isso, deve-se estudar os tipos diferentes de sensores para fazer a escolha correta. [4]

2.4.1 SENSOR INDUTIVO

Os sensores de proximidade indutivos funcionam gerando um campo eletromagnético a sua frente. Estes sensores executam uma comutação eletrônica quando um objeto metálico invade seu campo eletromagnético, causando a mudança de seu estado lógico. Os principais componentes presentes em um sensor indutivo são mostrados na Figura 2.3.

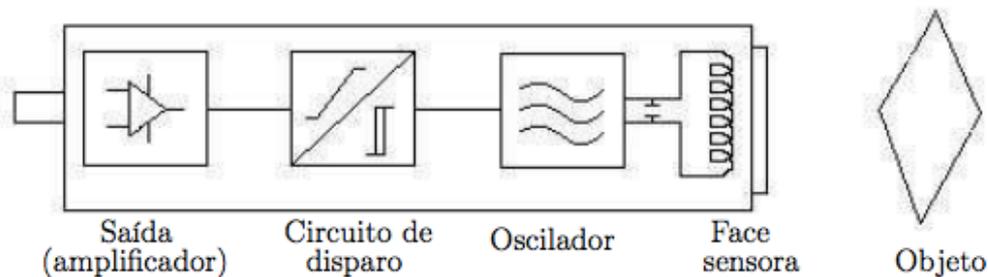


Figura 2.3: Principais componentes de um sensor indutivo.[5]

As principais vantagens dos sensores indutivos é sua grande robustez, não sendo afetados por umidade, poeira ou ambientes sujos. Além disso, não possuem partes móveis, evitando o desgaste mecânico. Sua repetitividade é bem alta, podendo ser usado para identificação de objetos que passam rapidamente. [6]

Suas desvantagens são por conta do curto alcance de detecção que este possui quando comparado a outros sensores, além disso, ele é capaz de identificar apenas objetos metálicos.

A sua instalação deve ser feita sempre observando a interferência de campos eletromagnéticos que podem atrapalhar o sinal, assim como o posicionamento de dois ou mais sensores muito próximos. [6]

2.4.2 SENSOR CAPACITIVO

Este sensor detecta a aproximação de materiais orgânicos, plásticos, pós, líquidos, madeiras, papéis, metais etc. Eles funcionam gerando um campo eletrostático criado por um oscilador controlado por capacitor e detectando mudanças neste campo causadas por um alvo que se aproxima da face ativa. Os principais componentes de um sensor capacitivo são mostrados na figura 2.4. [6]

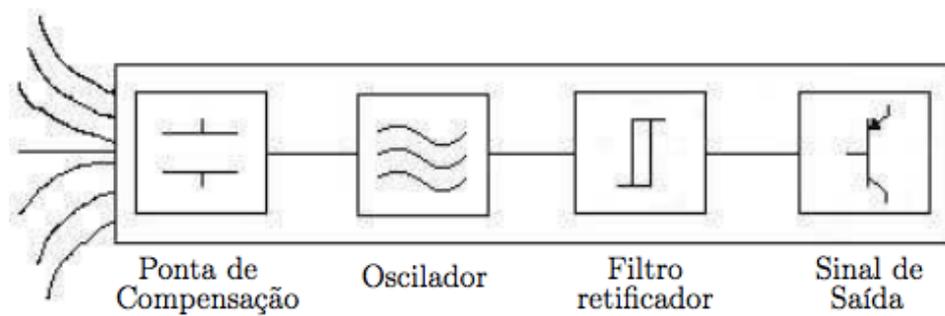


Figura 2.4: Principais componentes de um sensor capacitivo.[5]

Os sensores capacitivos são mais versáteis, podendo detectar metais e não metais, líquidos ou sólidos e podem detectar o material através de outros materiais não condutores, por exemplo se há líquido em uma garrafa plástica. Ele possui uma vida útil longa e apresenta muitas possibilidades para montagem. [6]

Suas desvantagens são a variação de distância de sensoriamento de acordo com o material, além de ser muito sensível a fatores ambientais e não apresentar uma boa seletividade de alvo. [6]

Assim como nos sensores indutivos, estes sofrem com interferências eletromagnéticas. Caso existam linhas de alta tensão perto do local por onde passa o cabo do sensor de proximidade, deve-se protegê-lo passando-o através de um duto metálico para evitar interferências.

2.4.3 SENSOR ULTRASSONICO

Um sensor ultrassônico usa o som de alta frequência para medir a distância de um objeto. A distância ao objeto é proporcional ao tempo de retorno. O tipo mais comum de dispositivo de excitação que pode gerar ondas ultrassônicas é um transdutor piezoelétrico. O princípio de funcionamento é baseado na transmissão e na reflexão de ondas, de modo a calcular a distância do objeto que refletiu a mesma. A figura 2.5 exemplifica este funcionamento.[7]

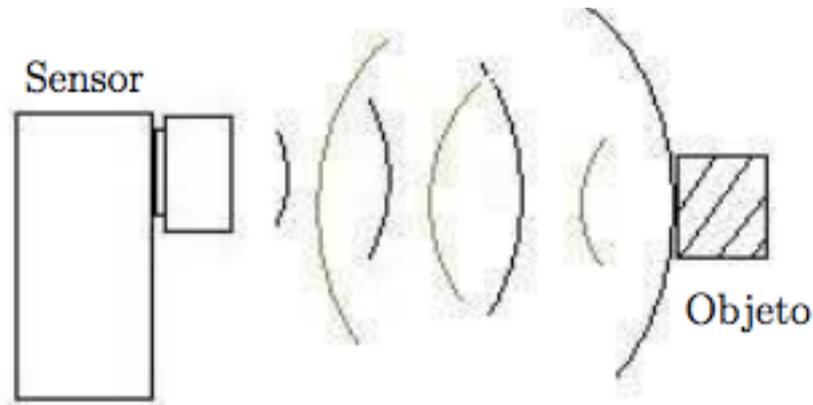


Figura 2.5: Princípio de funcionamento de sensor ultrassônico.[5]

O sensores ultrassônicos são capazes de detectar materiais sólidos, fluidos, grãos, têxteis, espumas e lã também são detectáveis, porém, com uma faixa de trabalho menor.[6]

É importante observar o alinhamento dos alvos, este deve estar perpendicular sensor, com um ângulo de incidência dentro de ± 3 graus para superfícies lisas.

As vantagens de se usar o sensor ultrassônico são:

- Capacidade de detecção de objetos à distância determinada de até 15 metros;
- Pode utilizar a saída da distância de objeto de forma digital ou analógica;
- Funcionamento constante sem manutenção;
- Não depende da cor da superfície e pode monitorar objetos transparentes, plásticos ou metálicos sem ajustes;
- Excelente repetitividade e precisão;
- Técnica mais indicada para monitoração de nível e de movimentos lineares.

Este sensor apresenta algumas desvantagens que devem ser observadas na hora de sua escolha:

- Devem ser orientados de forma perpendicular ao alvo;
- A superfícies do alvo tem que apresentar uma área mínima que varia em cada sensor;
- Possibilidade de interferências de ruídos sonoros de outras máquinas.
- Apresenta uma distancia mínima de detecção;
- Mudanças do meio ambiente como de temperatura, umidade, turbulência do ar ou partículas no ar, afetam a resposta destes dispositivos;
- Não opera no vácuo ou em ambientes de alta pressão.

2.4.4 SENSOR FOTOELÉTRICO

O sensor fotoelétrico, conhecido também por sensor óptico, baseia-se na transmissão e recepção de luz, podendo ser refletida ou interrompida pelo objeto a ser detectado. Este tipo de sensor é composto por dois circuitos básicos: um transmissor (LED – diodo emissor de luz), responsável pela emissão do feixe de luz, e o receptor (fototransistor ou fotodiodo), responsável pela recepção do feixe de luz. Os componentes de um sensor fotoelétrico são descritos na Figura 2.6.

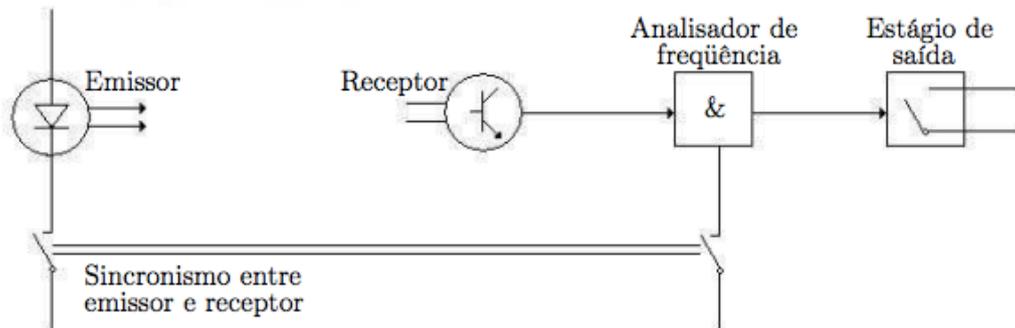


Figura 2.6: Principais componentes de um sensor fotoelétrico.[5]

As vantagens e desvantagens gerais desse tipo de sensor são:

- Vantagens:
 - Longa distância de detecção;
 - Resposta rápida;
 - Resolução elevada;
 - Luz pode ser enviada ou recebida através de fibra ótica. Sendo recomendadas para lugares com limitação de espaço;
 - Não recebe influência magnética.
- Desvantagens:
 - Vulneráveis a óleo e a poeira nas lentes;
 - Luz ambiente ou feixes de luz podem interferir no sensor;
 - Ponto de operação depende da superfície, cor, intensidade da luz e contraste ótico dos materiais.

2.4.5 SENSOR DE CONTATO COM ACIONAMENTO MECÂNICO

Este tipo de sensor é uma chave elétrica acionada mecanicamente pelo movimento dos elementos de trabalho, através do dispositivo atuador localizado no sensor.

Logo se vê que é necessário o contato físico entre o objeto a ser detectado e o atuador do sensor. Por isso, deve-se verificar se a força que o objeto poderá efetuar sobre o atuador é suficiente para acioná-lo e esse será um dos fatores importantes na escolha da chave correta.

Alguns sensores de contato possuem acionamento nas duas direções de operação, como mostrado na Figura 2.7. Existem sensores em que o acionamento é feito através de rolete escamoteável e ocorre somente em um sentido do movimento. A escolha do tipo de rolete vai depender da operação desejada.

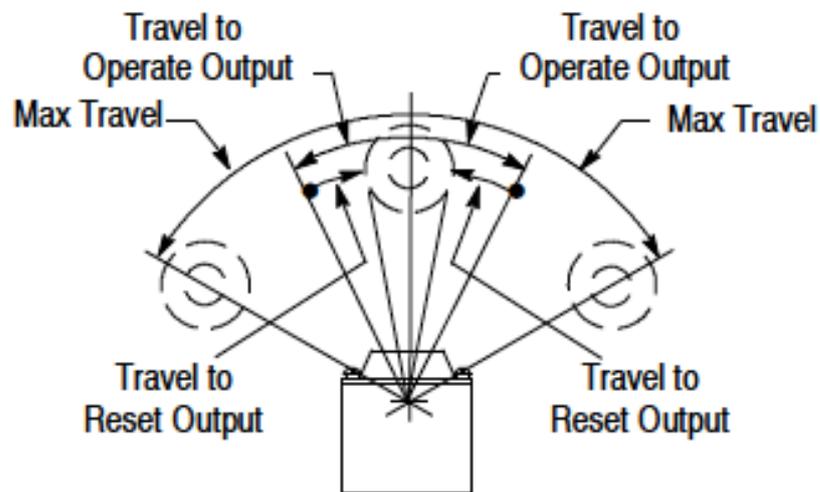


Figura 2.7: Posições possíveis de um sensor de contato.[8]

Uma das grandes vantagens das chaves fim-de-curso é o seu baixo custo por ser uma operação simples. Além disso, este sensor pode ser selado e utilizado em ambientes de condições extremas. Sua parte elétrica também possui diversas vantagens, como a imunidade a ruídos de interferências elétricas e eletromagnéticas, não apresenta corrente de vazamento e tem quedas de tensão muito baixas [4]. Por isso o seu uso é corriqueiro, mas deve-se considerar o número de operações por hora a que será submetida. Como a chave possui as partes mecânicas e o contato elétrico que se desgastam, deve-se calcular a sua vida útil de modo a evitar trocas constantes. Atualmente, as chaves fim-de-curso vêm sendo utilizadas em locais de baixo número de operações bem como em aplicações de segurança, em elevadores, por exemplo.

2.5 CONTROLADOR DE VELOCIDADE

Por muito tempo, motores CA eram usados apenas em aplicações de velocidade constante. Quando eram utilizados motores CA em algumas aplicações, como bombas de água, o controle de vazão era realizado por meio de estrangulamento de válvulas. Se a velocidade variável era realmente uma necessidade, motores CC ou de combustão interna eram escolhidos.

A criação de acionamentos de velocidade variável mudou este procedimento. À medida que eles foram evoluindo e se tornando mais confiáveis e com custo menor, motores CA com controle de velocidade passaram a ser utilizados em aplicações que antes eram inviáveis.

Como visto no tópico 2.3.3(acionamento de motores elétricos), muitos elevadores modernos utilizam motores CA com acionamento feito por variador de frequência.

O inversor de frequência altera a frequência da rede que alimenta o motor. Como a velocidade do motor está diretamente ligada à frequência da rede, alterando a frequência da rede de alimentação, é possível facilmente alterar a velocidade de rotação do motor de modo muito eficiente. Isso pode ser provado através da equação apresentada na figura 2.8.

$$n = \frac{120 \times f \times (1 - s)}{p}$$

onde:

n = velocidade em rotações por minuto (rpm)

f = frequência da rede em Hertz (Hz)

s = escorregamento

p = número de pólos

Figura 2.8: Equação da velocidade nominal de um motor trifásico.[9]

O uso de controles de frequência ajustável influencia no projeto e no desempenho dos motores CA, levando a efeitos positivos e negativos. Velocidades baixas diminuem a fadiga dos rolamentos e outros elementos girantes. A partida suave diminui os esforços nos elementos do motor.

Em geral, a vida útil do motor é aumentada ao utilizar controle de velocidade. Porém, devem-se considerar alguns fatores ao utilizar este tipo de acionamento. A operação do inversor de frequência pode fazer com que a tensão nominal do motor seja dobrada. O efeito

das harmônicas pode afetar a operação, o desempenho e a vida útil do motor ao gerar aquecimento extra.

Esses fatores negativos podem ser antecipados na hora da escolha do motor a ser controlado, escolhendo um motor com isolamento suficiente para suportar o dobro da tensão e verificar as harmônicas geradas pelo acionador para que o motor escolhido seja capaz de dissipar o calor extra gerado.[9]

2.6 CONTROLADOR LÓGICO PROGRAMÁVEL (CLP)

Antes do surgimento de controladores, os processos automatizados eram projetados com contatos programáveis. As instruções eram codificadas por meio das ligações elétricas dos contatos. A principal função era a seleção de operações a serem realizadas por meio da troca das ligações. Um exemplo deste tipo de painel pode ser visto na Figura 2.9.

Assim, este processo possuía uma operacionalidade muito baixa, além de alto consumo energético, difícil manutenção, dificuldade em fazer mudanças de comandos, por conta da necessidade de mudanças na fiação. Muitas vezes o processo de mudança demorava horas, tempo precioso para a indústria.

A indústria que tinha mais perdas nesse processo era a automobilística. Por conta de lançamentos de novos modelos a cada ano, muitos painéis de controle eram descartados pois os custos de alteração eram maiores que a instalação de novos painéis.

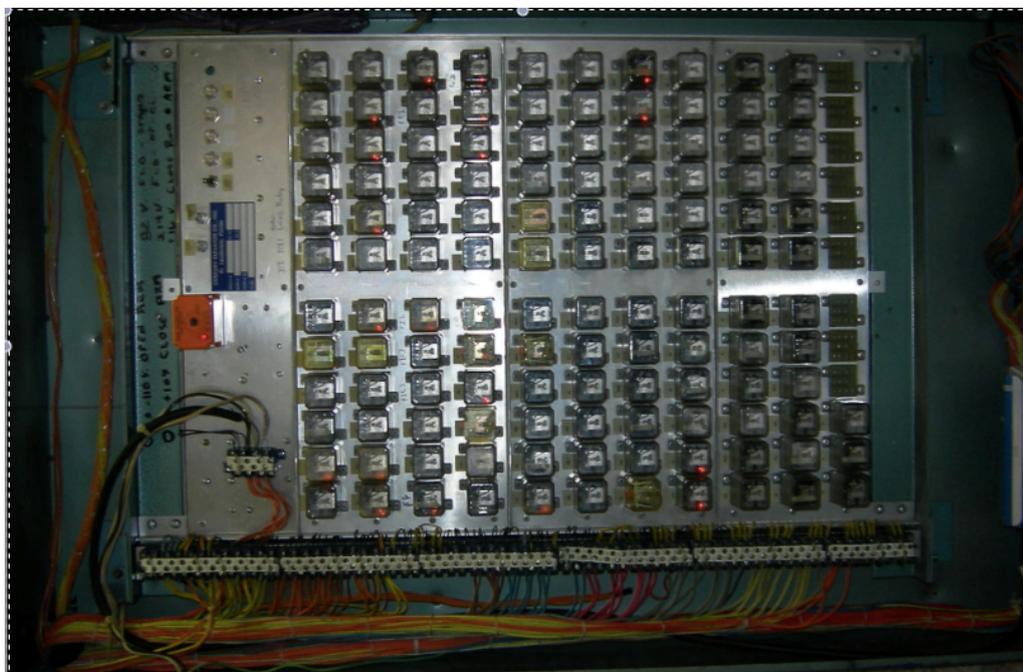


Figura 2.9: Painel de controle de elevador baseado em relés.[10]

Pensando nisso, a GM começou a trabalhar na ideia de um controlador programável. Eles identificaram a necessidade da indústria por algo mais elaborado, que pudesse fazer mudanças nos processos em pouco tempo. Em 1969 foi instalado o primeiro controlador na GM com apenas a função de intertravamento.

Ao longo dos anos, vários avanços foram feitos, acrescentando funções de sequenciamento, temporização, contagem, funções aritméticas. Hoje, por conta dos enormes avanços tecnológicos, o Controlador Programável evoluiu para um controlador universal de processos.

Sua versatilidade e custo faz com que ele seja usado em diversas atividades. O nome utilizado para esses controladores atualmente é controlador lógico programável (CLP). [9]

O CLP é um equipamento eletrônico utilizado em sistemas de automação flexível. Eles são usados em diversos ambientes, desde automação residencial a aplicações industriais. Este equipamento consiste em um processador, memória, uma linha de comunicação e portas de saídas e entradas para enviar comandos e receber sinais. O CLP lê os sinais de entrada e o processador, de acordo com a programação na memória, implementa funções específicas tais como, lógica, sequenciamento, temporização e aritmética para controlar através das saídas, diversos tipos de máquinas e processos. A visão geral do processo de funcionamento de um CLP é descrito na figura 2.10.

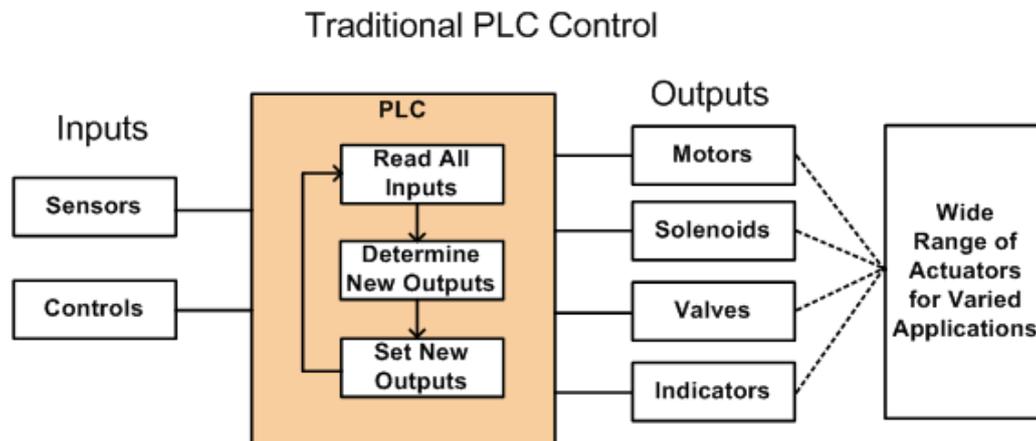


Figura 2.10: Visão geral de controle utilizando CLPs.[11]

2.6.1 LINGUAGENS DE PROGRAMAÇÃO DE CLPS

A norma IEC61131-3 (1993) define como padrão o uso de cinco linguagens. Duas delas, Lista de instruções (IL) e Texto estruturado (ST), são linguagens textuais e as outras três, Diagrama Ladder (LD), Diagrama de blocos funcionais (FDB) e gráfico de funções sequenciais (SFC – *Sequential Function Chart*), são linguagens gráficas. Vale ressaltar que

nem todos os controladores possuem os cinco tipos de linguagem. Em geral, os mais simples possuem pelo menos Lista de instruções e Diagrama Ladder.

- **Lista de instruções (IL)** : Consiste em uma série de instruções, sendo uma instrução por linha. Cada instrução vai se referir a um ou mais operadores. É uma linguagem muito próxima à linguagem de máquina, como pode ser visto na figura 2.11, sendo eficiente na resolução de pequenos problemas. Por ser uma linguagem de baixo nível, ela é utilizada para otimização de programas.

```
0001 STR      X001
0002 OR       X002
0003 AND NOT  X003
0004 AND NOT  C020
0005 OUT      C001
0006 STR      C001
0007 TMR V001 V002
0008 OUT      C050
```

Figura 2.11: Exemplo de código IL.[9]

- **Texto estruturado (ST)**: Trata-se de uma serie de instruções que possuem algumas funções comparadoras e periódicas, como IF/ELSE e WHILE. Se assemelha a linguagens de alto nível com C/C++, como pode ser observado na figura 2.12. Em geral, é uma linguagem mais fácil de programar que a Lista de instruções e sua semelhança com linguagens populares de alto nível torna ela muito utilizada.

```
0001
0002 IF LSH THEN
0003     BOBINA := TRUE;
0004 ELSE
0005     BOBINA :=FALSE;
0006 END_IF
0007
```

Figura 2.12: Exemplo de código ST.[9]

- **Diagrama Ladder (LD)**: O nome desta linguagem deve-se ao fato de que sua representação gráfica se assemelha a uma escada, tradução de ladder para o Português. A lógica a ser executada pelo controlador é descrita por um diagrama de interligação de contatos e bobinas organizadas em um diagrama de degraus, mostrado na figura 2.13. Esta foi a primeira linguagem de programação criada

para CLPs. Ela foi inspirada no diagrama de contatos elétricos, pois como os CLPs vieram para substituir os contatos elétricos, seria mais intuitivo para os técnicos e engenheiros entender desta maneira.

