



Universidade Federal de Campina Grande

Centro de Engenharia Elétrica e Informática

Curso de Graduação em Engenharia Elétrica

EZEQUIEL DE SOUZA BATISTA

**INSTRUMENTAÇÃO E CONTROLE DE ESTEIRA
TRANSPORTADORA UTILIZANDO CLP**

Campina Grande, Paraíba
Dezembro de 2015

EZEQUIEL DE SOUZA BATISTA

INSTRUMENTAÇÃO E CONTROLE DE ESTEIRA
TRANSPORTADORA UTILIZANDO CLP

*Trabalho de Conclusão de Curso submetido à
Unidade Acadêmica de Engenharia Elétrica da
Universidade Federal de Campina Grande
como parte dos requisitos necessários para a
obtenção do grau de Bacharel em Ciências no
Domínio da Engenharia Elétrica.*

Área de Concentração: Automação Industrial

Orientador:

Professor George Acioli Júnior, D.Sc.

Campina Grande, Paraíba
Dezembro de 2015

EZEQUIEL DE SOUZA BATISTA

INSTRUMENTAÇÃO E CONTROLE DE ESTEIRA TRANSPORTADORA UTILIZANDO CLP

*Trabalho de Conclusão de Curso submetido à Unidade
Acadêmica de Engenharia Elétrica da Universidade
Federal de Campina Grande como parte dos requisitos
necessários para a obtenção do grau de Bacharel em
Ciências no Domínio da Engenharia Elétrica.*

Área de Concentração: Automação Industrial

Aprovado em / /

Professor Avaliador
Universidade Federal de Campina Grande
Avaliador

Professor George Acioli Júnior , D.Sc.
Universidade Federal de Campina Grande
Orientador, UFCG

Dedico este trabalho aos meus pais: Vital Batista
Januário e Maria Nazaré de Souza Batista.

AGRADECIMENTOS

Agradeço primeiramente a Deus que esteve sempre presente em minha vida, que não me deixou fracassar mesmo nos momentos mais difíceis colocando anjos ao meu lado, na forma de amigos, familiares e desconhecidos.

Agradeço aos meus pais, Vital Batista Januário e Maria Nazaré de Souza Batista que sempre me apoiaram e me ensinaram tudo o que sei e tudo o que sou. Agradeço as minhas irmãs Katia Batista, Gessica Batista e Eleonore Batista pelo apoio e carinho.

Agradeço a minha querida sobrinha Rebeca Batista e a todos os meu familiares presentes e ausentes que sempre me fortaleceram com seus conselhos e me inspiraram com suas vidas.

Agradeço a meus irmãos Cleydstone Chaves e Jadailton Silva por toda ajuda, conselhos e carinho.

Agradeço a minha namorada, amiga e parceira Géssica Virgínio pelo apoio, carinho e compreensão.

Agradeço a meus amigos e irmãos "os manés" Mikhail Barros, Bianca Cartaxo, Elton Brasil, Mariana Ribeiro, Nelson Campos e em especial ao meu amigo Snadgyell Leal pelo acolhimento em sua casa nas várias madrugadas que precisei, pelas dicas e correções e por sempre estar disposto a me ajudar, mesmo quando cansado e cheio de ocupações. Agradeço ao meu amigos Jhonnathas Lima, Tulio Roberto, Julio Lima e demais amigos do Ed. Micheli por toda colaboração e torcida.

Agradeço aos meus amigos Rafael Bezerra, Moises Tavares e Kleber Oliveira que sempre me responderam com carinho a todas as perguntas que os fiz.

Agradeço ao meu amigo Simões Soares por toda sua considerável contribuição ao longo do trabalho e por sempre se mostrar disposto a me ajudar.

Por fim, agradeço ao caríssimo professor George Acioli Junior que contribuiu de maneira significativa na execução deste trabalho e ao professor Pericles Rezende de Barros pela oportunidade e confiança depositada.

*"Um guerreiro não desiste do que ele ama.
Ele encontra amor no que faz. Ser guerreiro não exige perfeição.
Ou vitória. Ou invulnerabilidade. Ele é a vulnerabilidade absoluta.
Essa é a única coragem verdadeira"*

Dan Millman

RESUMO

Este trabalho trata-se da construção, instrumentação, identificação e controle de uma esteira transportadora. O dispositivo escolhido para realizar o controle deste equipamento será o CLP do kit de treinamento ZTK900, fornecido pela empresa HI Tecnologia. A esteira transportadora possui como elementos atuadores um motor CC, sobre o qual será implementado um controlador PID para o controle de velocidade da esteira em ambos os sentidos de rotação, resistências elétricas, utilizadas no módulo de aquecimento, as quais também serão controladas pelo CLP e ventoinhas, que serão utilizadas no módulo de resfriamento, como elementos sensores possuirá o *encoder* que será fixado ao eixo do motor para medição de posição e velocidade, barreiras ópticas para auxiliar na identificação da posição do objeto e sensores de temperatura LM35. Fixados sobre a esteira transportadora tem-se dois módulos, um de aquecimento rápido e outro de resfriamento rápido, estes módulos também foram construídos para este trabalho, eles compõem parte de alguns processos indústrias, sendo assim serão neste trabalho objetos de identificação e controle. Para realizar a identificação dos equipamentos citados, serão utilizadas algumas ferramentas do *software* MATLAB, especificamente o *System Identification* e o *PID Tuner*. A comunicação entre o dispositivo de controle CLP e o PC para esta tarefa de identificação e controle será realizada utilizando o padrão OPC, já a comunicação para programação do CLP será realizada através do software SPDSW fornecido pela empresa HI Tecnologia, fabricante do CLP. O CLP usado será o ZAP900 do kit de treinamento ZTK900, este kit também fornecido pela empresa HI Tecnologia possui módulos que possibilitam a leitura do *encoder*, entradas analógica e digital, saída analógica digital e saída PWM que serão utilizadas neste trabalho.

Palavras-chave: Esteira Transportadora, PWM, CLP, OPC, PID, *System Identification*, *PID Tuner*, SPDSW, HI Tecnologia.

Abstract

This paper deals with the construction, instrumentation, identification and control of a conveyor belt. The chosen device to perform control of this equipment will be the CLP Training Kit ZTK900, provided by HI Technology company. The conveyor belt has as actuator elements a DC motor, on which is implemented a PID controller for the speed control of the belt in both directions of rotation, electrical resistors used in the heating module, which are also controlled by the CLP and fans, to be used in the cooling module, such as sensing elements possess the encoder to be attached to the motor shaft to measure position and speed, light screens to help identify the position of the object and LM35 temperature sensors. Fixed onto the conveyor belt has two modules, a rapid heating and other rapid cooling, these modules were also constructed for this paper, they comprise part of some industrial processes, and thus will be in this paper, objects of identification and control. To perform the identification of said equipments, some tools of MATLAB software will be used, specifically the System Identification and the PID Tuner. The communication between the CLP control device and PC for this task of identification and control will be performed using the OPC standard, since communication for CLP programming will be performed by SPDSW software provided by the company HI Technology, CLP manufacturer. The CLP will be used the ZAP900 training kit ZTK900, this kit also provided by the company HI Technology has modules that enable the reading of the encoder, analog and digital inputs, digital analog output and PWM output to be used in this paper.

Keywords: Belt Conveyor, PWM, CLP, OPC, PID, System Identification, PID Tuner, SPDSW, HI Technology.

LISTA DE ILUSTRAÇÕES

Figura 1: Processo em malha fechada utilizando um controlador PID.....	19
Figura 2: CLP <i>CompactLogix</i> com cartões de entradas e saídas.....	20
Figura 3: Unidades básicas do CLP.....	21
Figura 4: Rotina de operação da CPU.....	22
Figura 5: Imagem da janela principal do <i>software System Identification Toolbox</i>	24
Figura 6: Plataforma base da esteira transportadora.....	26
Figura 7: Desenho Técnico Plataforma esteira transportadora.....	27
Figura 8: Kit de treinamento ZTK900.....	28
Figura 9: Vista em corte longitudinal de um típico motor CC de potência elevada.....	28
Figura 10: Armadura de um motor CC.....	29
Figura 11: Motor de corrente contínua, (Motor CC) utilizado.....	30
Figura 12: <i>Encoder</i> fixado ao eixo do motor.....	31
Figura 13: Partes do <i>encoder</i> ww-12.....	32
Figura 14: <i>Encoder</i> ww-12.....	32
Figura 15: Especificações de instalação.....	33
Figura 16: Diagrama de tempo ww-12.....	33
Figura 17: O objeto interrompe o feixe entre o emissor e o receptor.....	34
Figura 18: Sensor óptico de barreira instalado na esteira transportadora.....	34
Figura 19: Diagrama de alimentação LM35 e saída de sinal de tensão.....	35
Figura 20: Pinos de configuração para o encapsulamento TO-92.....	36
Figura 21: LM35 instalado no módulo de resfriamento.....	36
Figura 22: Interior do Módulo de Aquecimento.....	37
Figura 23: Módulo de Aquecimento instalado sobre a esteira.....	37
Figura 24: Desenho técnico do módulo de aquecimento.....	38
Figura 25: Resistência 1400 W utilizada no módulo de aquecimento.....	38
Figura 26: Ventoinha 12 V DC.....	38
Figura 27: Descrição do funcionamento do módulo de aquecimento.....	39
Figura 28: Módulo de resfriamento.....	40
Figura 29: Módulo de Resfriamento, com detalhes a ventoinha e o sensor LM35.....	40
Figura 30: Desenho técnico módulo de resfriamento.....	41
Figura 31: Princípio de Funcionamento módulo de Resfriamento.....	41
Figura 32: Fonte de alimentação UPS1095-5A.....	42
Figura 33: Circuito elétrico do dispositivo de conversão do sensor óptico de barreira.....	44
Figura 34: Dispositivo de conversão do sensor óptico de barreira.....	44
Figura 35: Circuito do dispositivo conversor de 0 a 10 V para saída PWM.....	45
Figura 36: Dispositivo conversor de 0 a 10 V para saída PWM.....	46
Figura 37: Circuito do dispositivo de identificação de CLP desligado.....	46
Figura 38: Dispositivo de identificação de CLP desligado.....	47
Figura 39: Conexão do <i>encoder</i>	47
Figura 40: Dispositivo conversor de pulsos de tensão.....	48
Figura 41: Circuito do dispositivo conversor de pulsos de tensão.....	48
Figura 42: Dispositivo de conversão para sensor LM35.....	49
Figura 43: Dispositivo de conversão para sensor LM35.....	49
Figura 44: Dispositivo de Conversão Sinais PWM.....	50
Figura 45: Circuitos do dispositivo de conversão de sinais PWM.....	50
Figura 46: Desenho técnico da esteira com dispositivos integrados.....	51
Figura 47: Desenho técnico com identificação dos dispositivos adicionados.....	51
Figura 48: Esteira transportadora com dispositivos integrados.....	52
Figura 49: Tela inicial software SPDSW.....	54
Figura 50: Criando uma nova aplicação no SPDSW.....	54
Figura 51: Salvando a nova aplicação no SPDSW.....	55
Figura 52: Módulos de <i>hardware</i> associados ao controlador ZAP900 do kit ZTK900.....	55
Figura 53: Configuração de <i>hardware</i> passo 1.....	56
Figura 54: Configuração de <i>hardware</i> passo 2.....	57

Figura 55: Configuração de <i>hardware</i> passo 3.....	57
Figura 56: Configuração de <i>hardware</i> passo 4.....	57
Figura 57: Configuração de <i>hardware</i> passo 5.....	58
Figura 58: Configuração de <i>Hardware</i> passo 6.....	58
Figura 59: Editor Ladder do SPDSW.	59
Figura 60: Arquivo <i>Setup Driver</i> HS1.....	60
Figura 61: Tela Inicial de Instalação.	60
Figura 62: Seleção do diretório de instalação.....	61
Figura 63: Seleção da opção " <i>Server</i> ".	61
Figura 64: Conceito " <i>node name</i> ".	62
Figura 65: Seleção da pasta de instalação dos atalhos.....	62
Figura 66: Finalização da instalação do <i>Driver</i>	63
Figura 67: Mensagem de erro.....	63
Figura 68: Configuração da porta de comunicação COM1.	64
Figura 69: Acesso ao <i>driver</i> na pasta de atalho escolhida na instalação.....	64
Figura 70: Seleção da opção " <i>Use Local Server</i> ".	65
Figura 71: Ferramenta de configuração do Servidor.	65
Figura 72: Adição de um canal de comunicação.	66
Figura 73: Adição de um " <i>Device</i> ".	66
Figura 74: <i>Datablock</i> e suas principais configurações.	67
Figura 75: Teste de comunicação.	68
Figura 76: Navegador de bibliotecas do Simulink.....	69
Figura 77: Experimento de identificação realizado no Simulink via OPC.	69
Figura 78: Diagrama de Blocos do Sistema de controle de posição e velocidade.....	72
Figura 79: Sistema de controle de posição e velocidade proposto.	73
Figura 80: Diagrama de Blocos do Sistema de controle para o módulo de aquecimento.....	74
Figura 81: Sistema de controle de temperatura para o módulo de aquecimento.	75
Figura 82: Diagrama de Blocos do Sistema de controle para o módulo de resfriamento.	75
Figura 83: Sistema de controle de temperatura para o módulo de resfriamento.....	76
Figura 84: Bloco MOV.....	77
Figura 85: Trecho de Código utilizando o Bloco MOV.	77
Figura 86: Bloco FCT.....	78
Figura 87: Bloco TMR.	79
Figura 88: Bloco FQG.....	80
Figura 89: Bloco PID.	81
Figura 90: Tela de configuração de Blocos de controle PID.....	82
Figura 91: Ambiente de simulação Simulink.	84
Figura 92: Gráfico dos dados coletados para identificação do sistema.	85
Figura 93: Sinais PWM para controle da velocidade no sentido anti-horário.	86
Figura 94: Sinais PWM para controle da velocidade do sentido horário.....	86
Figura 95: Resposta do sistema, de controle de velocidade do sentido anti-horário, a entrada degrau.	87
Figura 96: Resposta do sistema, de controle de velocidade do sentido horário, a entrada degrau.	88
Figura 97: Ambiente do <i>System Identification Toolbox</i> , com os modelos estimados, para velocidade no sentido anti-horário.....	89
Figura 98: Modelos com os melhores ajustes comparados com o modelo real para a velocidade no sentido anti-horário.	89
Figura 99: Ambiente do <i>System Identification</i> com os modelos estimados, para velocidade no sentido horário.	90
Figura 100: Comparação dos melhores modelos estimados para a velocidade no sentido horário.	91
Figura 101: Ambiente de simulação no Simulink para identificação do módulo de aquecimento.	92
Figura 102: Gráfico da resposta do sistema em malha aberta à entrada degrau.	93
Figura 103: Ambiente do <i>System Identification Toolbox</i> com os modelos estimados, para o módulo de aquecimento.	94
Figura 104: Comparação dos melhores modelos estimados para o módulo de aquecimento.	94
Figura 105: Gráfico do experimento para Identificação do sistema que representa a dinâmica do módulo de resfriamento.	96
Figura 106: Experimento de identificação realizado com entrada degrau, para o módulo de resfriamento.	96
Figura 107: Ambiente do <i>System Identification Toolbox</i> com os modelos estimados, para o módulo de resfriamento.....	97

Figura 108: Resultado dos melhores modelos estimados para o módulo de resfriamento.	97
Figura 109: Gráfico da resposta ao sistema de malha fechada do módulo de resfriamento.	98
Figura 110: Sintonia para o controlador de velocidade do sentido anti-horário.	99
Figura 111: Parâmetros do controlador PI da sintonia para velocidade no sentido anti-horário.	99
Figura 112: Diagrama de blocos do controle em malha fechada implementado no ambiente Simulink. .	100
Figura 113: Controle de velocidade em malha fechada no sentido anti-horário da esteira transportadora.	101
Figura 114: Sintonia para o controlador de velocidade do sentido horário.	102
Figura 115: Parâmetros do controlador PI da sintonia para velocidade no sentido horário.....	102
Figura 116: Controle de velocidade em malha fechada no sentido horário da esteira transportadora.	103
Figura 117: Sintonia para o controle de temperatura do módulo de aquecimento.....	104
Figura 118: Parâmetros do controlador PI da sintonia para a temperatura do módulo de aquecimento. ..	104
Figura 119: Controle de temperatura em malha fechada do módulo de aquecimento.	106
Figura 120: Detalhes do ponto onde o controle atinge o regime permanente.....	106

LISTA DE TABELAS

Tabela 1: Especificações Técnicas para a fonte UPS1095-5A	43
Tabela 2: Descrição dos Parâmetros, das Entradas e da Saída.	78
Tabela 3: Relação do parâmetro P2 com o tempo contabilizado, para o bloco TMR.....	79
Tabela 4: Descrição dos parâmetros, das entradas e da saída para o bloco FQG.	80
Tabela 5: Descrição dos parâmetros do bloco PID.....	81
Tabela 6: Relação da saída B1 com as entradas A1 e A2, do bloco PID.....	82

LISTA DE ABREVIATURAS E SIGLAS

CLP	Controlador Lógico Programável
PLC	<i>Programmable logic controller</i>
CPU	Unidade de Processamento Central
CAN	<i>Controller Area Network</i>
PID	Proporcional Integral e Derivativo
OPC	<i>OLE for Process Control</i>
PC	Computador Pessoal
PWM	<i>Pulse Width Modulation</i>
LED	Diodo Emissor de Luz
IEC	Comunicação Eletrotécnica Internacional
IEC	<i>International Electrotechnical Commission</i>
HP	<i>Horse Power</i>
PV	<i>Process Variable</i>
MV	<i>Manipulated Variable</i>
SP	<i>Set Point</i>

SUMÁRIO

Agradecimentos.....	v
Resumo.....	vii
Lista de Ilustrações.....	ix
Lista de Tabelas.....	xii
Lista de Abreviaturas e Siglas.....	xiii
Sumário.....	xiv
1 Introdução.....	16
1.1 Objetivos.....	16
1.2 Motivação.....	17
2 Fundamentação Teórica.....	18
2.1 Controle Proporcional-Integral-Derivativo (PID).....	18
2.2 CLP (Controlador Lógico Programável).....	19
2.3 Comunicação OPC.....	22
2.4 Identificação de Sistemas.....	23
2.4.1 Introdução.....	23
2.4.2 <i>System Identification Toolbox</i>	24
2.4.3 Etapas básicas para identificação de sistemas.....	25
3 Desenvolvimento da Instrumentação.....	26
3.1 Esteira Transportadora.....	26
3.2 CLP ZAP900.....	27
3.3 Motor de Corrente Contínua.....	28
3.4 <i>Encoder</i>	30
3.5 Sensor óptico de barreira.....	33
3.6 Sensor de temperatura.....	35
3.7 Módulo de Aquecimento.....	36
3.8 Módulo de Resfriamento.....	39
3.9 Fontes de alimentação.....	42
3.10 Dispositivos de Conversão.....	43
3.10.1 Dispositivo de conversão do sensor óptico de barreira.....	44
3.10.2 Dispositivo de conversão para controle do motor cc.....	45
3.10.3 Dispositivo de identificação de CLP desligado.....	46
3.10.4 Dispositivo conversor de pulsos de tensão para o <i>encoder</i>	47
3.10.5 Dispositivo de conversão para sensor LM35.....	48
3.10.6 Dispositivo de conversão de controle PWM para resistências e ventoinhas.....	49
3.11 Esteira com dispositivos integrados.....	50
4 Comunicação CLP com o PC.....	53
4.1 Comunicação CLP com PC utilizando o <i>Software SPDSW</i>	53
4.2 Comunicação CLP com PC utilizando servidor OPC.....	59
4.2.1 Instalação do <i>Driver OPC</i>	59
4.2.2 Configuração do <i>Driver OPC</i>	63

4.3	Comunicação CLP com Simulink utilizando servidor OPC HSI	68
5	Projeto do Sistema de Controle.....	71
5.1	Definição das Malhas de Controle	71
5.1.1	Controle da posição e velocidade da Esteira.....	71
5.1.2	Controle de temperatura do Módulo de aquecimento	73
5.1.3	Controle de temperatura do Módulo de resfriamento	75
5.2	Descrição de código de programação das funções principais	76
5.2.1	Movimentação de Variáveis	76
5.2.2	Contador Rápido <i>Encoder</i>	77
5.2.3	Temporizador	78
5.2.4	Gerador de Frequência.....	80
5.2.5	Bloco de Controle PID	81
6	Sintonia das Malhas de Controle	83
6.1	Identificação de Sistemas.....	83
6.1.1	Identificação do sistema de controle de velocidade.....	83
6.1.2	Identificação do sistema de controle de Posição.....	91
6.1.3	Identificação do sistema do Módulo de Aquecimento	92
6.1.4	Identificação do sistema do Módulo de Resfriamento	95
6.2	Sintonia dos Controladores	98
6.2.1	Sintonia dos controladores de velocidade.....	99
6.2.2	Sintonia do controlador do módulo de aquecimento	103
7	Conclusão	107
8	Sugestões para Trabalhos Futuros	109
	Bibliografia.....	110

1 INTRODUÇÃO

Desde o século XIX, as esteiras transportadoras são utilizadas para o transporte de materiais e produtos. Nos dias atuais a sua presença é indispensável na maioria das indústrias, possibilitando a estes, uma maior agilidade no transporte e manufatura destes produtos.

Na época conhecida como "Revolução Industrial" as esteiras transportadoras possibilitaram, por exemplo, determinar o ritmo da produção, característica esta que muito agradou aos empresários do setor, garantindo sua participação massiva no ambiente industrial.

A esteira transportadora continua até os dias de hoje, presente em muitos processos industriais, agregando não apenas a função de transporte dos produtos manufaturados através de suas diversas células de produção, mas também a própria manufatura dos mesmos.

1.1 OBJETIVOS

O plano deste trabalho consiste na construção, instrumentação, identificação e controle dos sistemas básicos que compõe uma esteira transportadora, utilizando como dispositivo de controle, o CLP (Controlador Lógico Programável), e supervisão um PC (Computador Pessoal). Para a esteira transportadora será realizado o controle da velocidade em ambos os sentidos e controle de posição.

Sobre a esteira têm-se dois módulos construídos para compor um processo simples de manufatura. Estes módulos são: o módulo de aquecimento e módulo de resfriamento, estes foram escolhidos para compor um processo simples, devido a sua grande usabilidade, sendo de interesse deste trabalho o conhecimento da construção, instrumentação e controle dos mesmos. O controle destes módulos será realizado pelo mesmo dispositivo de controle utilizado no controle da velocidade e posição da esteira.

Como parte da sintonia dos controladores será utilizada as seguintes ferramentas do *software* MATLAB: *System Identification* para identificação dos sistemas, PID

Tuner para sintonia dos controladores e o Simulink para simulação e controle dos sistemas.

1.2 MOTIVAÇÃO

A razão deste trabalho constitui a busca do conhecimento da instrumentação, identificação e controle de uma esteira transportadora e dos módulos de aquecimento e resfriamento instalados sobre a esteira.

É de interesse neste trabalho familiarizar-se com os softwares que serão utilizados:

- SPDSW, utilizado para realizar a comunicação entre o computador e o CLP do kit ZTK900;
- HS1 *PowerTool*, servidor OPC, por exemplo, para possibilitar a comunicação do CLP com o MATLAB;
- Simulink, ambiente gráfico de simulação e controle do MATLAB;
- *System Identification Toolbox*, ferramenta do MATLAB de identificação de sistemas;
- *PID Tuner*, ferramenta do MATLAB de sintonia dos controladores.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo será apresentada de forma objetiva a base para a compreensão teórica dos dispositivos e processos utilizados para a elaboração deste trabalho.

2.1 CONTROLE PROPORCIONAL-INTEGRAL-DERIVATIVO (PID)

O controlador PID é o tipo de controle mais comumente utilizado nas indústrias. Ele foi a ferramenta padrão quando o controle de processo surgiu em 1940. O controlador PID sobreviveu a muitas mudanças tecnológicas, desde a mecânica e pneumática para microprocessadores através de válvulas, transistores e circuitos integrados [1].

O microprocessador tem tido uma influência dramática no controlador PID, praticamente todos os controladores PID feitos hoje são baseados em microprocessadores. Isso tem dado oportunidades para fornecer recursos adicionais como ajuste automático, o ganho de programação e adaptação contínua.

A combinação da ação de controle proporcional, ação de controle derivativo e ação de controle integral são denominadas ação de controle proporcional mais derivativa mais integral. Esta ação combinada possui as vantagens de cada uma das três ações de controle individuais [8]. O algoritmo que descreve o controlador com esta ação combinada é dada pela equação (1)

$$u(t) = K(e(t) + \frac{1}{T_i} \int_0^t e(t)dt + T_d \frac{de(t)}{dt}) \quad (1)$$

Na equação 1 temos as variáveis: $u(t)$ que representa o sinal de controle (*Manipulated Variable* - MV), $e(t)$ representa o erro, o qual é caracterizado pela diferença entre o valor de referência (*Set Point* - SP) usado para o controle e o valor medido na saída do processo (*Process Variable* -PV) (SP-PV).

O sinal de controle é composto pela soma de três termos: o termo **P** (que é proporcional ao erro), o termo **I** (que é proporcional a integral do erro) e o termo **D** (que é proporcional à derivada do erro). Os parâmetros do controlador são o ganho proporcional K , o tempo integral T_i e o termo derivativo T_d [2].

Na Figura 1 tem-se a representação de um processo em malha fechada utilizando um controlador PID.

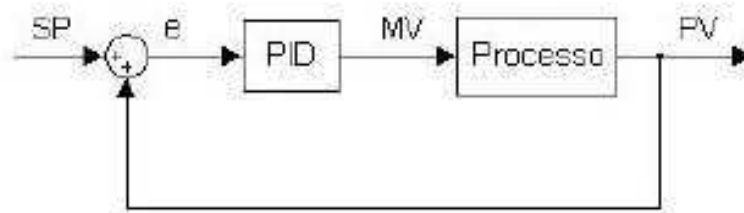


Figura 1: Processo em malha fechada utilizando um controlador PID.
Fonte: [2].

Aplicando a transformada de Laplace na equação 1, a lei de controle assumirá a forma apresentada na equação 2:

$$G(s) = K_p + \frac{K_i}{s} + K_d s \quad (2)$$

Nesta equação $G(s)$ é a função de transferência, K_p é o ganho proporcional do controlador, K_i o ganho integral do controlador e K_d o ganho derivativo do controlador.

Outra forma de expressar a equação 2 é como apresentado na equação 3:

$$G(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (3)$$

Onde: T_i é o tempo de integração do integrador e T_d é o tempo de derivação do derivador [2].

2.2 CLP (CONTROLADOR LÓGICO PROGRAMÁVEL)

A definição segundo a norma IEC 11311-1 afirma que o CLP é um sistema eletrônico digital, desenvolvido para uso em ambiente industrial, que usa memória programável para armazenamento interno de instruções do usuário.

Controladores Lógicos Programáveis, também chamados de Controladores Programáveis ou PLCs, Figura 2, pertencem a família dos computadores de estado sólido, usando circuitos integrados, em vez de dispositivos eletromecânicos para implementar funções de controle. Eles são capazes de armazenar instruções tais como: sequenciamento, temporização, contagem, aritmética, manipulação de dados e comunicação, para controlar máquinas e processos industriais [3].



Figura 2: CLP *CompactLogix* com cartões de entradas e saídas.

Os critérios de projeto para o primeiro controlador programável foram especificados pela divisão *Hydramatic* da *General Motors Corporation* em 1968 [3].

O seu principal objetivo era eliminar os altos custos associados com o inflexível sistema de controle a relé.

Algumas das especificações iniciais eram:

- Capacidade de trabalhar em ambientes industriais;
- As interfaces de entrada e saída devem ser facilmente substituídas;
- O controlador deve ser projetado de forma modular, para que subconjuntos possam ser facilmente removidos para substituição ou reparo;
- O sistema deve ser reutilizável;
- O método utilizado para programar o controlador deve ser simples, de modo que pudesse ser facilmente compreendido pelos funcionários da fábrica.

A seguir temos a lista de algumas recentes melhorias de *hardware* do CLP:

- Tempo de varredura menores;
- Grande quantidade de entradas e saídas;
- Processamento distribuído;
- Interfaces PID, rede CAN, Fieldbus entre outros;
- Interfaces especiais para a conexão direta de certos dispositivos (termopares, *strain gauges*, *encoders* entre outros);

Os avanços de *hardware* e os avanços de *softwares* levaram a PLCs mais poderosos com as seguintes características:

- Os PLCs incorporaram ferramentas de programação orientada a objetos baseadas no padrão IEC 1131-3;
- As linguagens de alto nível, como BASIC e C, foram implementadas em alguns módulos dos controladores para proporcionar uma maior flexibilidade de programação quando se comunica com dispositivos periféricos e manipulação de dados;
- Diagnóstico e detecção de falhas foram ampliados a partir do simples diagnóstico do sistema;
- Cálculo em ponto flutuante, que tornou possível realizar cálculos complexos em aplicações de controle que exigem aferição e cálculo estatístico.

Instruções de manuseio e manipulação de dados foram melhoradas e simplificadas para acomodar o controle complexo e aplicações de aquisição de dados que envolvem o armazenamento, rastreamento e recuperação de grande quantidade de dados [3].

Os controladores programáveis são capazes de comunicar-se com outros sistemas de controle, fornecendo relatórios de produção, programação da produção, e diagnosticar as suas próprias falhas e as da máquina ou processo. Essas melhorias têm feito os controladores programáveis contribuir de forma significativa no atendimento às demandas atuais para maior qualidade e produtividade [3].

Um controlador programável é formado basicamente de duas seções, Figura 3:

- Unidade central de processamento
- As interfaces de entrada e saída

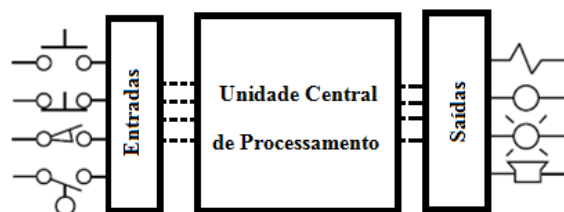


Figura 3: Unidades básicas do CLP.

A rotina de operação da CPU, Figura 4, é realizada basicamente em três processos: (1) leitura dos dados de entrada dos dispositivos de campo através das interfaces de entrada, (2) execução do programa armazenado no sistema de memória e (3) escreve ou atualiza os dispositivos de saída através das interfaces de saída (ciclo da CPU).

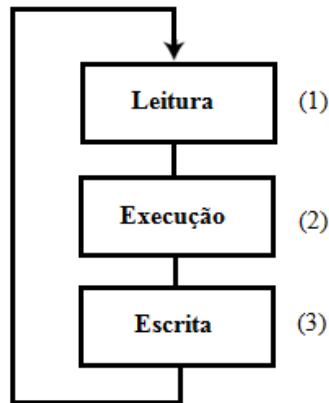


Figura 4: Rotina de operação da CPU.

2.3 COMUNICAÇÃO OPC

O OPC (OLE para Controle de Processos) consiste na especificação e no conjunto de interfaces baseadas no modelo COM (Modelo de Objeto Componente).

Historicamente, o primeiro padrão (originalmente chamado simplesmente de "OPC *Specification*" e agora de "*Data Access Specification*") surgiu na colaboração entre empresas líderes no mercado de Automação Industrial e a Microsoft.

Originalmente baseado no OLE COM da Microsoft e tecnologias DCOM, a especificação definia um conjunto de objetos, interfaces e métodos utilizados em aplicações de Controle de Processos e Automação da Manufatura para facilitar a interoperabilidade. A tecnologia COM/DCOM foi o ponto de partida para o desenvolvimento de *softwares* com o padrão OPC [18].