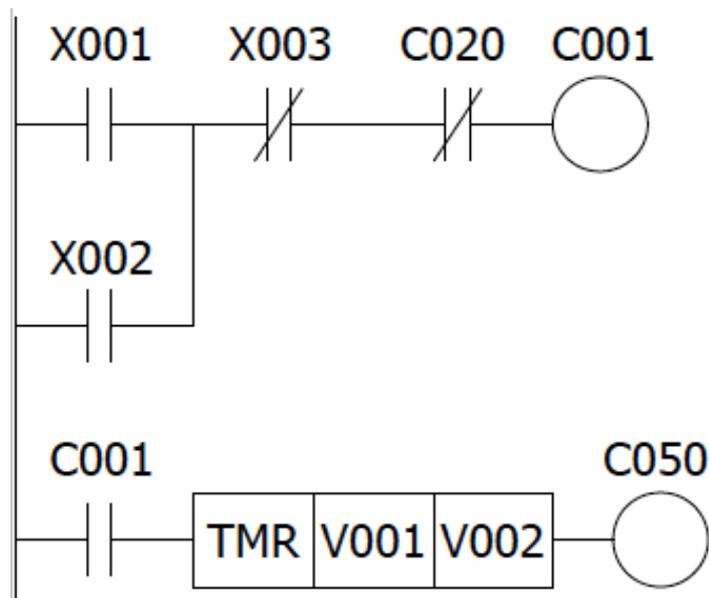


Figura 2.13: Exemplo de código LD.[9]

- **Diagrama de blocos funcionais (FDB):** É uma linguagem gráfica, cujos elementos são expressos por blocos interligados, semelhantes aos utilizados na eletrônica digital. Blocos como AND e OR são muito utilizados, mostrados na Figura 2.14. A partir de blocos mais simples é possível construir blocos complexos para funções muito utilizadas em determinado contexto.

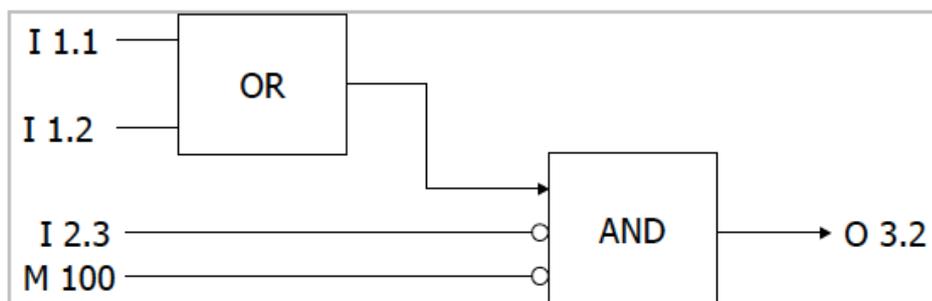


Figura 2.14: Exemplo de código FDB.[9]

- **Sequenciamento gráfico de funções(SFC):** é uma linguagem orientada graficamente que torna possível descrever a ordem cronológica de diferentes ações dentro de um programa. As ações, então, são distribuídas dentro de elementos, com transições entre eles, como pode ser visto na figura 2.15 Esta linguagem é muito versátil, pois dentro de cada elemento é possível programar utilizando as outras linguagens (IL, ST, LD, FDB) .

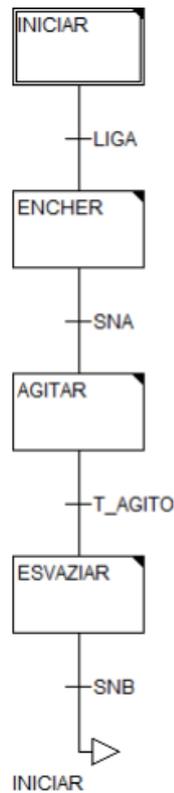


Figura 2.15: Exemplo de código SFC.[9]

2.7 REDES INDUSTRIAIS

Com o aumento da informatização da indústria e de seus processos e com a utilização da automação para criar soluções eficientes e baratas para estes ambientes, existe a necessidade de se conectarem todos os equipamentos presentes em um processo e permitir que eles se comuniquem entre si, trocando informações importantes para o melhor funcionamento possível.[12]

As redes industriais foram desenvolvidas para garantir essas características e elas possuem protocolos que podem ser visualizados como o idioma que os equipamentos utilizam entre si, permitindo que um seja capaz de entender as informações enviadas e recebidas de outros. É importante que haja uma padronização destes protocolos de modo que quaisquer tipos de equipamentos possam ser utilizados em um sistema.[13]

Algumas das mais importantes redes industriais que existem são:

- MODBUS;
- PROFIBUS;
- DeviceNet.

2.7.1 MODBUS

Desenvolvido pela Modicon em 1979, é uma estrutura de mensagem utilizada para estabelecer comunicação entre dispositivos mestre-escravo ou cliente-servidor.

A comunicação acontece quando o mestre inicia as transações (chamadas de *queries*) e os outros dispositivos (escravos) respondem de acordo com o pedido ou de acordo com a tarefa em questão.[12]

Neste tipo de comunicação, o mestre deve esperar um tempo suficientemente longo por uma resposta de qualquer escravo antes de abortar a transação. Caso ocorra um erro de transmissão, o escravo não formará a resposta e será necessário que o mestre tome providencias para solucionar este problema.[12]

A verificação de erro ocorre opcionalmente por meio da paridade de cada byte transmitido e obrigatoriamente pelos métodos LRC ou CRC. O LRC descarta os caracteres de início e fim da mensagem. O CRC descarta os bits de início, paridade e parada de cada byte.[13]

Existem três tipos de protocolo MODBUS, descritos abaixo e mostrados na figura 2.16:

- MODBUS TCP/IP: utilizado para sistemas de supervisão e controladores lógicos programáveis. O MODBUS é encapsulado em um protocolo TCP/IP e transmitidos através de redes Ethernet.
- MODBUS PLUS: utilizado para comunicação geralmente entre controladores lógicos programáveis, módulos de E/S, chaves de partida eletrônica de motores e interfaces homem máquina. Usa portas RS-485 com taxa de transmissão de 1 Mbps.
- MODBUS Padrão: utilizado para comunicação do controlador lógico programável com dispositivos de entrada e saída de dados, relés de proteção, controladores de processos, atuadores de válvulas e instrumentos eletrônicos inteligentes em geral. Usa RS-232 ou RS-485.

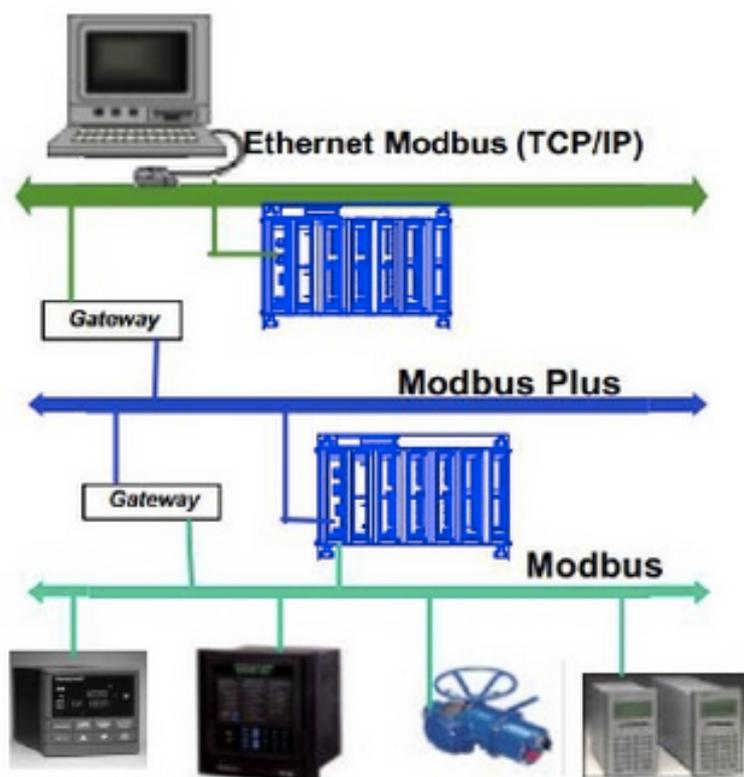


Figura 2.16: Tipos de protocolo MODBUS.[12]

2.7.2 PROFIBUS

Desenvolvido na Alemanha inicialmente pela Siemens em conjunto com a Bosch e a Klockner-Moeller em 1987, o PROFIBUS (Process Field Bus) é a proposta alemã para a padronização dos protocolos de redes industriais. Este protocolo utiliza o modelo de referência ISO/OSI reduzidos às camadas 1, 2 e 7 (física, de enlace e de aplicação) para atender aos requisitos de tempo de resposta.[13]

É uma rede de campo aberta, que independe dos fabricantes e está ao alcance de uma larga variedade de aplicações. A comunicação entre dispositivos de diferentes fabricantes acontece sem ajustes especiais e pode ser utilizada em tarefas que requerem comunicação em tempo real, alta velocidade e complexidade.[12]

Existem três perfis de comunicação diferentes:

- PROFIBUS – DP (Distributed Process): perfil mais utilizado, otimizado para velocidade, eficiência, baixos custos de ligação e pode ser utilizada para comunicação entre sistemas de automação e periféricos distribuídos.

- PROFIBUS – PA (Process Application): perfil que permite conectar sensores e atuadores em barramentos comuns e em áreas intrinsecamente seguras. Pode ser utilizado com tecnologia 2 fios.
- PROFIBUS – FMS (Fieldbus Message Especification): perfil que abre um amplo alcance de aplicações com grande flexibilidade e pode ser utilizado para tarefas de comunicação extensas e complexas.

2.7.3 DEVICENET

Apresentado em 1994, DeviceNet é uma implementação do protocolo *Common Industrial Protocol* (CIP) para redes de comunicação industrial. Desenvolvido originalmente pela Allen-Bradley, teve sua tecnologia transferida para a ODVA que, desde então, mantém, divulga e promove o DeviceNet e outras redes baseadas no protocolo CIP (DeviceNet, Ethernet/IP e ControlNet por exemplo). Além disso, utiliza o protocolo *Controller Area Network* (CAN) para enlace de dados e acesso ao meio (camadas 2 e 1 do modelo OSI/ISO).[12]

Utilizado principalmente na interligação de controladores industriais e dispositivos entra/saída (I/O), o protocolo segue o modelo produtor-consumidor, suporta múltiplos modos de comunicação e possui prioridade entre mensagens.[12]

É um sistema que pode ser configurado para operar tanto numa arquitetura mestre-escravo quanto numa arquitetura distribuída ponto a ponto. Além disso, define dois tipos de mensagens: I/O (dados de processo) e *explicit* (configuração e parametrização). Possui também mecanismos de detecção de endereços duplicados e isolamento dos nós em caso de falhas críticas.[12]

Este tipo de rede possui como principais características:

- Alta velocidade;
- Comunicação em nível de byte;
- Comunicação com equipamentos discretos e analógicos;
- Alto poder de diagnóstico dos dispositivos da rede.

Uma rede DeviceNet pode conter até 64 dispositivos, endereçados de 0 a 63. Qualquer um destes pode ser utilizado. Não há qualquer restrição, embora se deva evitar o 63, pois este costuma ser utilizado para fins de comissionamento. Pode-se ver uma rede DeviceNet com diversos dispositivos conectados na figura 2.17.[13]

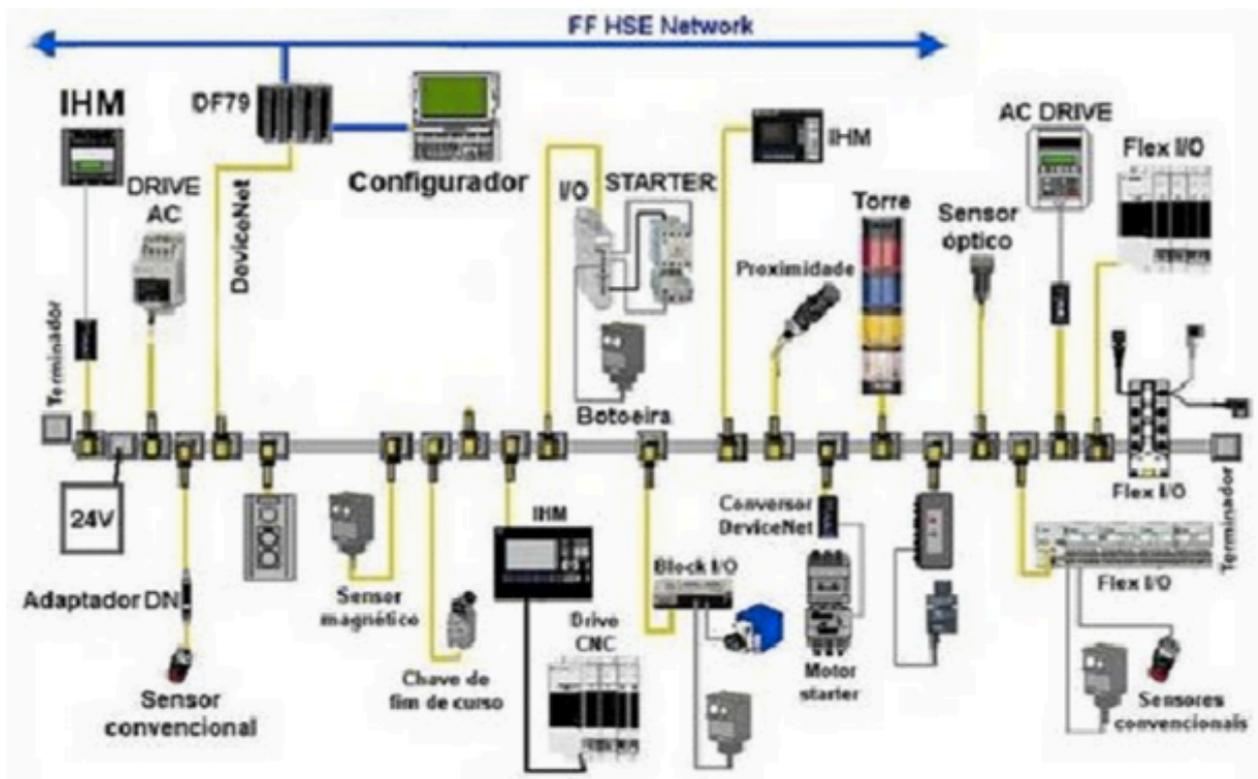


Figura 2.17: Exemplo de rede DeviceNet.[12]

2.8 ALGORITMOS DE CONTROLE

Assim como todas as indústrias, a indústria de elevadores está enfrentando dois grandes desafios: (a) a contínua pressão para que os custos de construção e instalação em edifícios sejam reduzidos; (b) o aumento da competitividade entre as empresas e a importância da diminuição do tempo de espera requer diversificação dos serviços e das estratégias de movimentação dos carros apresentados aos passageiros [14].

Portanto, pode-se observar que a eficiência, a produtividade e a segurança são aspectos primordiais que devem ser levados em conta quando se vai realizar ou modificar qualquer projeto.[15]

A maioria dos componentes físicos utilizados para movimentação de elevadores são invisíveis aos olhos dos passageiros e podem ser mudados a qualquer momento sem que os mesmos percebam tais mudanças. Porém, os serviços prestados pelos elevadores e a qualidade dos mesmos determinam como os passageiros visualizam e avaliam aqueles elevadores [14]. Atualmente, a maioria dos prédios possui tecnologias de comunicação distribuídas, com o intuito de centralizar todas as informações colhidas pelo sistema. Com isso, surgiu a possibilidade de implementar sistemas de controle grupos de elevadores. [15]

O objetivo principal do controle de grupos de elevadores é o de melhorar o desempenho do sistema, aumentando assim sua capacidade de transporte, reduzindo o consumo de energia e reduzindo o tempo de espera do passageiro. Este objetivo baseia-se na ideia de distribuir a movimentação e alocação de quais elevadores irão atender a quais chamadas, de modo a melhor atender a todos, diminuindo assim o tempo de espera, que consiste não só do tempo aguardando o elevador chegar, mas também o de ser entregue ao destino desejado. [16]

Em relação à alocação dos elevadores, existem dois métodos que são frequentemente utilizados para o atendimento das chamadas: alocação contínua e alocação imediata. A alocação imediata consiste em associar um elevador a determinada chamada, imediatamente depois desta ser registrada, sendo o passageiro notificado para qual elevador deve se dirigir. Desta maneira, o elevador associado àquela chamada fica obrigado a cumprir aquele trajeto, independentemente de acontecimento e chamadas futuras.[16]

Na alocação contínua, cada elevador é associado a apenas um comando por vez, e sempre ao chamado que possui maior prioridade ou menor custo de atendimento. Dessa maneira, o passageiro não saberá de imediato qual será o elevador que irá atendê-lo e deve aguardar um sinal luminoso indicar qual elevador ficou associado ao seu chamado. [16]

Antes de expor os tipos de algoritmos normalmente utilizados no controle de grupo de elevadores, é importante introduzir o conceito de padrão de chegada dos passageiros.

Dependendo do horário do dia, haverá maior tráfego de elevadores se movimentando de baixo para cima, normalmente pela manhã, no horário de chegada dos trabalhadores em um prédio comercial, sendo este chamado de *Up-peak*. Em algum outro horário ocorrerá o inverso, normalmente no final da tarde, quando os trabalhadores estão deixando o edifício, conhecido como *Down-peak* e durante a maior parte do tempo, a movimentação dos elevadores será entre andares do prédio, trabalhadores indo entre andares do prédio a trabalho, conhecido como *Interfloor* [17]. Estes padrões estão exemplificados na figura 2.18.

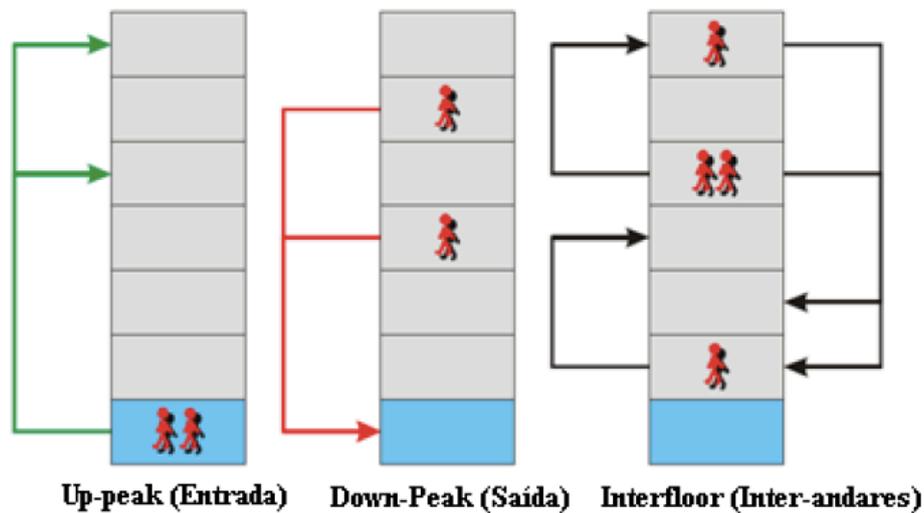


Figura 2.18: Fluxo de passageiros em um prédio. [15]

Ao contrário do que se pode imaginar, o *up-peak* não é simplesmente equivalente ao *down-peak*, pois o primeiro possui um único lugar de chegada e vários de saída dos passageiros enquanto que o segundo possui vários locais de chegada e um único de saída. Realizando uma análise dos dois, pode-se perceber que a complexidade do primeiro se torna muito maior e seu tempo de espera também, quando o tráfego de passageiros for elevado, pois com vários locais de parada, novos passageiros no local de entrada terão um tempo de espera elevado, enquanto que no segundo, a situação se mantém semelhante com tráfego alto ou baixo de passageiros. [17]

A otimização de algoritmos de controle vem sendo bastante estudada e desenvolvida nos últimos anos e algumas soluções foram desenvolvidas, baseadas em diversas técnicas, tais como: métodos heurísticos, lógica nebulosa e algoritmos genéticos. [16]

2.8.1 ALGORITMOS HEURÍSTICOS

Algoritmos heurísticos são aqueles que se baseiam em experiências prévias ou o “senso comum” para criar estratégias para solução dos problemas, ou seja, regras heurísticas associam conclusões e condições por meio de implicações lógicas como SE, ENTAO e etc. [15]

É importante observar que algoritmos heurísticos não produzem soluções ótimas para os problemas apresentados, porém, permitem grande redução de custo e tempo. Além disso, como este tipo de algoritmo não é baseado em modelos matemáticos nem são necessariamente restringidos pelos mesmos, custam menos tempo para serem desenvolvidos quando

comparados a modelos que são muito complexos, onde existe a presença de parâmetros variantes e/ou incertos. [15]

Alguns algoritmos heurísticos já conhecidos para controle de elevadores são os seguintes:

- Abordagem por zona;
- Abordagem baseado em pesquisa;

A abordagem por zona, desenvolvida pela companhia de elevadores Otis, tenta separar o edifício em zonas aproximadamente iguais em tamanho e aloca um elevador para atender cada uma destas zonas, com o objetivo de diminuir o tempo de espera entre a chamada e a chegada dos elevadores. Além disso, existe o conceito de controle de área, que consiste em atribuir a um elevador que já esteja a caminho de determinada zona atender uma chamada dentro da mesma, de modo a evitar que dois carros atendam ao mesmo chamado. [17]

A abordagem baseada em pesquisa, registra a chamada de algum passageiro e pesquisa dentre os possíveis carros, qual possui determinada característica otimizada em relação aos outros, como tempo médio de espera. Existem algoritmos baseados neste conceito que realiza uma média ponderada de determinadas características como tempo de espera, tempo de viagem, numero de passageiros para encontrar qual o melhor carro para atender ao chamado. [17]

2.8.2 ALGORITMOS GENÉTICOS

Algoritmos genéticos são algoritmos de otimização, baseados na teoria da seleção natural de Darwin e nas teorias da genética. Estes tipos de algoritmos utilizam estratégias de busca que são paralelas e estruturadas, mas que também possuem características de aleatoriedade no processo. [18]

Mesmo possuindo características de aleatoriedade, estes algoritmos caminham em busca de características que melhorem a execução do sistema como um todo por meio da avaliação de determinadas características a cada iteração, ou seja, a cada geração. [18]

O princípio de funcionamento do algoritmo genético é que através do critério de seleção, após várias gerações, o conjunto inicial gere indivíduos mais aptos. A maioria dos métodos de seleção é projetada para escolher aqueles indivíduos que possuem maior nota de aptidão, porém não excluem aqueles menos aptos, de modo a manter a diversidade da população. [18]

Para que se consigam gerar gerações novas, é necessária a criação de operadores, onde é esperado que estes causem uma melhora na adaptação da população ao longo do tempo. Os operadores são: cruzamento e mutação. Estes permitem que a nova geração seja completamente nova, mas garantindo também que elas possuam características dos seus pais. Para evitar que os melhores indivíduos sumam por completo da população, eles podem ser automaticamente colocados na próxima geração através da reprodução elitista. [18]

2.8.3 LÓGICA NEBULOSA(FUZZY)

A lógica utilizada normalmente para resolver problemas, não é muito compatível com a realidade, pois ela se baseia em conceitos booleanos, ou seja, em valores que são verdadeiros ou não. Considerando os problemas reais aplicados a indústrias, este tipo de abordagem não se mostra capaz de suportar as incertezas presentes em alguns aspectos. [19]

A lógica nebulosa, surgiu a partir do princípio da incerteza, em que esta forma de raciocínio se aproxima bastante da forma do pensamento humano, baseado em aproximações e cercado de incertezas e suposições. Este tipo de lógica busca atribuir graus de possibilidades para os elementos sendo abordados, de modo a evitar somente os valores *contido* ou *não contido*, por não satisfazerem o princípio da incerteza. [19]

Quando comparamos a lógica fuzzy com a lógica booleana, vemos que a primeira permite respostas no meio termo dos extremos, apresentando ainda a possibilidade de medir o grau de aproximação da solução correta e inferir algo que seja necessário. [14]

Ao aplicar o princípio da lógica nebulosa para controle de grupo de elevadores, ve-se que as variáveis dessa lógica possuirão uma granulosidade menor, representando o tráfego de elevadores como alta, médio ou baixa. Depois pode-se aplicar as regras desta lógica para determinar qual padrão de tráfego está predominante em determinado momento e, assim, inferir qual controle deve ser implementado de modo a otimizar o desempenho do sistema. [14]

Este tipo de lógica consiste basicamente na realização de três operações, sendo estas mostradas na figura 2.19.

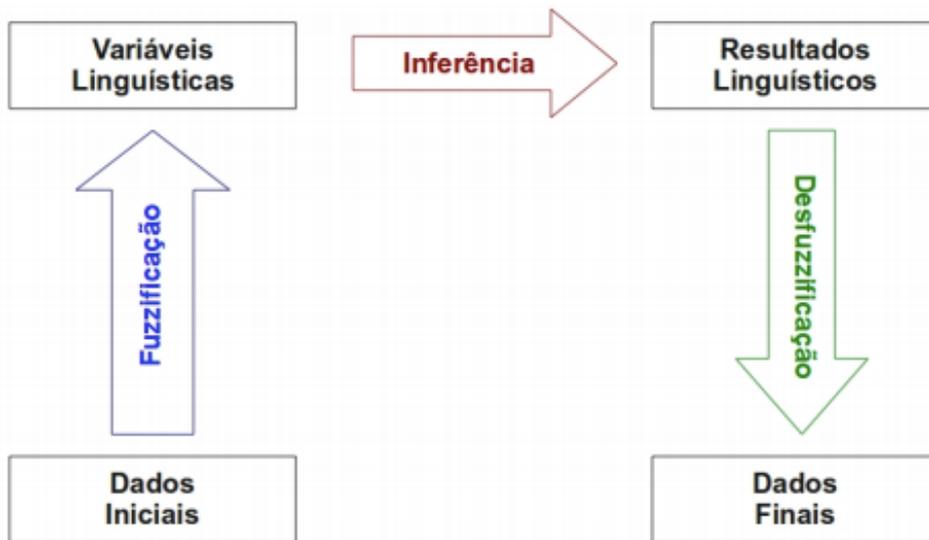


Figura 2.19: Lógica fuzzy e suas operações. [19]

Na etapa de fuzzificação, o problema é analisado e os dados são transformados em variáveis linguísticas, como pode ser visto na figura 2.20. No caso do problema de controle de elevadores, pode-se por exemplo analisar o padrão de tráfego dos mesmos e definir variáveis como tráfego baixo, tráfego médio e tráfego alto. [19]

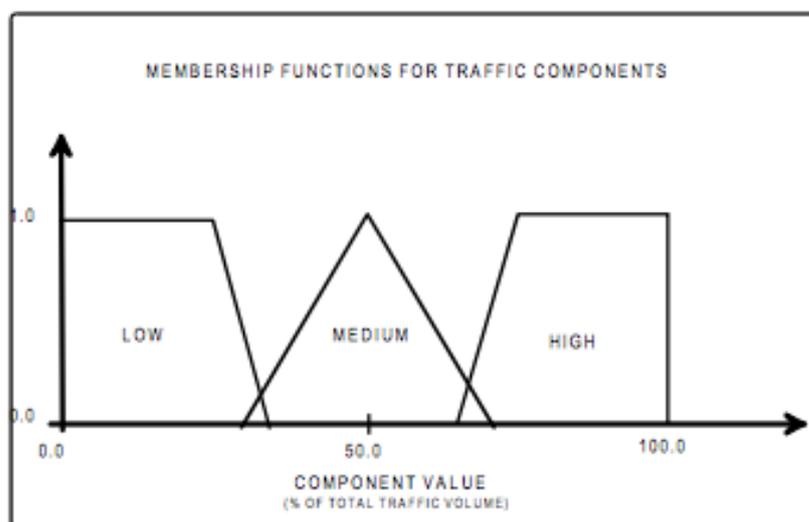


Figura 2.20: Fuzzificação de variáveis. [20]

Na segunda etapa, a de inferência, utilizam-se as variáveis criadas na primeira etapa para criar regras ou proposições da lógica através da análise das variáveis em conjunto. Além disso, esta etapa pode ser dividida em duas partes: a primeira conhecida como agregação, define a validade de uma regra e a segunda, composição, define o resultado obtido por meio de uma inferência. [19]

Pode-se por exemplo criar uma regra para controle de elevadores analisando os padrões de tráfego apresentados anteriormente. Caso o tráfego de *up-peak* seja alto, o de *down-peak* seja baixo e o *interfloor* também seja baixo, pode-se então criar uma regra chamada de *high up-peak* que resulta em uma lógica voltada a otimizar o controle de elevadores a transportar passageiros de baixo para cima do prédio. [19]

Por fim, a etapa de desfuzzificação consiste na transformação dos valores fuzzy em valores reais, ou seja, aplicaremos aos valores fuzzy lógicas de controle reais a serem implementadas no sistema. [19]

2.9 TECNICAS DE MODELAGEM

No desenvolvimento de sistemas, é de grande importância utilizar-se de técnicas de modelagem para investigar vários tipos de questões e acontecimentos que podem surgir durante a simulação do mesmo. Pode-se entender o modelo de um sistema como uma simplificação da realidade, e construímos estes para:

- Compreender melhor o sistema em questão;
- Ajudar na visualização do sistema;
- Especificar estrutura e/ou comportamento do sistema;
- Documentar as decisões tomadas;
- Proporcionar análises lógicas consistentes;

A aplicação deste tipo de técnica está se tornando cada vez mais ampla e variada, atingindo áreas como: sistemas de manufatura, sistemas de saúde, sistemas envolvendo recursos naturais, sistemas de transporte, sistemas de construção civil, sistemas computacionais, dentre outros.[21]

Dependendo do tipo de aplicação, pode-se utilizar de linguagens de modelagem diferentes para obter uma análise melhor do nosso sistema. Algumas das principais técnicas de modelagem são Redes de Petri, *Communicating Sequential Process (CSP)* e *Unified Modelling Language (UML)*. [21]

2.9.1 REDES DE PETRI

As redes de Petri foram desenvolvidas em 1962 por Carl Adam Petri como sua tese de doutorado que tratava da comunicação entre autômatos. Desde o princípio, tinham como objetivo a modelagem de sistemas com componentes concorrentes.[22]

A estrutura topológica das Redes de Petri é formada por três estruturas básicas: estados, transições e relações de fluxo. Pode-se entender como estados os componentes passivos do sistema, correspondendo as variáveis de estado.

Transições são consideradas os componentes ativos do sistema, correspondendo aos eventos que levam o sistema de um estado a outro. As relações de fluxo nada mais são do que a relação entre um estado e outro em relação aos eventos que ocorreram entre eles.[22]

Como a Rede de Petri é uma representação gráfica de um sistema, ela utiliza imagens pré-estabelecidas e definidas para representar cada uma das estruturas do sistema:

- Estados são representados por círculos;
- Transições são representadas por retângulos;
- Relações de fluxo são representadas por setas.

Todas estas estruturas e suas relações podem ser observadas na figura abaixo.

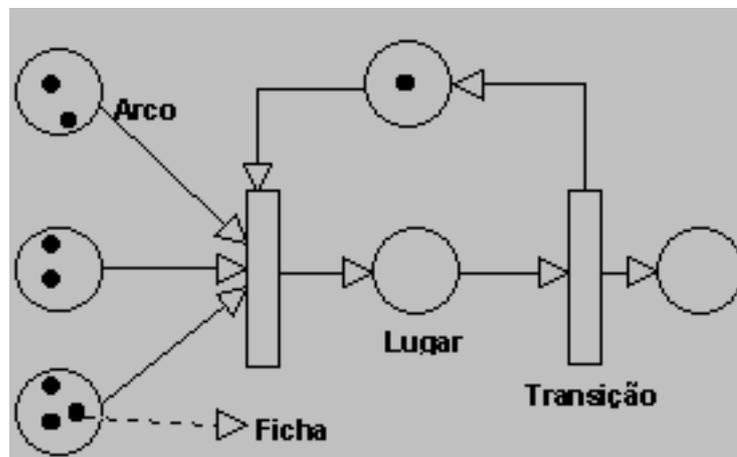


Figura 2.21: Elementos da Rede de Petri.

Na figura anterior, deve-se notar a presença de pontos, chamados de fichas, presente em alguns dos estados do sistema. Redes de Petri utilizam este conceito para definir quando uma transição esta habilitada para ocorrer, permitindo assim que a transição de um estado para outro aconteça somente quando todas as condições necessárias forem satisfeitas.[22]

2.9.2 CSP

Pode-se definir sistemas concorrentes como aqueles compostos por componentes independentes que necessitam de comunicação entre si e estes estão difundidos no nosso dia a dia em sistemas como internet, sistemas de defesa, sistemas de comunicação, redes de caixa eletrônico, etc. Desenvolvido nos anos 80 por Hoare na Universidade de Oxford, o CSP é uma linguagem capaz de descrever processos concorrentes de uma maneira mais abstrata.[23]

De modo a atender estes objetivos, o CSP utiliza uma notação matemática sucinta para descrever os processos e um alfabeto para controlar o nível de abstração destas descrições. Portanto, ela está fundamentada em dois elementos básicos: processos e eventos. Eventos são as entidades primitivas do sistema e que compõem os processos enquanto que estes são sequências de eventos e outros processos que descrevem uma ação do sistema.[23]

Processos podem ser combinados utilizando técnicas de composição como paralelismo, escolha interna e escolha externa. O conjunto de eventos que fazem parte de um processo definem um alfabeto para o mesmo, esta informação se torna de grande importância durante a composição de processos.

Além da representação matemática do CSP, ele também utiliza uma representação gráfica para visualizar os processos, esta é chamada de Diagramas de transição. Nesta representação utiliza-se círculos para representar estados e setas para representar eventos aos quais o processo reage.[23]

2.9.3 UML

Como uma linguagem utilizada para modelar sistemas orientados a objetos, sofreu grande concorrência no início de seu desenvolvimento. Entre 1989 e 1994 surgiram mais de 50 métodos diferentes para modelagem destes tipos de sistemas. Os maiores prejudicados com essa concorrência eram os usuários, que não possuíam uma metodologia padrão para desenvolvimento destes modelos.[24]

Após o ano de 1994 iniciaram-se os esforços para se unificar todas estas métodos de modelagem, com isso, ocorreu a união de três grandes criadores destas linguagens: Booch, Rumbaugh e Jacobson que, em 1996 lançaram a primeira versão da Linguagem Unificada de Modelagem (UML).[24]

O UML utiliza vários diagramas para estruturar os sistemas, divididos entre diagramas estruturais e de comportamento, que são utilizados para visualizar, especificar, construir e documentar os aspectos estáticos e dinâmicos do sistema, respectivamente.

Um diagrama nada mais é do que a representação gráfica de um conjunto de elementos como classes, interfaces, componentes e são usados para visualizar o sistema sobre diferentes perspectivas.

Alguns dos principais diagramas do UML são:

- Diagrama de Classes: mostra o conjunto de classes com seus atributos e métodos e os relacionamentos entre as classes.

- Diagrama de Objetos: descreve um conjunto de objetos e seus relacionamentos em um ponto do tempo.
- Diagrama de Componentes: indica os componentes do software e seus relacionamentos, ou seja, mostra quais tipos de arquivos fazem parte de um componente.
- Diagrama de Implantação/Utilização: organização do conjunto de elementos de um sistema para a sua execução.
- Diagrama de Uso de Casos: contém relacionamentos de dependência, generalização e associação e são basicamente usados para fazer a modelagem de visão estática.

3 MATERIAIS UTILIZADOS

3.1 ESTRUTURA MECÂNICA

A estrutura mecânica do elevador utilizada para a realização de testes está localizada no galpão do GRACO. Trata-se de uma estrutura metálica de aproximadamente 5,8 metros de altura.

Por ser usada apenas para a simulação de dois elevadores, a estrutura é simples e não contém alguns elementos fundamentais para elevadores de passageiros convencionais. Nesta estrutura estão montados dois elevadores que podem correr livremente desde a extremidade inferior até a superior através das guias. A cabine não possui porta, pois o carro não precisará subir com nenhum material. No topo da estrutura estão dois motores com tambores acoplados para tracionar os cabos de aço que sustentam os elevadores. Não é necessário o uso de contrapeso, pois a armação do carro é extremamente leve. Esta estrutura está exemplificada na Figura 3.1.

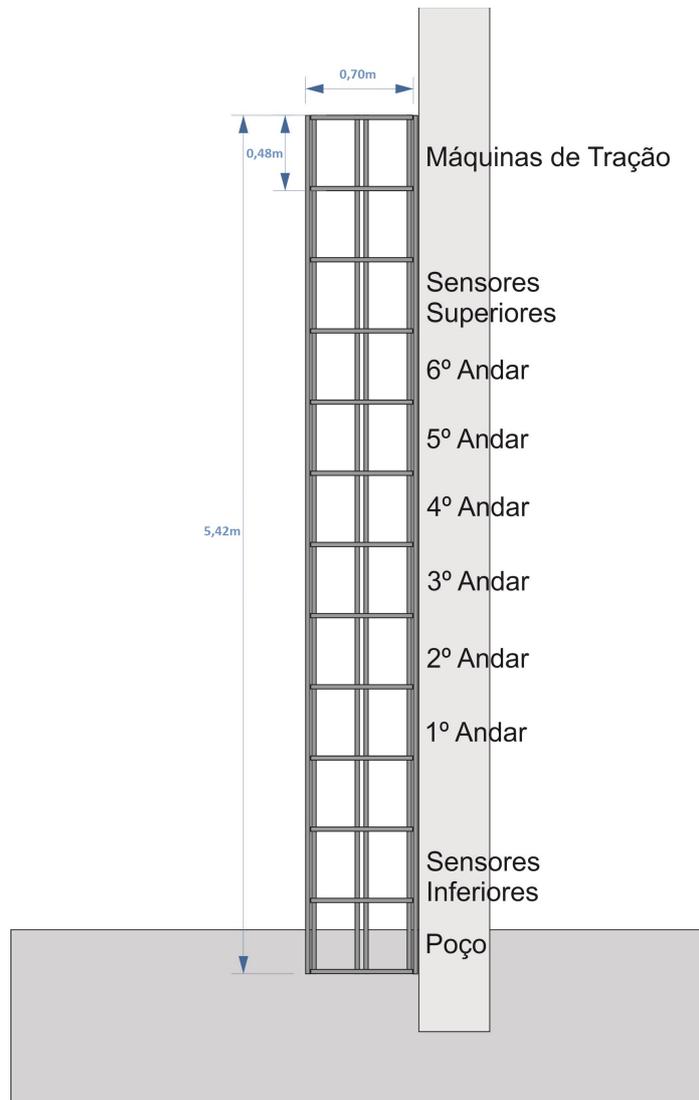


Figura 3.1: Desenho da estrutura mecânica.

3.2 MOTORES CA DE INDUÇÃO

Os motores são uma parte essencial para o funcionamento dos elevadores. O esforço mecânico de movimentação será feito por esses motores. Por isso deve-se ter cuidado na hora da escolha destes, pois eles devem ser capazes de suportar a carga e o esforço constante da operação de elevadores.

O motor de indução em conjunto com o variador de frequência é uma solução muito usada atualmente em edifícios modernos para o acionamento de elevadores, pois alia conforto com economia de energia em relação a motores CA sem o controle de velocidade. As paradas e acelerações suaves causam menos desgaste nos componentes de frenagem, cabos de tração, polias. [25].

A Norma de segurança apresenta o modo em que eles devem atuar.

NM 207 item 12.1 - Generalidade

Cada elevador deve possuir, pelo menos, uma máquina própria.

NM 207 item 12.4.1.1 - O elevador deve possuir um sistema de freada que opere automaticamente:

a) caso haja queda da fonte de energia principal;

b) caso haja queda da fonte de energia dos circuitos de controle.

Adequando o projeto ao item 12.1 da NM 207, são utilizados dois motores, um para cada elevador, para a movimentação do carro. Foram escolhidos motores fabricados pela WEG. Trata-se de um trifásico de indução do tipo gaiola e seu componentes podem ser observados na figura 3.2. A potência do motor é suficiente para a movimentação da cabine, pois esta não será testada para o levantamento de cargas pesadas.

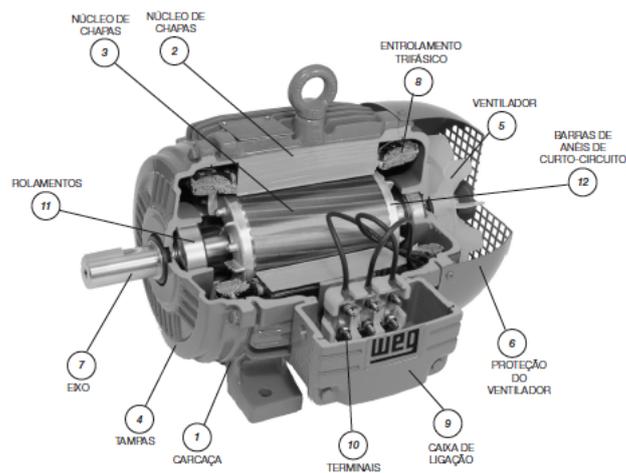


Figura 3.2: Descrição dos componentes de um motor CA de indução do tipo gaiola.

Eles são do tipo Motor-Freio, ou seja, caso ele não esteja submetido à alimentação, o freio é automaticamente acionado. Essa característica garante o cumprimento do item 12.4.1.1 da NM 207. O freio está localizado ao lado do ventilador e pode ser observado na figura 3.3, em que o motor está sem alimentação e portanto, ele está acionado[1]

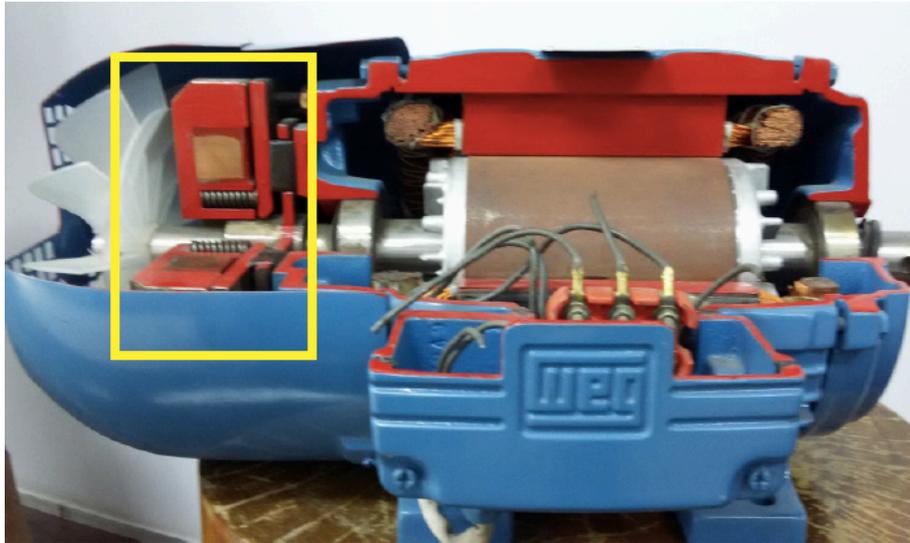


Figura 3.3: Indicação da posição do freio no motor.

3.2.1 ESPECIFICAÇÕES DO MOTOR

As especificações técnicas deste motor são obtidas a partir de sua placa identificadora. Todo motor elétrico possui uma placa identificadora, seguindo a norma NBR7094. Esta placa contém as informações técnicas do motor necessárias para a sua operação.

A partir da figura 3.4, pode-se extrair todas as informações relevantes ao projeto, sobre os motores utilizados. Essas informações serão usadas para realizar a configuração dos parâmetros do controlador de velocidade, e são dispostas na Tabela 3.1:

3 ~ AL71		26AG005 GW81443	
MOTOR INDUCAO - GAIOLA INDUCTION MOTOR - SQUIRREL CAGE		Hz 60	CAI N
kW(HP-cv) 0.18(0.25)		RPM min ⁻¹ 1060	
FS 1.15	ISOL INSL B Δ	K Ip/m 3.0	IP55
220/380 V		1.52/0.880	
REG DUTY S1		MAX AMB 40°C	ALT 1000
REND.x= 50.0		COS ψ 0.62	
IFS			
220 V 6 4 5 1 2 3 L1 L2 L3 Δ		380 V 6 4 5 1 2 3 L1 L2 L3 Y	

Figura 3.4: Placa identificadora do motor.

Tensão nominal (Ligação Delta)	220 V
Tensão nominal(Ligação Estrela)	380 V
Corrente nominal (Ligação Delta)	1,52 A
Corrente nominal (Ligação Estrela)	0,880 A
Frequência	60 Hz
Velocidade nominal	1060 RPM
Potencia nominal	0,18 kW ou 0,25 HP

Tabela 3.1: Valores obtidos da placa do motor.

3.2.2 RESFRIAMENTO DO MOTOR

As perdas são inevitáveis em qualquer motor. Grande parte dessas perdas são em forma de calor. Isso gera um aumento de temperatura na operação. Para que o aumento de temperatura não afete o funcionamento do motor, deve existir um sistema de resfriamento.

Os motores utilizam ventilação axial como sistema de refrigeração. A máquina é totalmente fechada, resfriada por ventilador e carcaça aletada. Esta máquina ainda possui canais axiais nos pacote de chapas do estator e do rotor, como pode ser visto na figura 3.5.

[26]

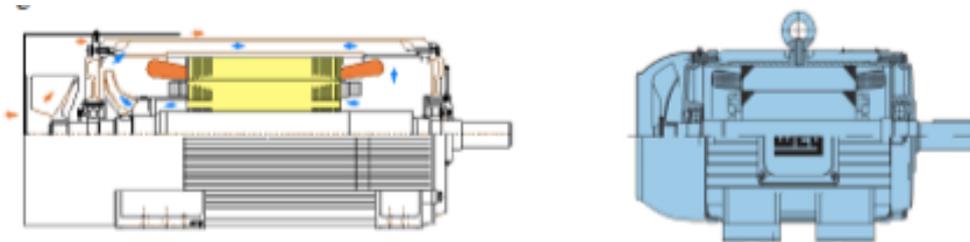


Figura 3.5: Componentes responsáveis pelo resfriamento do motor.