O OPC é a base tecnológica para a conexão eficiente de componentes de automação com dispositivos de *hardware* de controle e de campo [6]. Além disso, ele fornece a condição para integração de produtos de escritório e sistemas de informação sobre o nível da empresa, tais como Planejamento de Recursos Empresariais (ERP) e Sistema de Execução e Manufatura (MES). Os dados do processo no nível do campo podem ser apresentados em uma folha de Excel e dados de status e dados de produção

sobre o nível de controle podem ser arquivados em um banco de dados sem problemas, via OPC ou podem ser processados em um sistema de planejamento de produção.

O uso desta tecnologia significa que os fabricantes de servidores OPC e clientes não estão limitados com a complexidade, a funcionalidade e a execução da sua componente OPC, contanto que as definições adequadas para as interfaces e os métodos e regulamentos em relação à sua aplicação, são observados. Os mais diversos componentes OPC de diferentes fabricantes podem trabalhar juntos.

Nenhuma programação adicional para a adaptação das interfaces entre os componentes é necessária. Correlações complexas, por exemplo, dependências do componente de *software* em componentes de *hardware*, permanecem escondidas por trás dessa interface abstrata.

Os padrões OPC são especificações técnicas livremente acessíveis que definem conjuntos de interfaces padrão para diferentes campos de aplicação na tecnologia de automação. Estas interfaces permitem uma troca de dados altamente eficiente entre os componentes de diferentes fabricantes de *softwares*.

2.4 IDENTIFICAÇÃO DE SISTEMAS

2.4.1 INTRODUÇÃO

Identificação de sistemas é a área da modelagem matemática de sistemas a partir dos dados experimentais [9]. Uma abordagem utilizada para desenvolver modelos matemáticos para sistemas dinâmicos consiste na construção analítica dos modelos através de leis físicas e químicas inerentes aos processos analisados.

No ramo da engenharia, entretanto, muitas vezes não se tem acesso aos parâmetros exatos que descrevem os fenômenos, portanto se faz necessário uma aproximação destes que gerem resultados satisfatórios para uma dada aplicação.

Existe sempre uma relação entre complexidade e precisão dos modelos, incertezas são toleradas desde que não comprometam a robustez ou o desempenho do sistema [9].

O comportamento temporal de sistemas, tais como das áreas da engenharia elétrica, engenharia mecânica, engenharia de processos e de outras áreas adversas como

biologia, medicina e economia, por exemplo, podem ser descritos por modelos matemáticos [5].

No entanto, a aplicação da teoria de sistemas exige que os modelos matemáticos para o comportamento estático e dinâmico dos sistemas e dos seus elementos sejam conhecidos.

O processo de criação de um modelo adequado é chamado de modelagem, este possui duas abordagens gerais, ou seja, modelagem teórica e experimental, sendo que ambas possuem vantagens e desvantagens distintas.

2.4.2 SYSTEM IDENTIFICATION TOOLBOX

Para a tarefa da identificação dos sistemas utiliza-se o *software* MATLAB, especificamente a ferramenta *System Identification Toolbox*, Figura 5.

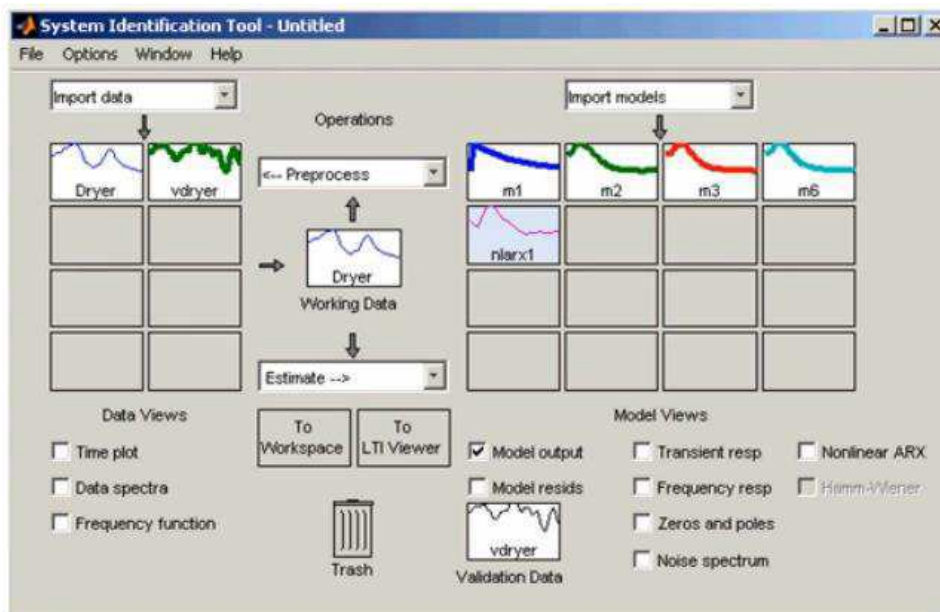


Figura 5: Imagem da janela principal do *software* *System Identification Toolbox*.

Fonte: [24].

De acordo com o guia (*System Identification Toolbox™ Getting Started Guide*, 2015), o *System Identification Toolbox* fornece funções do MATLAB, blocos do Simulink e um aplicativo para a construção de modelos matemáticos de sistemas dinâmicos a partir de dados de entrada e saída medidos. Podem-se usar dados no domínio do tempo e no domínio da frequência da entrada e saída para identificar em tempo contínuo e em tempo discreto funções de transferência, modelos de processos e modelos de espaço de estado.

Características Principais:

- Função de transferência, modelo de processo e de espaço de estado da identificação do modelo no domínio do tempo e dados de resposta no domínio da frequência;
- Estimativa de parâmetros de modelo *on-line*;
- Modelagem (AR, ARMA) e previsão;
- Identificação de modelos não lineares ARX e modelos Hammerstein-Weiner com entradas e saídas não lineares tais como saturação e zona morta;
- Identificação de sistemas "*grey-box*" para estimar modelos de uso definido;
- Estimação de *Delay*, filtragem e reconstrução de falta de dados.

2.4.3 ETAPAS BÁSICAS PARA IDENTIFICAÇÃO DE SISTEMAS

Identificação de sistema é uma metodologia para a construção de modelos matemáticos de sistemas dinâmicos usando medições de sinais de entrada e de saída do sistema [24].

O processo de identificação do sistema requer:

- Medição dos sinais de entrada e saída do sistema no domínio do tempo ou no domínio da frequência;
- Escolha de um modelo de estrutura;
- Aplicar um método de estimativa para estimar o valor dos parâmetros ajustáveis do modelo de estrutura escolhido;
- Avaliar o modelo estimado para ver se este é adequado para a aplicação desejada.

3 DESENVOLVIMENTO DA INSTRUMENTAÇÃO

Nesta etapa do projeto será realizada a instrumentação da esteira transportadora e dos Módulos de Aquecimento e Resfriamento, a instrumentação compreende o estudo para a escolha de quais dispositivos sensores e atuadores serão utilizados no projeto.

Neste capítulo será apresentada uma descrição destes dispositivos, assim como suas características técnicas, nem todos os dispositivos utilizados foram adquiridos prontos alguns foram construídos. Os dispositivos construídos serão descritos com um maior detalhamento para uma maior compreensão de suas funcionalidades e características.

3.1 ESTEIRA TRANSPORTADORA

A estrutura física da esteira transportadora utilizada neste trabalho foi reaproveitada de uma esteira da marca CALOI modelo FITNESS. Como o objetivo deste trabalho está relacionado à instrumentação e controle de uma esteira transportadora, esta serve muito bem para este propósito. Além da estrutura física também foi utilizado da esteira CALOI o motor CC, as polias, as correias e os roletes.

Após a retirada dos equipamentos que não seriam utilizados no projeto a esteira ficou da forma como apresentado na Figura 6.



Figura 6: Plataforma base da esteira transportadora.

A partir desta plataforma será construída uma esteira transportadora de pequeno porte, no entanto com alguns elementos sensores e atuadores com características semelhantes aos encontrados em esteiras transportadoras Industriais. Ao decorrer deste capítulo estes elementos serão descritos com maiores detalhes.

Para utilizar esta plataforma da esteira CALOI, como base para construção da esteira transportadora foi necessário construir o desenho técnico correspondente. O desenho foi elaborado a partir das medições da plataforma, em seguida foi utilizado o *software* AutoCAD 2014 para sua construção, Figura 7.

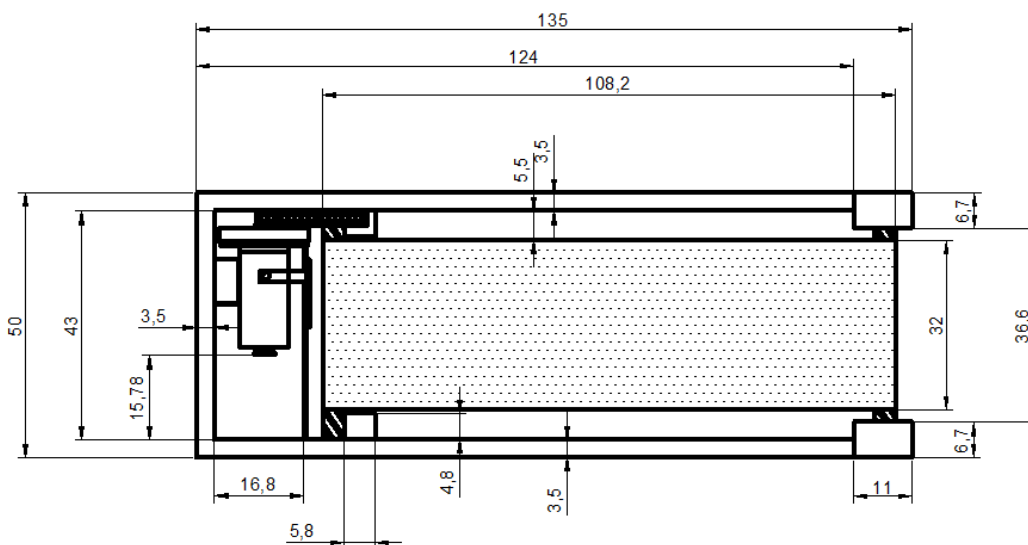


Figura 7: Desenho Técnico Plataforma esteira transportadora.

Todas as medidas do desenho técnico, ilustradas na Figura 7 estão na escala de centímetros (cm). Neste trabalho será dada ênfase aos dispositivos eletrônicos utilizados, sendo os detalhes das partes mecânicas não abordadas.

3.2 CLP ZAP900

Neste trabalho será utilizado o CLP ZAP900 do kit de treinamento ZTK900 da empresa HI Tecnologia, Figura 8, este kit foi escolhido para uma maior experiência com o mesmo e também por este possuir em seus módulos tipos de entradas e saídas desejadas para este trabalho, uma destas, por exemplo, é a entrada digital que pode ser configurada para ser uma entrada dedicada para o *encoder* incremental, outro motivo foi

acessibilidade a documentação e suporte, encontrados no site da empresa HI Tecnologia.



Figura 8: Kit de treinamento ZTK900.
Fonte: [17]

3.3 MOTOR DE CORRENTE CONTÍNUA

As máquinas elétricas rotativas assumem diversas formas e são conhecidas por diversos nomes: CC, síncronas, de ímã permanente, de indução, de relutância variável, de histerese, sem escovas, e assim por diante. Embora essas máquinas aparentem ser bem diferentes, os princípios físicos que regem o seu comportamento são bastante similares e frequentemente é útil pensar nelas em termos de um mesmo ponto de vista físico. Uma vista em corte longitudinal de um motor CC está ilustrada na Figura 9 [4].

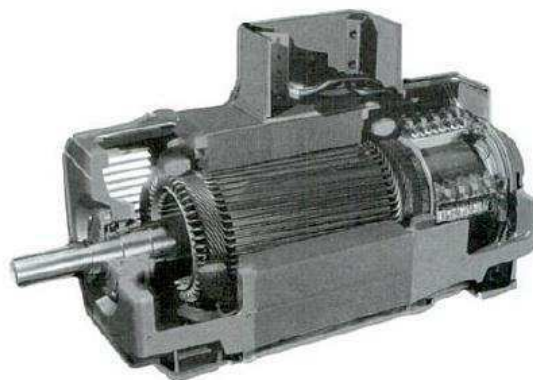


Figura 9: Vista em corte longitudinal de um típico motor CC de potência elevada.
Fonte: [4].

Em uma máquina CC, o enrolamento de armadura encontra-se na parte rotativa conhecida como rotor. A Figura 10 ilustra o rotor de uma máquina CC. O enrolamento de armadura de uma máquina CC consiste em muitas bobinas conectadas entre si para formar um laço fechado.

Quando o rotor está girando, um contato mecânico rotativo é usado para fornecer corrente ao enrolamento de armadura. Tipicamente, as máquinas síncronas e CC apresentam um segundo enrolamento (ou conjunto de enrolamentos) que conduz corrente contínua e que é usado para produzir o fluxo principal de operação da máquina.

Tal enrolamento é referido tipicamente como enrolamento de campo. Em algumas máquinas ímãs permanentes, são usados no lugar de enrolamentos de campo [4].

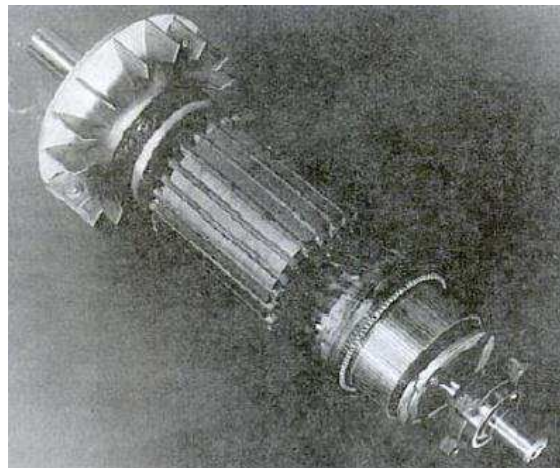


Figura 10: Armadura de um motor CC.

Fonte: [4].

O Motor de corrente contínua (Motor CC), Figura 11, utilizado neste trabalho foi reutilizado da Plataforma Original da esteira CALOI e possui os seguintes parâmetros:

- Modelo: MOTOR PMDC;
- Tensão: 180 V DC (podendo operar com tensões inferiores);
- Potência: 1HP;
- Fabricante: Universal Motors (Indústria Brasileira).



Figura 11: Motor de corrente contínua, (Motor CC) utilizado.

3.4 ENCODER

Os *encoders* (Geradores de Impulsos) são equipamentos eletromecânicos utilizados para conversão de movimentos rotativos ou deslocamentos lineares em impulsos elétricos de onda quadrada, gerando uma quantidade exata de impulsos por volta em uma distribuição perfeita ao longo dos 360 graus de giro do eixo [10].

Funcionam por barreiras de luz infravermelha onde a passagem de luz é interrompida ou liberada por um disco ótico fixo ao eixo com N pontos de transferência ao longo de sua circunferência, gerando impulsos luminosos que são convertidos em sinais elétricos e depois são reenquadrados e amplificados, gerando uma saída digital com número de pulsos igual à quantidade de pontos de transparência no disco ótico [10].

Os *encoders* podem ser utilizados em conjunto com indicadores ou programadores de posição linear ou angular, contadores, tacômetros, controladores lógicos programáveis ou conversores de frequência para sinais analógicos.

Fornecem medidas e controles precisos em comprimentos, velocidades de rotação, velocidades lineares, posicionamento angulares, volumes ou vazões de produtos líquidos, robótica e outras aplicações em processos diversos [10].

Existem dois tipos de *encoder*:

Encoder Incremental: O *encoder* incremental fornece normalmente dois pulsos quadrados defasados em 90°, que são chamados usualmente de canal A e canal B. A leitura de apenas um canal fornece somente a velocidade, enquanto que a leitura dos

dois canais fornece também o sentido do movimento. Outro sinal chamado de Z ou zero também está disponível e ele dá a posição absoluta “zero” do *encoder*. Este sinal é um pulso quadrado em que a fase e a largura são as mesmas do canal A. A resolução do *encoder* incremental é dada por pulsos por revolução (normalmente chamado de PPR), isto é, o *encoder* gera certa quantidade de pulsos elétricos por uma revolução dele próprio (no caso de um *encoder* rotativo) [11].

Encoder Absoluto: O princípio de funcionamento de um *encoder* absoluto e de um *encoder* incremental é bastante similar, isto é, ambos utilizam o princípio das janelas transparentes e opacas, com estas interrompendo um feixe de luz e transformando pulsos luminosos em pulsos elétricos. O *encoder* absoluto possui um importante diferencial em relação ao *encoder* incremental: a posição do *encoder* incremental é dada por pulsos a partir do pulso zero, enquanto a posição do *encoder* absoluto é determinada pela leitura de um código e este é único para cada posição do seu curso, conseqüentemente, os *encoders* absolutos não perdem a real posição no caso de uma eventual queda da tensão de alimentação (até mesmo se deslocados).

Quando voltar a energia ao sistema, a posição é atualizada e disponibilizada para o mesmo (graças ao código gravado no disco do *encoder*) e, com isso, não se precisa ir até a posição zero para saber a sua localização como é o caso do incremental [11].

Neste trabalho utilizamos o *encoder* incremental para determinarmos a velocidade do Motor CC e a posição do objeto sobre a esteira, o *encoder* foi fixado ao eixo do rotor para termos uma precisão maior e um melhor controle, Figura 12. O modelo do *encoder* utilizado corresponde ao "ww - 12 *encoder*" produzido pela empresa NU-BOTICS, na Figura 13 e Figura 14 tem-se as ilustrações deste modelo [12].



Figura 12: *Encoder* fixado ao eixo do motor.

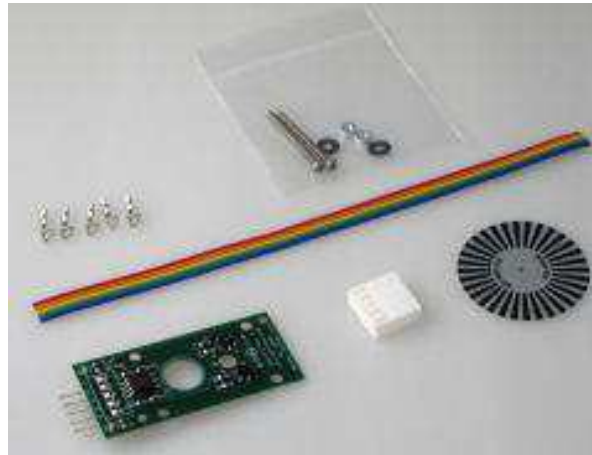


Figura 13: Partes do *encoder* ww-12.
Fonte:[12]



Figura 14: *Encoder* ww-12.
Fonte: [12]

O *encoder* utilizado é alimentado por uma faixa de tensão de +3.3 V a +5.5 V, consome uma corrente média de 34 mA e uma corrente máxima de 42 mA quando alimentado por uma tensão de +5 V. Possui uma resolução de 128 ciclos por revolução e dois canais de saída, canal A e canal B.

Para a instalação adequada do disco perfurado do *encoder* é preciso respeitar as recomendações fornecidas pelo fabricante, na Figura 15, temos as recomendações de distancias para o modelo ww-12 [12].

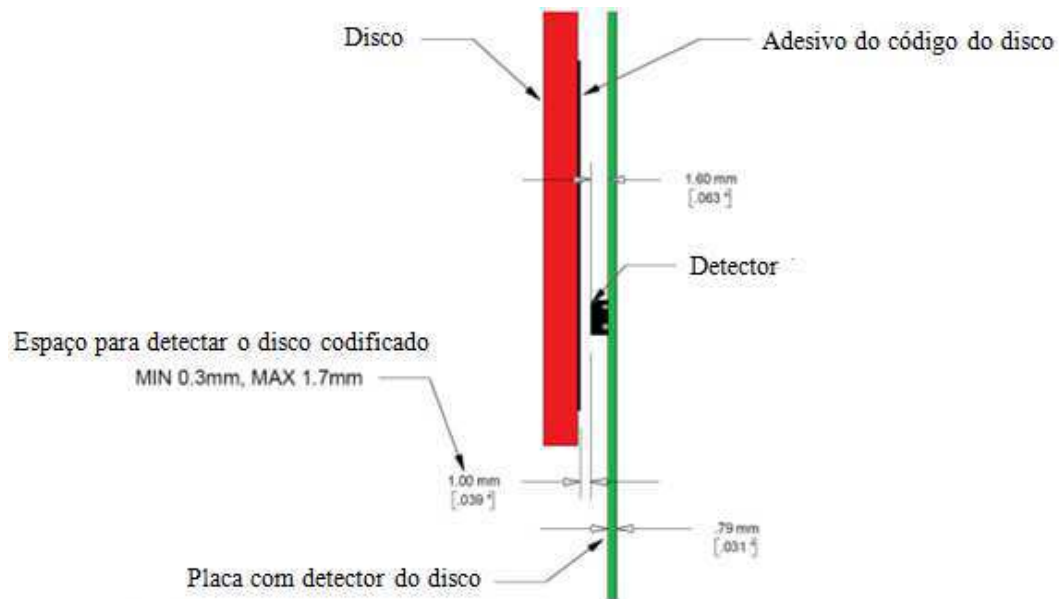


Figura 15: Especificações de instalação.
Fonte: [12]

Na Figura 16, é apresentado um diagrama de tempo que ilustra o canal A, canal B, a direção e o sinal de *Clock*.

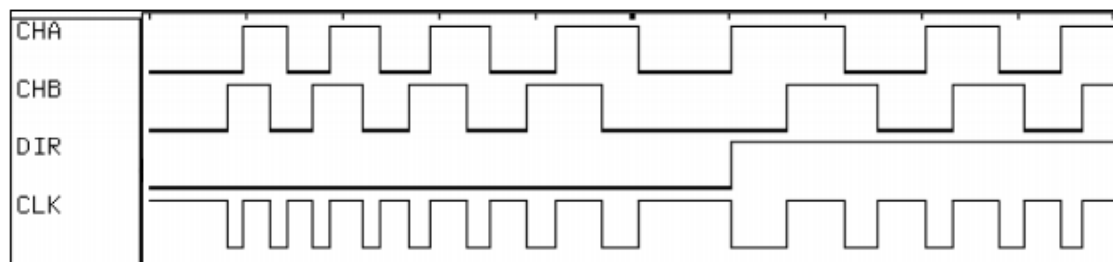


Figura 16: Diagrama de tempo ww-12.
Fonte: [12]

3.5 SENSOR ÓPTICO DE BARREIRA

Baseia-se na interrupção ou incidência de um feixe luminoso sobre um fotorreceptor, o qual provoca uma ação (comutação) eletrônica. A emissão de luz é proveniente da emissão de raios infravermelhos em grande parte das aplicações, bem como *laser*.

Para conseguir-se máxima eficiência a luz é modulada ou pulsada a uma frequência máxima de 1,5 kHz, frequência que será interpretada por um receptor óptico sincronizado a essa frequência o que imuniza o sistema de interferências da recepção luminosa do ambiente [13].

Neste tipo de transdutor, o emissor e o receptor encontram-se em dois dispositivos diferentes. Eles devem ser colocados em perfeito alinhamento de maneira tal que a luz emitida chegue ao receptor. Quando um objeto não transparente se interpõe entre o emissor e o receptor, a luz obviamente não chega, desativando o receptor, e entregando o estado correspondente na saída [14].

Na Figura 17, está ilustrado o modo de instalação de um sensor óptico de barreira.

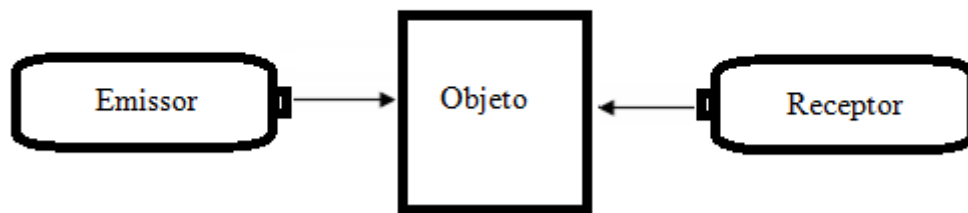


Figura 17: O objeto interrompe o feixe entre o emissor e o receptor.

No sensor de óptico de barreira foi utilizado com elemento emissor um LED auto brilho azul e um foto transistor TIL81 com circuito dedicado, como elemento receptor.

Na Figura 18 têm-se os sensores ópticos de barreira instalados na plataforma da esteira transportadora.



Figura 18: Sensor óptico de barreira instalado na esteira transportadora.

3.6 SENSOR DE TEMPERATURA

A temperatura é sem dúvida uma das grandezas mais medidas e controladas na indústria, o seu controle é realizado em diversos ambientes, sejam eles laboratórios de pesquisa e desenvolvimento, sejam em lares, proporcionando o conforto térmico ou mesmo presente na maioria dos eletrodomésticos encontrados neste ambiente, por exemplo: geladeira, ar condicionado, ferro de passar e etc.

Na indústria o controle e monitoramento da temperatura esta presente na maioria dos processos, seja de forma direta ou indireta, a forma direta seria o medição da temperatura por um elemento sensor para o controle de um processo, a forma indireta seria a medição da temperatura para auxiliar na calibração de algum elemento sensor ou transdutor.

Neste trabalho utilizou-se o dispositivo sensor de temperatura LM35 para as aplicações de controle de temperatura nos módulos de aquecimento e de resfriamento.

O sensor LM35 é um sensor de precisão, fabricado pela *National Semiconductor*, que apresenta uma saída de tensão linear relativa à temperatura em que ele se encontra no momento em que for alimentado por uma tensão de 4 a 20 V, tendo em sua saída um sinal de 10 mV para cada Grau Celsius de temperatura, Figura 19 [16].

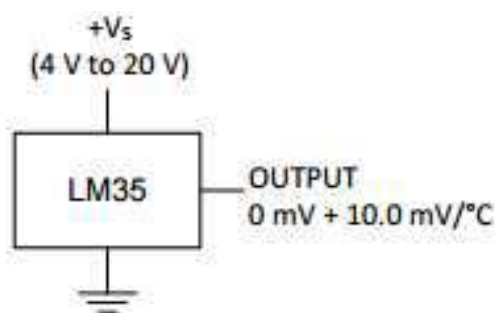


Figura 19: Diagrama de alimentação LM35 e saída de sinal de tensão.
Fonte: [16]

O LM35 não necessita de qualquer calibração externa para fornecer com exatidão, valores de temperatura com variações de $\frac{1}{4}^{\circ}\text{C}$ ou até mesmo $\frac{3}{4}^{\circ}\text{C}$ dentro da faixa de temperatura de -55°C a 150°C .

Este sensor poderá ser alimentado com fonte simples ou simétrica, dependendo do que se deseja com o sinal de saída, independente disso, a saída continuará sendo de $10\text{ mV}/^{\circ}\text{C}$. Para estas alimentações ele drena apenas $60\text{ }\mu\text{A}$ [15].

O LM35 que foi utilizado neste trabalho possui encapsulamento TO-92, Figura 20, o que oferece uma melhor relação custo benefício, por ser o mais barato dos modelos e propiciar a mesma precisão dos demais.

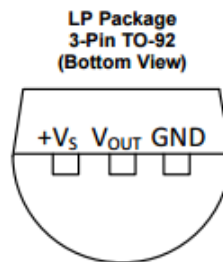


Figura 20: Pinos de configuração para o encapsulamento TO-92.
Fonte: [16]

Na Figura 21, tem-se o sensor LM35 instalado em um módulo de resfriamento.

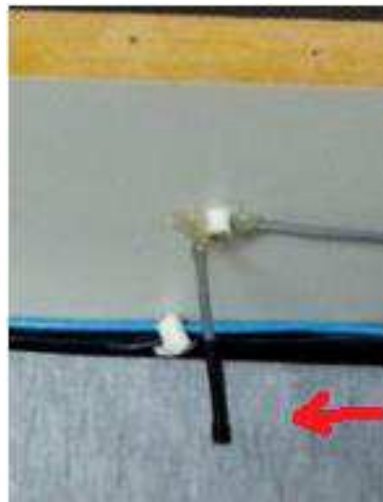


Figura 21: LM35 instalado no módulo de resfriamento.

3.7 MÓDULO DE AQUECIMENTO

Como o objetivo deste trabalho inclui a construção de um módulo de aquecimento, Figura 22, este será descrito em detalhes, a instalação deste módulo sobre a esteira transportadora está ilustrada na Figura 23.

Módulos de aquecimento ou dispositivo de aquecimento são comumente utilizados na indústria, eles são utilizados em diversos processos de aquecimento, por exemplo: aquecimento rápido associado a um rápido resfriamento para endurecimento de peças metálicas, ferramentas, moldes de peças plásticas, selagem de rótulos em garrafas, embalagens a vácuo entre outros.



Figura 22: Interior do Módulo de Aquecimento.

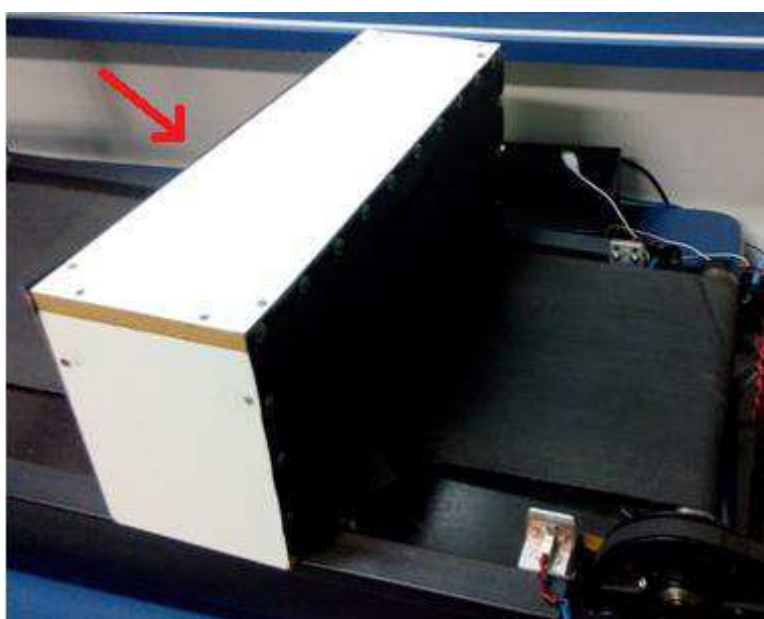


Figura 23: Módulo de Aquecimento instalado sobre a esteira.

O módulo de aquecimento foi construído de acordo com o desenho técnico apresentado na Figura 24, este módulo é composto basicamente por um sensor de temperatura LM35, para monitoramento da temperatura, duas resistências elétricas de 1400 W, Figura 25, duas ventoinhas alimentadas por uma tensão 12 V DC, Figura 26, para aumentar a eficiência da transferência de calor, suporte de madeira, metal e lonas.

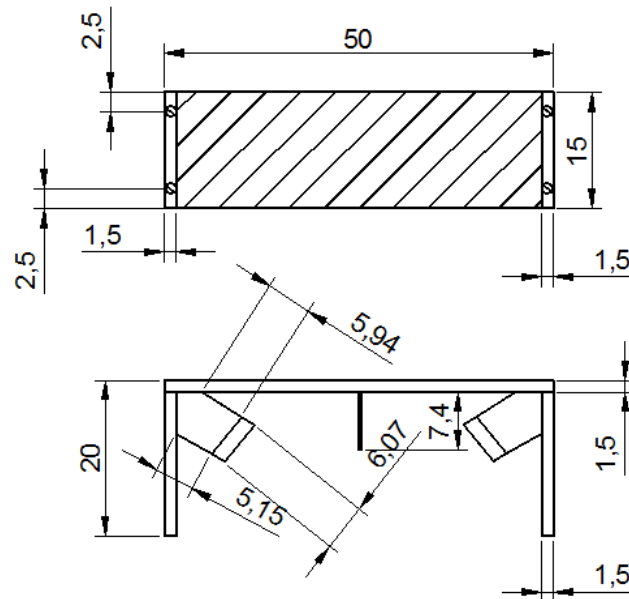


Figura 24: Desenho técnico do módulo de aquecimento.