3.3 SENSORES

Para a escolha dos sensores, leva-se em conta o ambiente no qual ele está inserido, o tipo de objeto que eles devem identificar e sua precisão.

Os sensores ficam localizados em cima da cabine do elevador, então eles estarão sujeitos a acumular muita sujeira. Com isso podemos descartar a utilização de sensores fotoelétricos, que são sensíveis a poeira e óleo nas lentes. Esses sensores são muito utilizados em sistemas antiesmagamento nas portas do elevador. Mas neste projeto, isso não se aplicará por conta da inexistência de portas.

Para identificar os pavimentos e locais de parada, utiliza-se a estrutura metálica. Sensores capacitivos são capazes de identificar metal, mas não possuem grande seletividade de alvos, podendo mandar falsos sinais devido a outros objetos.

Os sensores ultrassônicos possuem muitos requisitos no posicionamento da instalação e além disso possui pontos cegos que podem atrapalhar a identificação. O ambiente em que ele deve ser inserido também não é adequado pois a estrutura do elevador e as paredes podem causar reflexão dos sinais sonoros e implicar em interferência na leitura.

Os sensores indutivos são altamente indicados para identificação de alvos metálicos, possuem uma ótima robustez em relação a fatores ambientais e não possuem partes moveis, que causaria grande desgaste mecânico, sendo assim a escolha mais recomendada.

3.3.1 SENSOR ALLEN-BRADLEY

O projeto utiliza sensores indutivos de proximidades Allen-Bradley 871TM DeviceNet para a identificação de posições críticas para o controle do elevador. Este sensor é robusto, trabalhando em condições extremas de temperatura, vibração e choque. Ideal para a utilização em um sistema que está em constante movimento.



Figura 3.6: Sensor Indutivo de Proximidade da Allen-Bradley

Este sensor possui saídas analógicas e digitais e se comunica com o sistema através de uma rede DeviceNet com os controladores do sistema.

No Byte 1 se encontra a saída digital. temos algumas funcionalidades como objeto muito perto ou muito longe, detecção de movimento, além de informações sobre seu estado de operação. No Byte 2 é possível obter os dados da saída analógica, utilizando o modo Strobe.

Byte 1	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
	Sensor Output	Diagnostic	Coil Operational	Too Close	Too Far	Always In	Motion Detect	Counter Output
Byte 2	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
	Analog Output (Strobe Only)							

Tabela 3.2: Informações encontradas nos Bytes dos sensores.[27]

3.3.2 FAIXA DE TRABALHO

A faixa de trabalho é a distancia operacional no qual o sensor realizará a medição correta. O valor nominal dessa distancia é especificada no datasheet do sensor, utilizando seu alvo padrão, num local com temperatura em 20 °C.[27]

A distancia sensora efetiva pode diferir de acordo com as variações de tensão de alimentação do sensor e com as variações de temperatura do ambiente. Estes valores podem ir de 81% a 121% do valor nominal. Por isso, para garantir que o sensor sempre identificará o alvo, deve-se utilizar a distancia máxima de 81% do valor nominal. A distancia nominal do sensor é de 8mm, então a distancia em que seguramente pode-se operar, considerando condições mais adversas é de 6,5mm.[28]

O material do objeto que está sendo identificado, também influencia na distancia. De acordo com o material a distancia deve ser multiplicada por um fator de correção. Na tabela 3.2 podemos ver o fator de redução para alguns materiais. [27]

Material do Objeto	Fator de Correção
Aço Doce	1
Aço inoxidável	0,9...1
Latão	0,3...0,5
Alumínio	0,1...0,4
Cobre	0,4...0,6

Tabela 3.3: Fatores de redução para determinados materiais.

Assim o alcance especifico pode ser calculado por:

$$\text{(Alcance Especifico)} = \text{(Fator de Correção)} \times \text{(Alcance Nominal)}$$

3.3.3 ALVO PADRAO

O alvo padrão especifica as características e dimensões mínimas que o objeto a ser detectado deve possuir. Para o sensor indutivo, este alvo deve ser uma peça de 1mm de espessura. As dimensões laterais do alvo devem ser do tamanho da face sensor ou três vezes a distancia do sensor ao alvo, escolhendo o que for maior. No caso do sensor utilizado, a distancia sensora será no máximo de 6,5mm e a face do sensor possui 18mm. Assim, o alvo padrão deve possuir aproximadamente 20mm de dimensões laterais e 1mm de espessura como mostrado na figura abaixo.

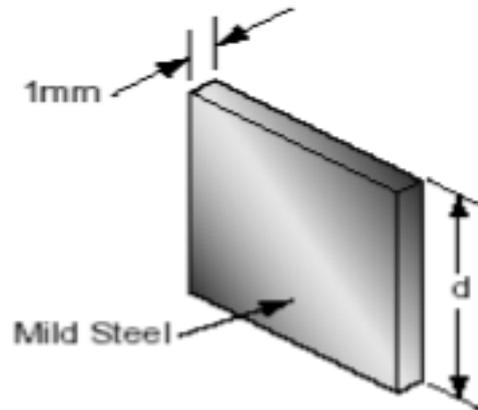


Figura 3.7: Alvo padrão ($d = 20\text{mm}$).[28]

3.3.3 POSICIONAMENTO DOS SENSORES

Os sensores indutivos devem seguir algumas recomendações para a sua montagem. Por serem sensíveis a metais, eles devem ser instalados em uma zona livre destas matérias, respeitando distancias mínimas, como mostrado na figura 3.8.

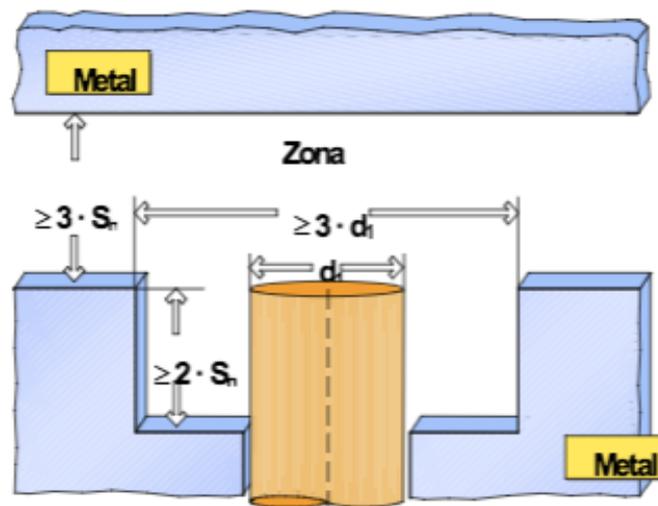


Figura 3.8: Distâncias necessárias para correto posicionamento dos sensores.[28]

Estes sensores não podem ficar muito próximos um do outro, pois um pode interferir na sensibilidade e até gerar falsos sinais. Então deve se respeitar uma distância mínima igual ao tamanho do diâmetro da face sensora entre dois sensores.

3.4 POWERFLEX 70

3.4.1 CONTROLADOR DE VELOCIDADE

O sistema utilizado para o acionamento dos motores é o VVVF (Variable Voltage – Variable Frequency), que significa Tensão variável e Frequência variável. Para isso será utilizado o controlador de velocidade Allen-Bradley PowerFlex 70. Este equipamento é responsável pelo controle da velocidade.

Assim, através do powerFlex 70, a velocidade é controlada em função de um padrão desejado; o que permite obter aceleração (V0 para V1) e desaceleração (V1 para V0) suaves do carro, evitando-se assim o salto na passagem da velocidade. Movimentos bruscos de parada ou acelerações muito altas não são desejáveis, seja para pessoas ou cargas transportadas.

O PowerFlex possui muitos detalhes para instalação e configuração. Internamente ele possui uma fonte de 24V que pode ser utilizada para alimentar a rede DeviceNet, para a comunicação com outros dispositivos de controle e aquisição de dados. Ele também pode fazer o controle manual do motor através de portas digitais. Ele possui entradas para as três fases de alimentação e um terra e fornece uma saída com três fases com frequência e tensão controladas. Este equipamento foi adquirido com uma interface homem máquina, mostrado na figura 3.9, que auxilia na seu manuseio.

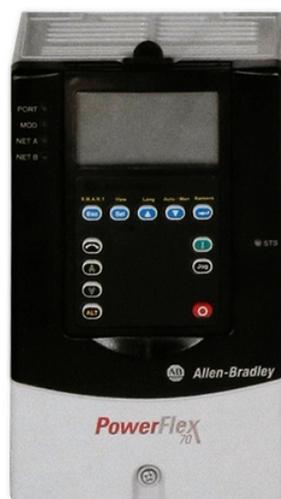


Figura 3.9: PowerFlex 70 Allen-Bradley.

3.4.2 DEFINIÇÃO DE PARAMETROS E FUNÇÕES

Para iniciar o sistema, o Controlador de velocidade deve-se passar para a configuração dos seus parâmetros e da inserção dos parâmetros do motor a ser controlado. Através do IHM (Interface Homem Máquina) interno é possível acessar diversas configurações. Esses parâmetros irão variar de acordo com do motor utilizado e sua finalidade. O controlador utiliza três tipos de parâmetro:

- ENUM Parameters: duas ou mais opções irão aparecer no visor para a escolha. Cada item terá uma numeração;
- Bit Parameters: Cada um desses parâmetros tem um bit associado as suas funções e condições. Assim a função estará desligada ou falsa no bit 0 e ligada ou verdadeira no bit 1.
- Numeric Parameters: Estes representam um único valor numérico a ser definido.

Os parâmetros são divididos em grupos para simplificar a programação destes pois são agrupados com parâmetros similares. Eles estão organizados em seis grupos de arquivos na definição básica e em sete grupos na definição avançada.

As definições dos parâmetros serão feitas em princípio no modo avançado. Os sete grupos de arquivo do modo avançado são: *Monitor, Motor Control, Speed Command, Dynamic Control, Utility, Communication e Inputs and Output*“.

Configurando os parâmetros é possível inserir informações do motor, tais como rotações por minuto, tensão, corrente máxima, número de pólos, potência e frequência máxima. Pode-

se escolher o tipo de curva na transição de velocidades, a duração das transições, a origem dos dados de controle do variador, o sentido de referência de rotação do motor e o ajuste de quais funções serão acessadas pelas entradas digitais. Feito isso o PowerFlex70 fará a parametrização do acionamento e então o sistema do motor estará pronto para ser utilizado.

3.5 CONTROLLOGIX 5555

O CLP usado é o ControlLogix 5555 da Allen-Bradley. Sua comunicação pode ser feita por diversos protocolos como Ethernet, DeviceNet e ControlNet. O sistema é modular, podendo ter diferentes combinações de entradas, saídas e módulos especiais, a serem escolhidos de acordo com a necessidade do sistema implementado.



Figura 3.10: CLP ControlLogix 5555 Allen-Bradley.

O controlador ControlLogix 5555 é muito robusto e foi desenhado para uso em ambientes com condições adversas. Este controlador é acoplado no Rack de módulos. Ele se comunica com outros dispositivos, neste caso pela rede DeviceNet.

A linha de controladores ControlLogix é programada por meio do software RSLogix, que permite criação de programas em diferentes linguagens, tais como Ladder, Grafset e Function Blocks. Independente da linguagem o programa é traduzido para linguagem de máquina e executado da mesma forma.

3.6 PANELVIEW

O PanelView é uma IHM que pode receber dados de controle do usuário, que serão enviados aos CLP em tempo real para alterar parâmetros, iniciar ou parar processos, comunicar mensagens a outros pontos da rede, verificar falhas, entre outras funcionalidades.

O PanelView pode desempenhar funções de monitoramento e controle, como mostrado na figura 3.11 e pode ser instalado onde se queira, comunicando com o CLP através da rede DeviceNet.

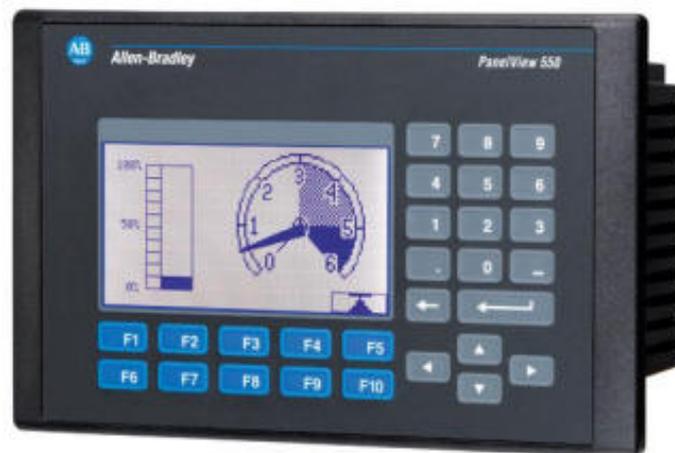


Figura 3.11: PanelView Allen-Bradley.

O programa PanelBuilder32 é uma aplicação dedicada que permite a criação de telas no PanelView e de transições entre estas telas. Os botões podem ser associados às variáveis de programa e saídas de dados podem ser fornecidas em tela em forma de barras, textos, gráficos e outros.

Com ele, podemos criar simulações de sistemas, mostrar em tempo real as ações do CLP e criar os botões de chamada do elevadores virtualmente. Isso é muito útil por conta do sistema não ter botões de chamada em seus andares.

3.7 REDE INDUSTRIAL

3.7.1 PROTOCOLO CAN

A rede DeviceNet é uma rede baseada em CAN, o que significa dizer que ela utiliza o protocolo CAN para troca de dados na rede.[12]

O protocolo CAN utiliza comunicação serial para transmitir os dados e descreve os serviços das camadas 1 e 2 do modelo (camada física e camada de enlace de dados respectivamente) OSI/ISO. A norma ISO 11898 define a camada física deste protocolo.

Na camada de enlace de dados são definidos os diferentes tipos de frames, a forma de detecção de erros, a validação e a arbitração das mensagens.[13]

Os frames de dados em uma rede baseada em CAN são compostos principalmente por um campo identificador que contém 11 bits de informação e um campo de dados, podendo este conter até 8 bytes de informação, mostrados na figura 3.10. Existe na especificação do modelo CAN 2.0 dois tipos de campo identificador: *standard* (11 bits) e *extended* (29 bits).

Identificador	8 bytes de dados							
11 bits	byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6	byte 7

Figura 3.12: Estrutura dos Bytes do protocolo CAN.

Além do frame de dados, existe o frame remoto (RTR frame) que não possui campo de dados, apenas o identificador. Ele funciona como uma requisição para que outro dispositivo da rede transmita o frame de dados desejado.[13]

Em uma rede CAN, qualquer elemento da rede pode tentar transmitir um frame para a rede em um determinado instante de tempo. Caso ocorra de dois elementos tentarem acessar a rede ao mesmo instante, aquele que enviar a mensagem mais prioritária conseguirá transmitir os dados. A prioridade da mensagem é definida pelo identificador do frame, quanto menor o valor deste identificador, maior a prioridade da mensagem. Um frame com campo identificador com valor 0 (zero) corresponde ao mais prioritário.[13]

Este protocolo define diversos mecanismos para controle de erros, tornando-se assim uma rede muito confiável e com baixo índice de erro de transmissão onde não se detecta o mesmo. Cada dispositivo neste tipo de rede deve ser capaz de identificar a ocorrência de erros e de informar aos demais elementos que um erro foi detectado.

Os dispositivos na rede CAN funcionam da seguinte maneira: possuem contadores internos onde são incrementados sempre que um erro de transmissão ou recepção é detectado e decrementado quando um telegrama é enviado ou recebido corretamente. Dependendo do valor encontrado neste contador interno, o dispositivo poderá apresentar os seguintes estados:

- **Warning:** quando este contador supera um limite inferior, significando uma elevada taxa de erros.
- **Error Passive:** quando este contador ultrapassa um limite maior, onde o dispositivo para de atuar na rede ao detectar que outro dispositivo mandou um telegrama com erro.
- **Bus Off:** neste último estado, o dispositivo não irá mais enviar ou receber telegramas.

3.7.2 CAN E DEVICENET

Somente as especificações definidas pelo protocolo CAN não são suficientes para determinar um significado para os dados que são enviados na rede. É necessária uma especificação para indicar como o identificador e os dados devem ser montados e como as informações devem ser trocadas, permitindo assim que os elementos da rede interpretem corretamente os dados que são transmitidos. Esse é exatamente o objetivo do DeviceNet.

3.7.3 DEVICENET

3.7.3.1 CAMADA FISICA

O protocolo de rede DeviceNet utiliza uma topologia de rede do tipo tronco/derivação aonde se permite que tanto a fiação de sinal quanto a de alimentação estejam presentes no mesmo cabo. Esta alimentação, fornecida por uma fonte conectada diretamente na rede, supre os transceivers CAN dos nodos e apresenta as seguintes características:

- 24Vdc;
- Saída CC isolada da entrada CA;
- Capacidade de corrente compatível com os equipamentos instalados.

Dependendo da taxa de transmissão utilizada, o tamanho total da rede irá variar. Podemos observar esta relação na tabela 3.4:

Taxa de transmissão	Tamanho da rede	Derivação	
		Máximo	Total
125kbps	500m	6m	156m
250kbps	250m		78m
500kbps	100m		39m

Tabela 3.4: Taxa de transmissão X tamanho da rede em DeviceNet.

Neste tipo de rede pode ocorrer reflexões de sinal na linha, e para evitar este tipo de fenômeno, recomenda-se a instalação de resistores de terminação nas extremidades do tronco da rede, pois a falta deste pode provocar erros intermitentes. As características desse resistor devem ser as seguintes:

- 121 Ω ;
- 0.25W;
- 1% de tolerância.

Em DeviceNet, diversos tipos de conectores podem ser utilizados, tanto selados quanto abertos. A definição do tipo a ser utilizado dependerá da aplicação e do ambiente de operação do equipamento.

3.7.3.2 CAMADA DE ENLACE DE DADOS

A camada de enlace de dados do DeviceNet é definida pela especificação do CAN, o qual define dois estados possíveis, dominante e recessivo (níveis lógicos 0 e 1 respectivamente). Um nodo pode levar a rede ao estado dominante se transmitir alguma informação. Assim, o barramento somente estará no estado recessivo se não houver nodos transmissores no estado dominante.[13]

CAN utiliza o CSMA/NBA para acessar o meio físico, o que significa dizer que, antes de transmitir, um nodo deve verificar se o barramento está livre. Caso esteja, então ele pode iniciar a transmissão do seu telegrama. Caso não esteja, deve aguardar. Se mais de um nodo acessar a rede simultaneamente, um mecanismo baseado em prioridade de mensagem entrará em ação para decidir qual deles terá prioridade sobre os outros. Este mecanismo é não destrutivo, ou seja, a mensagem é preservada mesmo que ocorra colisão entre dois ou mais telegramas.

CAN define quatro tipos de telegramas: dados, remoto, overload e erros. Destes, a DeviceNet utiliza somente o frame de dados e o frame de erros.

Dados são movimentados utilizando-se o frame de dados, a estrutura deste frame pode ser observado na figura 3.11.

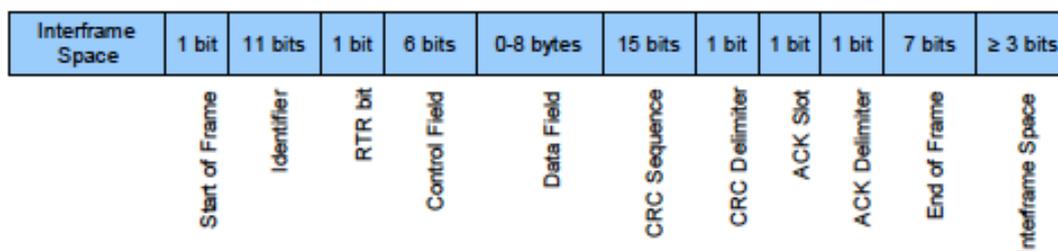


Figura 3.13: Frame de dados do CAN.

Já os erros são indicados através do frame de erros. CAN possui uma verificação e um confinamento de erros bastante robusto. Isto garante que um nodo com problemas não prejudique a comunicação na rede.

3.7.3.5 ARQUIVO DE CONFIGURAÇÃO

Todo nodo, ou seja, todo equipamento utilizado na rede DeviceNet possui um arquivo de configuração próprio chamado EDS. Neste arquivo estão presentes informações importantes sobre o funcionamento do dispositivo e este deve ser registrado no software de configuração de rede.

3.7.3.6 MODOS DE COMUNICAÇÃO

O DeviceNet possui dois tipos básicos de mensagens, I/O e *explicit*. Dependendo do tipo de dado utilizado, um ou outro será mais adequado:

- I/O: telegrama síncrono dedicado à movimentação de dados prioritários entre um produtor e seus consumidores. Podem ser divididos de acordo com o método de troca de dados, sendo os principais os seguintes:
 - *Polled*: método no qual o mestre envia telegramas a cada um dos escravos de sua lista (*scan list*). Cada escravo, após ter recebido a solicitação, responde prontamente a solicitação do mestre. Este processo se repete até que todos os escravos sejam consultados e, em seguida, reinicia-se o ciclo.
 - *Bit-Strobe*: neste método o mestre envia um telegrama de 8 bytes para a rede, onde cada bit desses 8 bytes correspondem a um escravo que, se endereçado, responderá de acordo com o programado.
 - *Change of state*: método em que a troca de dados entre mestre e escravo ocorre somente quando houver mudanças nos valores que estão sendo monitorados e/ou controlados até determinado limite de tempo ser atingido. Após este limite ter sido alcançado, os processos de transmissão e recepção de dados ocorrerão mesmo que não tenham ocorrido alterações. Este limite de tempo pode ser configurado no programa de configuração da rede.
 - *Cyclic*: este método é bastante semelhante ao anterior, sendo a única diferença na parte de produção e consumo de mensagens. Neste método, as trocas de dados ocorre em intervalos regulares de tempo, ocorrendo ou não alteração dos dados. Este intervalo de tempo também pode ser ajustado no programa de configuração da rede.
- *Explicit*: este tipo de telegrama é utilizado de uma maneira geral e não prioritariamente, ocorrendo principalmente em tarefas assíncronas como parametrização e configuração dos equipamentos.

3.7.3.7 CONJUNTO DE CONEXÕES PREDEFINIDAS MESTRE/ESCRAVO

Em DeviceNet emprega-se fundamentalmente mensagens ponto a ponto, porém, é bastante comum utilizar um esquema predefinido de comunicação baseado no mecanismo mestre/escravo.

Este esquema utiliza um movimento simplificado de mensagens do tipo I/O comum em aplicações de controle. A vantagem na utilização deste método é a baixa quantidade de requisitos e de recursos necessários para rodá-lo.

4 PROCEDIMENTOS

4.1 ANALISE PRÉVIA

A bancada didática, que possui os elevadores, motores e todos os equipamentos eletrônicos já estava montada. Porém, ela estava fora de uso e devido a falta de manutenção, muitos problemas foram surgindo. Alguns destes problemas já existiam desde o trabalho anterior feito nos mesmos equipamentos, em 2009. O trabalho cita problemas em diversas áreas, incluindo posicionamento dos sensores de final de curso, falha na detecção dos sensores indutivos, movimentação dos carros prejudicada por falta de lubrificação e falhas nos trilhos e problemas nas configurações dos inversores.[29]

Assim, identificou-se todos os defeitos e problemas do sistema que impediria seu funcionamento correto. Primeiramente, fez-se uma análise sobre a parte mecânica. A estrutura metálica, usada pela cabine, para movimentar entre os pavimentos estava muito suja e totalmente sem lubrificação.

Outra falha mecânica observada foi o posicionamento dos sensores de fim de curso. Estes estavam mal posicionado, atrapalhando o movimento da cabine e com isso, eles não seriam acionados, caso a cabine chegasse ao encontro dos sensores, gerando uma falha nos comandos de emergência.[29]

A posição dos sensores indutivos foram analisadas e possivelmente eles não fariam a detecção dos alvos corretamente. Por isso eles devem ser testados e caso o erro se confirme, eles devem ser reposicionados. Existem 3 sensores indutivos em cima de cada cabine, na hora do reposicionamento, deve se levar em conta a distancia mínima que estes sensores devem ficar em relação ao outro.

O cabeamento elétrico do sistema foi analisado e constatou-se a falta de etiquetas identificando as conexões, isso dificulta qualquer manutenção. Além disso, as extremidades dos cabos estão sem terminais e mal conectados aos bornes de entrada dos equipamentos. Verificou-se que alguns dos sensores não estavam conectados ao sistema.[29]

A rede de comunicação DeviceNet apresentou falhas, impossibilitando-a de se conectar com os vários equipamentos. Percebeu-se que o problema poderia ser devido às conexões estarem soltas ou por falta da resistência de fim de linha necessária nesse tipo de rede.[29]

4.2 LIMPEZA E LUBRIFICAÇÃO

A limpeza e lubrificação do sistema foi efetuada, visando a perfeita movimentação da cabine ao longo de todo o trilho da estrutura mecânica presente para teste.

Foi necessária a utilização do agente químico thinner para remover o excesso de impurezas que haviam se acumulado ao longo do tempo nos trilhos por onde os carros irão percorrer.

Além disso, realizamos a remoção do antigo lubrificante que havia sido utilizado para teste pois o mesmo já havia endurecido e se tornado pegajoso, impossibilitando a livre e boa movimentação da cabine.

Após realizada esta primeira etapa, utilizamos graxa azul ao longo de todo o trilho percorrido por eles, garantindo que não haviam falhas e verificando a movimentação da cabine ao longo do trilho.

Através do teste de movimentação, verificou-se que em alguns pontos, as cabines encontravam resistência para se mover , devido a falta de concordância entre os trilhos paralelos. Este problema foi solucionado, ajustando os guias de movimentação da cabine.

4.3 REPOSICIONAMENTO E FIXAÇÃO DOS SENSORES DE FINAL DE CURSO

Na estrutura que havia sido instalada previamente pela equipe que realizou testes anteriormente, verificamos que haviam problemas na fixação e no posicionamento dos sensores de final de curso em ambos elevadores presentes no laboratório.

Primeiramente, somente uma das cabines haviam sido utilizada para testes, portanto, foi necessário todo um reposicionamento dos sensores de final de curso de modo a permitir que ambas as cabines possuíssem acesso aos mesmo, uma vez que nas posições que os sensores estavam instalados anteriormente, o sensor pertencente ao trilho vizinho interferia na movimentação do outro.

Após encontrado uma posição adequada para estes sensores, como pode ser observado na figura 4.1, realizou-se a fixação, garantindo que a simples movimentação e acionamento dos mesmo pela cabine pertencente ao trilho não iria alterar a posição dos sensores, o que

ocasionaria falhas no período de testes. Ao final, foram feitos testes de movimentação dos carros para garantir o funcionamento dos sensores em caso de emergência.

Percebeu-se com estes testes, que a cabine possui um estrutura que entra em contato com os sensores. Porém esta estrutura está localizada na parte de baixo da cabine. O acionamento dos sensores superiores estavam prejudicados por conta da posição dessa estrutura. A solução para este problema foi colocar pedaços de metal na parte superior da cabine e assim acionar corretamente os sensores em caso de emergência.

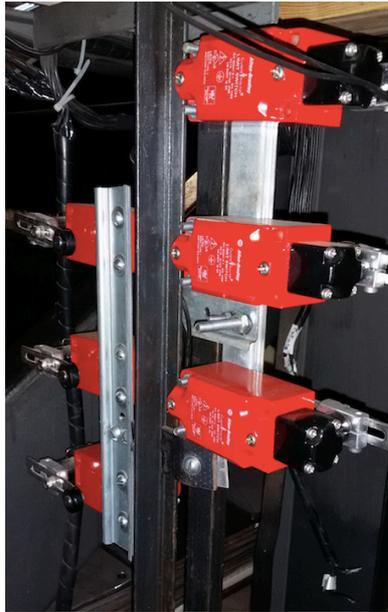


Figura 4.1: Sensores de final de curso reposicionados.

Os sensores indutivos localizados em cima da cabine do elevador também estavam com o posicionamento inadequado.

Um dos grandes problemas encontrado foi o posicionamento dos sensores indutivos. Os sensores estavam posicionados nas laterais, como mostrado na figura 4.2, que possuem muitos metais que poderiam interferir na leitura. Além disso, a estrutura não estava utilizando o alvo padrão para a identificação dos lugares. Então a leitura não estava sendo feita da maneira correta.



Figura 4.2: Sensores nas laterais do elevador.

Assim, os sensores foram direcionados para a parte da frente da estrutura, que possui metal, apenas nas mudanças de pavimentos. Alvos padrão para cada pavimento foram usinados no SG-09 e fixados na estrutura de mudança de pavimento, como visto na figura 4.3.

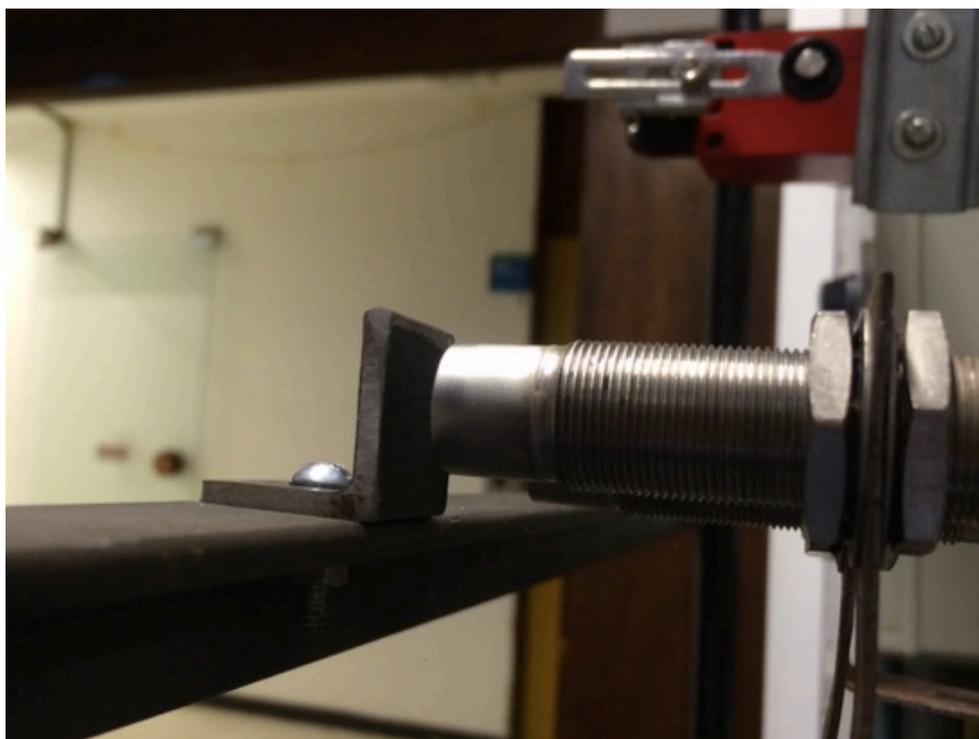


Figura 4.3: Foto dos alvos fixados com sensores indutivos reposicionados.

4.4 INSTALAÇÃO ELÉTRICA

Quando vamos analisar um sistema elétrico, sua documentação e organização são fundamentais para um entendimento completo. Por isso o cabeamento de todo o sistema foi estudado e documentado. Após fazer o levantamento de todas as conexões elétricas dos

equipamentos utilizados, assim como a de topologia da rede de comunicação, foi utilizado o software de desenho CorelDraw para fazer representações destas ligações, mostrados nas figuras 4.4 a 4.6.

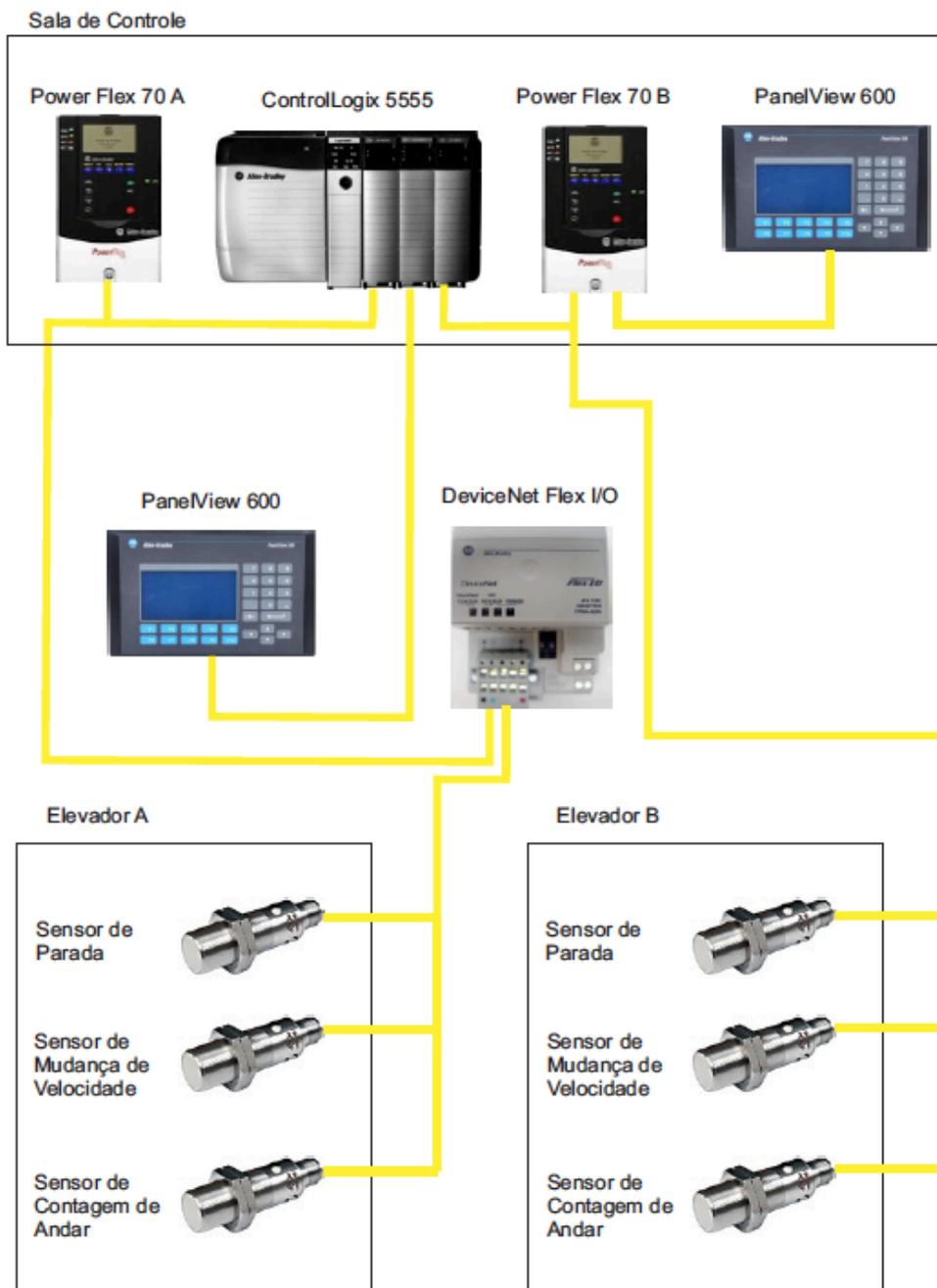


Figura 4.4: Topologia da rede DeviceNet.

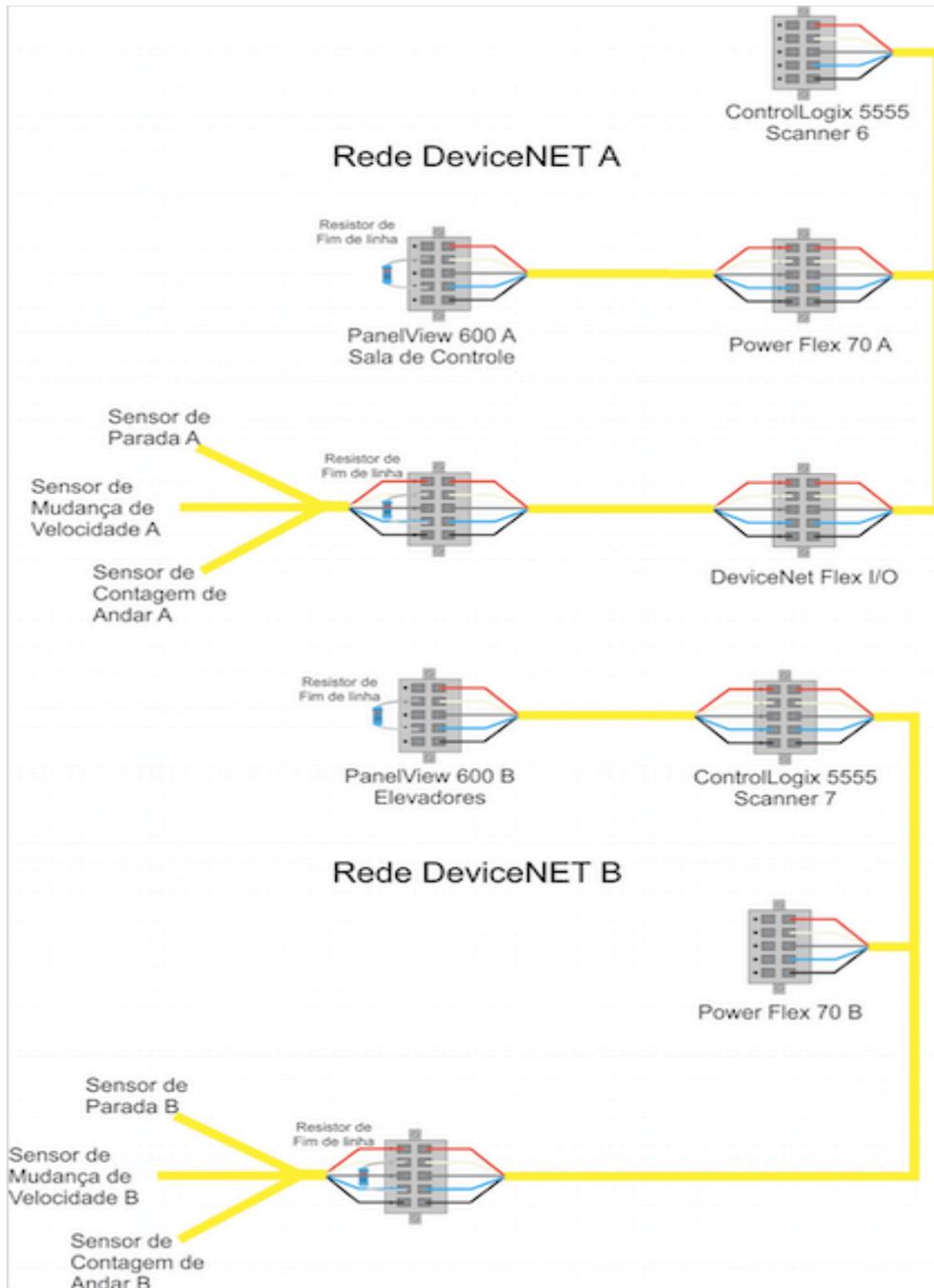
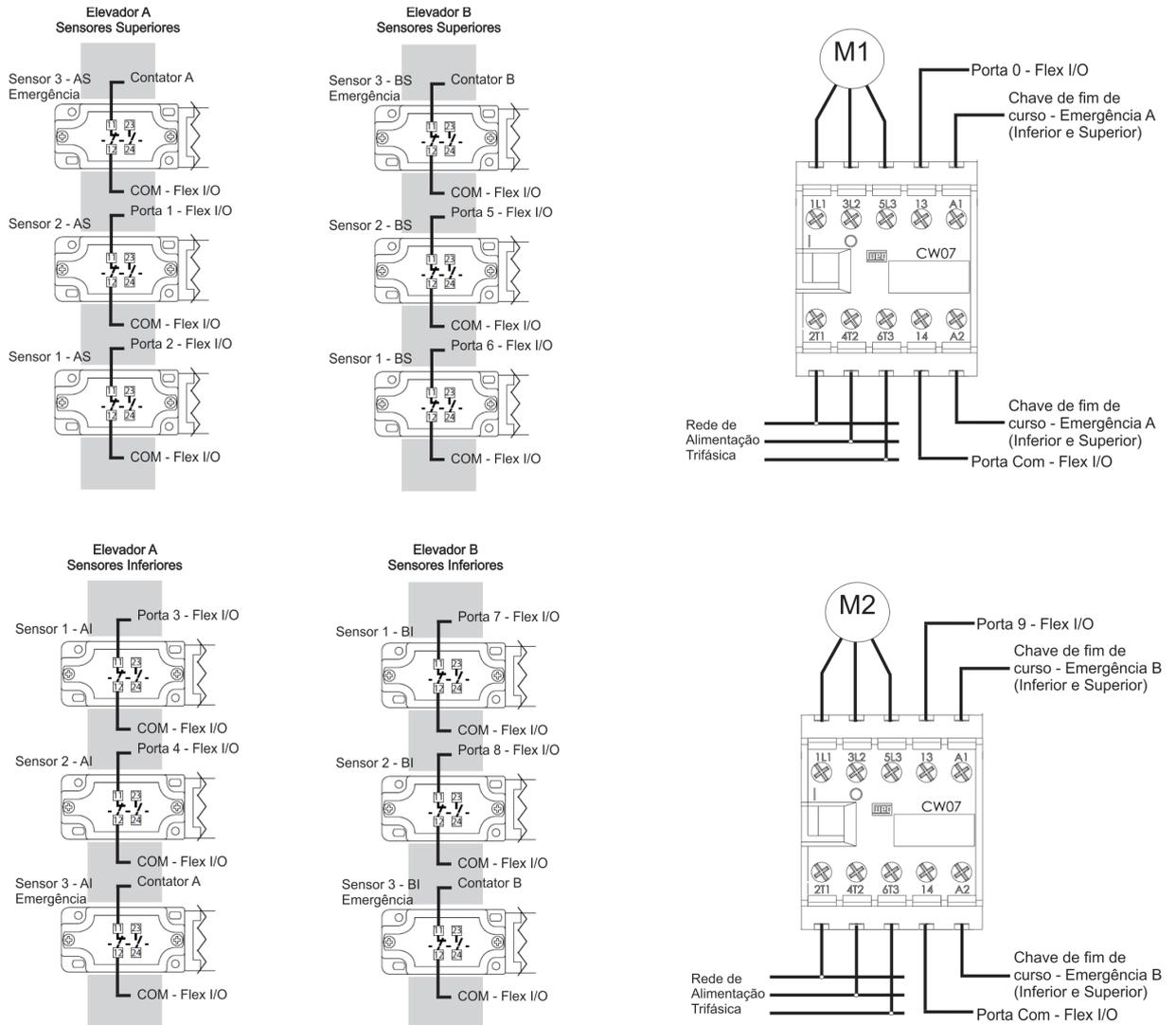
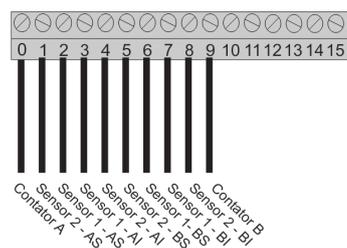


Figura 4.5: Conexões e montagem da rede de comunicação.



Flex I/O



- 0 - Contador A
- 1 - Sensor A Superior Parada
- 2 - Sensor A Superior Corta Velocidade
- 3 - Sensor A Inferior Corta Velocidade
- 4 - Sensor A Inferior Parada
- 5 - Sensor B Superior Parada
- 6 - Sensor B Superior Corta Velocidade
- 7 - Sensor B Inferior Corta Velocidade
- 8 - Sensor B Inferior Parada
- 9 - Contador B

Figura 4.6: Conexões e identificações no Flex I/O.

Com o projeto feito, deve-se identificar todos os cabos e ligações nos equipamentos de acordo com o definido em projeto. Assim o trabalho de manutenção é facilitado. Essa identificação também pode evitar erros de ligações, que podem danificar os equipamentos. Terminais devem ser acoplados nas extremidades dos cabos elétricos. Desta maneira, as conexões nos bornes serão seguras e menos suscetíveis a falhas.

O sistema anterior estava bem danificado, com ligações mal feitas ou até perdidas. Então uma restauração destas ligações se tornou necessária para a continuação do projeto. Como foi destacado anteriormente, a rede DeviceNet não estava se comunicando com os equipamentos conectados a ela. O principal motivo disto eram o estados das ligações, por isso todos pontos de conexão entres os equipamentos foram reparados.

Um bom exemplo para essa degradação do sistema é a ligação da comunicação dos três sensores indutivos posicionados no topo das cabines. A comunicação deles se juntam e seguem através de uma única via até o próximo ponto de conexão. Então é de extrema importância que a comunicação dos sensores com o sistema seja realizada para o controle correto do sistema. Nas condições que o sistema se encontrava isso não seria possível, como podemos ver na figura 4.6.

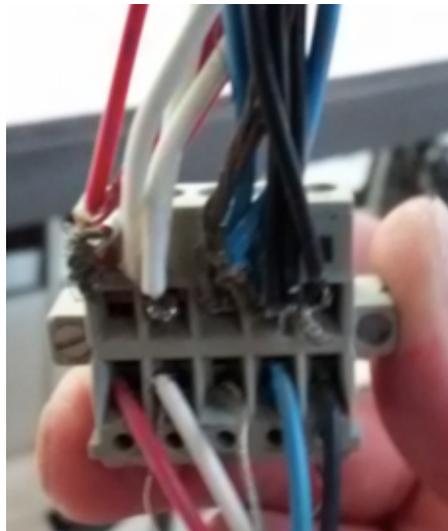


Figura 4.7: Conexão danificada dos sensores indutivos.

Por este ser um ponto que estará em constante movimento pois se encontra em conjunto com a cabine do elevador, foi necessário a reinstalação dessas ligações para uma maneira mais confiável. Então mudou-se o tipo de borne, para que três cabos pudessem ser conectados em apenas um borne e acoplou-se terminais para que estes não se soltassem tão facilmente, mostrados na figura 4.7.