Figura 25: Resistência 1400 W utilizada no módulo de aquecimento.



Figura 26: Ventoinha 12 V DC.

Na Figura 27, tem-se um esboço do princípio de funcionamento do módulo de aquecimento, as duas resistências são as fontes de calor e as ventoinhas farão com que o ar frio que se encontra na base da estrutura seja aspirado pela parte inferior das câmaras de aquecimento, onde estão localizadas as resistências elétricas, passando pelas resistências e facilitando a troca de calor com as mesmas e em seguida este ar aquecido é injetado em direção ao objeto que se pretende aquecer.

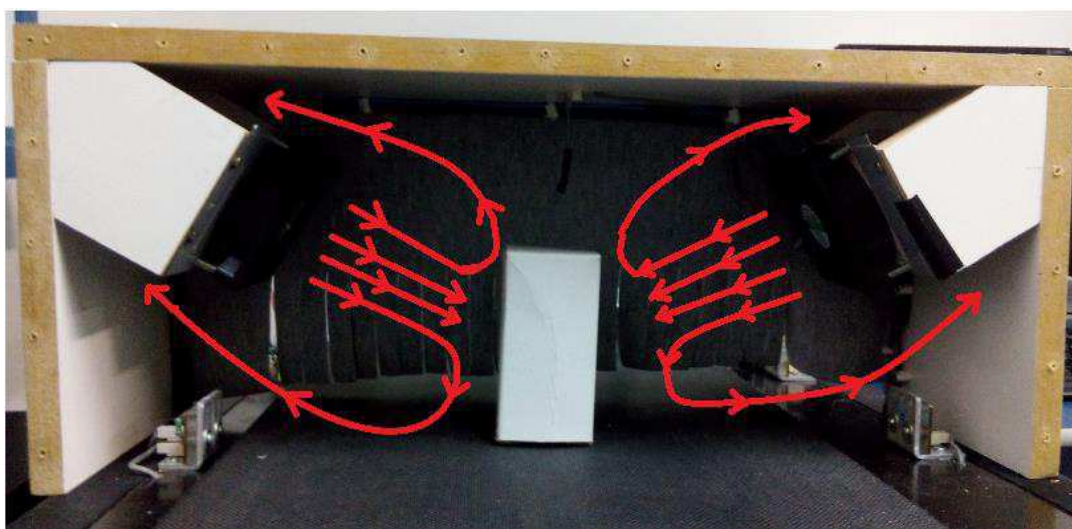


Figura 27: Descrição do funcionamento do módulo de aquecimento.

Como inicialmente este ar é frio, a cada passagem pelas câmaras de aquecimento ele vai ganhando calor, sendo assim após alguns ciclos, o ar estará aquecido a temperatura desejada. As câmaras de aquecimento tem o princípio de funcionamento de uma turbina de ar quente. As lonas que se encontram na entrada e na saída do módulo de aquecimento tem como função principal diminuir a transferência de calor do interior do módulo de aquecimento para o ambiente, caracterizando a função modular, onde o aquecimento acontece exclusivamente em seu interior.

3.8 MÓDULO DE RESFRIAMENTO

Assim como os módulos de aquecimento, os módulos de resfriamento estão presentes em diversos processos indústrias. A sua função é tão importante quanto à do processo de aquecimento, no entanto devido às características específicas do material que necessita perder calor e do ambiente em que este se encontra a função de resfriamento geralmente torna-se mais complexa.

Neste trabalho será construído um módulo de resfriamento de construção simples, porém que atenda as especificações de projeto, este é composto por um sensor de temperatura LM35, que será nosso elemento sensor de controle, e duas ventoinhas com características semelhantes as usados no módulo de aquecimento, que serão os elementos atuadores. Na Figura 28, tem-se a ilustração da parte interna do módulo de resfriamento proposto.



Figura 28: Módulo de resfriamento.

Na Figura 29, tem-se o módulo de resfriamento, com detalhes a ventoinha e o sensor de temperatura LM35, que se encontra fixado na parte superior, semelhante ao módulo de aquecimento, o módulo de resfriamento também possui lonas que melhoram a eficiência térmica.



Figura 29: Módulo de Resfriamento, com detalhes a ventoinha e o sensor LM35.

Na Figura 30 é apresentado o desenho técnico do módulo de resfriamento, todas as medidas estão na escala de centímetros.

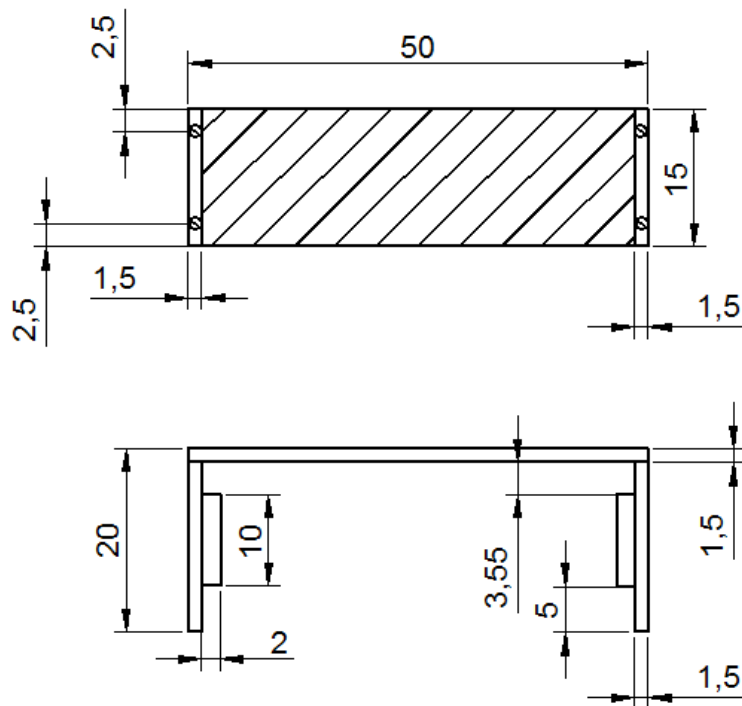


Figura 30: Desenho técnico módulo de resfriamento.

O funcionamento do módulo de resfriamento é apresentado na Figura 31, neste módulo as ventoinhas estão instaladas de forma que uma direciona o fluxo de ar para dentro da câmara de resfriamento e a outra direciona o fluxo de ar para fora da câmara, este efeito permite a criação de um túnel de vento, que para esta aplicação será utilizado para resfriar o objeto colocado em seu caminho.

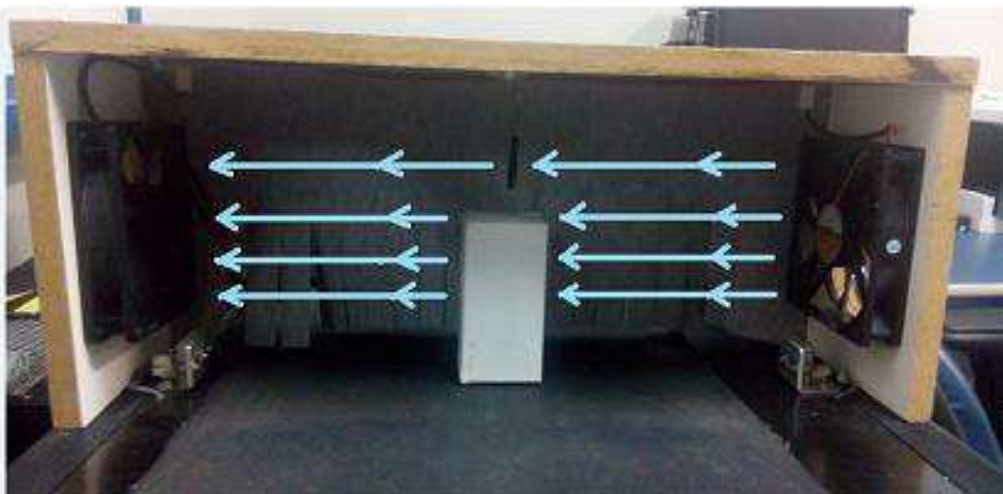


Figura 31: Princípio de Funcionamento módulo de Resfriamento.

3.9 FONTES DE ALIMENTAÇÃO

Para alimentar todo o sistema foram utilizadas fontes de alimentação de acordo com as especificações do projeto, como o ambiente de estudo e trabalho corresponde ao ambiente industrial, os sensores e atuadores são alimentados com a tensão de 24 V DC, tensão mais comumente utilizada neste ambiente para alimentação destes dispositivos.

Não foi necessário utilizar uma fonte de alimentação externa para o CLP, pois a fonte para este já se encontra integrada ao kit de treinamento ZTK900, sendo apenas necessária a alimentação do kit a rede 220 V AC. Os demais dispositivos sensores e atuadores foram alimentados com a fonte UPS1095-5A, Figura 32, que trata-se de uma fonte chaveada 24V-5A, desenvolvida para fixação em trilho DIN TS35 em armários de comando, possui fusível de proteção na entrada e proteção contra curto circuito e sobretensão na saídas [19].

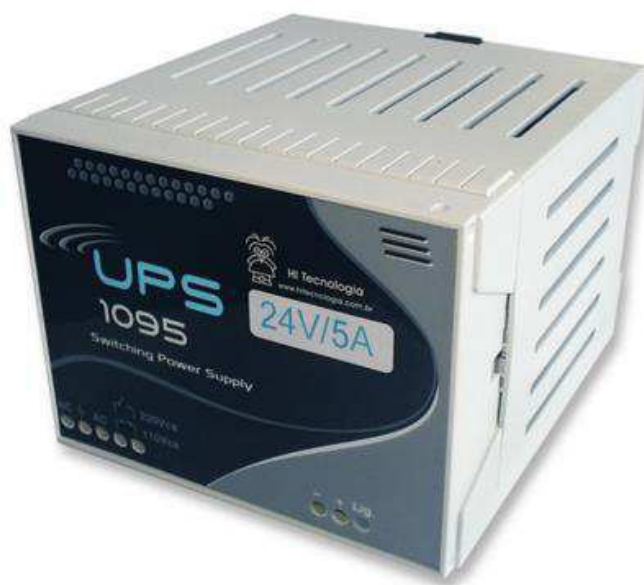


Figura 32: Fonte de alimentação UPS1095-5A.
Fonte: [19]

As especificações técnicas para a fonte UPS1095-5A estão apresentadas na Tabela 1:

Tabela 1: Especificações Técnicas para a fonte UPS1095-5A

Alimentação	110/220 VAC
Consumo	90VA
Saída regulada	24Vdc/5A
Regulação da Linha	menor que 0.8% com variação de 10% do valor nominal
Regulação da Carga	menor que 0.8% de 0 a 100% de carga máxima
Rendimento	Aproximadamente 80%
Proteção	Curto Circuito tipo <i>foldback</i> , superaquecimento, sobre tensão e fusível na entrada
Temperatura de operação	0 a 65°C
Temperatura de Estocagem	-25 a 80°C
Umidade relativa	<80% sem condensação
Peso	0,4kg
Grau de proteção	IP20
LED's	LED de indicação de saída energizada
Bornes	3 bornes de entrada e 2 bornes de saída com fixação por parafusos para terminais tipo pino
Dimensões	100(L) x 75(A) x 110(P) mm

Fonte: [19].

3.10 DISPOSITIVOS DE CONVERSÃO

Para realizar a comunicação entre o CLP e os elementos sensores e atuadores acima citados, foi necessário construir dispositivos que convertessem os sinais de tensão e corrente fornecidos por estes elementos, para sinais de tensão e corrente compatíveis com as entradas e saídas disponíveis no kit de treinamento ZTK900, como essas

entradas possuem as mesmas características de tensão e corrente usualmente utilizadas em dispositivos de controle industrial, os sensores e atuadores utilizados neste trabalho juntamente com seus dispositivos de conversão podem ser utilizados por outros CLPs.

3.10.1 DISPOSITIVO DE CONVERSÃO DO SENSOR ÓPTICO DE BARREIRA.

Os sensores ópticos de barreira utilizados neste trabalho correspondem basicamente: ao LED azul como elemento emissor e ao foto transistor TIL81 como elemento receptor, no entanto esses elementos não podem ser conectados diretamente ao dispositivo de controle. Para que seja realizada a comunicação destes elementos com o dispositivo de controle, foi construído um circuito elétrico dedicado, Figura 33, e a partir deste circuito o dispositivo de conversão, Figura 34.

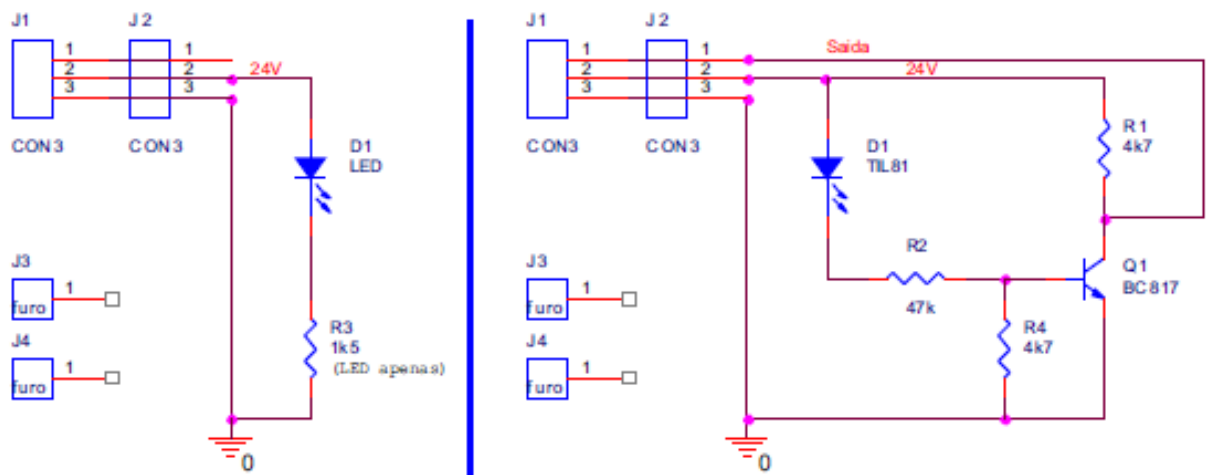


Figura 33: Circuito elétrico do dispositivo de conversão do sensor óptico de barreira.

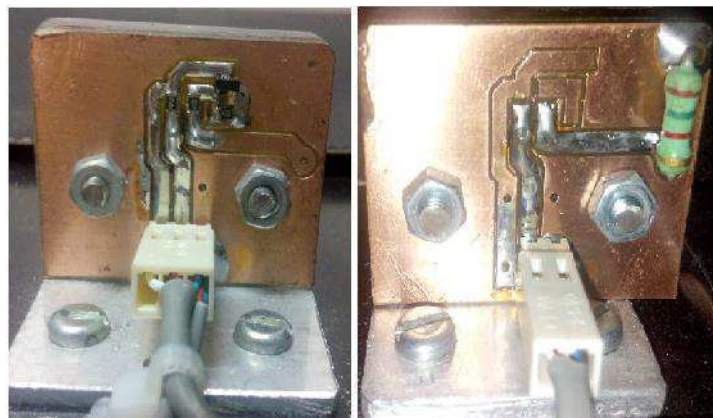


Figura 34: Dispositivo de conversão do sensor óptico de barreira.

Tem-se para este dispositivo um sinal de saída de tensão de 0 e 24 V, este sinal de tensão é correspondente ao estado de detecção e não detecção de um objeto pelo sensor óptico de barreira.

3.10.2 DISPOSITIVO DE CONVERSÃO PARA CONTROLE DO MOTOR CC

A partir das especificações do motor CC utilizado, foi construído um dispositivo de conversão de sinais, que a partir de um sinal analógico de tensão de 0 a 10 V, fornecido por uma saída analógica do kit ZTK900, possuísse uma saída PWM correspondente, adequada para o controle da velocidade e sentido de rotação do motor.

Como a inversão no sentido de rotação do motor que aciona a esteira faz parte dos objetivos deste trabalho e a velocidade desejada para a esteira, podia ser obtida com um valor PWM inferior a 50%, utilizou-se a saída PWM de 0 a 49% correspondente ao controle da velocidade em um sentido de rotação (sentido horário) e a saída PWM de 50 a 100% correspondente ao controle da velocidade no sentido inverso de rotação (sentido anti-horário), com esta solução, o controle da velocidade e a inversão do sentido de rotação do motor foram realizados utilizando o mesmo circuito, Figura 35, dispositivo de conversão, Figura 36.

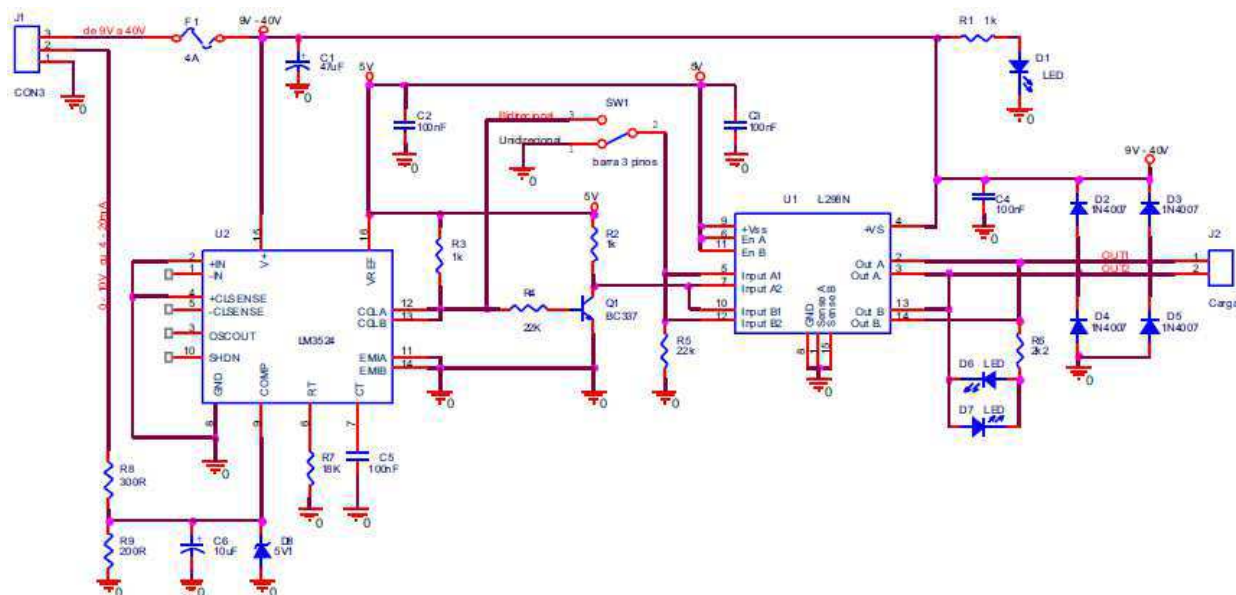


Figura 35: Circuito do dispositivo conversor de 0 a 10 V para saída PWM.

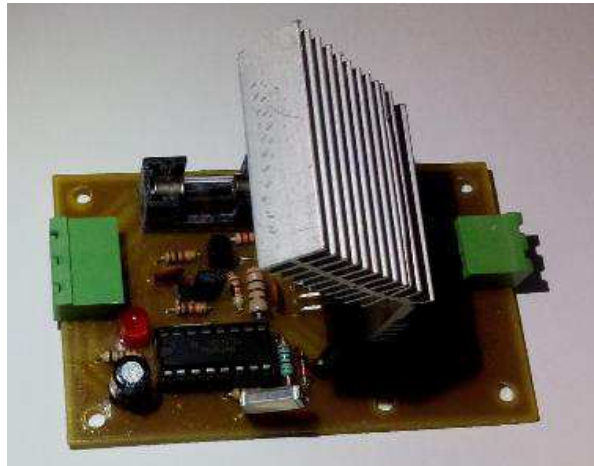


Figura 36: Dispositivo conversor de 0 a 10 V para saída PWM.

3.10.3 DISPOSITIVO DE IDENTIFICAÇÃO DE CLP DESLIGADO

Durante os teste realizados com o dispositivo conversor citado acima e o motor CC, verificou-se que o valor analógico correspondente ao sinal de tensão 0 V fornecido pela saída analógica do controlador, também era obtido quando o mesmo estava desligado, como este valor de 0 V era utilizado como entrada válida para o dispositivo conversor de 0 a 10 V para saída PWM, e não se desejava perder resolução de controle, optou-se por utilizar um circuito, Figura 37, que identificasse quando a tensão 0 V fosse obtida com o dispositivo de controle desligado, este dispositivo está ilustrado na Figura 38.

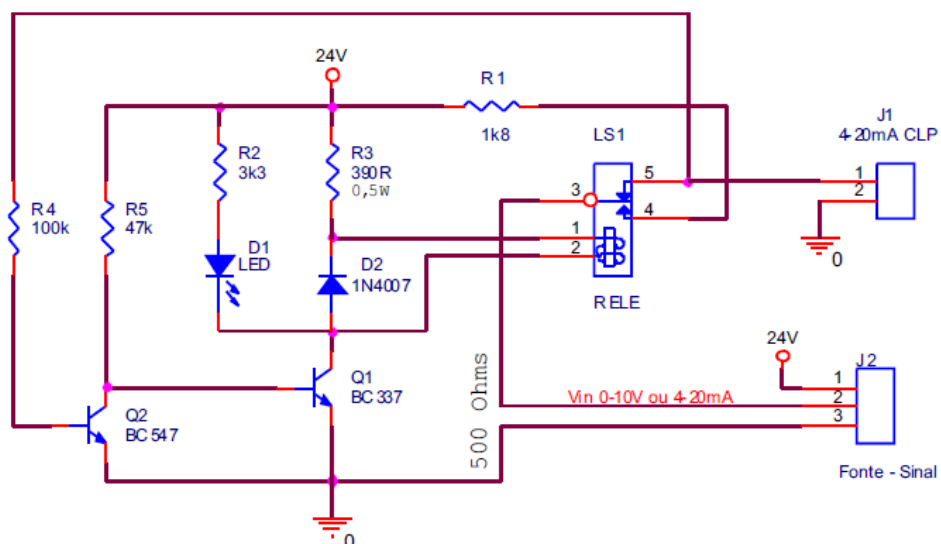


Figura 37: Circuito do dispositivo de identificação de CLP desligado.



Figura 38: Dispositivo de identificação de CLP desligado.

3.10.4 DISPOSITIVO CONVERSOR DE PULSOS DE TENSÃO PARA O *ENCODER*

O *encoder* que foi utilizado neste projeto possui como saída, pulsos de tensão de 5 V, no entanto a entrada digital existente no kit ZTK900 para *encoders* necessita para um bom funcionamento um sinal entrada com pulsos de tensão de 24 V, na Figura 39, tem-se a ilustração do esquema de ligação elétrica do *encoder* com o kit ZTK900, fornecido pela fabricante do kit [20].

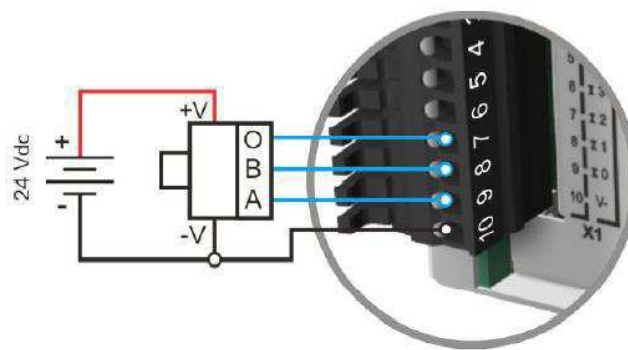


Figura 39: Conexão do *encoder*.
Fonte: [20].

Para converter o sinal digital de tensão 0 a 5 V fornecido pelo encoder para um sinal digital de tensão 0 a 24 V recomendado pelo fabricante do controlador, foi construído um dispositivo conversor de pulsos de tensão, Figura 40, utilizando o circuito ilustrado na Figura 41.

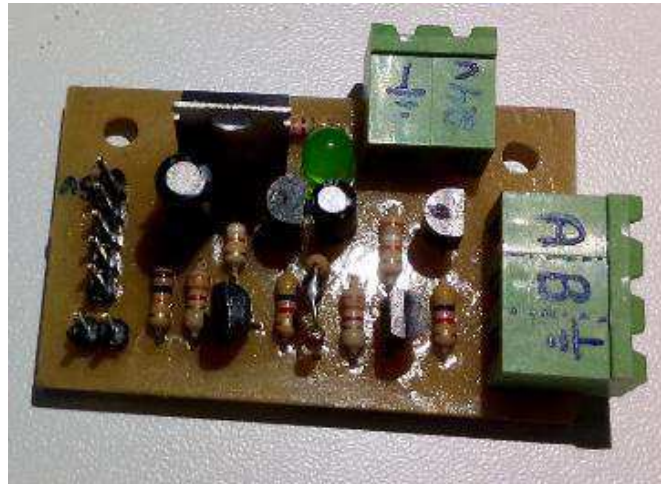


Figura 40: Dispositivo conversor de pulsos de tensão.

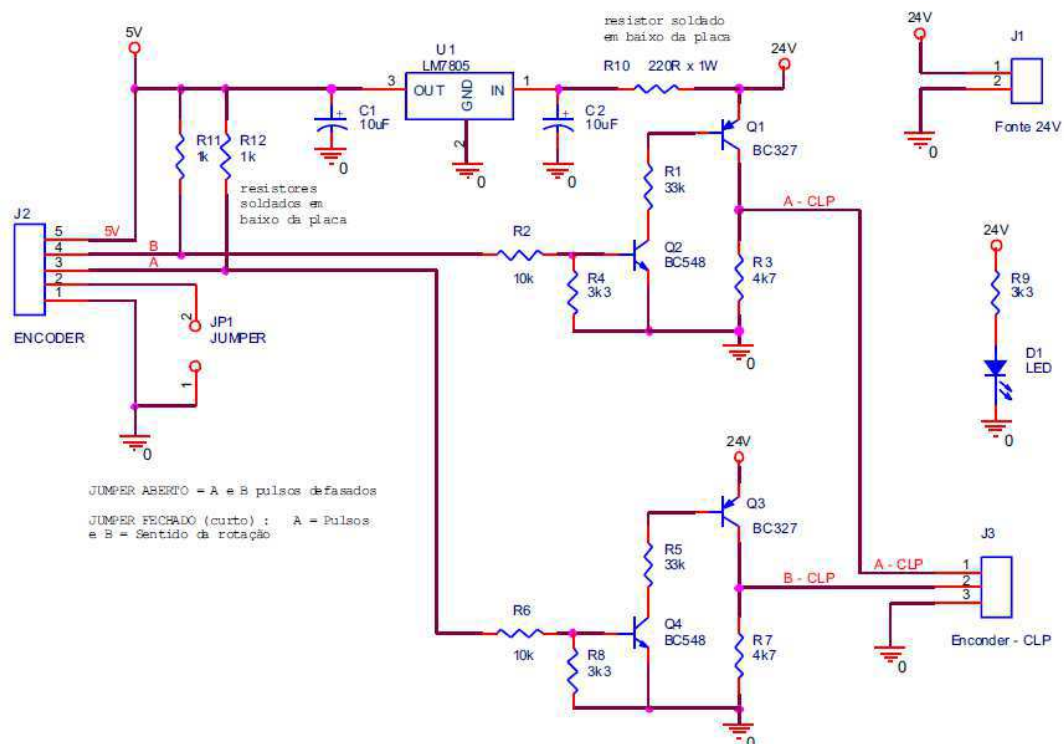


Figura 41: Circuito do dispositivo conversor de pulsos de tensão.

3.10.5 DISPOSITIVO DE CONVERSÃO PARA SENSOR LM35

Apesar do sensor de temperatura LM35 já possuir uma saída de tensão proporcional à temperatura, esta saída fornece valores de tensão que não podem ser identificados com precisão pela entrada analógica do CLP, ocasionando erros de leitura nas faixas de temperaturas desejadas, para solucionar este problema, foi construído um dispositivo que basicamente funciona como um amplificador analógico de tensão, Figura 42, de acordo com o circuito mostrado na Figura 43.

A saída analógica de tensão fornecida por este dispositivo conversor corresponde à faixa de 0 a 10 V, este valor de tensão é utilizado como entrada analógica para o dispositivo de controle.



Figura 42: Dispositivo de conversão para sensor LM35.

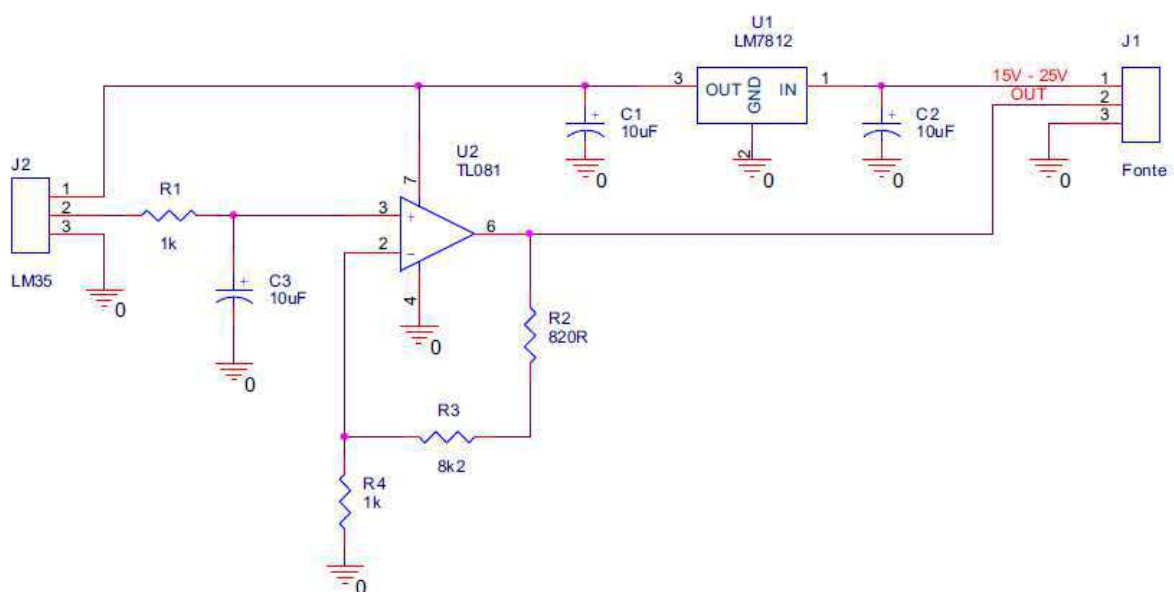


Figura 43: Circuito do dispositivo de conversão para sensor LM35.

Como o sensor de temperatura LM35, será utilizado tanto no módulo de aquecimento quanto no módulo de resfriamento, foi construído um único dispositivo que contem dois circuitos semelhantes ao ilustrado na Figura 43.

3.10.6 DISPOSITIVO DE CONVERSÃO DE CONTROLE PWM PARA RESISTÊNCIAS E VENTONHAS.

Para controlar a temperatura nos módulos de aquecimento e resfriamento, necessita-se primeiramente controlar a tensão fornecida às resistências e a tensão fornecida às ventoinhas. Para tal, foi construído um dispositivo de conversão de sinal PWM, Figura 44, que basicamente funciona como um amplificador de sinal PWM, já

que o sinal de entrada deste dispositivo é do tipo PWM e o sinal de saída é PWM. Este dispositivo de conversão é composto pelos circuitos ilustrados na Figura 45.