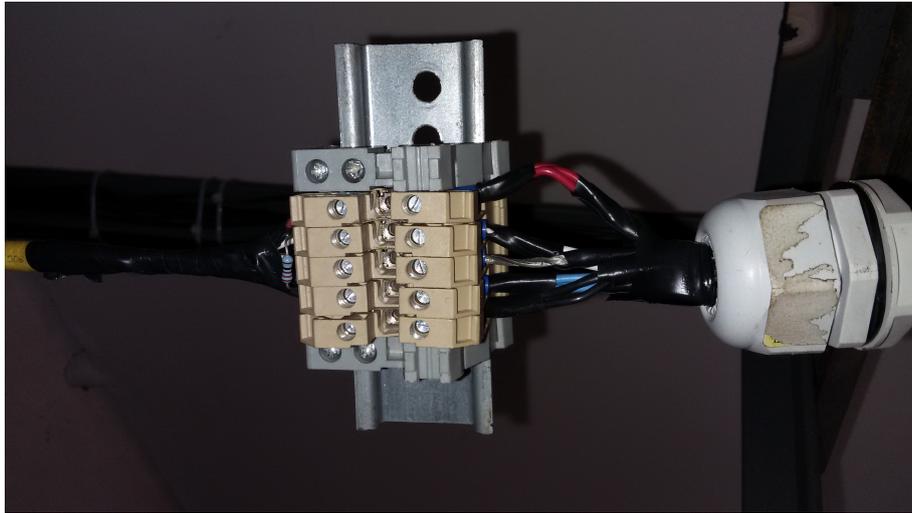


Figura 4.8: Conexão ajustada dos sensores indutivos.

A ligações dos sensores de final de curso e da rede DeviceNet que chegavam ao Flex I/O apresentavam condições ruins: cabos soltos, sem terminais e com a identificação apagada. Fez-se então o mapeamento de todos os cabos dos sensores que chegavam ao Flex I/O e a revitalização dos mesmo, com a devida identificação e terminais que garantem que os cabos não se soltem facilmente como podemos observar nas figuras 4.8 e 4.9 a seguir.



Figura 4.9: Conexões danificadas no Flex I/O.

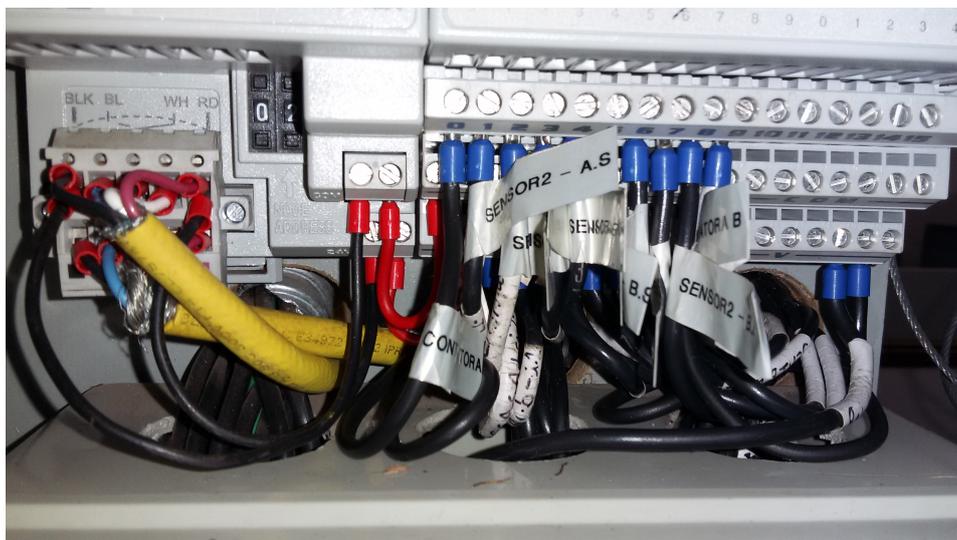


Figura 4.10: Conexões no Flex I/O após ajustes.

Mesmo após todos os reparos, uma das redes ainda não se conectava ao CLP. No tópico 3.7.3.1, as características da camada física da rede DeviceNet são descritas. Com essas informações, buscamos identificar qual poderia ser o problema. Esta rede é do tipo tronco/derivação e seu tamanho máximo é de 500m. Cada derivação pode ter no máximo 6m. Porém, a rede analisada não ultrapassava esses limites. O tamanho dos troncos e derivações de cada elevador podem ser vistos nas imagens abaixo.

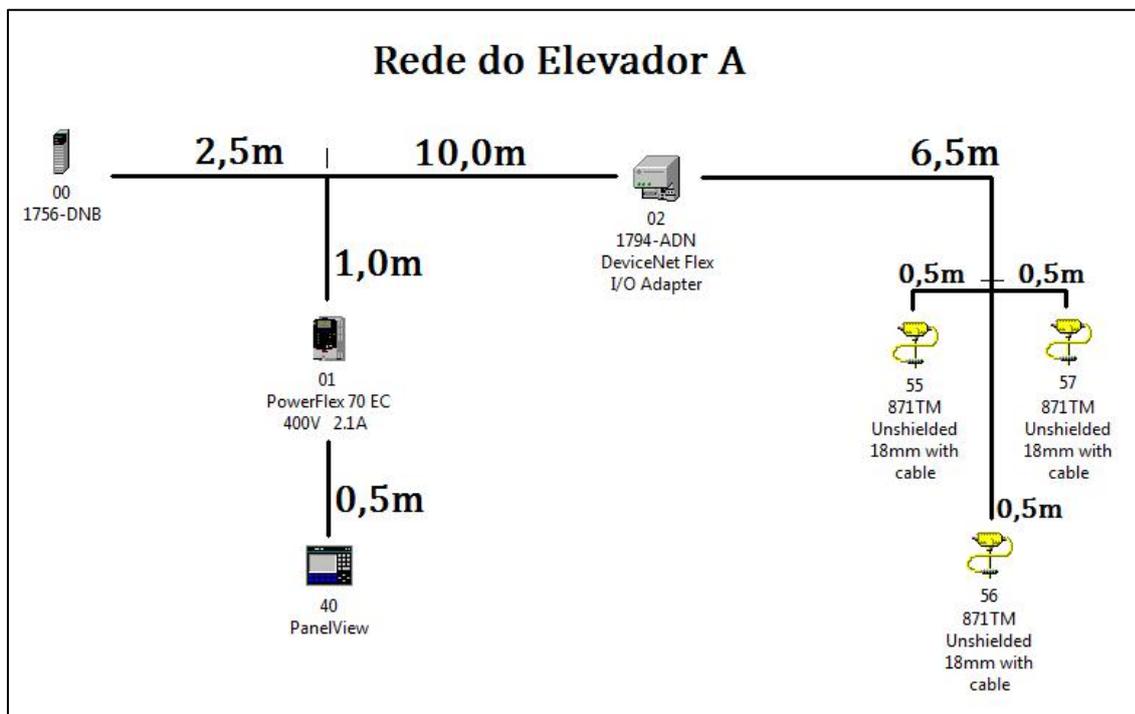


Figura 4.11: Representação da rede de comunicação para o elevador A.

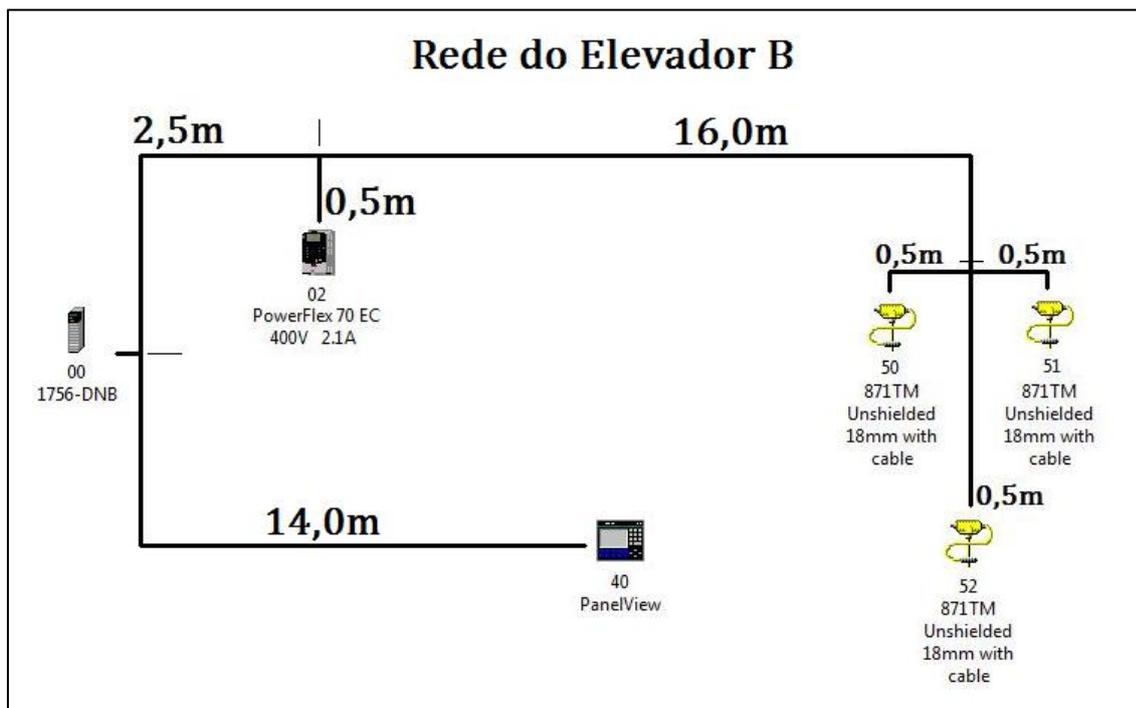


Figura 4.12: Representação da rede de comunicação para o elevador B.

Outro detalhe é a necessidade de uma resistência nas extremidades do tronco da rede, para evitar reflexões de sinais na rede. Verificou-se que em um dos troncos, essas resistências não existiam e por isso seu funcionamento poderia estar prejudicado. Assim, instalou-se resistências de 121Ω em cada extremidade da rede e esta passou a funcionar de maneira correta.

4.5 ENTRADAS E SAIDAS DO CLP

O CLP é responsável por receber as informações dos sensores e equipamentos da rede (entradas) e enviar comandos a esses equipamentos (saídas). Cada equipamento possui inúmeros tipos de entrada e alguns podem receber comandos. No Anexo 1, pode se encontrar todos os bits de entradas e saídas definidos no sistema com seu respectivo endereço referente ao CLP.

As entradas utilizadas na programação foram as seguintes:

- Sensores Indutivos
 - Bit 0 – Indicador de presença
 - Bit 1 – Diagnostico
- Flex I/O
 - Bit 0 - Contator Elevador A

- Bit 1 – Sensor de Parada Elevador A
- Bit 2 – Sensor de Diminuição de velocidade Elevador A
- Bit 3 - Sensor de Diminuição de velocidade Elevador A
- Bit 4 – Sensor de Parada Elevador A
- Bit 5 – Sensor de Parada Elevador B
- Bit 6 – Sensor de Diminuição de velocidade Elevador B
- Bit 7 - Sensor de Diminuição de velocidade Elevador B
- Bit 8 – Sensor de Parada Elevador B
- Bit 9 - Contator Elevador B
- PowerFlex 70 A e B
 - Bit 6 – Diagnostico Alarme
 - Bit 7 – Diagnostico Falha
- PanelView
 - Word 0 – Bits 0 ao 5 – Chamadas Externas 1-6
 - Word 1 – Bits 0 ao 5 – Chamadas Internas Elevador A 1-6
 - Word 2 – Bits 0 ao 5 – Chamadas Internas Elevador B 1-6

As saídas utilizadas foram:

- PowerFlex 70 A e B
 - Bit 0 – Parar
 - Bit 1 – Iniciar
 - Bit 3 – Limpar falhas
 - Bits 4 e 5 – Direção
 - Bit 6 – Controle Local
 - Bits 8 e 9 – Aceleração
 - Bits 10 e 11 – Desaceleração
 - Bit 12 ao 14 – Velocidade
- PanelView
 - Word 0 – Bit 0 – Liberação Chamada Interna A
 - Word 0 – Bit 1 – Liberação Chamada Interna B

4.6 CONFIGURAÇÃO DOS PARAMETROS DO INVERSOR

Para colocar os motores em funcionamento através do variador de frequência, deve-se definir os parâmetros do drive.

Foi utilizado a configuração no modo Avançado, por isso o primeiro parâmetro a ser definido é o de número 196 [Param Access Lvl] do grupo “Utility“. Definido em 1 para a opção Avançado mostrado na figura 4.12. Esta opção permite, diferentemente da opção 0 – Básico, mudar os parâmetros de comunicação, utilizados para mandar e receber dados pela rede DeviceNet.



Figura 4.13: Definindo o parâmetro Access Lvl.

Os parâmetros são divididos em arquivos. Cada arquivo agrupo parâmetros de mesma categoria, tornando a configuração do sistema mais fácil e intuitiva. Cada arquivo e suas principais funções estão descritos abaixo e o valor de cada parâmetro pode ser encontrado no anexo 2.

Monitor: Este arquivo possui apenas parâmetros do tipo leitura. São parâmetros de leitura dos dados do motor, por isso estes não precisam ser definidos

Motor Control: é o arquivo de definição dos parâmetros do controle do motor. Este arquivo é dividido em dois grupos. No grupo “Motor Data“ as informações técnicas do motor serão definidas de acordo com a placa de identificação do motor. No grupo “Torq Attributes“ são definidos o modo de performance do torque e máximas tensão e frequência.

Speed command: Neste grupo são definidos os valores das velocidades de referencia desejadas para o sistema. Ele é dividido em três grupos, sendo um para a limitação máxima e

mínima das velocidades, velocidades de referencia e velocidades predefinidas para serem usadas pelo drive.

Dynamic Control: Arquivo para a definição do controle dinâmico. Inclui aceleração e desaceleração, frenagem, parada, entre outros. Este arquivo é dividido em cinco grupos.

Utility: possui parâmetros de utilidade como a linguagem, funções de limpar os parâmetros, salvar e carregar arquivos de parâmetros. Normalmente é o primeiro arquivo a ser definido, pois as preferencias do sistema são definidas neste arquivo.

Communication: Define todos os parâmetros de comunicação da rede, como por exemplo, taxa de transmissão, porta de recebimento e envio de dados, permissões de acesso aos comandos.

Inputs e Outputs: Define a maneira que as entradas e saídas do drive serão usadas. As estradas e saídas podem ser do tipo analógicas ou digitais.

Todos os parâmetros de cada grupo de arquivos do modo Avançado– Controle Padrão para este projeto, estão listados na tabela em anexo.

4.6.1 CURVAS DE ACELERAÇÃO

Uma das principais características do Powerflex 70 para um sistema de elevadores é o controle de velocidade. Não só pelo fato de que a velocidade pode ser alterada por comandos desejados, mas pela maneira que esta mudança é realizada. No arquivo Dynamic Control, pode-se definir os tempos de aceleração e desaceleração do motor. Isso permite partidas e paradas sem trancos.

De acordo com a NM 207, paragrafo 9.7.4, o retardamento médio recomendado é de $0.2g_n$ ou $1,96 \text{ m/s}^2$. A velocidade máxima atingida pelo carro é de $0,03 \text{ m/s}$. Assim, escolheu-se para os parâmetros de tempo de aceleração e desaceleração, o valor de $0,5\text{s}$, que resulta em uma desaceleração máxima de $0,06\text{m/s}^2$.

O PowerFlex 70 tem a função de um soft-starter melhorado. Soft-starters são utilizados basicamente para partidas de motores de indução CA tipo gaiola, em substituição aos métodos estrela-triângulo, chave compensadora ou partida direta. Tem a vantagem de não provocar trancos no sistema, limitar a corrente de partida, evitar picos de corrente e ainda incorporar parada suave e proteções. A partida de um soft starter se dá gradativamente, podendo ser descrito por uma função $F(t) = At$, como observado na imagem 4.13, utilizando o tempo de aceleração de $0,5\text{s}$

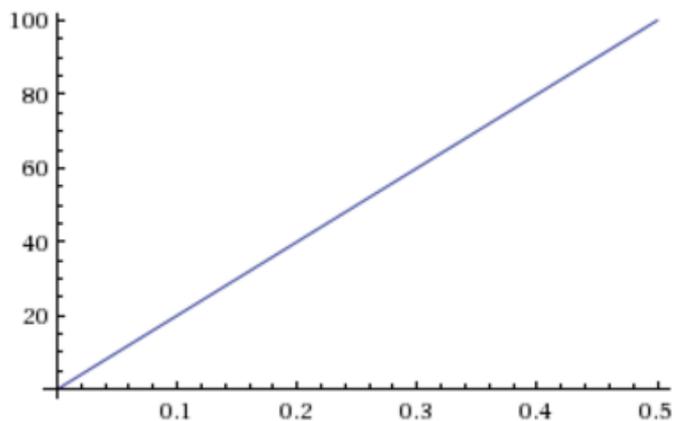


Figura 4.14: Partida soft-starter do controlador de velocidade.

Além da função básica de um soft starter, O PowerFlex permite a configuração do parâmetro S-curve, que permite que essa aceleração seja uma curva mais suave ainda, intensificando todos os benefícios do soft starter mencionados anteriormente. Podemos verificar o comportamento desta curva na imagem abaixo.

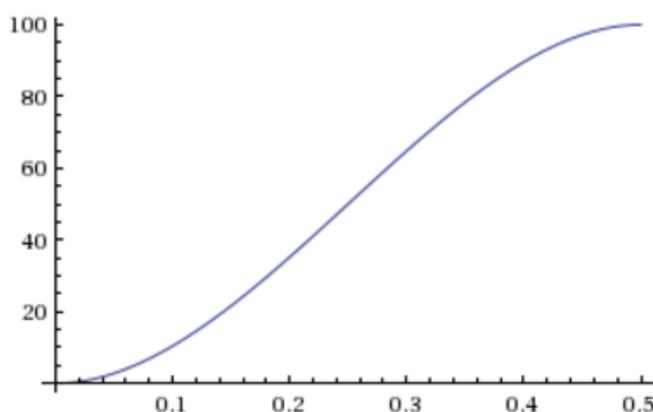


Figura 4.15: Partida utilizando S-Curve.

A curvatura vai depender da definição do parâmetro 146, descrito na imagem 4.15. O tempo de aceleração e desaceleração também aumenta de acordo com o valor definido. No caso dos elevadores, foi definido uma S Curve de 15%, levando o tempo de aceleração para 0,575.

146	<p>[S Curve %]</p> <p>Sets the percentage of accel or decel time that is applied to the ramp as S Curve. Time is added, 1/2 at the beginning and 1/2 at the end of the ramp.</p>	<p>Default: 0.0%</p> <p>Min/Max: 0.0/100.0%</p> <p>Units: 0.1%</p>
-----	---	--

Figura 4.16: Parâmetro de configuração 146 do PowerFlex 70.

4.6 ESTRATÉGIA DE CONTROLE

Anteriormente ao desenvolvimento do software propriamente dito, é de suma importância analisar a estratégia que será abordada para o desenvolvimento do mesmo. Como apresentado na seção 2.8, existem vários algoritmos que podem ser implementados para o transporte de passageiros utilizando elevadores.

Como este projeto tem por objetivo principal entregar um sistema funcional, integrando todas as partes, não será necessário o desenvolvimento de algoritmos mais sofisticados e otimizados para transporte de passageiros, até mesmo porque o protótipo não possui nem escala real nem funcionamento real. Além disso, a estrutura mecânica atualmente instalada possui apenas 6 andares, o que não restringe muito o tempo de movimentação e/ou de espera dos supostos passageiros.

Das estratégias apresentadas anteriormente, o algoritmo heurístico baseado em zonas, desenvolvido primeiramente pela empresa de elevadores Otis, se mostra como bastante interessante para ser aplicada nesta situação, por conta de seu simples desenvolvimento e adequação às características do projeto.

De acordo com essa lógica, todos os andares devem ser divididos em zonas onde cada um dos elevadores irá operar, atendendo somente chamadas dentro daquela zona. A partir desse momento, ambos os elevadores poderiam levar seus passageiros para quaisquer andar, retornando finalmente à sua zona de origem.

Para o projeto em questão, dividiu-se os 6 andares disponíveis para movimentação em duas zonas, a primeira correspondente aos três primeiros andares e a segunda correspondente aos três andares restantes.

Em seguida, atribui-se ao elevador A (elevador da esquerda), a cobertura de chamadas da primeira zona, e por consequência ao elevador B (da direita), a cobertura de chamadas da segunda zona.

Portanto, o primeiro elevador poderia receber chamadas externas dos andares 1, 2 e 3 enquanto que o segundo elevador receberia chamadas externas dos andares 4, 5 e 6. Foi atribuído a cada um dos elevadores, um andar dentro de sua zona onde seria seu local de origem; para o elevador A foi o segundo andar e para o elevador B o quinto andar.

Podemos entender como chamada externa aquela em que o elevador não possui nenhum passageiro ainda em seu interior, este se movimentará para o andar da chamada de modo a aguardar a entrada do passageiro e a sua chamada de destino, que chamaremos de chamada interna.

Após a movimentação dos elevadores para qualquer um dos andares dentro de sua zona e a suposta entrada do passageiro, cada um dos elevadores poderá receber uma chamada interna para qualquer um dos seis andares disponíveis, levando os passageiros de suas zonas para qualquer andar da edificação.

Por fim, ao chegar ao andar de destino do passageiro e aguardar por um tempo sua saída da cabina, retornaremos os elevadores às suas zonas e andar de origem para que possam aguardar novas chamadas externas, reiniciando o processo.

4.7 MODELAGEM E PROGRAMAÇÃO

4.7.1 MODELAMENTO POR REDE DE PETRI

Redes de Petri são melhor utilizadas em sistemas com eventos e transições bem definidos, resultando em estados também bem definidos. CSP utiliza a ocorrência sequencial de operações para definir o próximo e os possíveis estados do sistema e a UML é utilizada principalmente no desenvolvimento de software, utilizando conceitos de linguagens orientadas a objeto para descrever as relações do sistema.

Dentre as técnicas de modelagem existentes, utilizaremos as Redes de Petri por terem um forte apelo gráfico, muito adequado para o projeto e verificação de diversos sistemas a eventos discretos, como os que deseja-se implementar. Outro fator importante para a escolha das Redes de Petri como modelo é devido a sua lógica e dinâmica serem muito similar com a linguagem de programação Ladder.

Com a estratégia de controle bem traçada, devemos então realizar o desenvolvimento da modelagem do sistema, verificando as possibilidades de implementação e se todos os casos necessários são alcançáveis e aplicáveis.

Como explicado na seção anterior, utilizaremos uma estratégia de controle baseado em zonas, onde cada elevador receberá chamadas somente se elas pertencerem à mesma e em seguida, serão capazes de se deslocar para quaisquer andar e em seguida retornar a seu andar de origem.

Podemos perceber de início, que a modelagem para o elevador A e para o elevador B serão bem semelhantes, uma vez que estes realizam o mesmo comportamento porém, cobrindo andares diferentes. Apesar disso, foi realizado a modelagem para ambos numa mesma Rede de Petri.