Figura 44: Dispositivo de Conversão Sinais PWM.

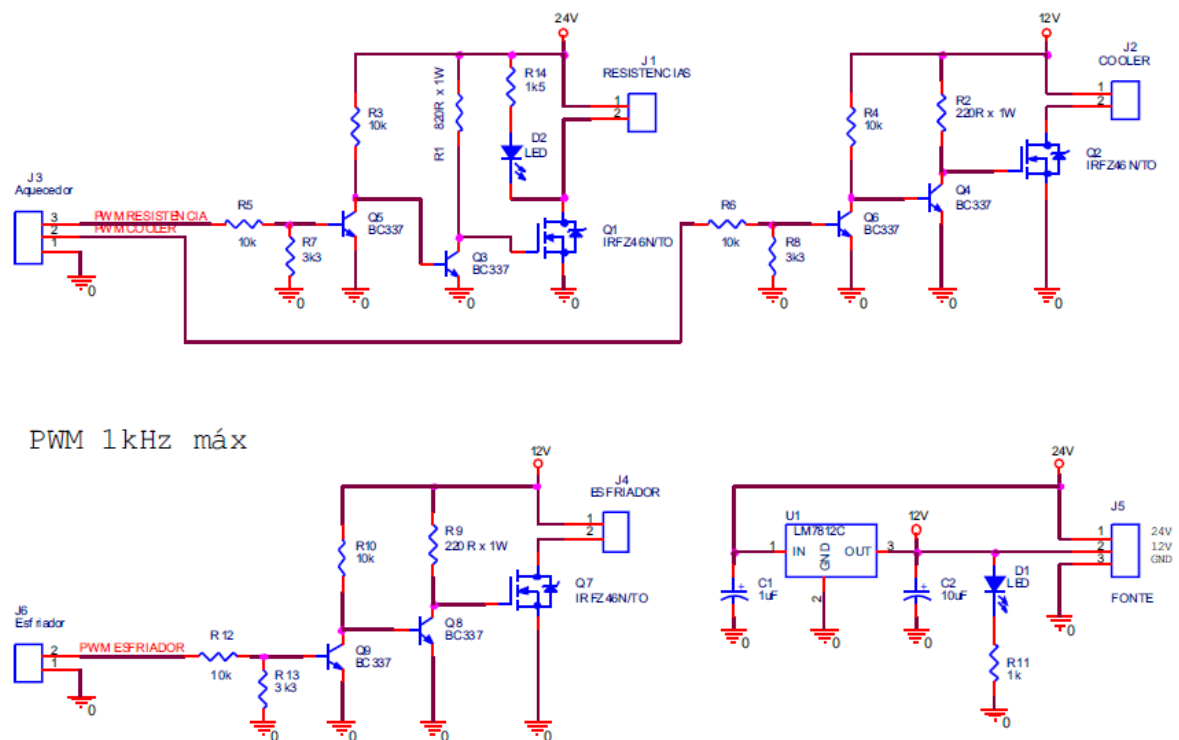


Figura 45: Circuitos do dispositivo de conversão de sinais PWM.

3.11 ESTEIRA COM DISPOSITIVOS INTEGRADOS

Após a integração dos dispositivos acima apresentados, o desenho técnico da esteira transportadora foi alterado para adicionar tais dispositivos, o desenho está agora de acordo com o apresentado na Figura 46.

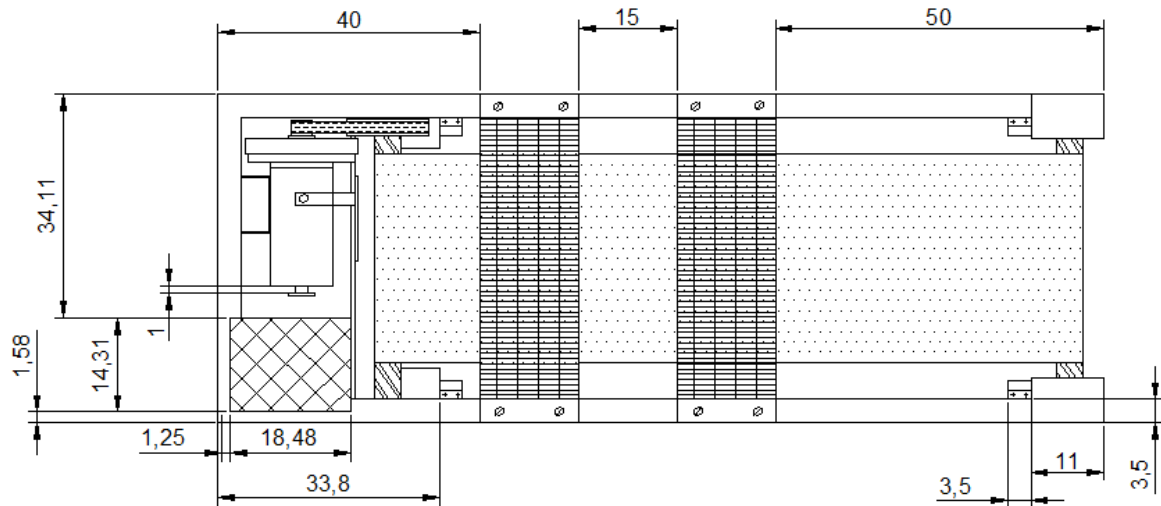


Figura 46: Desenho técnico da esteira com dispositivos integrados.

Na Figura 47, tem-se o desenho da esteira transportadora com a indicação dos dispositivos instalados.

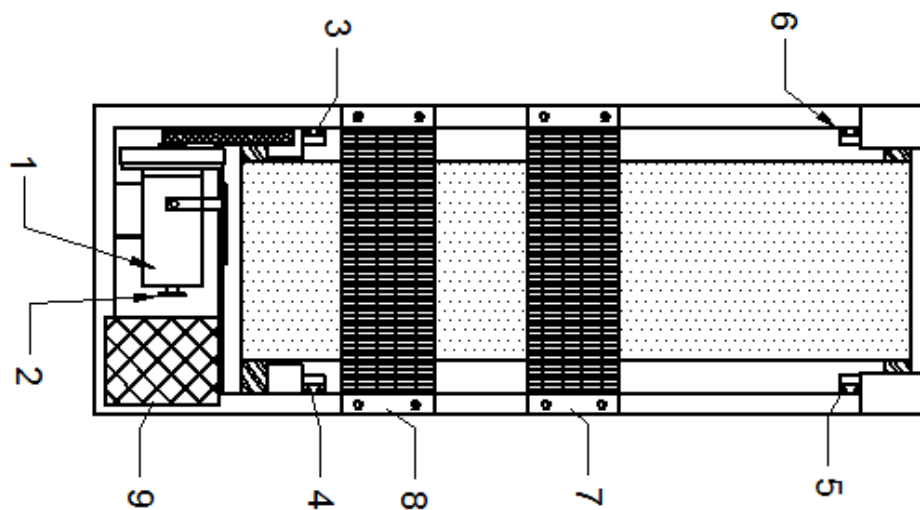


Figura 47: Desenho técnico com identificação dos dispositivos adicionados.

Os dispositivos identificados são os seguintes:

- 1 - Motor DC;
- 2 - *Encoder*;
- 3, 4, 5 e 6 - Sensores de barreira óptica;
- 7 - Módulo de Resfriamento;
- 8 - Módulo de Aquecimento;

- 9 - Suportes para fixação de dispositivos conversores e fontes de alimentação.

Na Figura 48, tem-se a esteira transportadora com seus respectivos dispositivos adicionados, de acordo com o desenho técnico apresentado na Figura 47.



Figura 48: Esteira transportadora com dispositivos integrados.

4 COMUNICAÇÃO CLP COM O PC

Para um bom funcionamento de um sistema de controle é fundamental que exista uma comunicação confiável entre os dispositivos que o compõe sejam eles sensores, transdutores, atuadores, controladores e etc. Em seguida será apresentado como realizar a comunicação do CLP com o PC (Computador Pessoal).

4.1 COMUNICAÇÃO CLP COM PC UTILIZANDO O *SOFTWARE*

SPDSW

Para a comunicação do kit de treinamento ZTK900 (controlador) como o PC, utiliza-se o *software* SPDSW que se trata de um ambiente de programação para os controladores da HI Tecnologia. O SPDSW é o ambiente de *software* desenvolvido para a configuração, programação, depuração, documentação de programas, e supervisão dos controladores da HI Tecnologia [21].

Integrado em um único aplicativo, o SPDSW oferece todos os recursos encontrados nos pacotes de programação para PLC's de grande porte, incluindo depuração, monitoramento *on-line* de variável e completa integração com as IHM's (Interface Homem Máquina) da HI [21].

O ambiente de programação da HI Tecnologia está disponível para *download* gratuitamente no site empresa. Na Figura 49, temos a ilustração da tela inicial do *software* SPDSW.



Figura 49: Tela inicial software SPDSW.
Fonte: [21].

Em seguida será descrito os passos para a criação de um novo projeto no SPDSW, assim como a configuração dos módulos de *hardware* associados ao controlador, para maiores detalhes consultar o documento EAT0100300 disponível no site da HI Tecnologia [22].

Para a criação de uma nova aplicação, no SPDSW, selecione a opção "Projeto | Novo", no menu principal do SPDSW, conforme ilustrado na Figura 50 [22]:

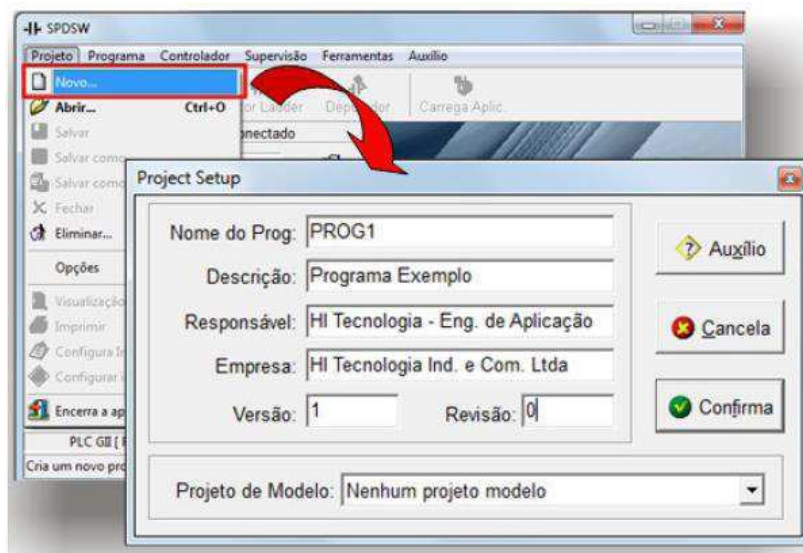


Figura 50: Criando uma nova aplicação no SPDSW.
Fonte: [22].

No formulário apresentado, devem-se preencher os parâmetros de identificação associados à nova aplicação.

Após criar uma nova aplicação, pode-se salvá-la no computador. Neste caso, selecione a opção "Projeto | Salvar", no menu principal do SPDSW, conforme ilustrado na Figura 51.

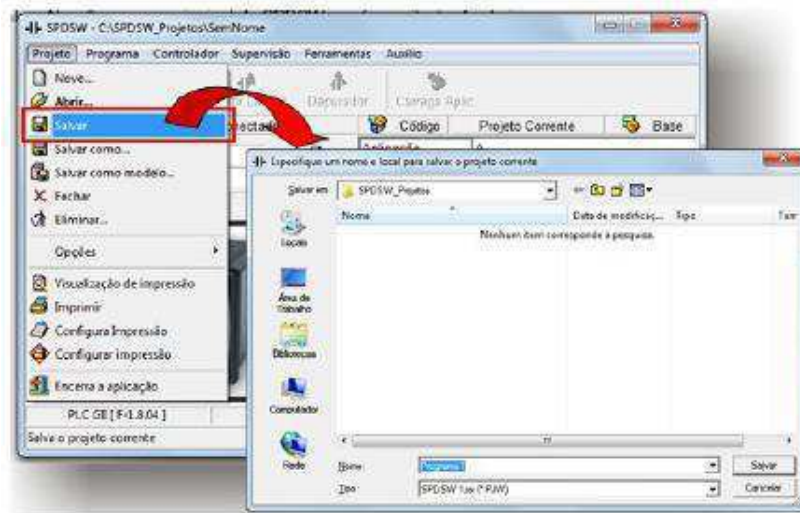


Figura 51: Salvando a nova aplicação no SPDSW.
Fonte: [22].

No formulário apresentado selecione o local onde desejar salvar os arquivos fontes associados à aplicação e especifique um nome para a mesma.

Após salvar a nova aplicação, é preciso configurar o *hardware* associado ao controlador a ser utilizado, ZAP900 associado ao kit de treinamento ZTK900. O controlador é composto pelos seguintes módulos de *hardware*, apresentados na Figura 52.

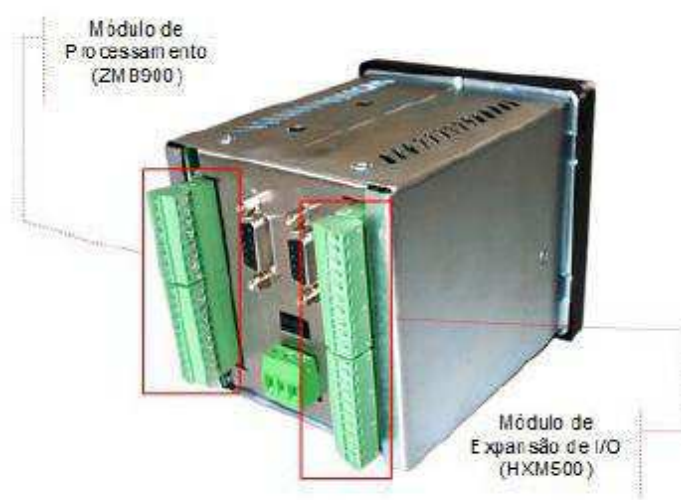


Figura 52: Módulos de *hardware* associados ao controlador ZAP900 do kit ZTK900.
Fonte: [22].

- Módulo de Processamento: corresponde ao módulo com a CPU do controlador e também disponibiliza alguns sinais de I/O. Existe um único módulo de processamento disponível, denominado ZMB900;
- Módulo de expansão de I/O: corresponde ao módulo de expansão de I/O. Deve-se utilizá-lo quando os I/O disponíveis no módulo ZMB900 não são suficientes para a aplicação. Existem diversos módulos de expansão, porém o padrão para o ZTK900 corresponde ao módulo HXM500, o que disponibiliza I/O digitais e analógicas.

Uma vez definido a configuração de *hardware* a ser utilizada no controlador ZAP900, basta defini-la no projeto de aplicação. Selecione a opção "Programa | Controlador Programável" no menu principal do SPDSW, conforme ilustrado na Figura 53.

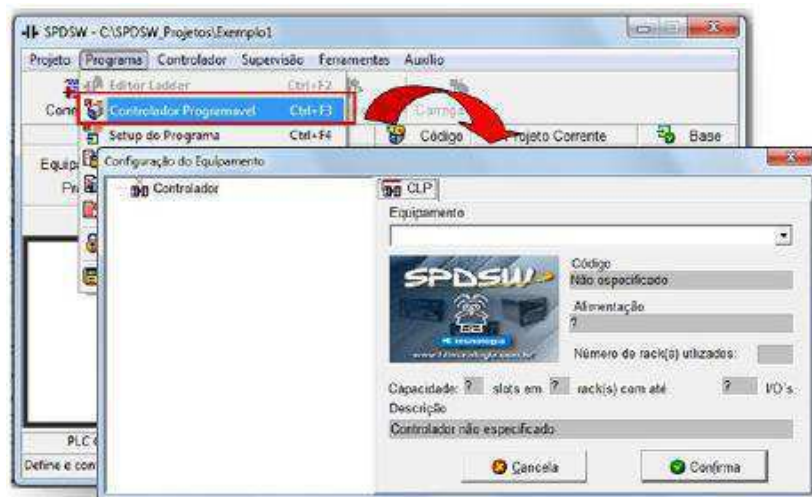


Figura 53: Configuração de *hardware* passo 1.
Fonte: [22].

Em seguida no formulário apresentado, selecione o controlador ZAP900, como apresentado na Figura 54.

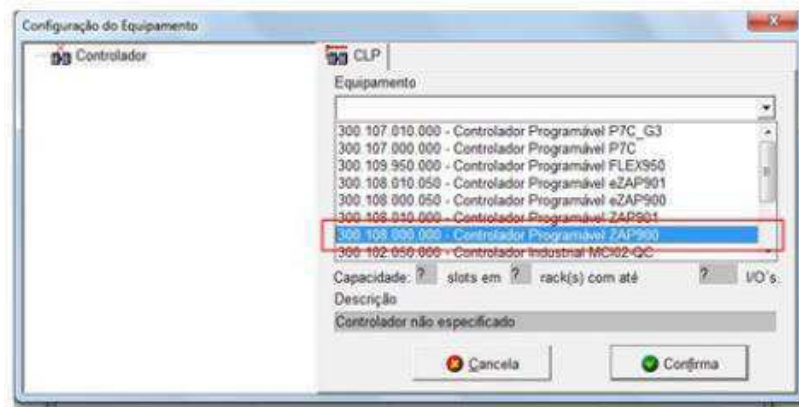


Figura 54: Configuração de *hardware* passo 2.
Fonte: [22].

Após a seleção do controlador ZAP900, apresenta-se o respectivo controlador, na árvore de composição do controlador, incluindo o módulo de processamento ZMB900, Figura 55.

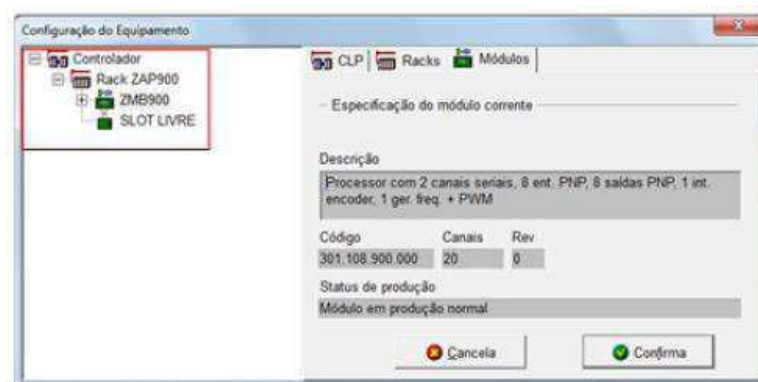


Figura 55: Configuração de *hardware* passo 3.
Fonte: [22].

O passo seguinte consiste em configurar o módulo de expansão de I/O do controlador ZAP900. Para tal seleccione o item "SLOT_LIVRE" na árvore de composição do controlador, Figura 56.

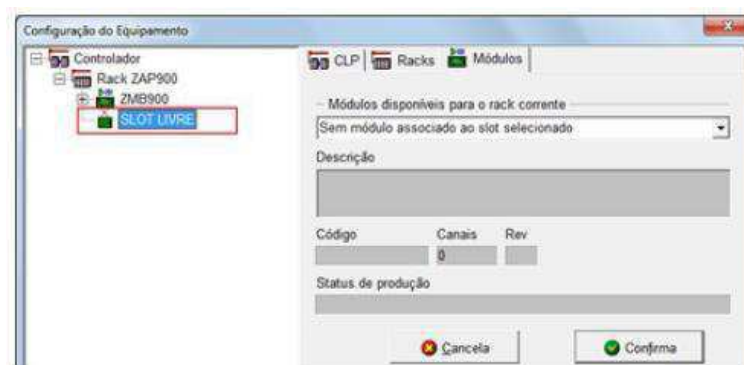


Figura 56: Configuração de *hardware* passo 4.
Fonte: [22].

Na aba "CLP", selecione o módulo de expansão correspondente, utilizado no controlador ZAP900. Por exemplo, o módulo de expansão HXM500, Figura 57.

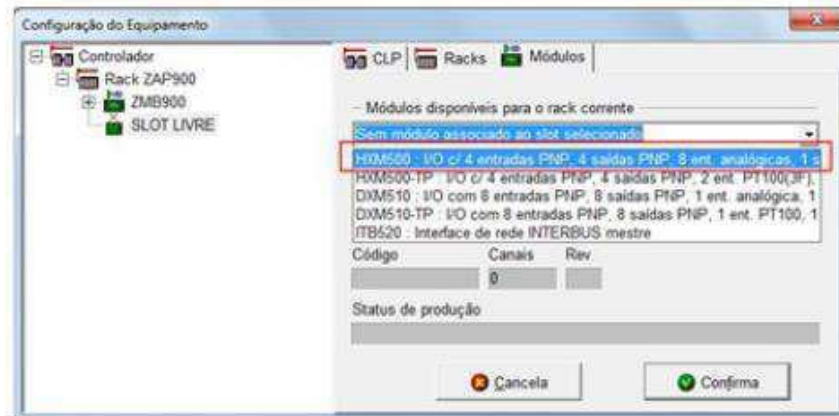


Figura 57: Configuração de *hardware* passo 5.
Fonte: [22].

Após a seleção do módulo de expansão, o mesmo aparece na árvore de configuração do projeto, bem como uma respectiva descrição associada ao mesmo. A configuração associada ao *hardware* do controlador está completa, basta selecionar a opção "Confirma", para concluir esta etapa de configuração, Figura 58.

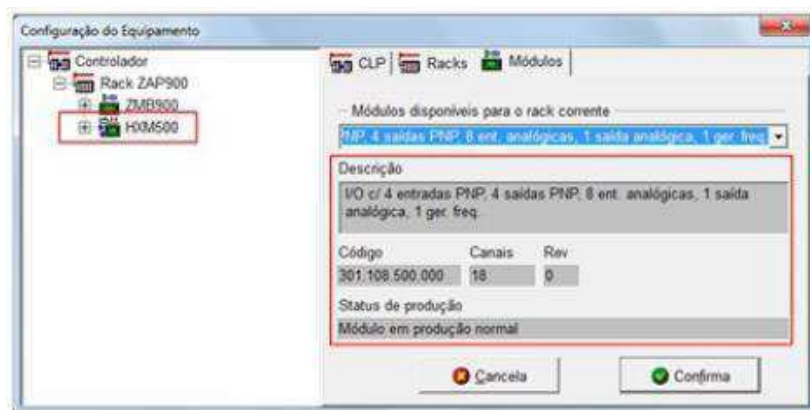


Figura 58: Configuração de Hardware passo 6.
Fonte: [22].

Uma vez definida a configuração do controlador a ser utilizado na aplicação, pode-se iniciar a edição das lógicas Ladder associadas à aplicação. Para selecionar o editor Ladder selecione a opção "Programa | Editor Ladder" no menu principal do SPDSW, ou através da seleção do botão de comando "Editor Ladder", disponível na barra de comandos do SPDSW, conforme ilustrado na Figura 59.

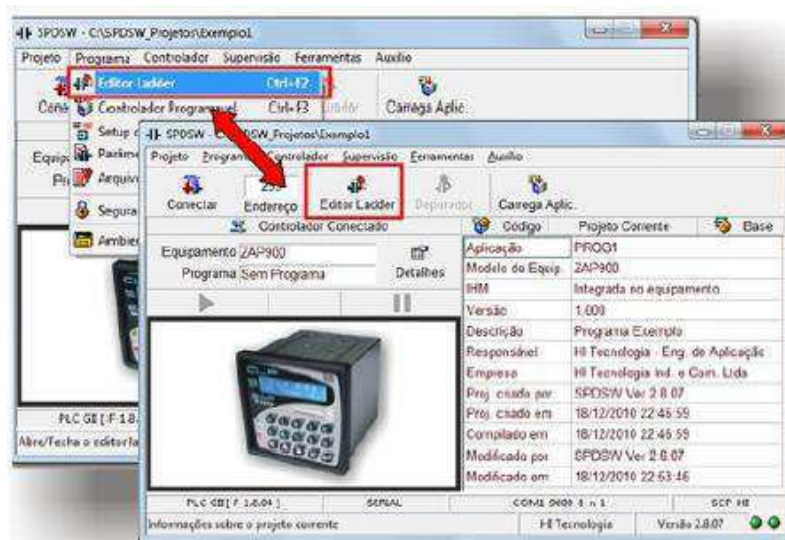


Figura 59: Editor Ladder do SPDSW.

Fonte: [22].

4.2 COMUNICAÇÃO CLP COM PC UTILIZANDO SERVIDOR OPC

A especificação OPC possibilita a substituição de todos os drives diferentes necessários para a comunicação entre dispositivos de fabricantes diferentes, por um único cliente OPC. A ideia é que cada equipamento possua seu servidor OPC. Este servidor possibilita a interface entre o protocolo do equipamento e o cliente OPC, assim, independente das particularidades de cada protocolo, entre o cliente e o servidor OPC existe um padrão para as trocas de dados [18].

4.2.1 INSTALAÇÃO DO DRIVER OPC

O *driver* HS1, desenvolvido pela HI Tecnologia, é um servidor que possibilita a comunicação serial. Este *driver* só poderá ser utilizado em plataformas com:

- Windows NT 4.0 com *Service Pack* 6a ou superior;
- Windows 2000 com *Service Pack* 3 ou superior;
- Windows XP com *Service Pack* 1 ou superior.

Neste trabalho utilizou-se um PC com Sistema Operacional *Windows 7 Professional*, e o *driver* HS1 foi instalado em módulo de compatibilidade com o *Windows XP Service Pack 1* seguindo as instruções do documento de instalação [18].

Os pacotes de instalação de cada Servidor OPC estão disponíveis para *download* no site da HI Tecnologia. Nesta página, selecione o *link* "Drivers de Comunicação" e faça o *download* do arquivo HS1OPC.zip para o HS1. Descompacte em um diretório conveniente e execute o arquivo *Setup*.

Dentro de cada arquivo de extensão zip é possível encontrar a instalação dos Servidores OPC e o pacote "OPC Core Components" disponibilizado também pela OPC Foundation. Este pacote possui módulos e bibliotecas utilizados por clientes OPC para acesso a servidores OPC. Sistemas de supervisão e outros clientes OPC normalmente instalam este pacote automaticamente [18].

A seguir os passos para o processo de instalação do *driver* HS1:

1. Execute o arquivo *Setup* presente no diretório criado para descompactar os arquivos de instalação, Figura 60;

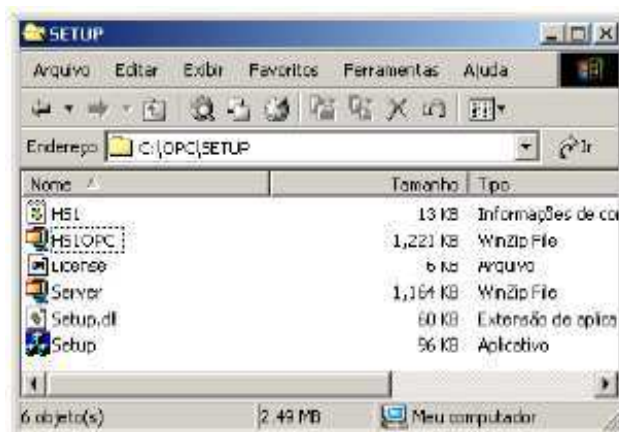


Figura 60: Arquivo *Setup Driver* HS1.

Fonte: [18].

2. Em seguida aparecerá a tela inicial de Instalação, Figura 61.

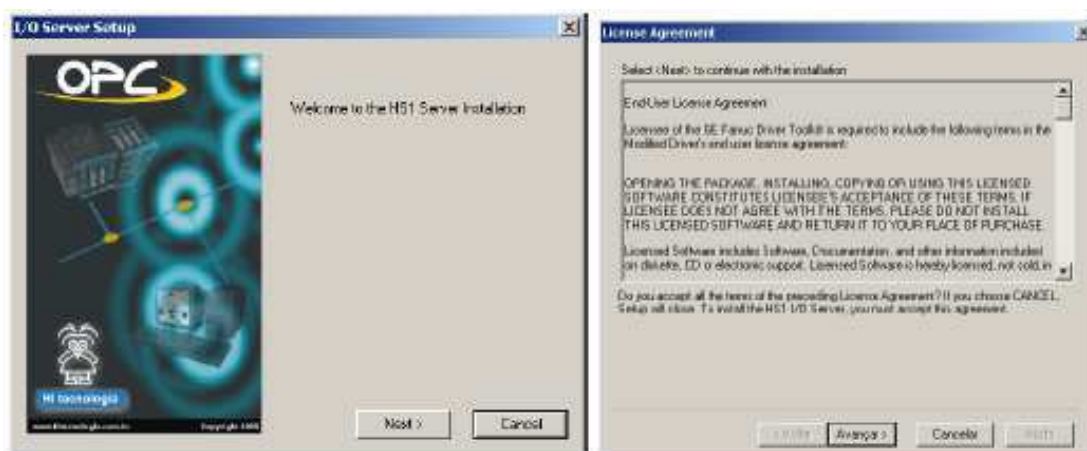


Figura 61: Tela Inicial de Instalação.

Fonte: [18].

3. Observação importante: é uma recomendação do fabricante que o *driver* não seja instalado em um diretório que possua espaços no nome, por exemplo:

- C:\Arquivos de programas\OPC\ (Forma incorreta);
- C:\Arquivos_de_programas\OPC\ (Forma correta).

Se não for seguida essa observação, pode acontecer do cliente OPC não encontrar o servidor HS1.

Seleção do diretório de instalação, Figura 62.



Figura 62: Seleção do diretório de instalação.

Fonte: [18].

4. Selecione a opção "Server". Este documento não fará a abordagem da utilização deste *Driver* como "Cliente", Figura 63.

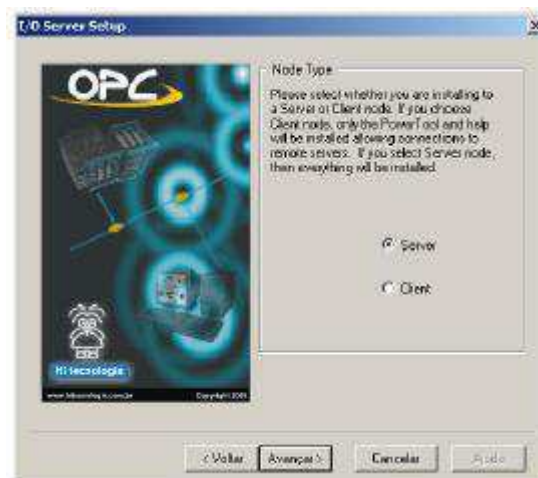


Figura 63: Seleção da opção "Server".

Fonte: [18].

5. Se o supervisor ou aplicação cliente não utilizar deste conceito, avance para o próximo passo, Figura 64.



Figura 64: Conceito "node name".
Fonte: [18].

6. Selecione a Pasta para instalação dos atalhos de acesso ao *Driver*, Figura 65. Pressione "Concluir" ou retorne e altere alguma opção dos passos anteriores.

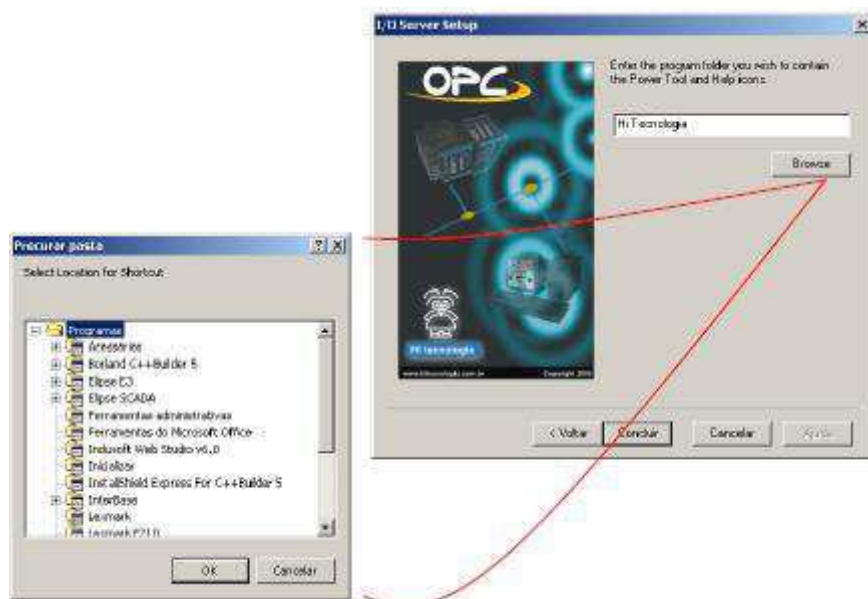


Figura 65: Seleção da pasta de instalação dos atalhos.
Fonte: [18].

7. Esta mensagem, Figura 66, indica que a instalação do *Driver* finalizou sem problemas.

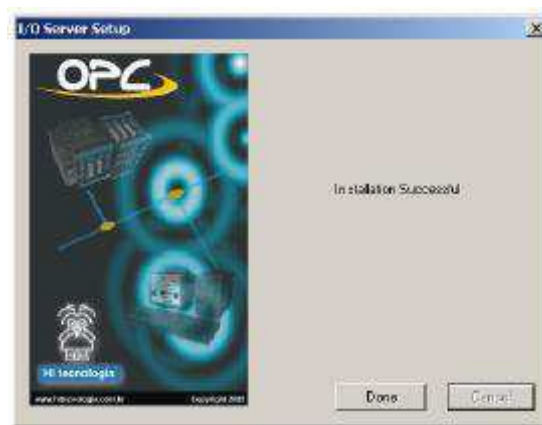


Figura 66: Finalização da instalação do *Driver*.
Fonte: [18].

8. Algumas configurações do *Windows* podem ocasionar o aparecimento da seguinte mensagem de erro ao final da instalação, Figura 67. Esta mensagem de erro não significa que o Servidor foi instalado incorretamente ou que apresentará problemas durante a instalação [18].

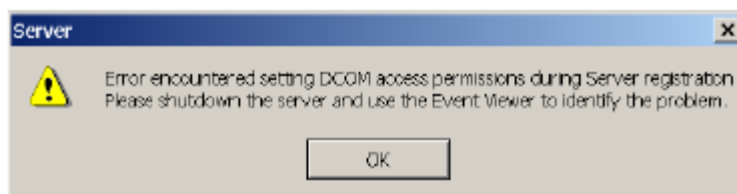


Figura 67: Mensagem de erro.
Fonte: [18].

4.2.2 CONFIGURAÇÃO DO *DRIVER* OPC

A Figura 68 apresenta a configuração da porta de comunicação COM1 do PLC da HI Tecnologia, visualizadas através do aplicativo SPDSW. Algumas das configurações do *driver* dependem destas informações. As destacadas em AZUL alteram diretamente algumas configurações do *driver*, as destacadas em VERMELHO são essenciais para que o *driver* funcione e devem permanecer com a configuração apresentada [18].

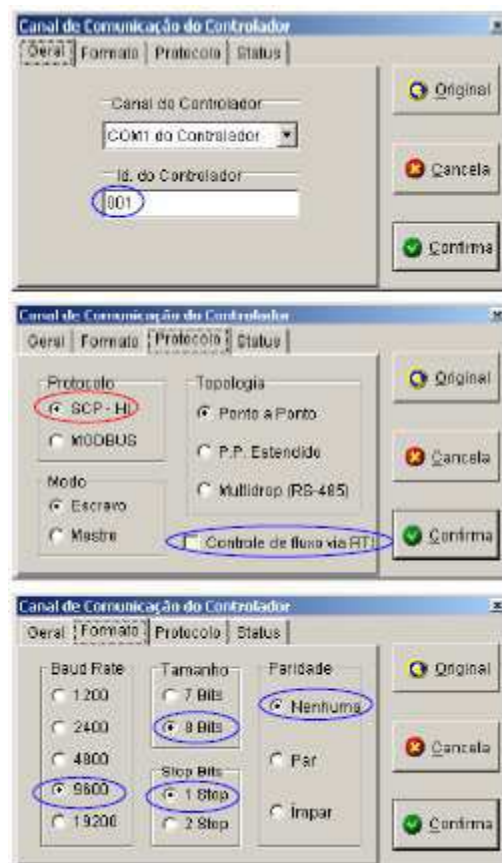


Figura 68: Configuração da porta de comunicação COM1.
Fonte: [18].

As configurações a seguir consideram que o computador está ligado a um único CLP. O acesso ao servidor pode ser feito de diversas maneiras, a mais fácil é acessar o *driver* na pasta de atalho escolhida na instalação. Na Figura 69 é apresentada este acesso.

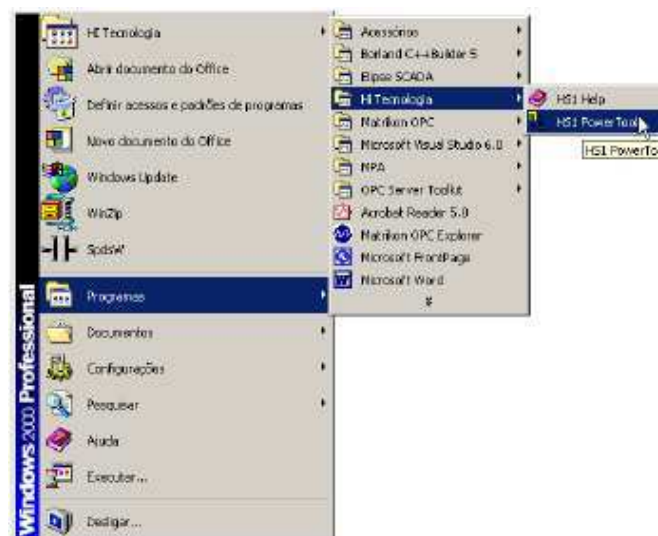


Figura 69: Acesso ao *driver* na pasta de atalho escolhida na instalação.
Fonte: [18].

Na Figura 70 é apresentada duas opções. Selecione a opção "*Use Local Server*". Isto significa que o servidor OPC está instalado na sua máquina.

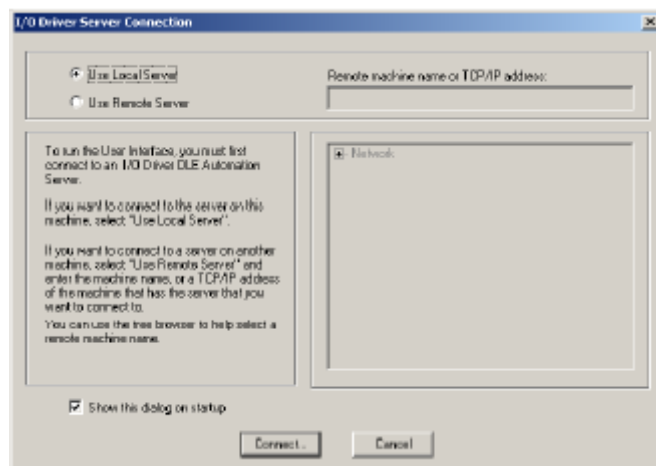


Figura 70: Seleção da opção "*Use Local Server*".
Fonte: [18].

É apresentado na Figura 71, os principais conceitos disponíveis na ferramenta de configuração do servidor.

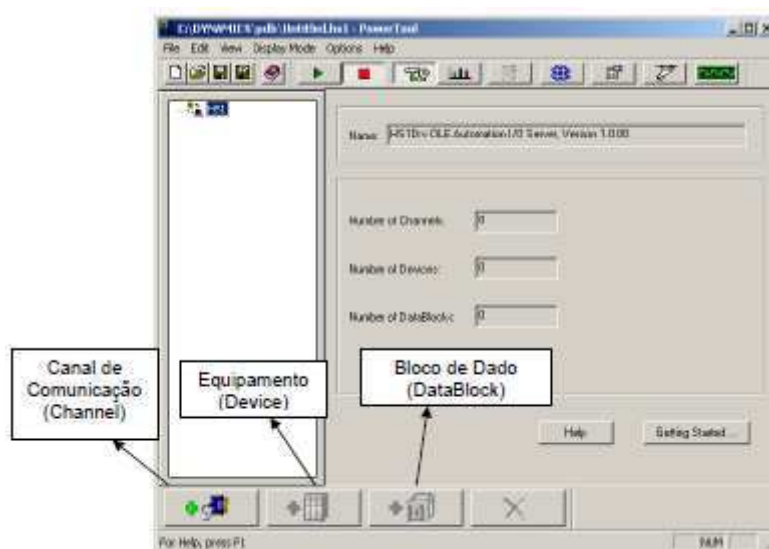


Figura 71: Ferramenta de configuração do Servidor.
Fonte: [18].

É indicado na Figura 72 como adicionar um canal de comunicação e as principais configurações disponibilizadas neste elemento.

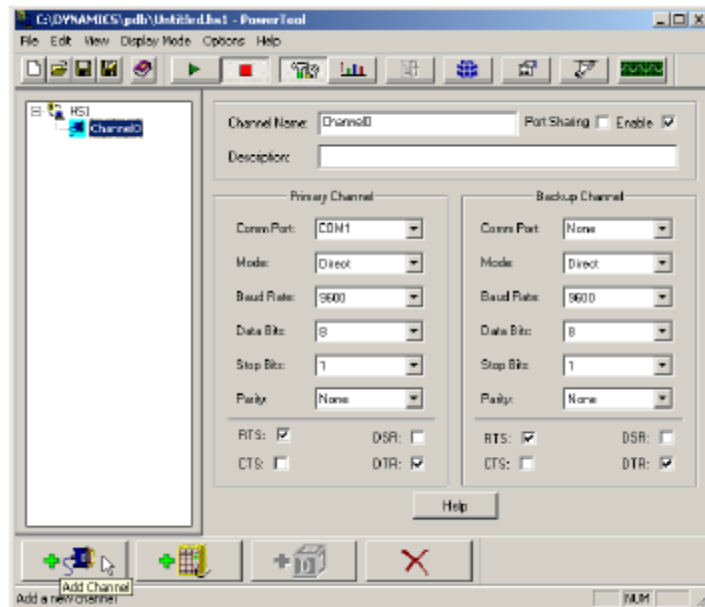


Figura 72: Adição de um canal de comunicação.
Fonte: [18].

Se as configurações do canal de comunicação do controlador forem mantidas iguais às figuras do início da seção, a única configuração necessária é a da porta de comunicação. Enquanto o endereço do equipamento é transparente às aplicações clientes, o nome será utilizado para mapear cada um destes endereços e, assim, para configurar a aplicação cliente, é necessário sempre conhecer o nome do "*Device*" e qual CLP ele está relacionado [18].

Na Figura 73 é apresentado como adicionar um "*Device*" e as configurações disponíveis para o HS1:

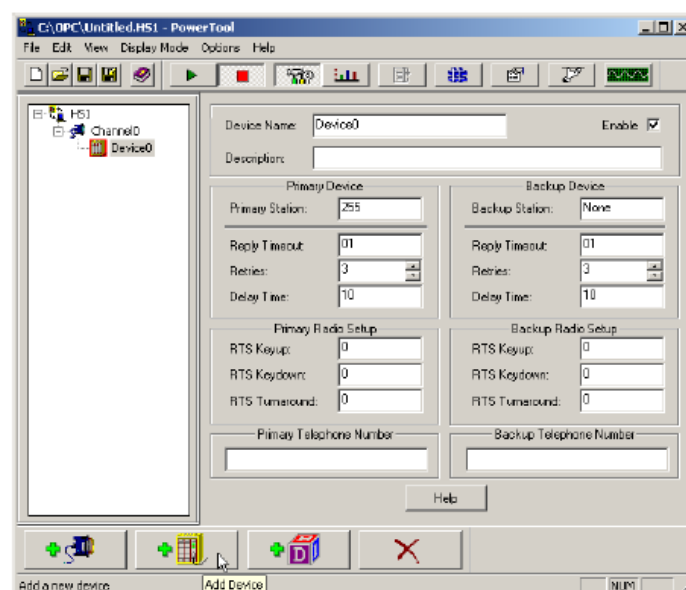


Figura 73: Adição de um "*Device*".
Fonte: [18].

Bloco de Dados (*Datablock*): Este elemento representa a região da memória contendo os dados lidos do controlador. São organizados de acordo com o tipo de memória do controlador que se deseja atribuir ao *Datablock*. Para os controladores da HI estão disponíveis três tipos de memória, tipo R (valores lógicos), Tipo M (inteiros com sinal: 16 bits) e Tipo D (ponto flutuante: 32 bits).

A Figura 74 ilustra como adicionar um novo *Datablock* e suas principais configurações.



Figura 74: *Datablock* e suas principais configurações.
Fonte: [18].

1. Endereço Inicial, Final e tamanho do *Datablock*;
2. "*Primary Rate*" representa o tempo em que o *driver* OPC envia mensagens ao controlador para atualizar o valor do bloco e dados. O valor *default* deste campo é 1 segundo. Este valor deve ser administrado quando se deseja otimizar a comunicação entre controlador e *driver* OPC;
3. "*Access Time*" é o tempo que o *driver* espera para tentar enviar uma nova mensagem quando ocorre um erro de comunicação;
4. É utilizado para adicionar blocos de dado ao Equipamento;
5. "*Enable*" habilita/desabilita que o *driver* envie mensagens de comunicação, no tempo especificado, ao controlador para obter o valor das memórias correspondentes aos endereços configurados no bloco D dado.

Para certificar-se de que as comunicações estão ocorrendo sem problemas, acesse as estatísticas conforme apresentado na Figura 75:

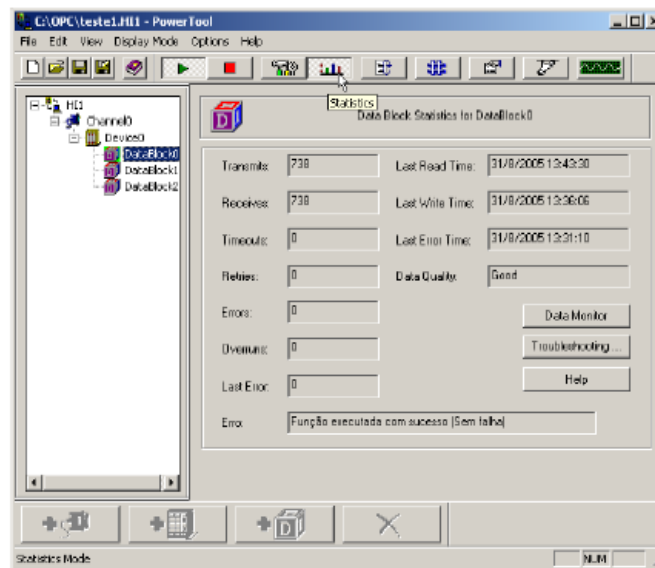


Figura 75: Teste de comunicação.
Fonte: [18].

Se a qualidade do dado (*Data Quality*) apresentar a identificação "Good", as mensagens correspondentes a este bloco de dados estão ocorrendo sem problemas, verifique também que o número de transmissões é igual ao número de recepções ("*Transmits*" e "*Receives*"). Isto significa que nenhuma mensagem enviada de leitura deste bloco de dados não recebeu resposta ou recebeu resposta erro.

Para maiores detalhes na configuração do Servidor OPC HS1, recomenda-se a leitura do documento ENA00033, referência [18].

4.3 COMUNICAÇÃO CLP COM SIMULINK UTILIZANDO

SERVIDOR OPC HS1

De acordo com o guia do usuário, o Simulink é um ambiente de diagramas de blocos para simulação de projetos baseados em modelos. Ele suporta projeto no nível do sistema, simulação, geração automática de código e verificação contínua de sistemas embarcados.

O Simulink é integrado com MATLAB, sendo capaz de incorporar algoritmos do programa principal em seus modelos e exportar resultados de simulação para o

MATLAB, possibilitando análises posteriores. A Figura 76 ilustra o navegador de bibliotecas do Simulink [23].

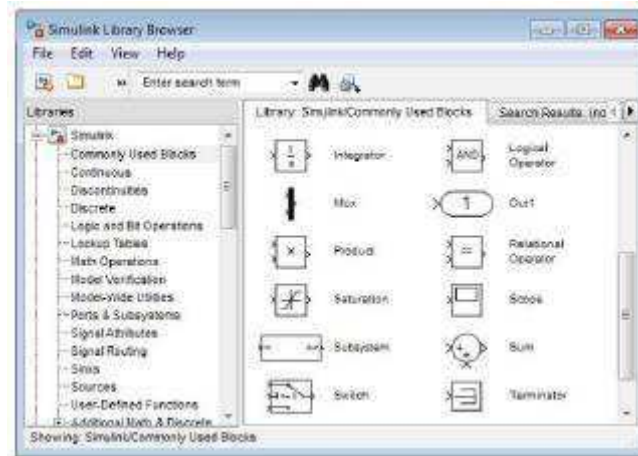


Figura 76: Navegador de bibliotecas do Simulink.
Fonte: [23].

Após a comunicação com Servidor OPC HS1, será realizada a comunicação OPC com a ferramenta Simulink do Software MATLAB, esta comunicação é de extrema importância neste trabalho, pois possibilita a realização de experimentos para identificação e controle dos sistemas.

A Figura 77 ilustra um exemplo da configuração dos blocos utilizados no Simulink para realizar um dos experimentos de identificação utilizado neste trabalho.

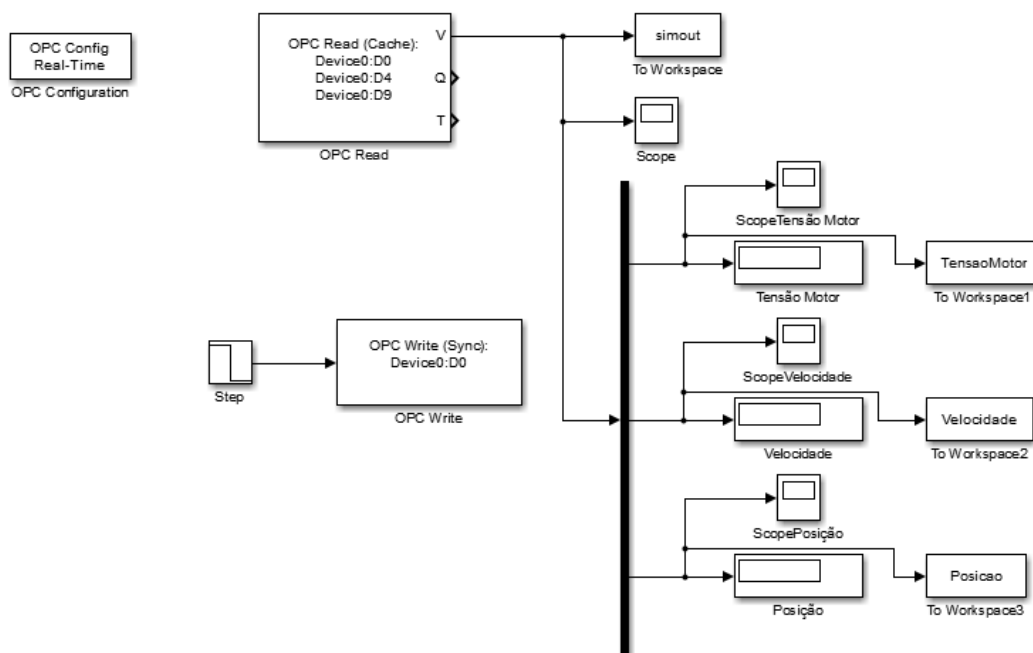


Figura 77: Experimento de identificação realizado no Simulink via OPC.
Fonte: [23].

Os blocos necessários para realizar a comunicação OPC são:

- *OPC Configuration*;
- *OPC Read*;
- *OPC Write*.

A seguir a descrição dos passos de um exemplo, de como realizar a comunicação do CLP com o Simulink por Servidor OPC:

1. Fechar o programa SPDSW ou outro que esteja se comunicando com o CLP;
2. Executar o programa do Servidor OPC HS1;
3. Com o programa HS1 aberto, clicar no Botão *Connect*;
4. Abrir o arquivo.HS1 já configurado (seção 4.2.2);
5. Ainda no HS1, habilitar o "*Channel*", clicando em "*Enable*", em seguida clicando no botão "*PLAY*" e testar a comunicação clicando no botão "*Statistics*"
6. Minimizar a janela do Servidor OPC HS1;
7. Executar o *software* MATLAB;
8. Abrir a ferramenta Simulink;
9. Clicar no botão *new model* e salvar o arquivo com extensão.slx;
10. No Simulink na biblioteca *OPC Toolbox*, selecionar os blocos: *OPC Configuration*, *OPC Read* e *OPC Write*;
11. Clicar duas vezes sobre o bloco *OPC Config Real-Time*;
12. Clicar em *Configure OPC Clients/Add/select*;
13. Selecione: *Intellution.HS1OPC*;
14. Em seguida clicar em *ok/ok/close/ok*;
15. Clicar duas vezes sobre o bloco *OPC Read*;
16. Em seguida clicar em *Add Itens/Device0/D4*;
17. Clicar no botão ">>";
18. Em seguida clicar em *ok/ok*;
19. Por fim clicar em *Play*.

5 PROJETO DO SISTEMA DE CONTROLE

Concluída a etapa de instrumentação e comunicação dos dispositivos como apresentado nos capítulos anteriores, inicia-se neste capítulo a etapa de projeto dos sistemas de controle, para tal será necessário definir as malhas de controle e descrever as principais funções do código de programação que serão utilizadas.

5.1 DEFINIÇÃO DAS MALHAS DE CONTROLE

As malhas de controle definidas neste trabalho estão relacionadas às malhas de controle da posição e da velocidade da esteira, malha de controle da temperatura do módulo de aquecimento e a malha de controle da temperatura do módulo de resfriamento. Todas as malhas de controle citadas são do tipo controle de malha fechada, na qual o sinal de saída é utilizado para determinar o sinal de controle que será aplicado ao processo.

5.1.1 CONTROLE DA POSIÇÃO E VELOCIDADE DA ESTEIRA

Para o controle da posição e velocidade da esteira será realizado o controle de posição e velocidade do motor CC que está conectado mecanicamente a esteira, para o motor CC a variável de comando que permitirá alterar a velocidade, será a tensão aplicada a armadura do motor, e a variável de controle serão pulsos fornecidos pelo *encoder* que está fixado ao eixo do motor.

O objetivo do controle da velocidade do motor é manter a velocidade medida igual à velocidade de referência dada à variação da carga mecânica e do controle de posição, e que o objeto que esteja sobre a esteira, esteja na posição desejada. Para o controle da velocidade será utilizado um controlador PID, que será implementado no CLP. Na Figura 78 é apresentado um diagrama de blocos básicos do sistema motor e controlador com realimentação.

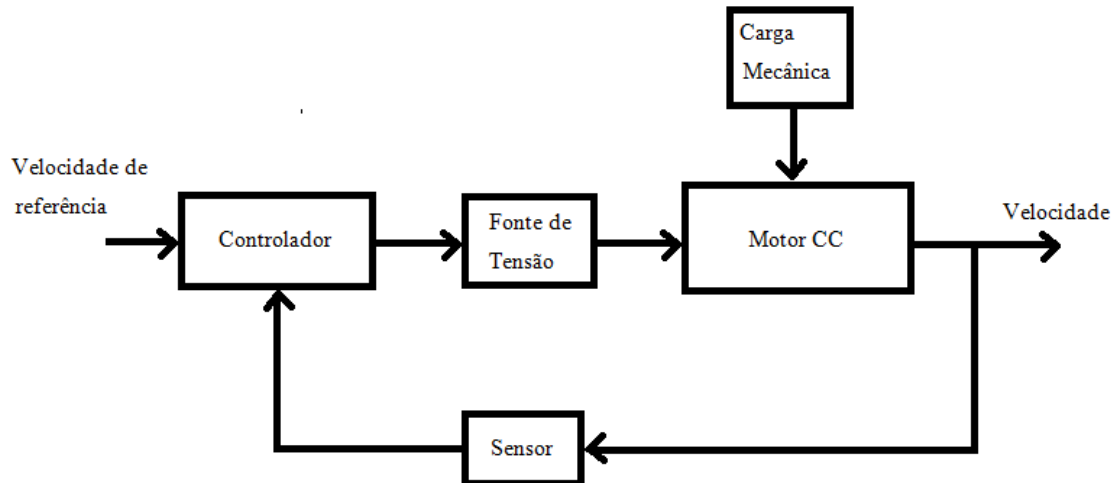


Figura 78: Diagrama de Blocos do Sistema de controle de posição e velocidade.

O controle da velocidade será realizado da seguinte forma: o *encoder* incremental, fixado ao eixo do motor, fornece um sinal na forma de pulsos de tensão, com amplitude de 0 a 5 V, ao dispositivo de conversão de pulsos, o sinal fornecido por este dispositivo com amplitude de 0 a 24 V é lido pelas entradas digitais dedicadas para leitura de *encoder* do kit de treinamento ZTK900, o qual possui o controlador CLP ZAP900, no programa contido neste controlador o sinal lido nas entradas dedicadas será tratado e comparado ao valor de referência para velocidade pré-configurado, a diferença entre estes valores corresponde ao erro, este valor de erro é convertido pelo CLP em um valor de saída de tensão entre 0 e 10 V, este valor de tensão será enviado ao dispositivo de conversão de controle do motor CC o qual fornecerá ao motor CC um valor PWM correspondente, e a este estará relacionado um valor de velocidade que será medido pelo *encoder* incremental, fechando a malha de controle.

Para o controle da posição foram utilizados os valores das velocidades medidas pelo CLP e estes valores serão integrados para determinar a posição. Para auxiliar este controle foram utilizados três conjuntos de sensores ópticos de barreira, que possibilitarão a identificação do objeto nas posições onde estes sensores estão instalados. Na Figura 79, tem-se a configuração do sistema de controle de velocidade e posição proposto.

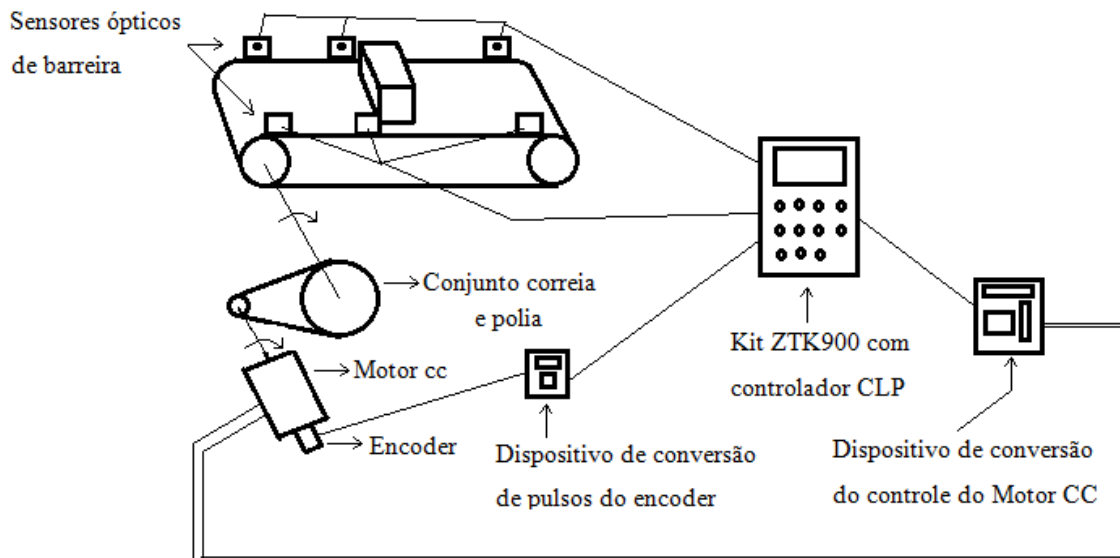


Figura 79: Sistema de controle de posição e velocidade proposto.

5.1.2 CONTROLE DE TEMPERATURA DO MÓDULO DE AQUECIMENTO

O módulo de aquecimento é composto por um elemento sensor, no caso o sensor de temperatura LM35, e dois elementos atuadores que constituem duas câmaras de aquecimento, formadas por uma resistência de chuveiro elétrico de 1400 W e uma ventoinha de 12V cada uma, como ilustrado na Figura 26, seção 3.6. Apesar de cada câmara possuir dois elementos atuadores, resistência elétrica e ventoinha, apenas um destes terá ação de controle, no caso a resistência elétrica, o outro elemento, a ventoinha será alimentada com uma tensão contínua de 12 V, mantendo um fluxo de ar constante em direção ao objeto que se pretende aquecer.

Sendo assim teremos nesta malha de controle o dispositivo de comando, a resistência elétrica, que possuirá como variável de comando o sinal PWM fornecido pelo dispositivo de conversão de sinais PWM, e o dispositivo de controle, o sensor de temperatura LM35, com variável de controle a tensão fornecida pelo mesmo, para esta malha, a perturbação será a temperatura do ambiente, juntamente com a perda de calor por processos de condução, assim como a temperatura do objeto que será aquecido, o ar será neste trabalho utilizado como meio de condução de calor neste processo. Na Figura 80, tem-se o diagrama de blocos do sistema e controlador com realimentação.

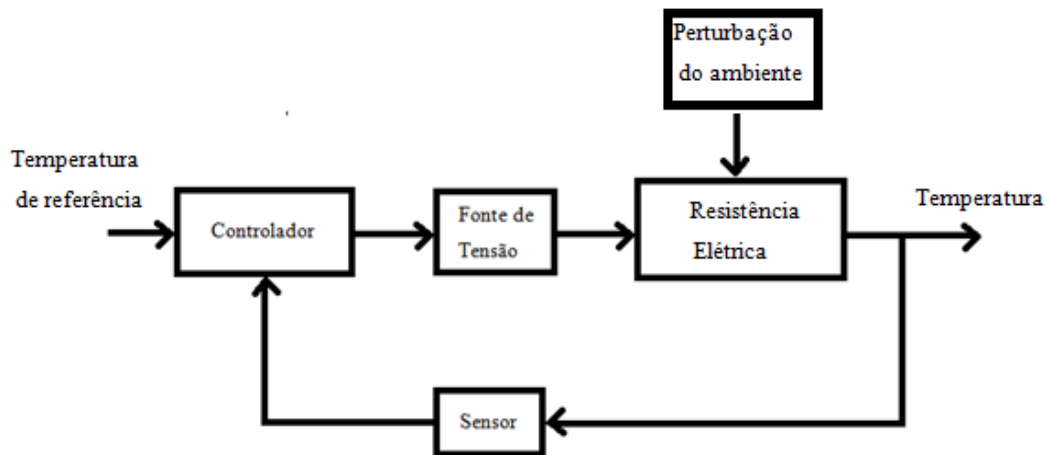


Figura 80: Diagrama de Blocos do Sistema de controle para o módulo de aquecimento.

O controle de temperatura do módulo de aquecimento será realizado da seguinte forma: o sensor de temperatura LM35 fornecerá um valor de tensão proporcional à temperatura registrada pelo mesmo, este valor de tensão é fornecido ao dispositivo de conversão para o sensor LM35, em seguida o sinal fornecido por este dispositivo é enviado para o kit ZTK900, o qual possui o controlador CLP ZAP900, no CLP este sinal é tratado e comparado com o valor de temperatura de referência pré-configurado no programa do CLP, a diferença entre o valor da temperatura medida e da temperatura de referência é convertido em um sinal PWM de controle fornecido pelo CLP, este sinal é enviado ao dispositivo de conversão de sinais PWM, o qual gera um sinal PWM de potência para as resistências elétricas, com auxílio das ventoinhas, que aumentam a eficiência da troca de calor entre o ar e as resistências, este sinal PWM de potência é convertido em calor pelas resistências elétricas, conseqüentemente a temperatura do módulo de aquecimento irá aumentar e esta será sentida pelo sensor de temperatura, fechando a malha de controle de temperatura do módulo de aquecimento.