Como esta serve para modelar sistemas de eventos discretos, é de grande importância perceber quais são os eventos principais que farão parte do processo de movimentação dos elevadores e garantir que estes sejam seguidos, se dando a separação entre um evento e outro através de transições.

Após extensa análise do sistema, foi destacado como eventos principais os seguintes:

- Elevador A/B em *idle*;
- Elevador A no andar 1, 2 ou 3;
- Elevador B no andar 4, 5 ou 6;
- Elevador A movimentando para andar 1, 2, 3, 4, 5 ou 6;
- Elevador B movimentando para andar 1, 2, 3, 4, 5 ou 6;
- Elevador A retornando a origem;
- Elevador B retornando a origem.

Além disso, é necessária a definição das transições principais, que são:

- Chamadas externas no andar 1, 2, 3, 4, 5 ou 6;
- Chamadas internas no Elevador A para andar 1, 2, 3, 4, 5 ou 6;
- Chamadas internas no Elevador B para andar 1, 2, 3, 4, 5 ou 6.

De uma maneira geral, podemos modelar a movimentação completa de todo o sistema utilizando praticamente estas características, como pode ser observado na figura abaixo.

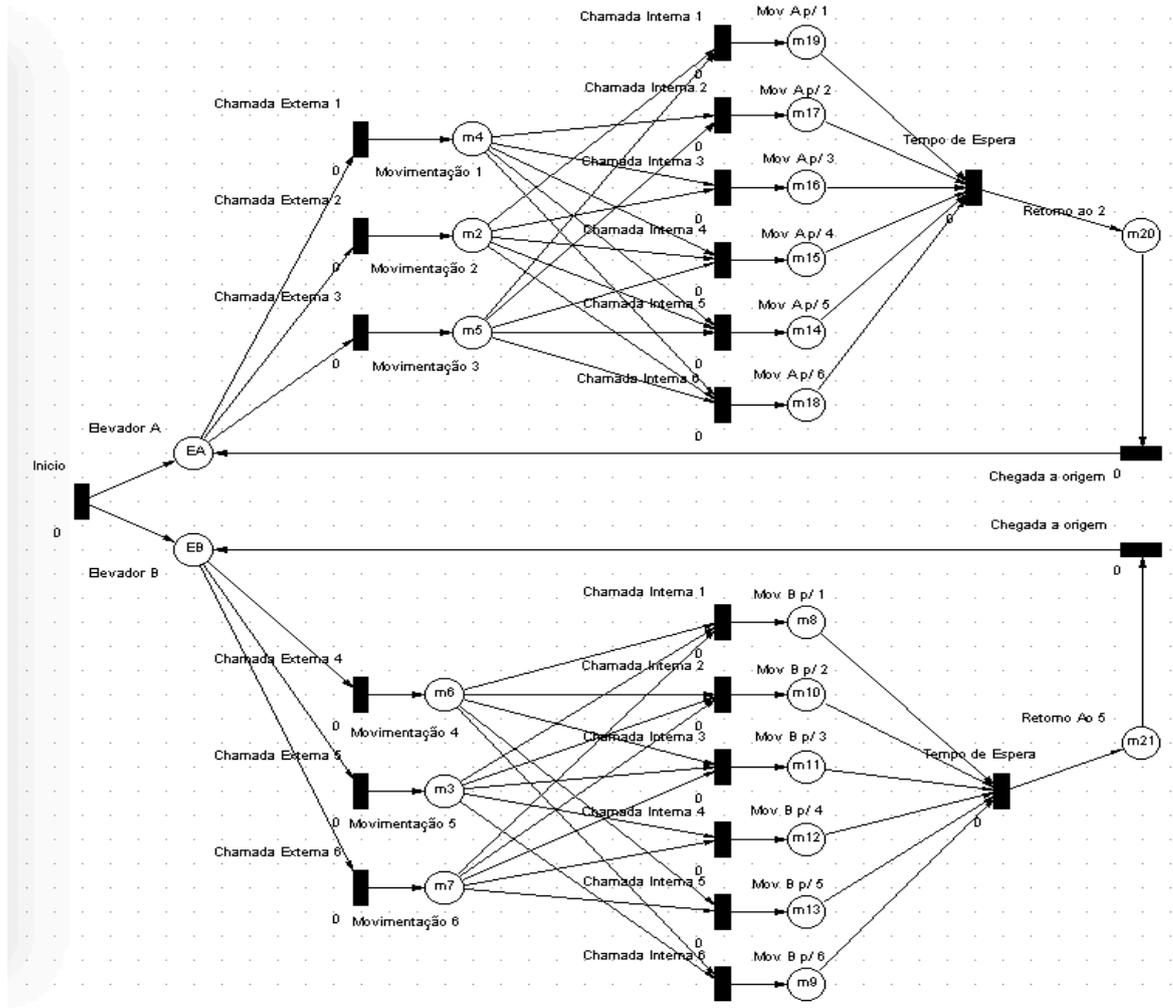


Figura 4.17: Rede de Petri para a movimentação completa do sistema.

É importante mencionar também que, para simplificação da visualização e para diminuir a poluição da imagem anterior, suprimimos nela a parte responsável pela movimentação dos elevadores para os respectivos andares de destino.

Podemos observar na figura abaixo que, para qualquer tipo de movimentação que for necessária para qualquer um dos elevadores, a mesma sequência de passos pode ser seguida, simplificando bastante a modelagem para todo o sistema e a implementação final em linguagem de programação.

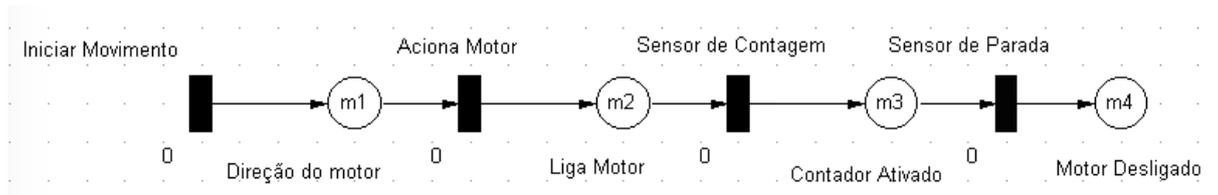


Figura 4.18: Sequência de passos para movimentação entre-andares.

A partir da análise da Rede de Petri e utilizando o conceito de redes marcadas, podemos verificar que nesta estratégia de controle somente uma chamada por vez é aceita

pelo elevador, necessitando assim que ele percorra todo o ciclo e retorne ao andar de origem para que receba uma nova chamada.

4.7.2 DIAGRAMA LADDER

A linguagem Diagrama Ladder é uma linguagem visual, que não possui o mesmo poder de implementação quanto linguagens textuais. Porém a facilidade de interpretação e rapidez com que pode-se implementar e executar são atrativos. Além disso, por ser a linguagem mais usada, ela possui vasto suporte técnico. Esses fatores fazem dela uma ótima escolha para a operação de elevadores.

O problema da linguagem Diagrama Ladder é que a complexidade aumenta à medida que o sistema cresce. Assim, um sistema que exibe muitas entradas e muitas saídas, além de inúmeras ligações dificulta a manutenção do código, gerando erros difíceis de serem localizados. Por isso é recomendado utilizar um modelo formal para desenvolver o algoritmo, antes da programação.

Utilizando a modelagem desenvolvida em Redes de Petri, a lógica de programação fica mais simples e intuitiva de ser desenvolvida. Pode-se ver no exemplos a seguir que a programação foi desenvolvida de acordo com a modelagem e isso facilitou muito a compreensão e organização, além de requerer menos tempo para pensar na lógica. Com isso, não se encontrou nenhuma dificuldade na linguagem Ladder por conta do tamanho do código ou em relação a sua manutenção. A programação completa pode ser encontrada no anexo 3.

Primeiramente, desenvolveu-se a lógica das chamadas externas, como é mostrado na figura 4.18. No momento em que a chamada externa é realizada, uma variável interna é acionada e a movimentação do elevador está liberada. Um timer de 1500ms é utilizado para desligar a chamada após sua ativação. Esse timer é importante, pois esse é o tempo necessário para que o carro inicie o movimento e o sensor de parada perca o contato com o alvo, impedindo este, de parar o movimento. Esta lógica é repetida para todas a chamadas externas.

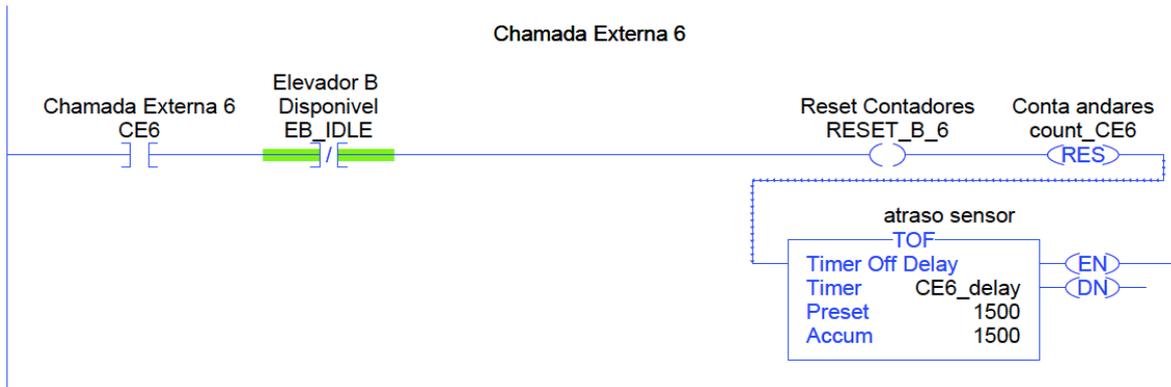


Figura 4.19: Lógica em ladder para chamadas externas.

A parte de movimentação do carro foi realizada de acordo com a rede de Petri secundária, que foi desenvolvida genericamente para esta finalidade. Na figura 4.19, observa-se a lógica de movimentação do quinto andar ao primeiro andar. Este bloco é repetido para todas as movimentações, com apenas algumas mudanças no contador de andares e na direção do movimento.

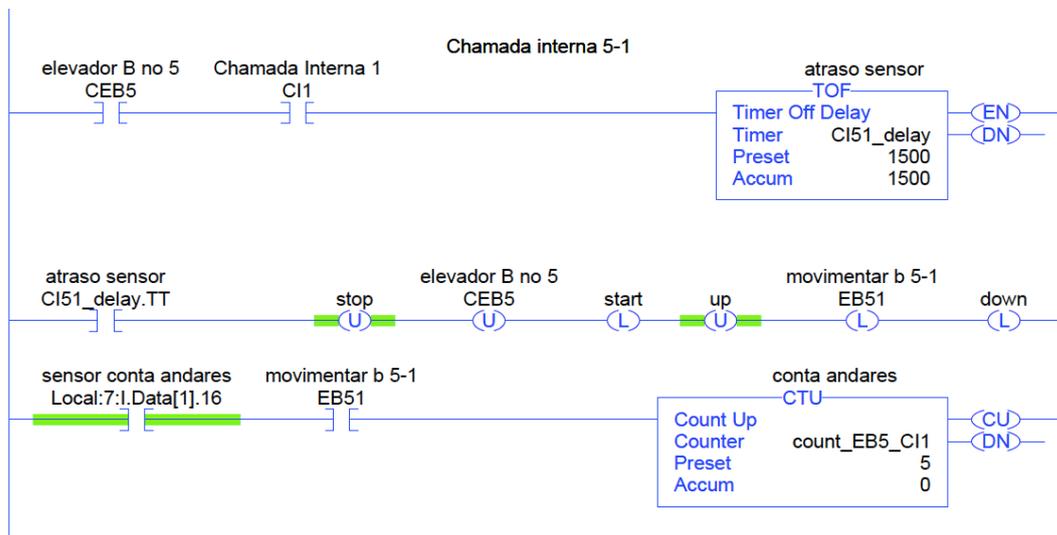


Figura 4.20: Movimentação do quinto ao primeiro andar.

Por fim, o elevador deve retornar a sua posição de origem, de acordo com a estratégia definida. Na figura 4.20, o retorno acontece do segundo andar ao quinto andar. Na lógica, tem um timer definido para este elevador ficar no andar ao qual a chamada interna foi designada. Caso o elevador em questão possuísse portas, seria adicionada a entrada de um sensor fotoelétrico da porta em serie com o timer. Dessa maneira, o elevador só iniciará seu movimento após a liberação da porta e o termino do tempo de espera.

Apos o tempo de espera, a movimentação do carro é iniciada e a lógica segue o mesmo padrão que a figura 4.19, mudando apenas o valor do contador de andares e a direção do movimento.

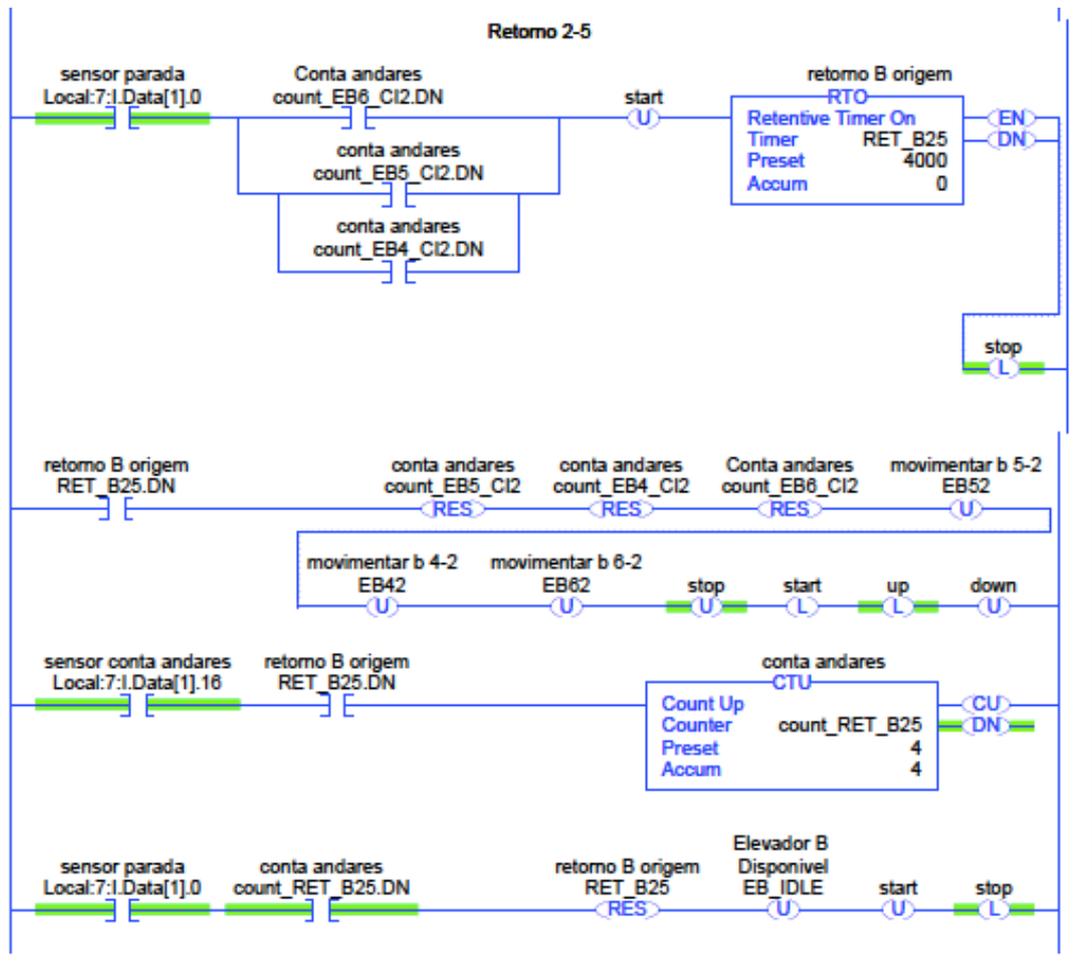


Figura 4.21: Retorno do segundo para o quinto andar.

4.8 PANELVIEW

O programa desenvolvido para o CLP necessita de informações sobre as chamadas dos elevadores. Em um elevador real, essas informações são enviadas através das botoeiras externas e internas dos elevadores. Como o nosso sistema não possui esse componentes instalados devido ao seu tamanho reduzido, houve a necessidade de programar esta IHM.

Usando o software PanelBuilder 32 foi possível construir a tela mostrada na figura 4.21. Esta figura apresenta todos os botões de chamadas externas e os botões de chamada interna de cada elevador. Quando o elevador está habilitado para receber chamadas internas, ou seja, ele chegou ao destino após uma chamada externa, existe um indicativo na tela sobre isso. Para realizar esta programação foram utilizados 3 Words(24 bits) de input e 1 Word(8 bits) de output.

Cada botão criado é ligado aos inputs do CLP e ao clicar em um deles, gera uma informação que será usada pela programação. O CLP manda informações sobre a

disponibilidade dos elevadores, para que ocorra o aviso de liberação do uso das chamadas internas. O detalhamento de cada bit está no Anexo 1 – Bits de entradas e saídas do CLP.

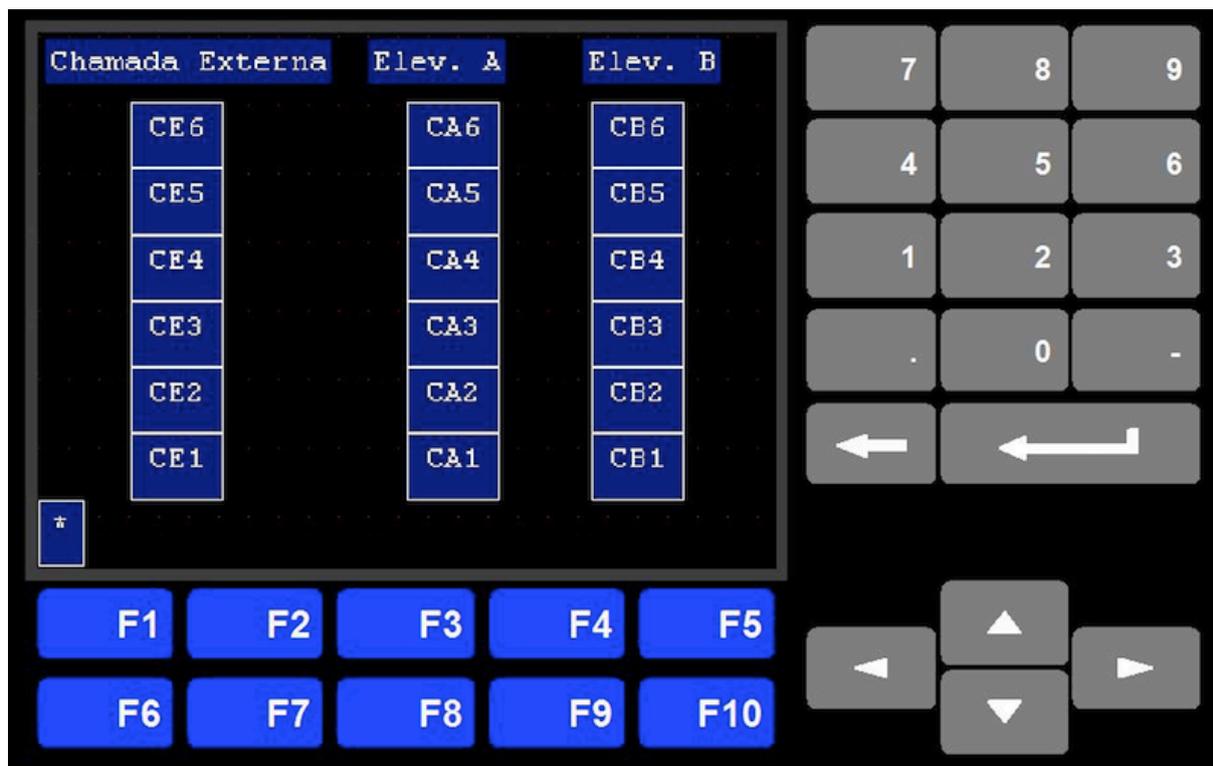


Figura 4.22: PanelView desenvolvido.

5 CONCLUSÃO

Neste trabalho, executado em várias etapas, foi realizado a análise e reconstrução dos equipamentos utilizados; a análise, identificação e montagem de diagramas dos equipamentos e circuitos elétricos; o planejamento estratégico e modelagem do controle que será utilizado e a programação do mesmo no CLP.

Na primeira etapa, já existia a estrutura física do sistema e a instalação mecânica da maior parte dos equipamentos, assim, foi feito um estudo intenso sobre o funcionamento dos mesmos, de modo a preparar as etapas seguintes sempre de acordo com a utilização correta dos equipamentos.

Apesar de que a estrutura física já existir, foi constatado que seria necessária algumas mudanças para o perfeito funcionamento dos elevadores. Primeiramente foi feita uma limpeza nos trilhos para garantir uma movimentação suave e constante das cabines. A seguir, foi necessário o reposicionamento dos sensores de final de curso além da melhor fixação dos mesmos para evitar qualquer falha da lógica do sistema quando esta fosse implementada. Por fim, foi feito testes com o sistema elétrico ainda desconectado, verificando que toda a parte mecânica estava pronta para utilização.

Na segunda etapa, verificou-se todo o cabeamento que havia instalado no sistema, etiquetando os fios com suas respectivas funcionalidades para garantir que numa eventual necessidade e/ou imprevisto, não seria perdido tempo com a reinstalação dos mesmos, uma vez que eles e suas posições já estariam corretamente sinalizadas. Além disso, criou-se um diagrama da instalação elétrica, permitindo manter a documentação do sistema atualizado. Os sensores foram colocados em rede, após resolver os problemas que nela existiam por conta da falta de resistência de fim de linha, e pode-se analisar que a posição deles era desfavorável para a detecção dos andares.

O reposicionamento dos sensores foi realizado para garantir a detecção. Os alvos que eles identificam foram fabricados e instalados e essa detecção foi testada para todos os andares. Com os alvos em posição, os testes tiveram 100% de detecção e não houve nenhuma falsa detecção. Mesmo que a amostragem dos testes não tenha sido muito alta, isso mostrou que o sistema estava confiável.

Na terceira etapa, que consistiu na modelagem do sistema, foi analisado diferentes tipos de algoritmos de controle de elevadores, como a lógica Fuzzy, algoritmos genéticos e algoritmos heurísticos. Analisou-se que o algoritmo mais adequado seria o heurístico baseado em zonas, por conta de sua simplicidade e eficiência para o projeto executado neste trabalho.

Com o algoritmo definido, o próximo passo foi realizar a modelagem do sistema. Esta foi feita por Redes de Petri, tendo sido muito interessante trabalhar com esse tipo de modelagem. A sua lógica e desenvolvimento similares à programação do CLP tornou o desenvolvimento do código muito tranquila e eficiente.