Na **Error! Reference source not found.**, está ilustrado o sistema de controle de temperatura para o módulo de aquecimento.

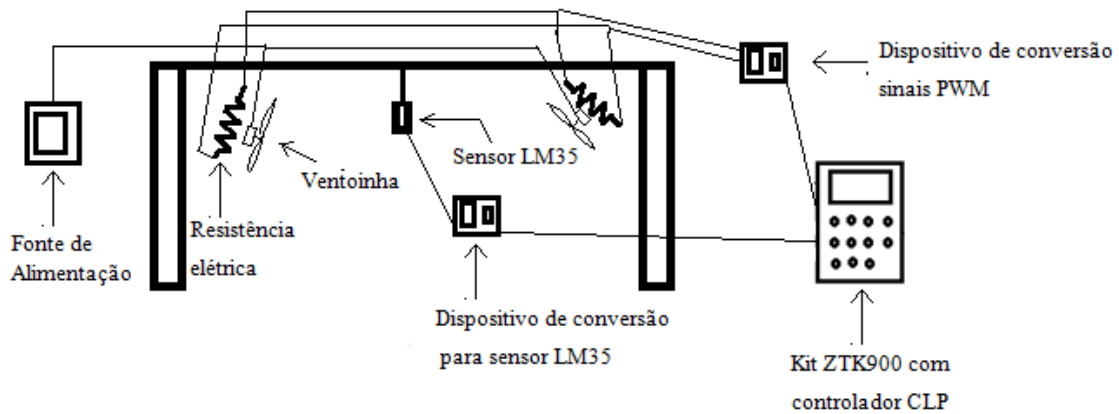


Figura 81: Sistema de controle de temperatura para o módulo de aquecimento.

5.1.3 CONTROLE DE TEMPERATURA DO MÓDULO DE RESFRIAMENTO

O módulo de resfriamento é composto por um elemento sensor, no caso o sensor de temperatura LM35, e dois elementos atuadores, duas ventoinhas de 12 V cada uma.

Na Figura 82 está ilustrado o diagrama de blocos que representa a malha de controle para o módulo de resfriamento.

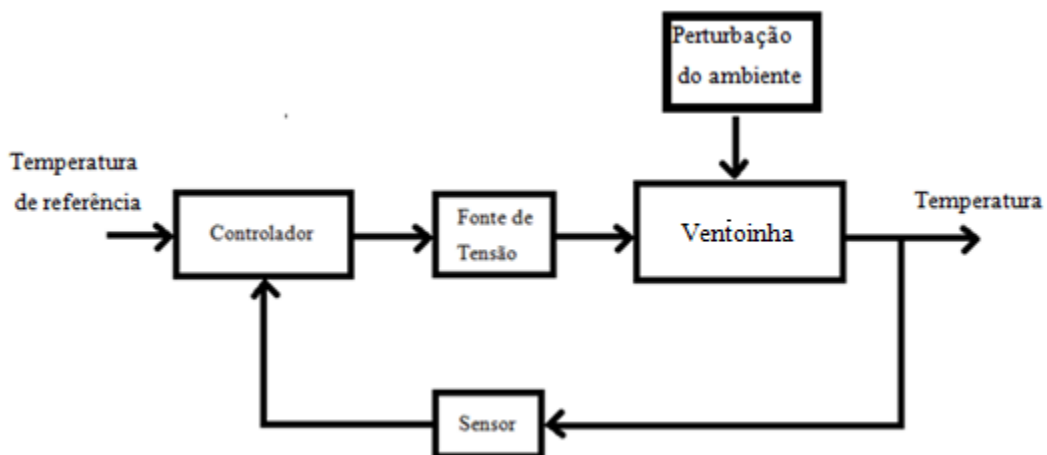


Figura 82: Diagrama de Blocos do Sistema de controle para o módulo de resfriamento.

O controle de temperatura do módulo de resfriamento será realizado da seguinte forma: o sensor de temperatura LM35 fornecerá um valor de tensão proporcional à temperatura registrada pelo mesmo, este valor de tensão é fornecido ao dispositivo de conversão para o sensor LM35, em seguida o sinal fornecido por este dispositivo é enviado para o kit ZTK900, o qual possui o controlador CLP ZAP900, no CLP este sinal é tratado e comparado com o valor de temperatura de referência pré-configurado

no programa do CLP, a diferença entre o valor da temperatura medida e da temperatura de referência é convertido em um sinal PWM de controle fornecido pelo CLP, este sinal é enviado ao dispositivo de conversão de sinais PWM, o qual gera um sinal PWM de potência para as ventoinhas, este sinal PWM é proporcional a velocidade das ventoinhas que formam um túnel de vento com uma maior intensidade e, conseqüentemente faz com que o objeto que esteja no caminho deste túnel de vento perca calor com mais facilidade, isto implica que a temperatura do módulo de resfriamento irá diminuir e esta será sentida pelo sensor de temperatura, fechando a malha de controle de temperatura do módulo de resfriamento.

Na Figura 83, está ilustrado o sistema de controle de temperatura para o módulo de resfriamento.

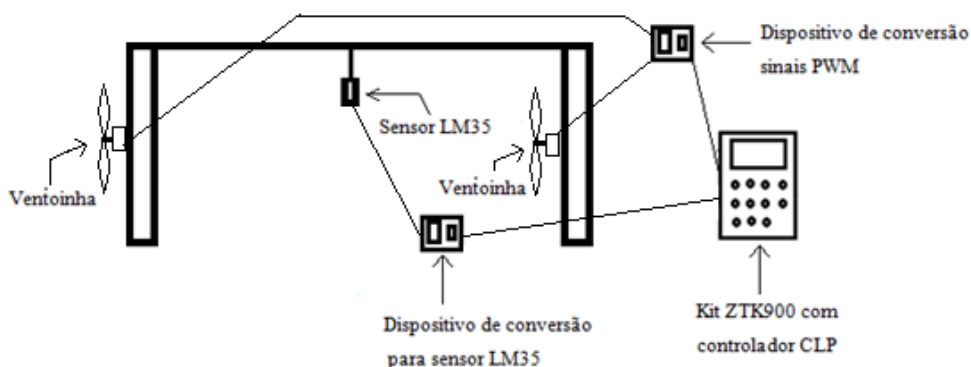


Figura 83: Sistema de controle de temperatura para o módulo de resfriamento.

5.2 DESCRIÇÃO DE CÓDIGO DE PROGRAMAÇÃO DAS FUNÇÕES

PRINCIPAIS

Neste tópico serão descritas as principais funções utilizadas neste trabalho na programação do CLP, a linguagem de programação é o LADDER, a compreensão destas funções é de extrema importância para seu correto funcionamento. As funções abordadas aqui são pertencentes ao *software* SPDSW e compatíveis com os controladores da HI Tecnologia.

5.2.1 MOVIMENTAÇÃO DE VARIÁVEIS

O bloco MOV, Figura 84, possibilita a inicialização de valores de memórias, a transferência de dados entre memórias, textos, entradas e saídas do PLC e a conversão

de tipos de dados entre vários formatos. Este único bloco implementa a funcionalidade normalmente encontrada em vários blocos de outros PLC's do mercado. Essencialmente o bloco de movimentação transfere o valor do operando P1 para o operando P2.

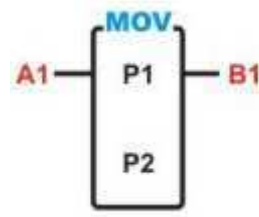


Figura 84: Bloco MOV.
Fonte: [21].

No programa desenvolvido, esta função foi utilizada em muitos trechos de código, para uma utilização correta deste bloco é necessário inicialmente uma compreensão dos tipos de variáveis e suas particularidades. Na Figura 85, está ilustrado um trecho do código, onde a função MOV é utilizada para mover o valor lido pela entrada analógica E0007 para uma variável auxiliar de 32 bits, realizando uma cópia deste valor para esta variável resolvendo, por exemplo, problemas de conflitos de *hardware* de leitura e escrita.

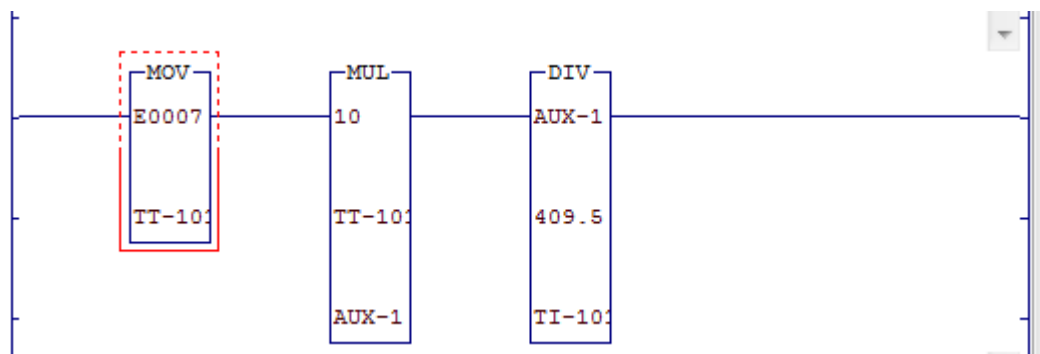


Figura 85: Trecho de Código utilizando o Bloco MOV.

5.2.2 CONTADOR RÁPIDO *ENCODER*

O bloco FCT, Figura 86, permite acesso aos módulos de interface para *Encoder* (contador em quadratura) e contador rápido, disponíveis na linha de PLC's. Com este bloco é possível, após configurar as entradas digitais I0000, I0001 e I0002 como entradas para *encoder* incremental, realizar de forma simples a leitura do *encoder* incremental.

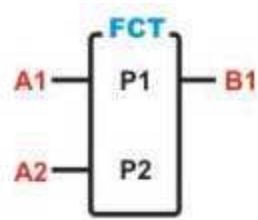


Figura 86: Bloco FCT.
Fonte: [21].

Na Tabela 2, está apresentando para este bloco, a descrição dos parâmetros, das entradas e a saída.

Tabela 2: Descrição dos Parâmetros, das Entradas e da Saída.

Parâmetro	Tipo(s)	Valores	Descrição
P1	T	0.. (*1)	Identificador do canal de encoder. Deve começar com T0 para o primeiro canal e ser incrementado seqüencialmente para cada novo canal incorporado. (*1) O valor máximo de T depende da quantidade de canais de encoder disponíveis no controlador selecionado.
P2	M	0.. Max(M)	Memória com o número de pulsos corrente do encoder. Este valor será incrementado ou decrementado em função do sentido de rotação do encoder podendo assumir os valores de -32768 a +32767.

Entradas	Função	Estado	Descrição
A1	Habilitação	Inativo	Nesta condição o bloco estará bloqueado
		Ativo	Nesta condição o bloco estará operacional passando a depender da entrada A2 para contar ou não os pulsos do encoder.
A2	Ativação / Reset	Inativo	Mantém o módulo em Reset fazendo o parâmetro P2 = 0 e ignorando os pulsos gerados pelo encoder
		Ativo	Se esta entrada estiver ativa juntamente com a entrada A1 o bloco estará apto a contar os pulsos do encoder. Neste caso a cada novo pulso o valor do parâmetro P2 será incrementado ou decrementado dependendo do sentido de rotação do eixo do encoder.

Saída	Função	Estado	Descrição
B1	Em Reset / Contando	Inativo	A saída permanece neste estado até que as entradas A1 e A2 sejam ativadas.
		Ativo	Indica que o bloco está gerando frequência. A saída permanecerá nesta condição até que seja dado um novo reset no bloco através da entrada A2.

Fonte: [21].

5.2.3 TEMPORIZADOR

O bloco que implementa esta função é o bloco TMR, Figura 87. Em diversas situações em processos de controle é necessário o sincronismo da taxa de aquisição de dados compartilhados por dispositivos diferentes. Um temporizador pode ser utilizado para realizar tal sincronismo.

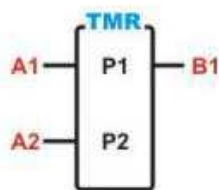


Figura 87: Bloco TMR.
Fonte: [21].

O bloco TMR é o responsável por temporizar eventos dentro de um programa, ele opera, basicamente, ativando a saída após a contagem de um tempo pré-programado.

Este é representado por um bloco funcional com duas entradas (A1 e A2) e uma saída (B1) e utiliza dois operadores (P1 e P2). O operador P1 que aparece na parte superior do bloco guarda o valor corrente do tempo e deve ser obrigatoriamente uma memória inteira (M) ou memória inteira longa (L). O parâmetro P2 contém o valor total do tempo, e se o parâmetro P1 for inteiro (M) pode ser uma memória inteira (M) ou uma constante inteira (K), e se o parâmetro P1 for inteiro longo (L) pode ser uma memória inteira longa (L) ou uma constante inteira longa (Y).

Este tempo é especificado no parâmetro P2 e seu valor é dado em múltiplos de 0,01 segundos, ou seja, um intervalo de 1 segundo será contabilizado se o valor de P2 for igual a 100, como apresentado na Tabela 3. Os parâmetros P1 e P2 devem ser do mesmo tipo ou inteiro ou inteiro longo. O parâmetro P2 pode ser do tipo inteiro (M ou K) ou inteiro longo (L ou Y), sendo exemplos de tempos limites do timer:

Tabela 3: Relação do parâmetro P2 com o tempo contabilizado, para o bloco TMR.

Valor de P2	Tipo de parâmetro P1 e P2	Tempo contabilizado
0	inteiro ou inteiro longo	0 seg
1	inteiro ou inteiro longo	10 ms
2	inteiro ou inteiro longo	20 ms
100	inteiro ou inteiro longo	1000 ms (1 seg)
5000	inteiro ou inteiro longo	50000 ms (50 seg)
32767	inteiro ou inteiro longo	327,67 seg (05 min 27 seg 67 ms)
-32768	inteiro ou inteiro longo	327,68 seg (05 min 27 seg 68 ms)
-1	inteiro	655,35 seg (10 min 55 seg 35 ms)
2147483647	inteiro longo	21474836,47 seg (5965 hor 13 min 56 seg 47 ms)

Fonte: [21].

5.2.4 GERADOR DE FREQUÊNCIA

O bloco FQG, Figura 88, permite acesso aos módulos de geração de frequência programável, disponíveis em alguns PLCs da linha. Este bloco será utilizado para gerar o sinal PWM para o controle das resistências e das ventoinhas.

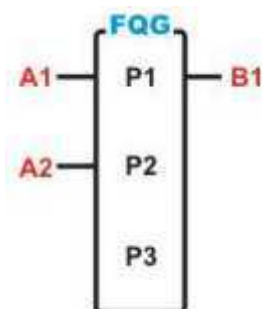


Figura 88: Bloco FQG.
Fonte: [21].

Para que este bloco funcione de forma correta é importante definir adequadamente seus parâmetros e como os demais definir suas entradas e saídas. Na Tabela 4, tem-se a descrição destes parâmetros, das entradas e da saída, para o bloco FQG.

Tabela 4: Descrição dos parâmetros, das entradas e da saída para o bloco FQG.

Parâmetro	Tipo(s)	Valores	Descrição
P1	T	0.. (*1)	Identificador do canal do bloco gerador de frequência. Deve começar com T0 para o primeiro bloco e ser incrementado seqüencialmente para cada novo módulo incorporado. (*1) O valor máximo de T depende da quantidade de canais de frequência programável disponíveis no controlador selecionado.
P2	M K	0..Max(M) 0..FMax 0..Max(K) 0..FMax	Define o valor da frequência a ser gerada pelo módulo. Os valores máximos e mínimos dependem do tipo de módulo de hardware utilizado.
P3	M K	0..Max(M) 0..Max 0..Max(K) 0..Max	Define o valor do ciclo de trabalho (Duty Cycle) utilizado. É especificado em forma de percentual, podendo variar de 0 a 99% ou de 0 a 999 (99,9%). Este recurso não está disponível em todos os módulos. Quando não disponível, o sinal de saída possuirá ciclo de trabalho de 50%.

Entradas	Função	Estado	Descrição
A1	Ativação	Inativo	Nesta condição o bloco não é processado.
		Ativo	Nesta condição o bloco estará operacional passando a depender da entrada A2.
A2	Desliga	Inativo	Bloqueia a saída de geração da frequência.
	Liga	Ativo	Habilita a geração da frequência e ciclo de trabalho programados em P2 e P3.

Saída	Função	Estado	Descrição
B1	Ativação	Inativo	A saída permanece neste estado as entradas A1 e A2 sejam ativadas.
		Ativo	Indica que já havendo geração de frequência. A saída permanecerá nesta condição até que seja dado um novo reset no bloco através da entrada A2.

Fonte: [21].

5.2.5 BLOCO DE CONTROLE PID

O objetivo deste bloco PID, Figura 89, é executar um algoritmo de controle Proporcional, Integral e Derivativo (PID). Este elemento é representado por um bloco de duas entradas e uma saída, com um operador. O operador (P1) especifica o número do bloco de controle PID, de 0 a 7. Os parâmetros de controle são inseridos durante a fase de edição do programa através do comando "Blocos Contr." da janela de comandos.

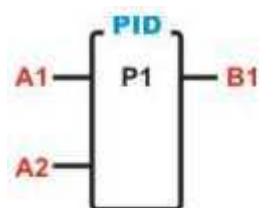


Figura 89: Bloco PID.
Fonte: [21].

Na Tabela 5, tem-se a descrição dos parâmetros, entradas e saída do bloco PID.

Tabela 5: Descrição dos parâmetros do bloco PID.

Parâmetro	Tipo(s)	Valores	Descrição
P1	T	0..n (T)	Identifica o bloco de controle. G1, GII: Pode estar na faixa entre 0 a 7, suportando até 8 blocos de controle PID permitidos em um programa de aplicação GII-Duo, G3: Pode estar na faixa entre 0 a 31, suportando até 32 blocos de controle PID permitidos em um programa de aplicação

Entradas	Função	Estado	Descrição
A1	Ativação	Inativo	Entrada de habilitação do bloco de controle PID. Sempre que esta entrada estiver desenergizada o controle não estará sendo executado.
		Ativo	Entrada de habilitação do bloco de controle PID. Sempre que esta entrada estiver energizada o controle estará sendo executado
A2	Ativação	Inativo	Entrada de modo de operação (Manual/Automático) do bloco de controle PID. Sempre que esta entrada estiver desenergizada o controle estará sendo executado em modo manual (veja no item "Descrição" as implicações do modo manual e automático).
		Ativo	Entrada de modo de operação (Manual/Automático) do bloco de controle PID. Sempre que esta entrada estiver energizada o controle estará sendo executado em modo automático (veja no item "Descrição" as implicações do modo manual e automático).

Saída	Função	Estado	Descrição
B1	Ativação / Falha	Inativo	O bloco de controle não estará sendo executado ou está em falha.
		Ativo	O bloco de controle estará sendo executado.

Fonte: [21].

Na Tabela 6, é apresentado o comportamento da saída B1 do bloco PID, em relação às entradas A1 e A2. O estado X é irrelevante.

Tabela 6: Relação da saída B1 com as entradas A1 e A2, do bloco PID.

A1	A2	Ganhos	Saída B1
Inativo	X	X	Inativo
Ativo	Manual	X	Inativo
Ativo	Automático	Valores Inválidos	Inativo
Ativo	Automático	Valores válidos	Ativo

Fonte: [21].

O estado da entrada do bloco (A1) será refletido na saída (B1) sem qualquer alteração. Se a entrada deste bloco estiver energizada, a malha de controle estará operando em modo AUTOMÁTICO e em modo MANUAL caso contrário. No modo automático o bloco de controle estará atuando diretamente na variável de saída da malha PID, enquanto no modo manual o último valor da saída será mantido constante. O bloco de controle PID para ser utilizado precisa de vários parâmetros que devem ser fornecidos pelo usuário. Para ter acesso aos parâmetros de controle do bloco PID é necessário posicionar o cursor do editor em cima do bloco PID e teclar-se "CTRL+T". Através deste comando, têm-se acesso à janela de "Configuração dos Blocos de Controle PID", Figura 90. A especificação dos parâmetros deve ser feita para todos os blocos inseridos no programa do usuário.



Figura 90: Tela de configuração de Blocos de controle PID.

Fonte: [21].

6 SINTONIA DAS MALHAS DE CONTROLE

Neste capítulo será realizada a etapa de identificação dos modelos matemáticos dos sistemas a serem controlados. No capítulo 5, mas especificamente na seção 5.1, foi apresentado e definido as malhas de controle que serão utilizadas neste trabalho. Apesar da possibilidade de se obter os modelos matemáticos dos sistemas dinâmicos a partir de princípios físicos, optou-se por obtê-los pela medição dos sinais de entrada e saída dos sistemas dinâmicos. Após esta etapa de identificação será definida uma técnica de controle e sintonia do controlador.

6.1 IDENTIFICAÇÃO DE SISTEMAS

Para a tarefa de identificação de sistemas, será utilizada a ferramenta *System Identification Toolbox* do *software* MATLAB, com esta ferramenta é possível construir modelos usando sinais de entrada e saída no domínio do tempo, frequência, reposta de dados, sinais de séries temporais e espectros de séries temporais. Neste trabalho serão utilizados os sinais de entrada e saída no domínio do tempo.

6.1.1 IDENTIFICAÇÃO DO SISTEMA DE CONTROLE DE VELOCIDADE

Após estabelecer todas as conexões necessárias entra a Planta, CLP e o PC, para a tarefa de identificação de sistemas, foi realizada a coleta dos dados da dinâmica do sistema de controle de velocidade. Para tal, foi preciso inicialmente realizar alguns pequenos ajustes na planta, afinal para obtenção de um bom modelo do sistema é preciso obter bons dados do comportamento do sistema.

Os ajustes foram os seguintes:

- Centralização e fixação do *encoder* ao eixo do motor CC. Foi observado que o posicionamento correto do *encoder* ao eixo do motor interferia na qualidade do sinal fornecido pelo mesmo, com este ajuste foi possível diminuir alguns sinais de interferência identificados nos primeiros experimentos;

- Lubrificação dos enrolamentos e limpeza da correia da esteira, foi identificado que durante o funcionamento da esteira esta produzia um som característico de falta de lubrificação e obstrução de seu movimento. Esta resistência ao movimento apresentou-se mais evidente devido o fato da tensão aplicada ao motor CC ser aproximadamente 13% da sua tensão nominal, no caso 180 V DC, com isso seu rendimento e torque foram reduzidos, após a lubrificação foi observado uma diminuição dos ruídos capturados nas medições de velocidade do motor;
- Sincronização das taxas de amostragem, para que não haja perda de dados no processo de aquisição dos mesmos, a sincronia das taxas de amostragem entre os softwares e hardwares que o compõe é fundamental.

Após a realização destes pequenos ajustes, utilizando o Simulink conectado ao CLP via OPC, foram coletados os dados: de velocidade escalar do motor CC (consequentemente da esteira) e do valor digital (0 - 4095) fornecido ao CLP, o qual fornece um valor de tensão (0 - 10V) na sua saída analógica correspondente. Este valor fornecido ao CLP foi endereçado a uma variável específica em seu programa utilizando a mesma identificação no ambiente Simulink utilizado, Figura 91. Os dados foram plotados no gráfico como apresentado na Figura 92.

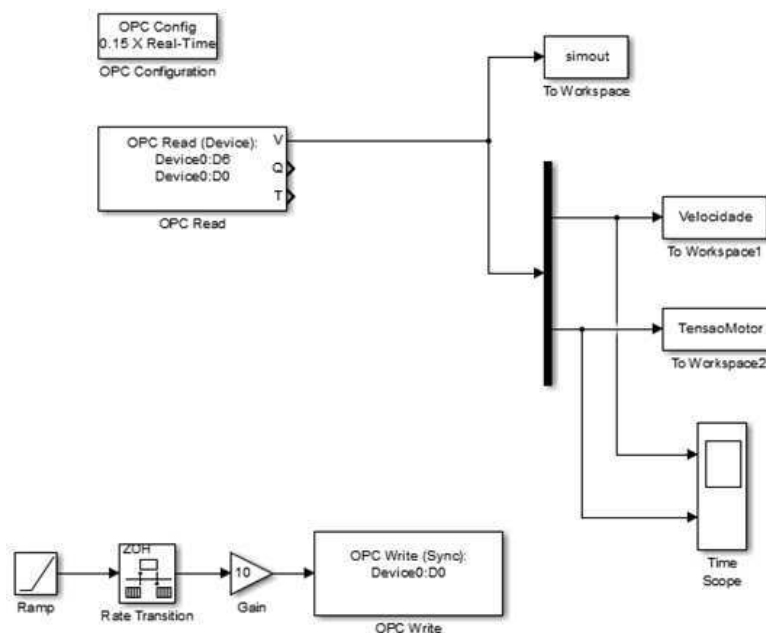


Figura 91: Ambiente de simulação Simulink.

Fonte: [23].

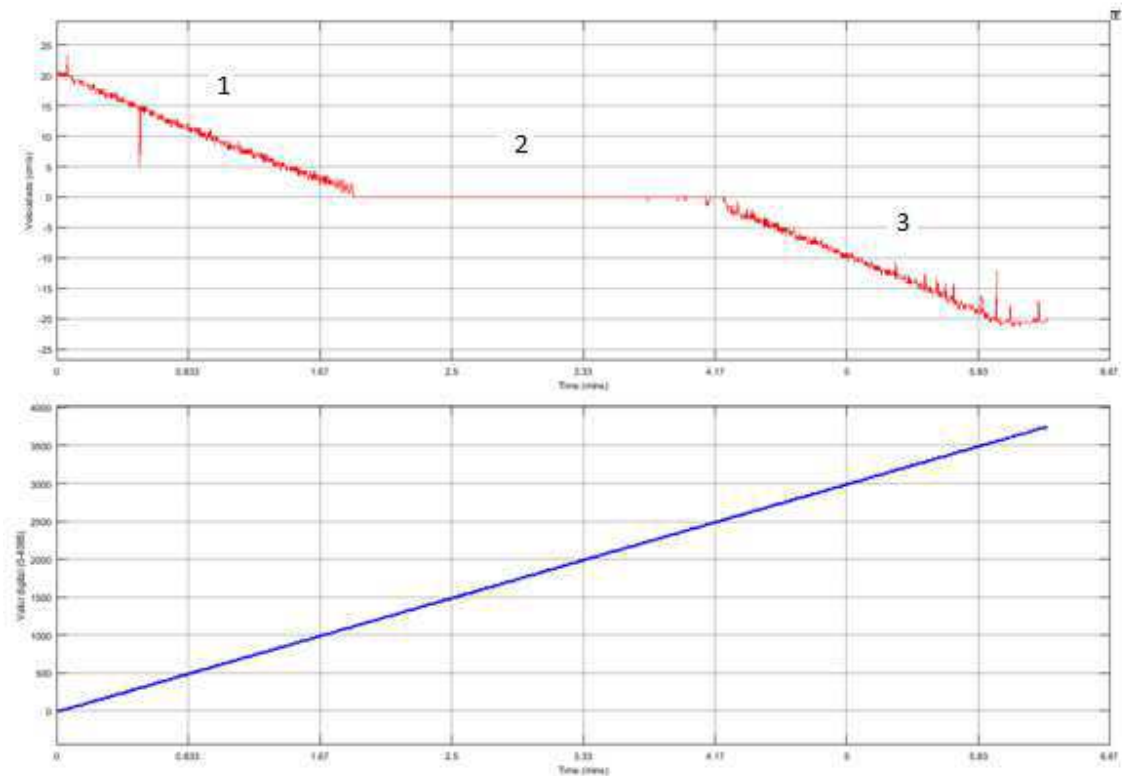


Figura 92: Gráfico dos dados coletados para identificação do sistema.
Fonte: [23].

Analisando o gráfico apresentado na Figura 92, do sistema da malha aberta de controle de velocidade, pode-se observar que o sistema pode ser representado por um modelo não linear. Para sistemas de controle sistemas não lineares não são facilmente controlados, sendo o desejado, sistemas que representem modelos lineares. Ainda analisando este gráfico, pode-se observar que este modelo não linear pode ser representado por dois modelos lineares descartando a área 2 do gráfico, para os objetivos de controle neste trabalho a área 2 pode ser descartada, pois a mesma representa a "zona morta", o instante em que a esteira se encontra parada, sendo assim pode-se realizar o controle de velocidade usando a área 1 e a área 3 do gráfico. Onde a área 1 representa a velocidade no sentido anti-horário (valor positivo de velocidade) e área 2 representa a velocidade no sentido horário (valor negativo de velocidade).

Realizando novamente o experimento de identificação, para melhor representar as áreas 1 e 2 do gráfico apresentado na Figura 92, e utilizando como sinal de controle valores de sinal PWM de 0 a 100% para o controle da velocidade no sentido anti-horário e valores de sinal PWM de 0 a -100% para o controle da velocidade no sentido horário, foram obtidos os seguintes gráficos ilustrados nas figuras: Figura 93 para sentido anti-horário e Figura 94 para o sentido horário.

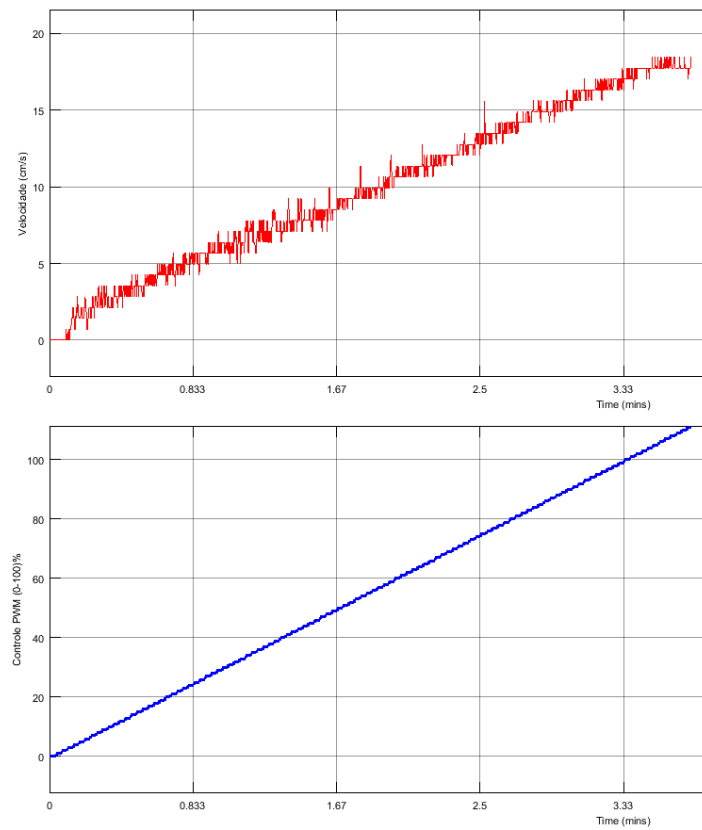


Figura 93: Sinais PWM para controle da velocidade no sentido anti-horário.
Fonte: [23].

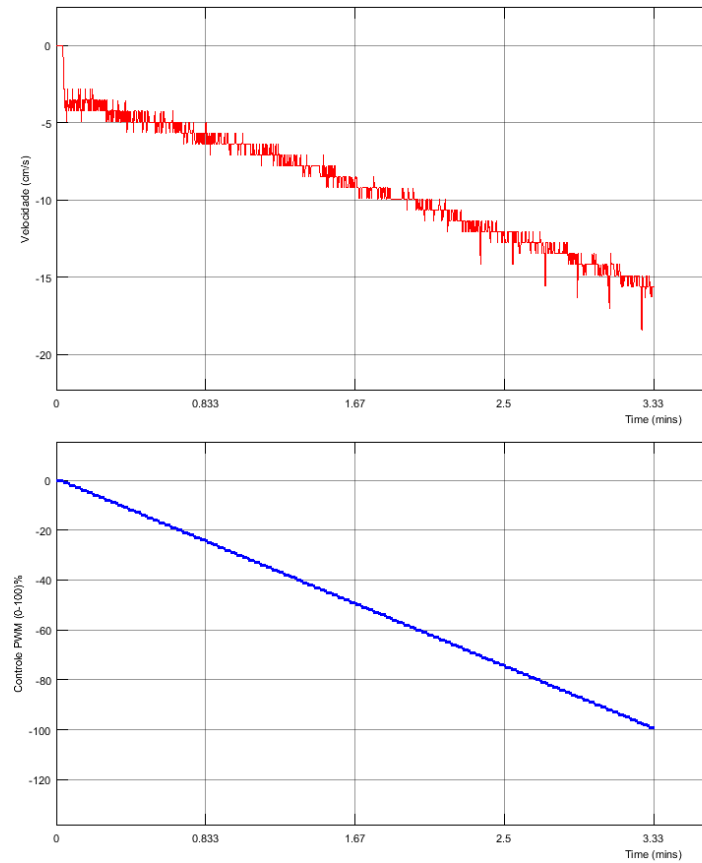


Figura 94: Sinais PWM para controle da velocidade do sentido horário.
Fonte: [23].

Observando os gráficos apresentados nas figuras 93 e 94, pode-se verificar que os sistemas observados podem ser considerados sistemas lineares.

Para melhor estimar o modelo matemático do sistema de controle de velocidade, será aplicado um sinal degrau com valores na faixa de operação de velocidade que se deseja trabalhar.

Na Figura 95, tem-se o gráfico da resposta do sistema, em velocidade (cm/s), a entrada degrau, com valor inicial sinal PWM de 20% e valor final com sinal PWM de 50% para sistema de controle de velocidade no sentido anti-horário.

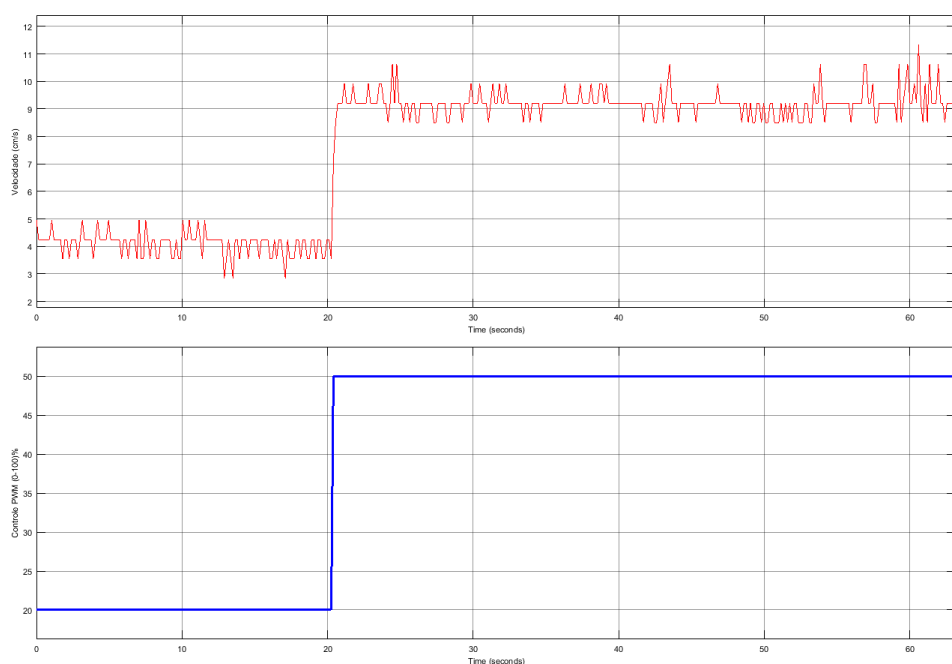


Figura 95: Resposta do sistema, de controle de velocidade do sentido anti-horário, a entrada degrau.
Fonte: [23].

Na Figura 96, tem-se o gráfico da resposta do sistema, em velocidade (cm/s), a entrada degrau, com valor inicial sinal PWM de -20% e valor final com sinal PWM de -60% para sistema de controle de velocidade no sentido horário.

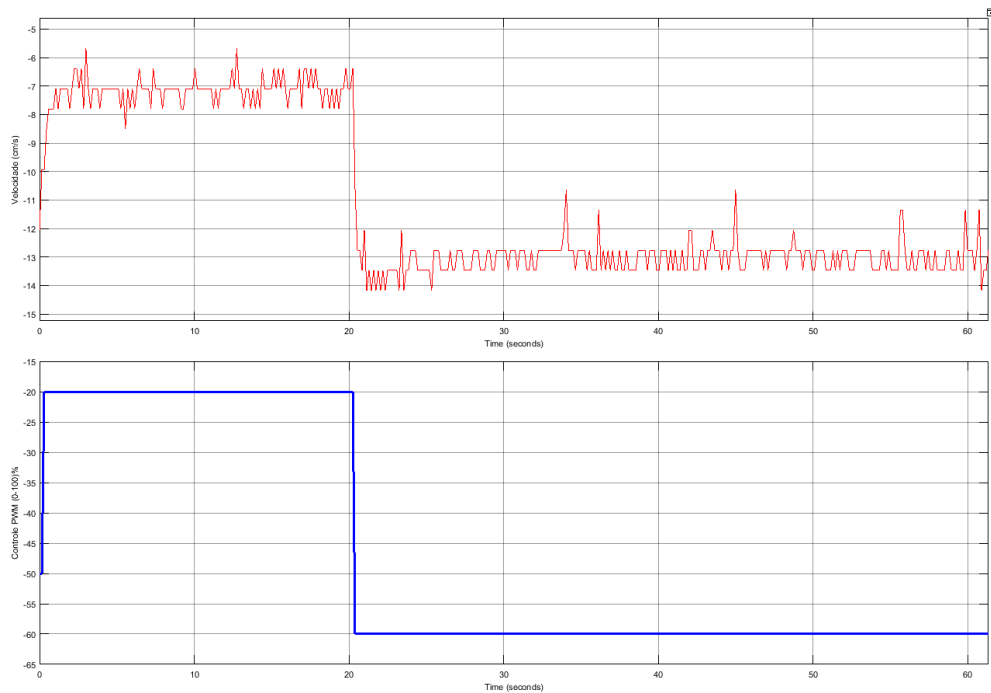


Figura 96: Resposta do sistema, de controle de velocidade do sentido horário, a entrada degrau.
Fonte: [23].

Em seguida os dados que contém os valores de entrada e saída destes sistemas serão importados da área de trabalho do MATLAB para software de identificação de sistemas, *System Identification Toolbox*.

Inicialmente será feito o tratamento dos dados, para o sistema do controle da velocidade no sentido anti-horário. No ambiente da ferramenta, *System Identification Toolbox*, os dados serão importados como dados no domínio do tempo e será ajustado o valor da taxa de amostragem para o valor utilizado durante todo o experimento.

Em seguida será realizado um pré-processamento destes dados na própria ferramenta de identificação de sistemas, este pré-processamento é importante para obter uma melhor aproximação do modelo matemático em relação ao modelo real (medido).

No ambiente de trabalho da ferramenta *System Identification Toolbox*, os modelos lineares que foram estimados para o sistema de controle de velocidade no sentido anti-horário em malha aberta, estão ilustrados na Figura 97, e os resultados da comparação dos melhores modelos, comparados ao modelo real, estão ilustrados na Figura 98.

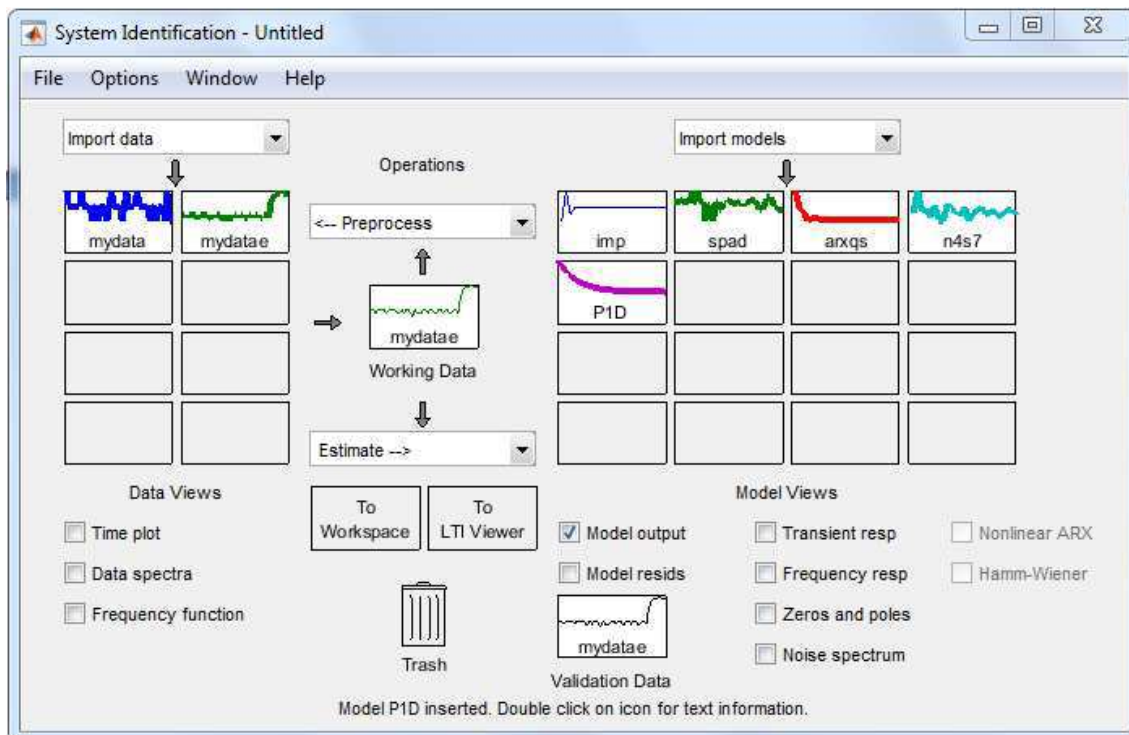


Figura 97: Ambiente do *System Identification Toolbox*, com os modelos estimados, para velocidade no sentido anti-horário.

Fonte: [24].

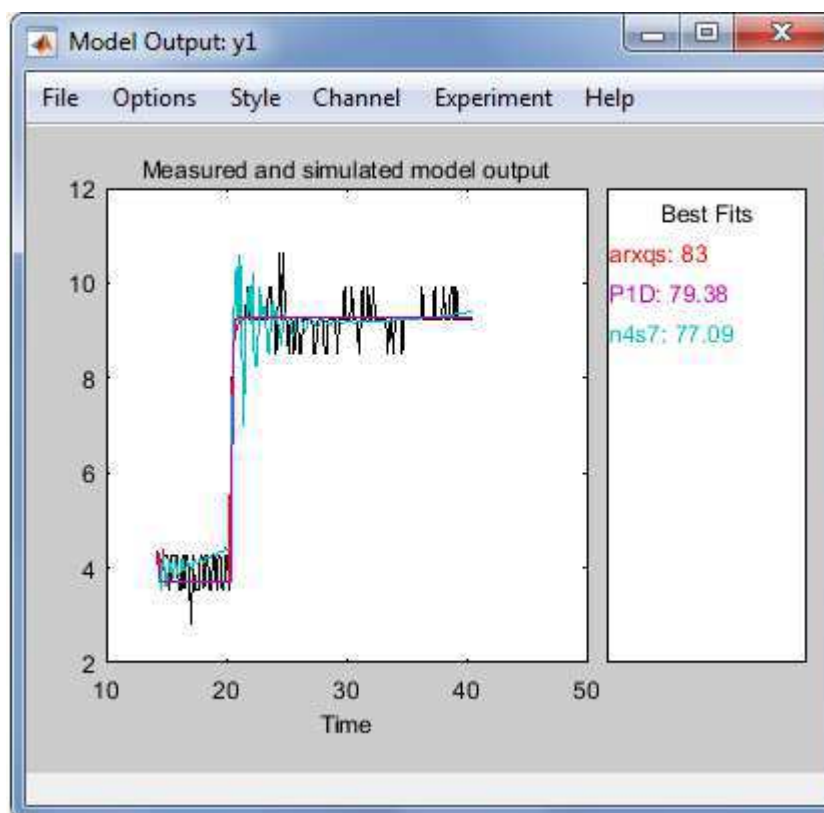


Figura 98: Modelos com os melhores ajustes comparados com o modelo real para a velocidade no sentido anti-horário.

Fonte: [24].

O gráfico ilustrado na Figura 98, apresenta os melhores modelos estimados, pode-se observar neste gráfico que os modelos possuem aproximações semelhantes para o modelo real, no entanto o modelo que possui o melhor ajuste é o modelo arxqs, este modelo usa o algoritmo arx auto-regressivo.

O modelo matemático arxqs (de quarta ordem) estimado para o sistema em malha aberta pode ser representado pelas seguintes equações:

$$A(z)y(t) = B(z)u(t) + e(t)$$

$$A(z) = 1 - 0.1552 z^{-1} + 0.02051 z^{-2} - 0.1437 z^{-3} + 0.009633 z^{-4}$$

$$B(z) = 0.1112 + 0.02867 z^{-1} + 0.02212 z^{-2} - 0.02635 z^{-3}$$

Como se pode observar na Figura 98, o modelo arxqs possui uma aproximação de 83% do modelo real.

Concluída esta etapa de identificação do sistema de controle de velocidade no sentido anti-horário, será realizada a identificação do sistema de controle de velocidade no sentido horário de forma semelhante.

Na Figura 99, tem-se a ilustração do ambiente da ferramenta *System Identification Toolbox*, com os modelos lineares que foram estimados para o sistema de controle de velocidade no sentido horário em malha aberta.

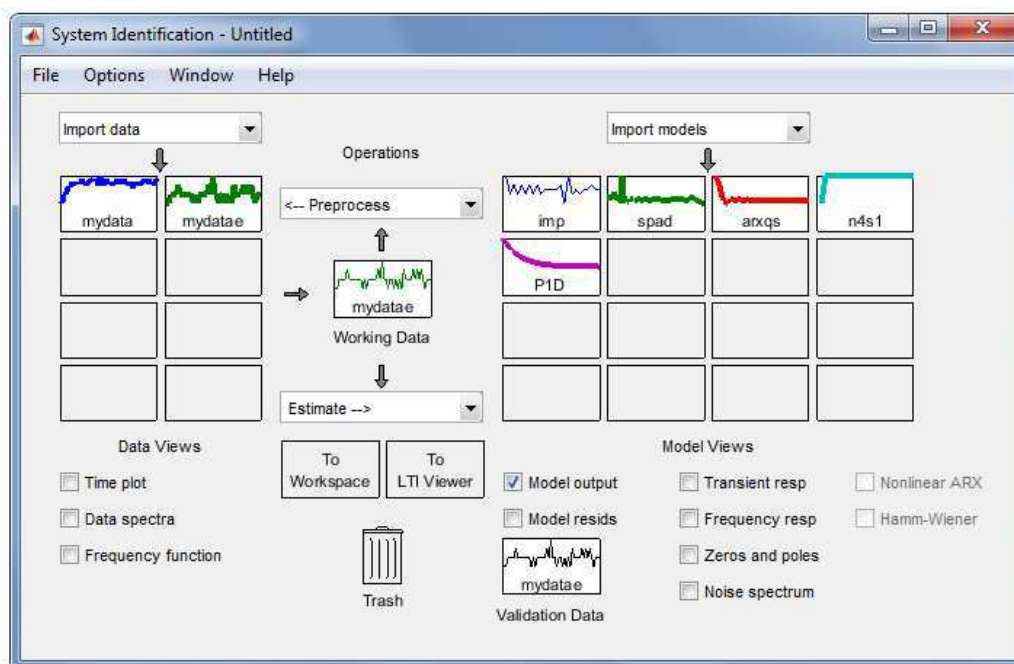


Figura 99: Ambiente do *System Identification* com os modelos estimados, para velocidade no sentido horário.

Fonte: [24].

Os resultados da comparação dos melhores modelos estimados estão ilustrados na Figura 100.

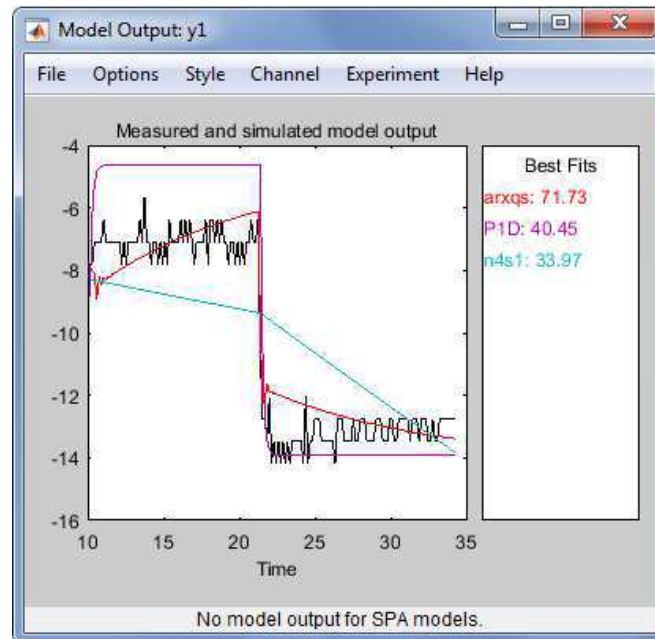


Figura 100: Comparação dos melhores modelos estimados para a velocidade no sentido horário.
Fonte: [24].

O gráfico ilustrado na Figura 100, apresenta os melhores modelos estimados, pode-se observar neste gráfico que os modelos possuem aproximações semelhantes para o modelo real, no entanto o modelo que possui o melhor ajuste é o modelo arxqs, este modelo usa o algoritmo arx auto-regressivo.

O modelo matemático arxqs (de quarta ordem) estimado para o sistema em malha aberta pode ser representado pelas seguintes equações:

$$A(z)y(t) = B(z)u(t) + e(t)$$

$$A(z) = 1 - 0.2748 z^{-1} - 0.5457 z^{-2} - 0.0734 z^{-3} + 0.07923 z^{-4}$$

$$B(z) = 0.0965 + 0.03222 z^{-1} - 0.07135 z^{-2} - 0.05106 z^{-3}$$

Como se pode observar na Figura 100, o modelo arxqs possui uma aproximação de 71.73% do modelo real.

6.1.2 IDENTIFICAÇÃO DO SISTEMA DE CONTROLE DE POSIÇÃO

O controle de posição será realizado por um controlador ON/OFF, as posições que serão identificadas são as posições que serão identificadas correspondem as

posições onde estão instalados os sensores de barreira ótica. A identificação desse sistema corresponde simplesmente a resposta desses sensores de sinal ON/OFF, quando um objeto estiver entre sua barreira ótica.

6.1.3 IDENTIFICAÇÃO DO SISTEMA DO MÓDULO DE AQUECIMENTO

Para a identificação do sistema que representa a dinâmica do módulo de aquecimento as etapas seguidas foram semelhantes às seguidas na identificação do sistema de controle de velocidade. Utilizando a estrutura de diagrama de blocos construída no ambiente de simulação da ferramenta Simulink, Figura 101, foi realizada identificação do sistema do módulo de aquecimento.

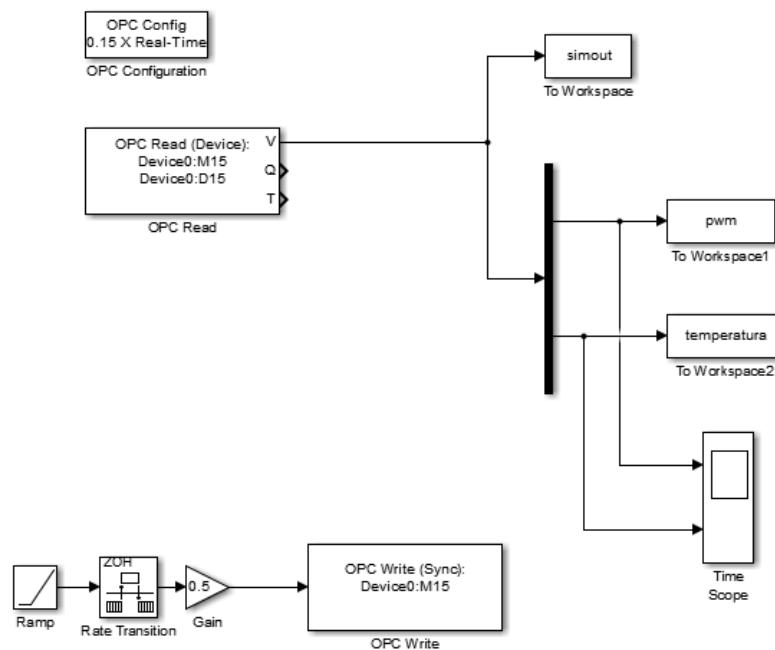


Figura 101: Ambiente de simulação no Simulink para identificação do módulo de aquecimento.
Fonte: [23].

Devido à dinâmica esperada para os sistemas térmicos ser bastante lenta, em relação aos sistemas elétricos, por exemplo, foram tomados alguns cuidados para a realização dos testes de identificação deste tipo sistema. Se estes cuidados experimentais não fossem observados possivelmente o sistema não seria identificado de forma correta. Primeiramente considera-se que o módulo de aquecimento (sistema a ser identificado) possui uma eficiência térmica não conhecida e o ambiente onde o módulo de aquecimento está instalado possui um aparelho de ar condicionado controlando a

temperatura da sala. Essas considerações são importantes para que se possa compreender melhor a resposta do sistema. Para atenuar estes problemas, que podem ser considerados, por exemplo, perturbações para o sistema, foram tomadas as seguintes medidas: para o valor de temperatura da sala ajustado para 24°C de acordo com o valor informado pelo ar condicionado foi aplicado no sistema uma entrada com valor PWM de 20% constante, observou-se que a temperatura do sistema equivalente a esta entrada convergia em malha aberta para um valor próximo de 31°C. Logo em seguida foi aplicada uma entrada degrau ao sistema com valor inicial PWM de 20% e final de 30%. O gráfico da resposta do sistema em malha aberta para essa entrada a degrau está ilustrado na Figura 102.

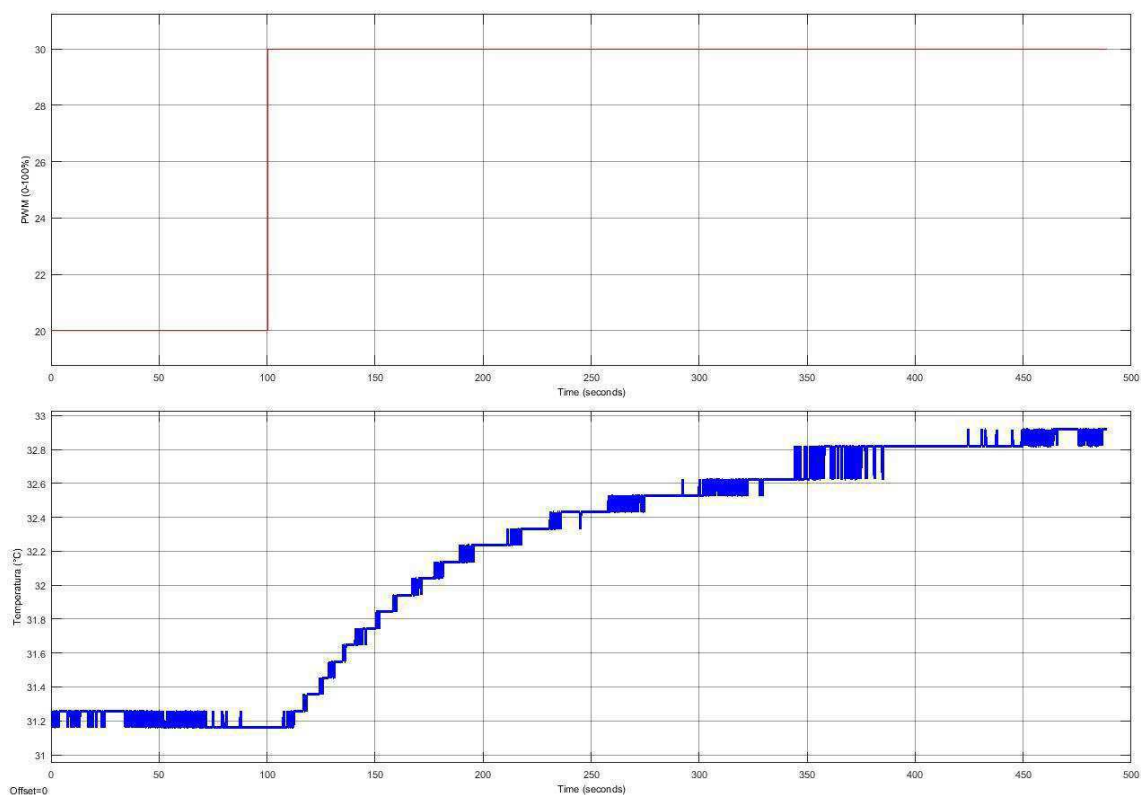


Figura 102: Gráfico da resposta do sistema em malha aberta à entrada degrau.
Fonte: [23].

Os dados com a resposta do sistema em malha aberta a entrada degrau, serão importados da área de trabalho do MATLAB para software de identificação de sistemas *System Identification Toolbox*.

No ambiente de trabalho da ferramenta *System Identification Toolbox*, os modelos lineares que foram estimados para o sistema de controle de temperatura do

módulo de aquecimento em malha aberta, estão ilustrados na Figura 103, e os resultados dos melhores modelos comparados ao modelo real, estão ilustrados na Figura 104.

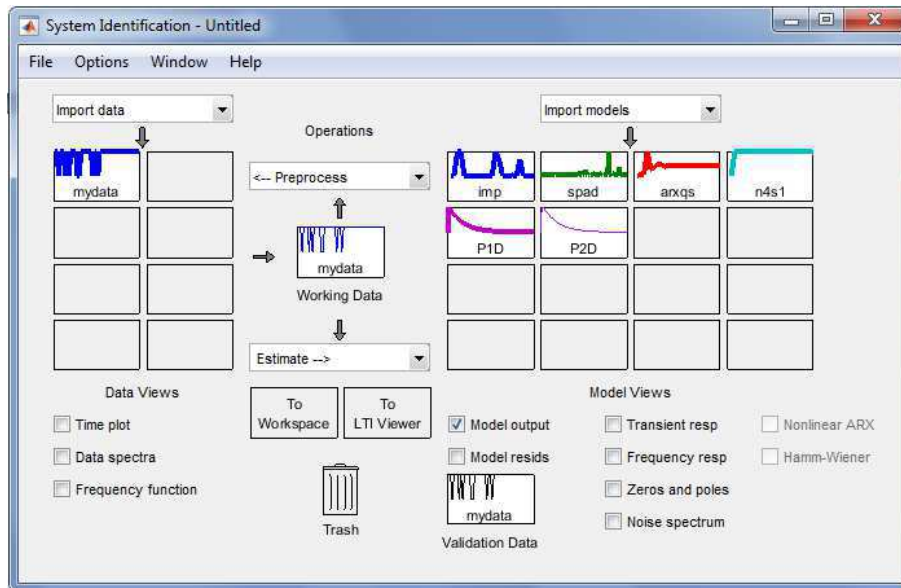


Figura 103: Ambiente do *System Identification Toolbox* com os modelos estimados, para o módulo de aquecimento.
Fonte: [24].

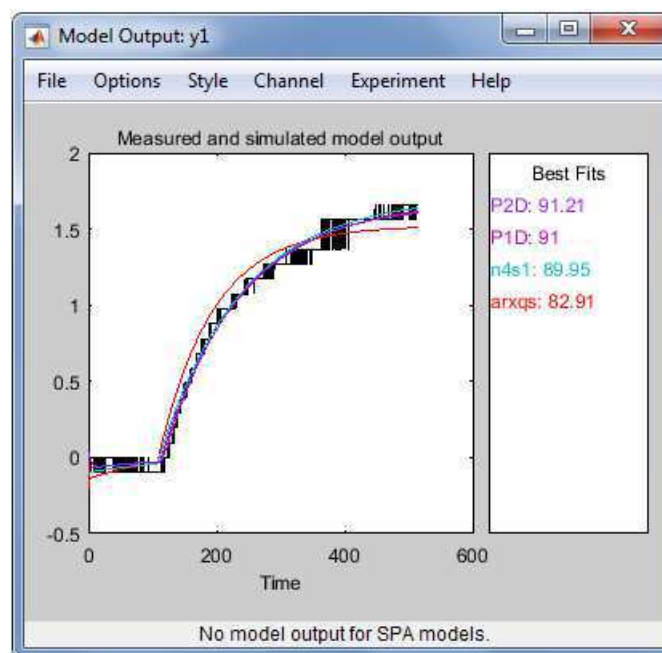


Figura 104: Comparação dos melhores modelos estimados para o módulo de aquecimento.
Fonte: [24].

Na Figura 104, tem-se os melhores modelos estimados, no entanto o modelo que possui o melhor ajuste é o modelo P2D. O modelo matemático P1D (modelo de

processo com função de transferência) estimado para o sistema em malha aberta pode ser representado pela seguinte função de transferência com seus respectivos parâmetros:

$$G(s) = \frac{Kp}{(1 + Tp1 * s)(1 + Tp2 * s)} * \exp(-Td * s)$$

$$Kp = 0.16614$$

$$Tp1 = 116.88$$

$$Tp2 = 4.0007$$

$$Td = 4.5$$

Como se pode observar na Figura 104, o modelo PID possui uma aproximação de 91.21% do modelo real.

6.1.4 IDENTIFICAÇÃO DO SISTEMA DO MÓDULO DE RESFRIAMENTO

Desejava-se para este módulo aplicar um controle PID ou PI, no entanto observaram-se alguns problemas transitórios na identificação do sistema que o representa. Alguns desses problemas são: as ventoinhas não eram acionadas no mesmo instante, possivelmente devido a características de funcionamentos não uniforme a tensões abaixo da sua tensão nominal e um fator de qualidade de rendimento inferior para este caso, outro problema era que o túnel de vento formado por essas ventoinhas possuía uma temperatura proporcional a temperatura ambiente a qual não possuía sempre os mesmos valores, influenciando diretamente nos resultados obtidos. Apesar destes problemas, foram realizados experimentos para se observar a dinâmica do sistema em questão, estes experimentos foram realizados com o ar condicionado da sala ajustado para a temperatura de 22°C.

Na Figura 105, tem-se o gráfico do experimento que auxiliou na identificação dos problemas citados, os testes foram realizados utilizando uma configuração de blocos no ambiente da ferramenta Simulink, semelhante ao ilustrado na Figura 91.