Ao desenvolver a programação no CLP, houve a necessidade de conectar com todos os dispositivos da rede. Ao longo desse processo, testamos as entradas e saídas conectadas diversas vezes. O PowerFlex 70, cuja função é o acionamento e o controle dos motores, teve seus parâmetros revisados diversas vezes para melhor se adequar ao projeto. Durante essa adequação, observou-se algumas funções específicas de curvas de aceleração. Função fundamental para deixar a desaceleração do elevador de acordo com a norma de transporte vertical de passageiros, NM 207.

Percebeu-se a partir das curvas de aceleração, que não haveria a necessidade de um sensor indutivo apenas para fazer a transição de uma velocidade alta para uma velocidade menor. O próprio PowerFlex 70 já estava fazendo isso de maneira mais eficiente e sem a necessidade de colocar mais um sensor na operação. Contatou-se isso a partir de medições de velocidade e aceleração e fez-se mudanças nos parâmetros de acordo com esses cálculos.

Outra constatação que teve-se durante a programação foi que não haveria a necessidade de um sensor para o local de parada e um sensor para a contagem dos andares. Utilizando o mesmo sinal, em diferentes funções, o CLP é capaz de fazer a contagem de andares e a parada do elevador no local desejado com apenas um sensor indutivo.

O trabalho realizado no ano de 2009, nessa mesma estrutura, fez um estudo de viabilidade financeira nos equipamentos. Cada sensor indutivo possuía um custo de R\$548,79. Retirando dois sensores de cada elevador, gerou uma economia de R\$ 2.195,16.

Por fim, com a programação preparada e otimizada, teve-se a necessidade de conectar entradas para as chamadas internas e externas dos elevadores. Assim desenvolveu-se um programa para o PanelView 600. Entretanto não foi possível receber as entradas e saídas desse equipamento. O mesmo conectou-se normalmente a rede DeviceNet, sendo possível fazer o download dos arquivos programados ao equipamento. Tentou-se consultar todo material possível encontrado no site do fabricante, porém esbarrou-se na falta de material por se tratar de um produto fora de linha desde 2004. Mesmo com a ajuda do Professor Guilherme Carvalho, não foi possível obter as entradas e saídas do equipamento.

Após o desenvolvimento do sistema como um todo, pode-se observar que foram desenvolvidos bons conhecimentos em várias áreas distintas, não se limitando a somente um

tipo de conhecimento, dimensionando e executando projetos em varias áreas de atuação, o que é pratica constante na engenharia.

5.1 TRABALHOS FUTUROS

Desde o princípio, o objetivo principal deste trabalho era o de detalhar o processo de desenvolvimento de um sistema de elevadores, exemplificando e explicando todas as escolhas de modo que ao final, teríamos como resultado um sistema funcional, robusto e sem falhas.

Alcançamos com sucesso o objetivo proposto e, apesar de alguns contratemplos, fomos capazes de desenvolver um projeto com estrutura mecânica e elétrica completamente funcional e documentada; além disso, fomos capazes de solucionar os problemas presente na rede industrial de comunicação o que nos permitiu com sucesso aplicar a lógica de controle a modelagem idealizada para o projeto, realizando uma programação bastante robusta em linguagem de programação ladder.

Desta forma, vemos agora várias possibilidades de estudos diferentes que podem ser realizados utilizando como base o projeto que foi desenvolvido ao longo deste documento.

Num primeiro instante, seria bastante interessante estudar as estratégias de controle abordadas aqui, além de outras existentes no mercado, e desenvolver projetos com base nas estatísticas de edificações do mundo real, de modo a otimizar a locomoção destes elevadores diminuindo assim custos operacionais e tempo de espera dos usuários, além de garantir que estes tenham uma visita agradável ao edificio.

No CLP utilizado para este projeto, existem varias estruturas lógicas pre-implementadas em seu próprio software que permitem o desenvolvimento destas lógicas mais avançadas, sem que haja a necessidade da implementação das mesmas utilizando somente as estruturas básicas, facilitando assim todo o processo.

Outro ponto que ficou sujeito a melhora foi desenvolvimento da IHM com os botões de chamadas dos elevadores. Infelizmente a produtora dos PanelViews nao oferece mais suporte aos mesmos, impossibilitando assim que nossas duvidas fossem sanadas e portanto não fomos capazes de entregar um modelo funcional.

6 BIBLIOGRAFIA

- 1 NBR NM 207 – Elevadores elétricos de passageiros.
- 2 ELEVADORES ATLAS SCHINDLER. Manual de Transporte Vertical em Edifícios.
- 3 CHARLES K. ALEXANDER; MATTHEW N. O. SADIKU. Motores Elétricos e Acionamentos: Série Tekne (2013).
- 4 BONACORSO, N. G.; NOLL, V. Automação Eletropneumática - 11.^a ed., São Paulo: Erica, 2004.
- 5 MAZAROPPI, MARCELO. Sensores de movimento e presença. UFRJ
- 6 CEFET-ES. Sensores de proximidade.
- 7 FRADEN, JACOB - Handbook of Modern Sensors: Physics, Designs, and Applications
- 8 ALLEN-BRADLEY .Datasheet DeviceNET sensors
- 9 WEG. Automação de Processos Industriais.
- 10 Imagem retirada da internet <<http://mycondoprojects.com/category/elevator>>
- 11 Imagem retirada da internet< https://www.pc-control.co.uk/distributed_control.htm>
- 12 SANTOS, KLEBER . Redes Industriais
- 13 STEMMER, MARCELO RICARDO. Sistemas distribuídos e redes de computadores para controle e automação industrial, Universidade Federal de Santa Catarina.
- 14 KOEHLER, JANA E OTTIGER, DANIEL. An AI-Based Approach to Destination Control in Elevators,.
- 15 PATIÑO F., ALVARO ANTONIO. Estudo e Simulação de Técnicas de controle de tráfego de grupo de elevadores usando automação industrial[Distrito Federal]2010.
- 16 SOBRAL, THAIS. Estudo e Análise de desempenho de algoritmos para elevadores inteligentes,.
- 17 CRITES,ROBERT e H BARTO ,ANDREW G Elevator Group Control Using Multiple Reinforcement Learning Agents. University of Massachusetts

- 18 CARVALHO, ANDRÉ. Algoritmos Genéticos. Universidade Federal de Pernambuco.
- 19 AGUADO, ALEXANDRE GARCIA E CANTANHEDE, MARCO ANDRÉ. Lógica Fuzzy.
- 20 SIIKONEN, MARJA-LIISA. Elevator Group Control with Artificial Intelligence.
- 21 GORMIDE ERNANDO. Fundamentos de modelagem de sistemas
- 22 DR. MARRANGHELLO, NORIAN . Redes de Petri: Conceitos e aplicações.
- 23 SANDERS, JW . An introduction to CSP.
- 24 VARGAS, THÂNIA. A história de UML e seus diagramas.
- 25 ATLAS SCHINDLER. Elevadores residenciais.
- 26 WEG. Inversores de Frequência.
- 27 ALLEN-BRADLEY. DeviceNet Sensors .
- 28 ROCKWELL AUTOMATION. Automação elétrica de processos industriais .
- 29 NUNES, T. C. O., RODRIGUES, V. H. (2009). Estudo da Viabilidade Técnica e Financeira da Aplicação da Rede Industrial DeviceNet na Construção e Operação de Elevadores de Passageiros. Trabalho de Graduação em Engenharia de Controle e Automação, Publicação FT TG-016/2009 Faculdade de Tecnologia, Universidade de Brasília, Brasília, DF, 124p.

7 ANEXOS

ANEXO 1 – BITS E/S DO CLP

Saídas do Scanner 6:

6:O Data [0] PowerFlex 70 Nó 01

7	6	5	4	3	2	1	0
MOP Increment	Local Control	Direction	Direction	Jog	Jog	Start	Stop
15	14	13	12	11	10	9	8
MOP Decrement	Reference Select	Reference Select	Reference Select	Decel Rate	Decel Rate	Accel Rate	Accel Rate

6:O Data [1] PanelView 600 Nó 40

7	6	5	4	3	2	1	0
						Elevador B em espera	Elevador A em espera

Saídas do Scanner 7:

7:O Data [0] PowerFlex 70 Nó 02

7	6	5	4	3	2	1	0
MOP Increment	Local Control	Direction	Direction	Jog	Jog	Start	Stop
15	14	13	12	11	10	9	8
MOP Decrement	Reference Select	Reference Select	Reference Select	Decel Rate	Decel Rate	Accel Rate	Accel Rate

Entradas do Scanner 6:

6:I Data [0] PowerFlex70 N° 01

7 At speed	6 Fault	5 Alarm	4 Decel	3 Accel	2 Command direction	1 Active	0 Ready
15 Reference Source	14 Reference Source	13 Reference Source	12 Reference Source	11 Reference Source	10 Local Control	9 Local Control	8 Local Control

6:I Data [1] Sensor Indutivo N° 55

7	6 Motion	5 Counter output	4 Too Far	3 Too Close	2 Major Faults	1 Diagnostic	0 State Bit
---	-------------	---------------------	--------------	----------------	-------------------	-----------------	----------------

6:I Data [2] Módulo Flex IO 1794-AND N° 02

23 Sensor BI corta Velocidade	22 Sensor BS corta Velocidade	21 Sensor AS Parada	20 Sensor AI Parada	19 Sensor AI corta Velocidade	18 Sensor AS corta Velocidade	17 Sensor AS Parada	16 Contator Emergência Elevador A
31	30	29	28	27	26	25 Contator Emergência Elevador B	24 Sensor BI Parada

6:I Data [3] PowerFlex N° 40

7	6	5 Chamada Externa 6° Andar	4 Chamada Externa 5° Andar	3 Chamada Externa 4° Andar	2 Chamada externa 3° Andar	1 Chamada Externa 2° Andar	0 Chamada Externa 1° Andar
15	14	35 Chamada Elevador A 6° Andar	12 Chamada Elevador A 5° Andar	11 Chamada Elevador A 4° Andar	10 Chamada Elevador A 3° Andar	9 Chamada Elevador A 2° Andar	8 Chamada Elevador A 1° Andar
24	23	22 Chamada Elevador B 6° Andar	20 Chamada Elevador B 5° Andar	19 Chamada Elevador B 4° Andar	18 Chamada Elevador B 3° Andar	17 Chamada Elevador B 2° Andar	16 Chamada Elevador B 1° Andar

Entradas do Scanner 7:

7:I Data [0] PowerFlex70							Nó 01
7 At speed	6 Fault	5 Alarm	4 Decel	3 Accel	2 Command direction	1 Active	0 Ready
15 Reference Source	14 Reference Source	13 Reference Source	12 Reference Source	11 Reference Source	10 Local Control	9 Local Control	8 Local Control

7:I Data [1] Sensor Indutivo							Nó 50
7	6 Motion	5 Counter output	4 Too Far	3 Too Close	2 Major Faults	1 Diagnostic	0 State Bit

ANEXO 2 – PARAMETROS DO INVERSOR

No.	Parameter Name	Default	Value
1	Output Freq	Read Only	read only
2	Commanded Freq	Read Only	read only
3	Output Current	Read Only	read only
4	Torque Current	Read Only	
5	Flux Current	Read Only	
6	Output Voltage	Read Only	
7	Output Power	Read Only	
8	Output Powr Fctr	Read Only	
9	Elapsed MWh	Read Only	
10	Elapsed Run Time	Read Only	
11	MOP Frequency	Read Only	
12	DC Bus Voltage	Read Only	read only
13	DC Bus Memory	Read Only	
14	Elapsed KWh	Read Only	
15	Torque Estimate	Read Only	
16	Analog In1 Value	Read Only	
17	Analog In2 Value	Read Only	
22	Ramped Speed	Read Only	
23	Speed Reference	Read Only	
24	Commanded Torque	Read Only	
25	Speed Feedback	Read Only	
26	Rated kW	Read Only	
27	Rated Volts	Read Only	
28	Rated Amps	Read Only	
29	Control SW Ver	Read Only	
40	Motor Type	0	
41	Motor NP Volts	Based on Drive	380
42	Motor NP FLA	Based on Drive	0.880
43	Motor NP Hertz	Based on Drive	60
44	Motor NP RPM	Based on Drive	1060
45	Motor NP Power	Based on Drive	0.25
46	Mtr NP Pwr Units	Based on Drive	0
47	Motor OL Hertz	Motor NP / 3	20
48	Motor OL Factor	1,00	
49	Motor Poles	4	
50	Motor OL Mode	0	
53	Motor Cntl Sel	0	0
54	Maximum Voltage	Based on Drive	Based on Drive
55	Maximum Freq	110.0 or 130.0	69
56	Compensation	0x1xx11011	
57	Flux Up Mode	0	
58	Flux Up Time	0.00 secs	
59	SV Boost Filter	500	
61	Autotune	3	3

62	IR Voltage Drop	Based on Drive	
No.	Parameter Name	Default	Value
63	Flux Current Ref	Based on Drive	
64	Ixo Voltage Drop	Based on Drive	
66	Autotune Torque	50,00%	
67	Inertial Autotune	0	
69	Start/Acc Boost	Based on Drive	
70	Run Boost	Based on Drive	
71	Break Voltage	Motor NP Volts x 0.25	
72	Break Frequency	Motor NP Hertz x 0.25	
80	Feedback Select	0	
81	Minimum Speed	0.0 Hz	0
82	Maximum Speed	50.0 or 60.0 hz	60
83	Overspeed Limit	10.0 Hz	
84	Skip Frequency 1	0.0 Hz	
85	Skip Frequency 2	0.0 Hz	
86	Skip Frequency 3	0.0 Hz	
87	Skip Freq Band	0.0 Hz	
88	Speed/Torque Mod	1	
90	Speed Ref A Sel	2	2
91	Speed Ref A Hi	[Maximum Speed]	[Maximum Speed]
92	Speed Ref A Lo	0.0 Hz	0
93	Speed Ref B Sel	11	11
94	Speed Ref B Hi	[Maximum Speed]	[Maximum Speed]
95	Speed Ref B Lo	0.0 Hz	0.0 Hz
96	TB Man Ref Sel	1	1
97	TB Man Ref Hi	[Maximum Speed]	[Maximum Speed]
98	TB Man Ref Lo	0.0 Hz	0.0 Hz
100	Jog Speed 1	10.0 Hz	10.0 Hz
101	Preset Speed 1	5.0 Hz	5.0 Hz
102	Preset Speed 2	10.0 Hz	10.0 Hz
103	Preset Speed 3	20.0 Hz	20.0 Hz
104	Preset Speed 4	30.0 Hz	15.0 Hz
105	Preset Speed 5	40.0 Hz	30.0 Hz
106	Preset Speed 6	50.0 Hz	15.0 Hz
107	Preset Speed 7	60.0 Hz	30.0 Hz
108	Jog Speed 2	10.0 Hz	
116	Trim % Setpoint	0,0%	
117	Trim In Select	2	
118	Trim Out Select	x000	
119	Trim Hi	60.0 Hz	
120	Trim Lo	0.0 Hz	
121	Slip RPM @ FLA	Based on Motor NP RPM	
122	Slip Comp Gain	40,0	
123	Slip RPM Meter	Read Only	
124	PI Configuration	0	

125	PI Control	0	
No.	Parameter Name	Default	Value
126	PI Reference Sel	0	
127	PI Setpoint	50,0%	
128	PI Feedback Sel	2	
129	PI Integral Time	2.00 Secs	
130	PI Prop Gain	1,00	
131	PI Lower Limit	"- Max Frequency"	
132	PI Upper Limit	"+ Max Frequency"	
133	PI Preload	0.0 Hz	
134	PI Status	Read Only	
135	PI Ref Meter	Read Only	
136	PI Fdback Meter	Read Only	
137	PI Error Meter	Read Only	
138	PI Output Meter	Read Only	
139	PI BW Filter	0.0 R/s	
140	Accel Time 1	10,0 Secs	0,5
141	Accel Time 2	10,0 Secs	1
142	Decel Time 1	10,0 Secs	0,5
143	Decel Time 2	10,0 Secs	1
145	DB While Stopped	0	
146	S Curve %	0,0%	15%
147	Current Lmt Sel	0	0
148	Current Lmt Val	Rated Amps x 1.5	Rated Amps x 1.5
149	Current Lmt Gain	250	
150	Drive OL Mode	3	
151	PWM Frequency	4 kHz	
152	Droop RPM @ FLA	0.0 RPM	
153	Regen Power Lim	-50,00%	
154	Current Rate Lim	400%	
155	Stop/Brk Mode A	1	1
156	Stop/Brk Mode B	0	1
157	DC Brake Lvl Sel	0	0
158	DC Brake Level	Based on Drive	Based on Drive
159	DC Brake Time	0,0 sec	0,0 sec
160	Bus Reg Ki	450	
160	Bus Reg Gain	450	
161	Bus Reg Mode A	1	1
162	Bus Reg Mode B	4	4
163	DB Resistor Type	0	0
163	DB Resistor Type	2	2
164	Bus Reg Kp	1500	
165	Bus Reg Kd	1000	
166	Flux Braking	0	
167	Powerup Delay	0.0 Secs	
168	Start At PowerUp	0	0

169	Flying Start En	0	
No.	Parameter Name	Default	Value
170	Flying StartGain	4000	
174	Auto Rstrt Tries	0	0
175	Auto Rstrt Delay	1,0 Secs	1,0 Secs
178	Sleep Wake Mode	0,0 Secs	
179	Sleep Wake Ref	2,0 Secs	
180	Wake Level	6.000 mA, 6.000 Volts	
181	Wake Time	1.0 Sec	
182	Sleep Level	5.000 mA, 5.000 Volts	
183	Sleep Time	1.0 Sec	
184	Power Loss Mode	0	0
185	Power Loss Time	0.5 Secs	0.5 Secs
187	Load Loss Level	200,00%	
188	Load Loss Time	0.0 Secs	
189	Shear Pin Time	0.0 secs	
190	Direction Mode	0	0
192	Save HIM Ref	0	
192	AutoMan Cnfg	0101	
193	Man Ref Preload	0	
194	Save MOP Ref	00	
195	MOP Rate	1.0 Hz/s	
196	Param Access Lvl	0	1
197	Reset To Defaults	0	0
198	Load Frm Usr Set	0	1
199	Save To User Set	0	1 - > 0
200	Reset Meters	0	
201	Language	0	7
202	Voltage Class	Based on Drive	
203	Drive Checksum	Read Only	
204	Dyn UsrSet Cnfg	00	
205	Dyn UserSet Sel	00	
206	Dyn UserSet Actv	0000	
209	Drive Status 1	Read Only	
210	Drive Status 2	Read Only	
211	Drive Alarm 1	Read Only	
212	Drive Alarm 2	Read Only	
213	Speed Ref Source	Read Only	
214	Start Inhibits	Read Only	read only
215	Last Stop Source	Read Only	
216	Dig In Status	Read Only	read only
217	Dig Out Status	Read Only	read only
218	Drive Temp	Read Only	
219	Drive OL Count	Read Only	
220	Motor OL Count	Read Only	
221	Mtr OL Trip Time	Read Only	

222	Drive Status 3	Read Only	
No.	Parameter Name	Default	Value
223	Status 3 @ Fault	Read Only	
224	Fault Frequency	Read Only	
225	Fault Amps	Read Only	
226	Fault Bus Volts	Read Only	
227	Status 1 @ Fault	Read Only	
228	Status 2 @ Fault	Read Only	
229	Alarm 1 @ Fault	Read Only	
230	Alarm 2 @ Fault	Read Only	
234	Testpoint 1 Sel	499	
235	Testpoint 1 Data	Read Only	
236	Testpoint 2 Sel	499	
237	Testpoint 2 Data	Read Only	
238	Fault Config 1	00x0001001x10	00x0001001x10
240	Fault Clear	0	
241	Fault Clear Mode	1	
242	Power Up Marker	Read Only	
243	Fault 1 Code	Read Only	
244	Fault 1 Time	Read Only	
245	Fault 2 Code	Read Only	
246	Fault 2 Time	Read Only	
247	Fault 3 Code	Read Only	
248	Fault 3 Time	Read Only	
249	Fault 4 Code	Read Only	
250	Fault 4 Time	Read Only	
259	Alarm Config 1	00001111x111111	
270	DPI Data Rate	0	125 Kbps
271	Drive Logic Rslt	Read Only	
272	Drive Ref Rslt	Read Only	
273	Drive Ramp Rslt	Read Only	
274	DPI Port Select	0	5
275	DPI Port Value	Read Only	
276	Logic Mask	111111	
277	Start Mask	111111	
278	Jog Mask	111111	
279	Direction Mask	111111	
280	Reference Mask	111111	
281	Accel Mask	111111	
282	Decel Mask	111111	
283	Fault Clr Mask	111111	
284	MOP Mask	111111	
285	Local Mask	111111	
288	Stop Owner	Read Only	
289	Start Owner	Read Only	
290	Jog Owner	Read Only	

291	Direction Owner	Read Only	
No.	Parameter Name	Default	Value
292	Reference Owner	Read Only	
293	Accel Owner	Read Only	
294	Decel Owner	Read Only	
295	Fault Clr Owner	Read Only	
296	MOP Owner	Read Only	
297	Local Owner	Read Only	
298	DPI Ref Select	0	
300	Data In A1	0	
301	Data In A2	0	
302	Data In B1	0	
303	Data In B2	0	
304	Data In C1	0	
305	Data In C2	0	
306	Data In D1	0	
307	Data In D2	0	
308	HighRes Ref	0	
310	Data Out A1	0	
311	Data Out A2	0	
312	Data Out B1	0	
313	Data Out B2	0	
314	Data Out C1	0	
315	Data Out C2	0	
316	Data Out D1	0	
317	Data Out D2	0	
320	Anlg In Config	0	0
321	Anlg In Sqr Root	0	
322	Analog In 1 Hi	10.000 Volt	10.000 Volt
323	Analog In 1 Lo	0.000 Volt	0.000 Volt
324	Analog In 1 Loss	0	
325	Analog In 2 Hi	10.000 Volt	10.000 Volt
326	Analog In 2 Lo	0.000 Volt	0.000 Volt
327	Analog In 2 Loss	0	
340	Anlg Out Config	0	
341	Anlg Out Absolut	1	
342	Analog Out1 Sel	0	0
343	Analog Out1 Hi	10.0 Volt	10.0 Volt
344	Analog Out1 Lo	0.0 Volt	0.0 Volt
354	Anlg Out1 Scale	0	
361	Digital In1 Sel	4	Not Used
362	Digital In2 Sel	5	Not Used
363	Digital In3 Sel	18	Not Used
364	Digital In4 Sel	15	Not Used
365	Digital In5 Sel	16	Not Used
366	Digital In6 Sel	17	Not Used

377	Anlg Out1 Setpt	0.0 Volts	
No.	Parameter Name	Default	Value
379	Dig Out Setpt	0	
380	Digital Out1 Sel	1	1
381	Dig Out1 Level	0	0
382	Dig Out1 OnTime	0.0 Secs	
383	Dig Out1 OffTime	0.0 Secs	
384	Digital Out2 Sel	4	4
385	Dig Out2 Level	0	0
386	Dig Out2 OnTime	0.0 Secs	
387	Dig Out2 OffTime	0.0 Secs	
411	DigIn DataLogic	xx000000xx000000	

ANEXO 3 – PROGRAMAÇÃO LADDER