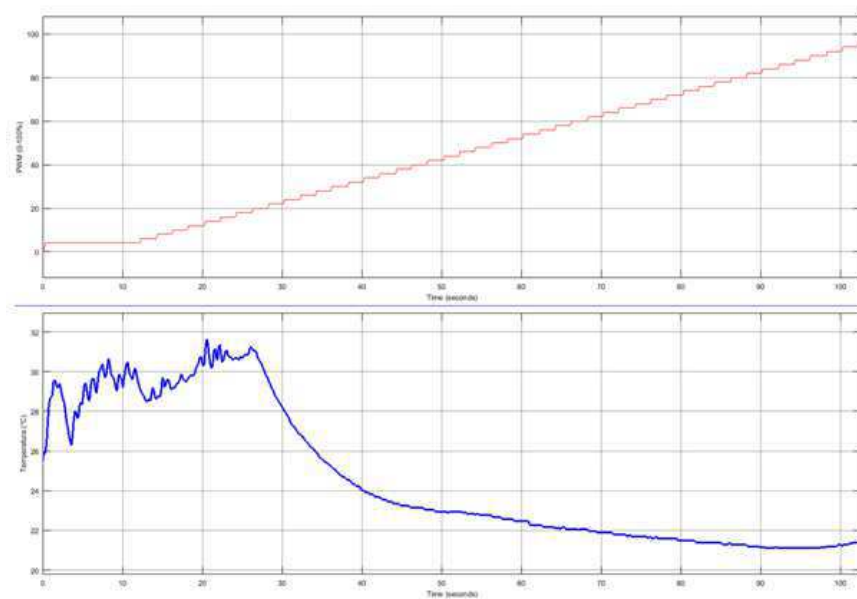


Figura 105: Gráfico do experimento para Identificação do sistema que representa a dinâmica do módulo de resfriamento.
Fonte: [23].

Neste experimento foi utilizada uma entrada tipo rampa, aplicada a variável no programa do CLP que corresponde ao valor do *duty cycle* do PWM, e como variável de saída o valor da temperatura em graus Celsius obtidos pelo sensor de temperatura do módulo de resfriamento. Em seguida foi realizado outro experimento, Figura 106, utilizando como entrada um sinal degrau na faixa de 0 a 100% do sinal PWM.

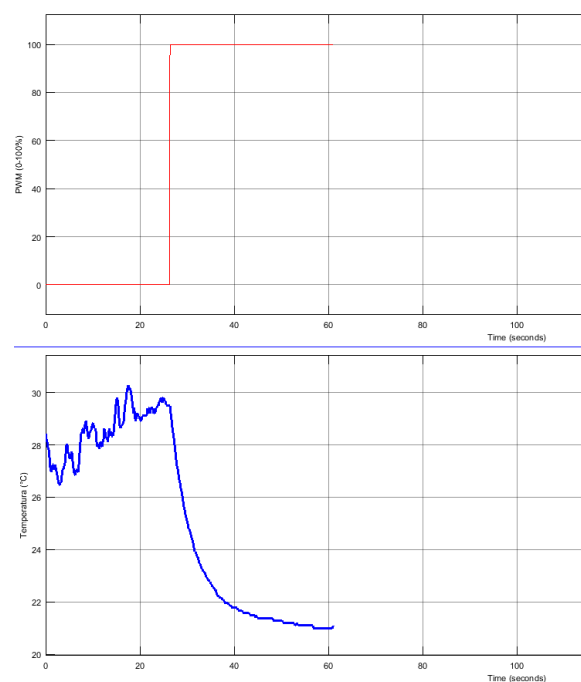


Figura 106: Experimento de identificação realizado com entrada degrau, para o módulo de resfriamento.
Fonte: [23].

Apenas a título de aprendizagem, considerou-se o sistema identificado no experimento da entrada a degrau, um sistema do tipo linear em seguida foi-se realizado a identificação dos melhores modelos para este sistema. A identificação destes modelos foi realizada utilizando a ferramenta *System Identification Toolbox*, e os modelos encontrados estão representados na Figura 107.

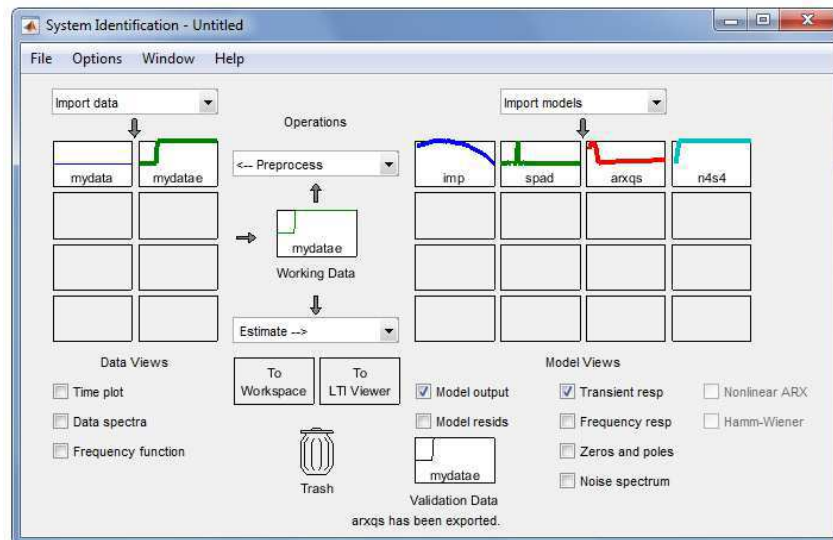


Figura 107: Ambiente do *System Identification Toolbox* com os modelos estimados, para o módulo de resfriamento.

Fonte: [24].

Os resultados da comparação dos melhores modelos estimados estão ilustrados na Figura 108.

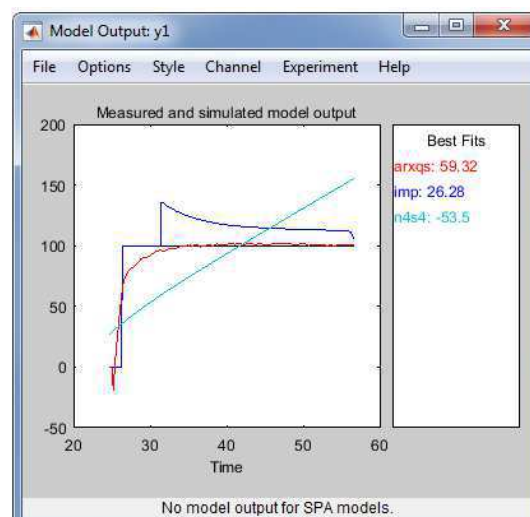


Figura 108: Resultado dos melhores modelos estimados para o módulo de resfriamento.

Fonte: [24].

Como esperado os resultados de correlação não foram satisfatórios para os modelos estimados, por questões práticas e como observado experimentalmente, foi decidido para o módulo de resfriamento, que será utilizado um controle *ON/OFF*, o qual atende satisfatoriamente as necessidades do projeto.

O controlador *ON/OFF* foi implementado no programa Ladder do CLP, este controle funciona da seguinte forma se a temperatura a temperatura identificada pelo sensor estiver maior que 26°C as ventoinhas do módulo de refrigeração são ligadas e permanecem ligadas até a temperatura for menor ou igual a 25°C, o gráfico resposta do sistema de malha fechada do controle *ON/OFF* está ilustrado na Figura 109.

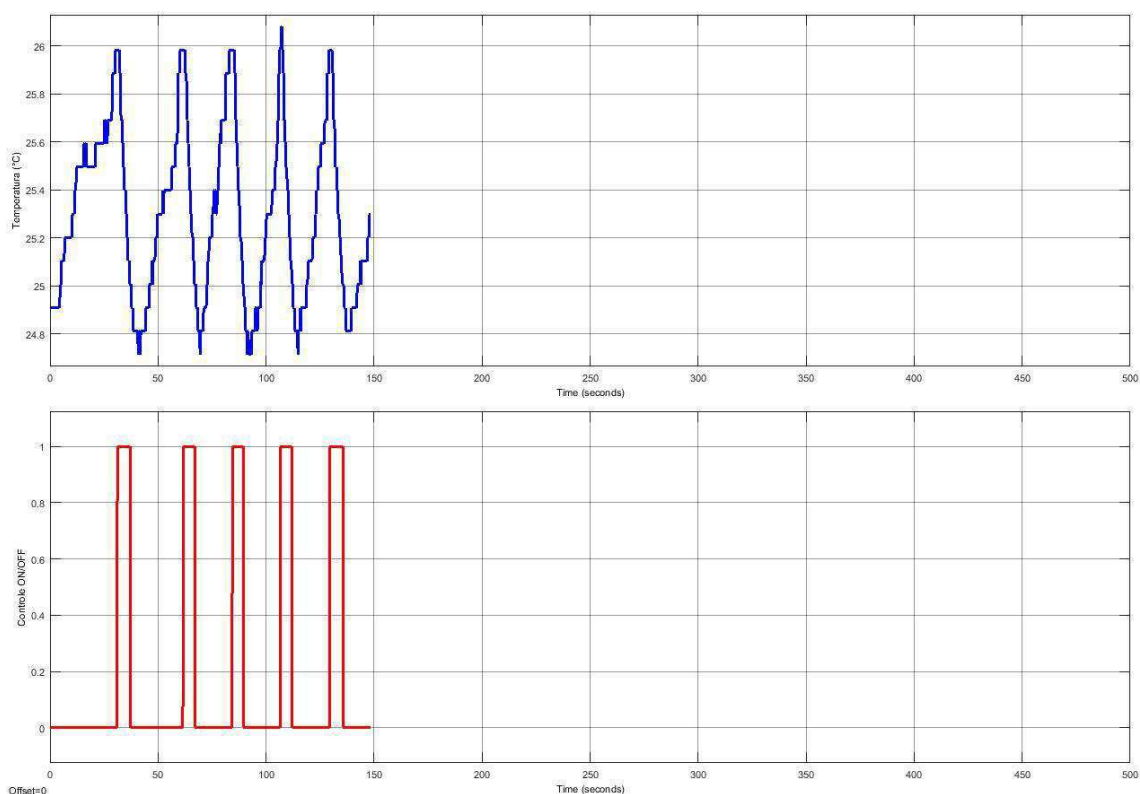


Figura 109: Gráfico da resposta ao sistema de malha fechada do módulo de resfriamento.

Fonte: [23].

6.2 SINTONIA DOS CONTROLADORES

Realizado a identificação do sistema, e obtido o modelo matemático que irá representá-lo, será realizada a sintonia do controlador obtido, para esta tarefa, o modelo que foi obtido na ferramenta *System Identification Toolbox* será exportado para a área de trabalho do MATLAB e em seguida será importado para o ambiente da ferramenta *PID Tuner* também do MATLAB.

6.2.1 SINTONIA DOS CONTROLADORES DE VELOCIDADE

Utilizando a interface gráfica da ferramenta *PID Tuner* é possível obter uma sintonia para um controlador mais rápido ou mais lento, assim como mais robusto ou mais agressivo, neste trabalho o controlador PI que foi utilizado para o controle de velocidade em ambos os sentidos.

A sintonia para o controle de velocidade no sentido anti-horário está de acordo com a apresentada na Figura 110, com os parâmetros do controlador apresentados na Figura 111. Nesta sintonia procurou-se obter um controlador estável mais rápido e mais robusto.

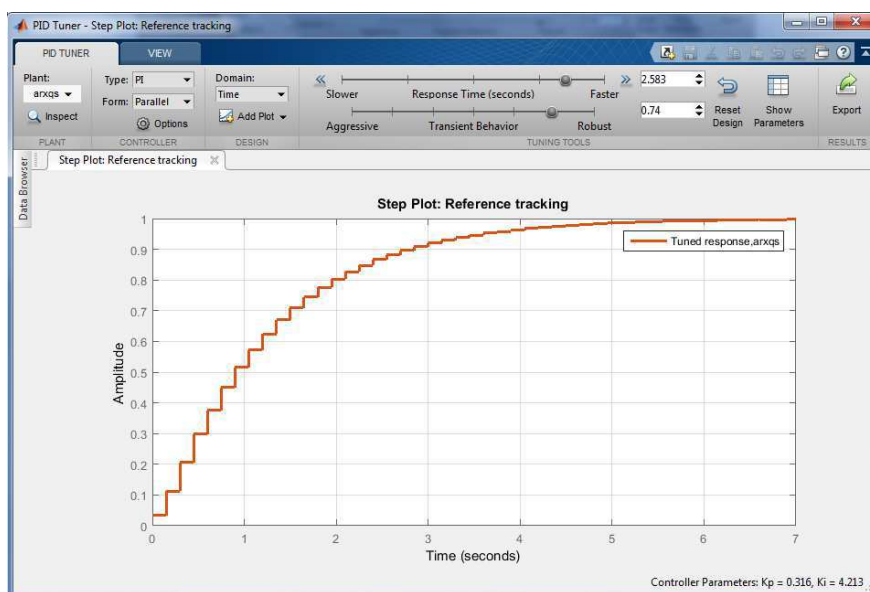


Figura 110: Sintonia para o controlador de velocidade do sentido anti-horário.

Fonte: [25].

Controller Parameters	
	Tuned
Kp	0.31599
Ki	4.2132
Kd	
Tf	

Performance and Robustness	
	Tuned
Rise time	2.55 seconds
Settling time	4.65 seconds
Overshoot	0 %
Peak	1
Gain margin	Inf dB @ NaN rad/s
Phase margin	85.8 deg @ 0.774 rad/s
Closed-loop stability	Stable

Figura 111: Parâmetros do controlador PI da sintonia para velocidade no sentido anti-horário.

Fonte: [25].

Como se pode observar na Figura 111, nos parâmetros do controlador após a sintonia, o tempo de subida corresponde a 2.55 segundos, o tempo de estabilização há 4.65 segundos e não possui *overshoot*.

Utilizando o ambiente do Simulink foram implementados o controle em malha fechada da velocidade no sentido anti-horário, os ganhos do controlador e a faixa de operação foram adicionados ao bloco *PID Controller*, o circuito com diagrama de blocos do controle de velocidade em malha fechada, está como ilustrado na Figura 112.

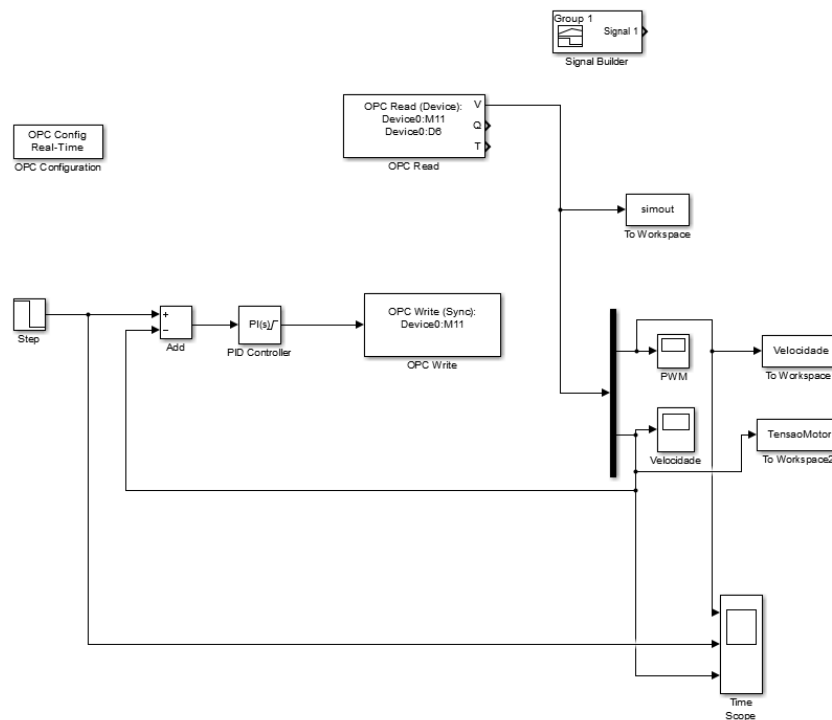


Figura 112: Diagrama de blocos do controle em malha fechada implementado no ambiente Simulink.
Fonte: [23].

Na Figura 113, tem-se a ilustração gráfica do controle de velocidade, em malha fechada no sentido anti-horário, da esteira transportadora, o valor de SP (*Set Point*) utilizado corresponde à velocidade 10 cm/s e está representado pela linha verde do gráfico, velocidade versus tempo, a velocidade controlada, que no instante da entrada do sinal degrau com o valor do SP, possuía um valor próximo de 0 cm/s, indicando que a esteira estava parada, convergiu para o valor de referência, nota-se ainda observando o gráfico da velocidade que o controle de velocidade em malha fechada foi obtido como esperado e no gráfico, valor PWM versus tempo, pode-se observar que a variação do valor PWM, no caso variável MV (*Manipulated Variable*), aplicado ao motor CC é

proporcional a velocidade, como anteriormente ilustrado no gráfico apresentado na Figura 93.

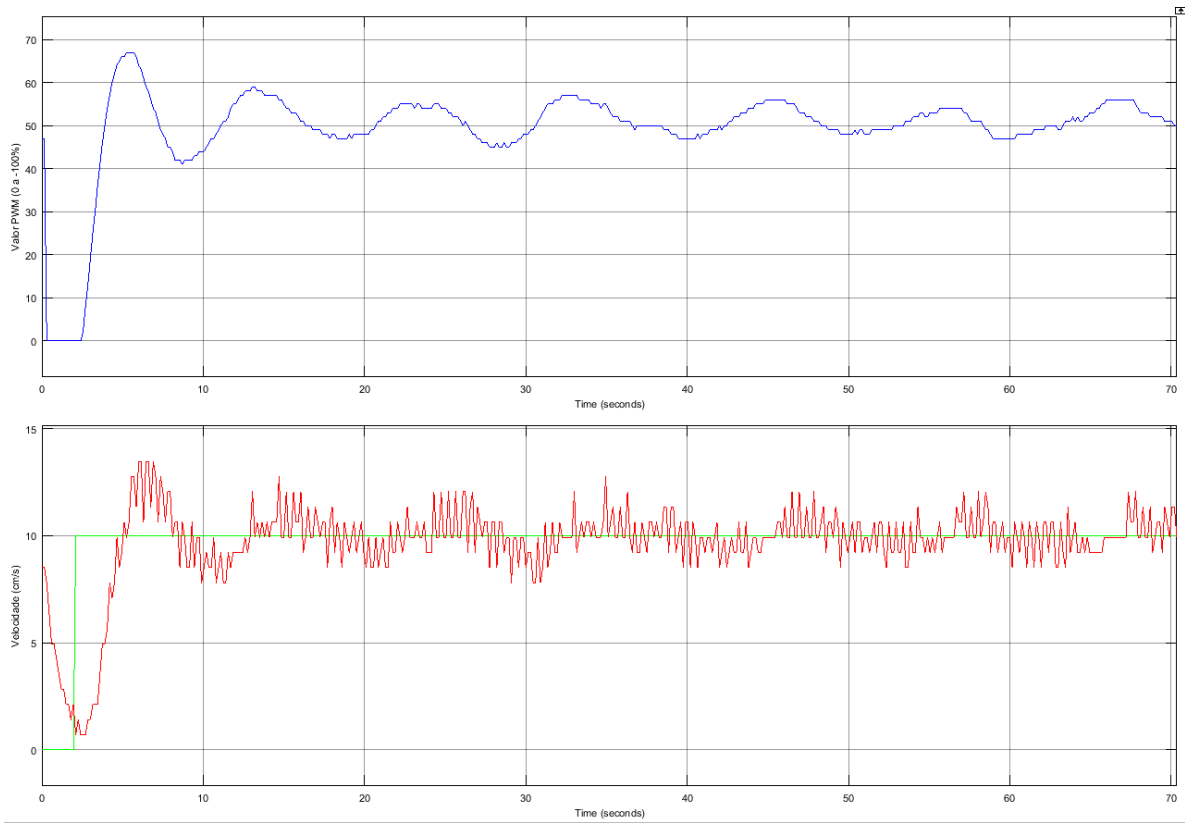


Figura 113: Controle de velocidade em malha fechada no sentido anti-horário da esteira transportadora.
Fonte: [23].

Os mesmos experimentos realizados para sintonizar o controlador para a velocidade no sentido anti-horário foram utilizados para a sintonia para o controle de velocidade no sentido horário. Na Figura 114, está ilustrado no ambiente do software *PID Tuner* a sintonia para este controlador, com os parâmetros do controlador apresentados na Figura 115. De forma semelhante à sintonia do controlador de velocidade no sentido anti-horário, procurou-se obter um controlador estável mais rápido e mais robusto.

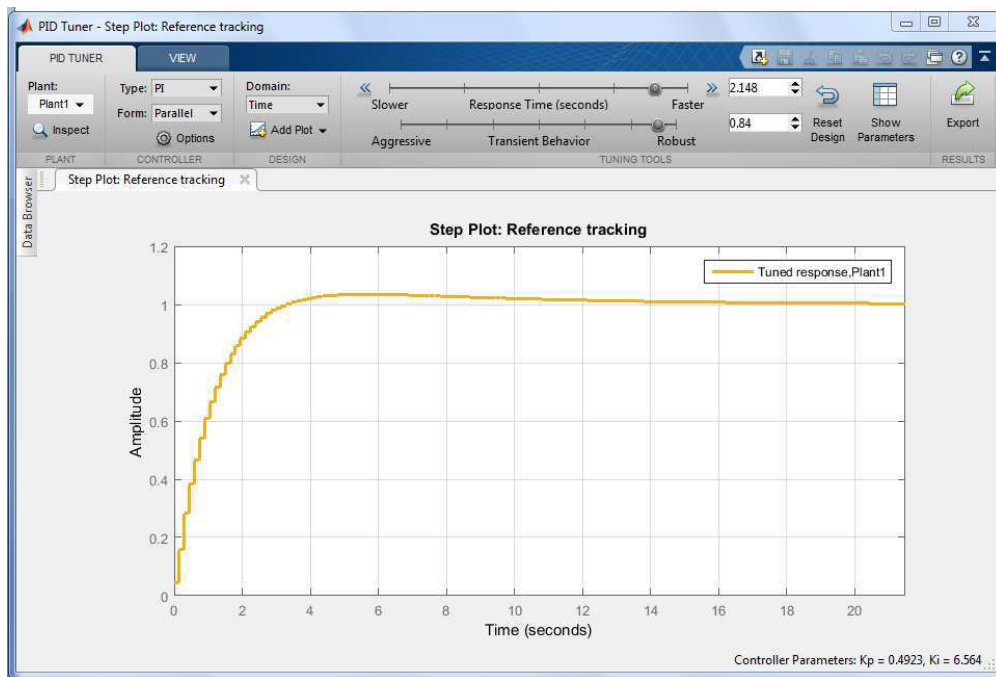


Figura 114: Sintonia para o controlador de velocidade do sentido horário.
Fonte: [25].

Controller Parameters	
	Tuned
K_p	0.49227
K_i	6.5635
K_d	
T_f	

Performance and Robustness	
	Tuned
Rise time	1.95 seconds
Settling time	10.4 seconds
Overshoot	3.46 %
Peak	1.03
Gain margin	Inf dB @ NaN rad/s
Phase margin	84.7 deg @ 0.931 rad/s
Closed-loop stability	Stable

Figura 115: Parâmetros do controlador PI da sintonia para velocidade no sentido horário.
Fonte: [25].

Como se pode observar na Figura 115, nos parâmetros do controlador após a sintonia, o tempo de subida corresponde a 0.0547 segundos, o tempo de estabilização a 0.297 segundos e possui *overshoot* de 6.69%.

O mesmo ambiente do Simulink que foi implementado o controle em malha fechada da velocidade no sentido anti-horário, Figura 112, foi utilizado para a velocidade no sentido horário, com a diferença dos valores limites de saturação

superiores e inferiores ajustados no software para os valores 0 e -100 respectivamente, assim como o valor do SP para uma faixa negativa de velocidade (sentido horário), em seguida os ganhos do controlador e a faixa de operação foram adicionados ao bloco *PID Controller*.

Na Figura 116, tem-se a ilustração gráfica do controle de velocidade, em malha fechada no sentido horário, da esteira transportadora, o valor de SP utilizado corresponde à velocidade -10 cm/s e está representado pela linha verde do gráfico, velocidade versus tempo, a velocidade controlada, que no instante da entrada do sinal degrau com o valo do SP, possuía um valor próximo de 0 cm/s, indicando que a esteira estava parada, convergiu para o valor de referência, no gráfico, valor PWM versus tempo, pode-se observar que a variação do valor PWM, no caso variável MV, aplicado ao motor CC é proporcional a velocidade, como anteriormente ilustrado no gráfico apresentado na Figura 94.

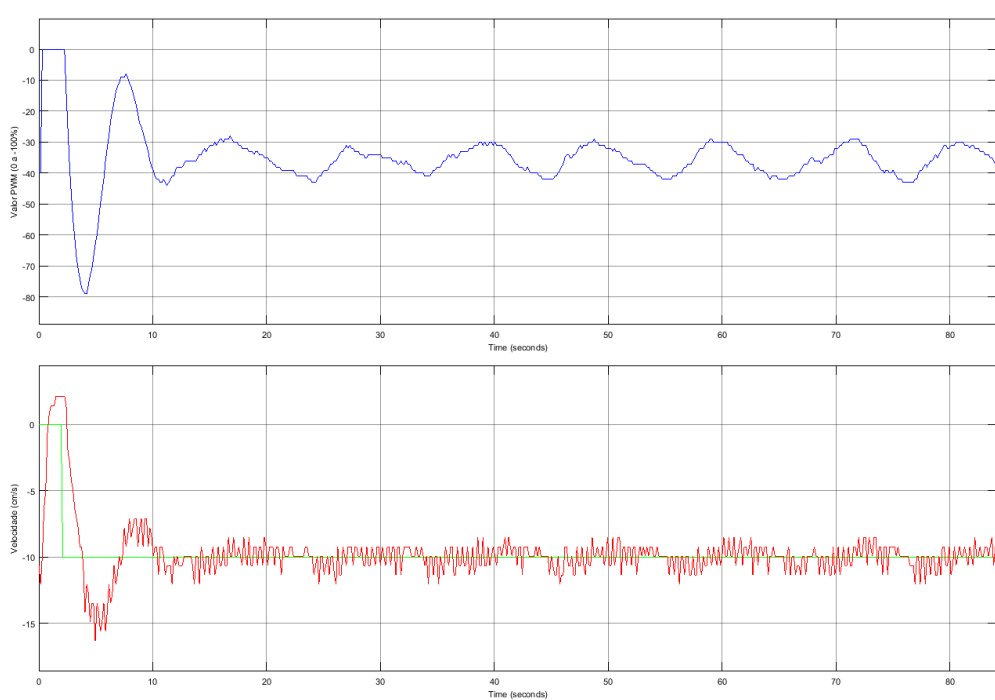


Figura 116: Controle de velocidade em malha fechada no sentido horário da esteira transportadora.
Fonte: [23].

6.2.2 SINTONIA DO CONTROLADOR DO MÓDULO DE AQUECIMENTO

Será utilizado para controle da temperatura do módulo de aquecimento um controlador do tipo PI, devido ao comportamento do sistema observado. Assim como para os outros sistemas identificados neste trabalho, será utilizada a ferramenta PID

Tuner para sintonia do controle de temperatura do módulo de aquecimento. A sintonia para este controle de temperatura está de acordo com a apresentada na Figura 117, com os parâmetros do controlador apresentados na Figura 118. Nesta sintonia procurou-se obter um controlador estável mais lento e mais robusto, atendendo as características do sistema que possui uma dinâmica lenta.

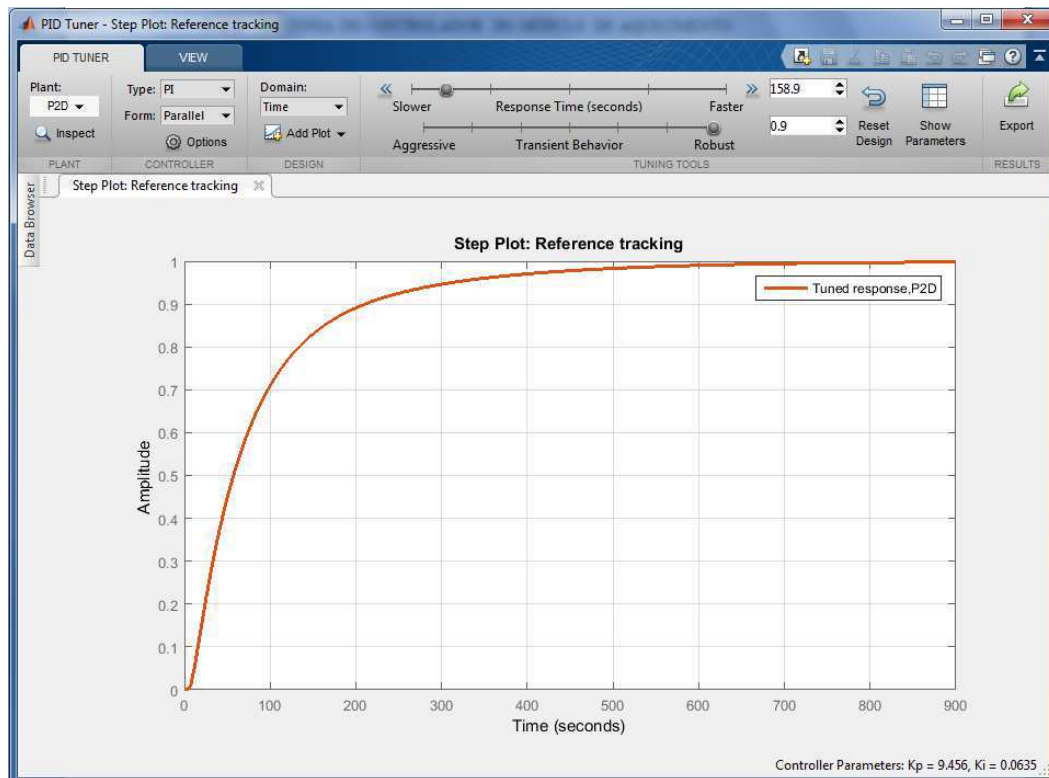


Figura 117: Sintonia para o controle de temperatura do módulo de aquecimento.
Fonte: [25].

Controller Parameters	
	Tuned
K_p	9.456
K_i	0.063499
K_d	
T_f	

Performance and Robustness	
	Tuned
Rise time	195 seconds
Settling time	468 seconds
Overshoot	0 %
Peak	1
Gain margin	25.6 dB @ 0.201 rad/s
Phase margin	90 deg @ 0.0126 rad/s
Closed-loop stability	Stable

Figura 118: Parâmetros do controlador PI da sintonia para a temperatura do módulo de aquecimento.
Fonte: [25].

Como se pode observar na Figura 118, nos parâmetros do controlador após a sintonia, o tempo de subida corresponde a 195 segundos, o tempo de estabilização há 468 segundos e sem *overshoot*, caracterizando um sistema relativamente lento, comparado aos outros sistemas identificados neste trabalho.

Na Figura 119, tem-se o comportamento do sistema de controle em malha fechada em relação ao *Set Point*, como esperado o controle PI convergiu para um valor próximo do valor de *Set Point*, a variação do valor de temperatura sobre o valor de referencia deve-se aos seguintes fatores:

- Ambiente no qual está a esteira com o módulo de aquecimento possui um ar condicionado o qual estava programado para a temperatura de 24 °C, no entanto a temperatura do ambiente oscila bastante devido ao calor dissipado pelas máquinas contidas neste, assim como o movimento das pessoas que trabalham;
- O sistema térmico possui uma dinâmica lenta que impossibilita uma resposta rápida de controle;
- O sensor de temperatura utilizado possui também um atraso no seu tempo de resposta as variações de temperatura;
- A tensão fornecida na saída do sensor quando medida pelo CLP é convertida pelo seu conversor A/D, o valor digital tratado no CLP será a cada operação matemática realizada, incrementado de um erro de cálculo numérico;
- O valor de controle que corresponde ao *duty cycle* do PWM gerado no bloco FQG da programação Ladder do CLP, devido às características dos parâmetros deste bloco, tem que ser do tipo inteiro, o que acaba prejudicando na resolução da variável controlada.

Apesar destes fatores, como se pode observar na Figura 120, a variação em torno do valor de *Set Point* está na faixa de $\pm 0.2^{\circ}\text{C}$, que para os objetivos do trabalho é tolerável.



Figura 119: Controle de temperatura em malha fechada do módulo de aquecimento.
Fonte: [23].

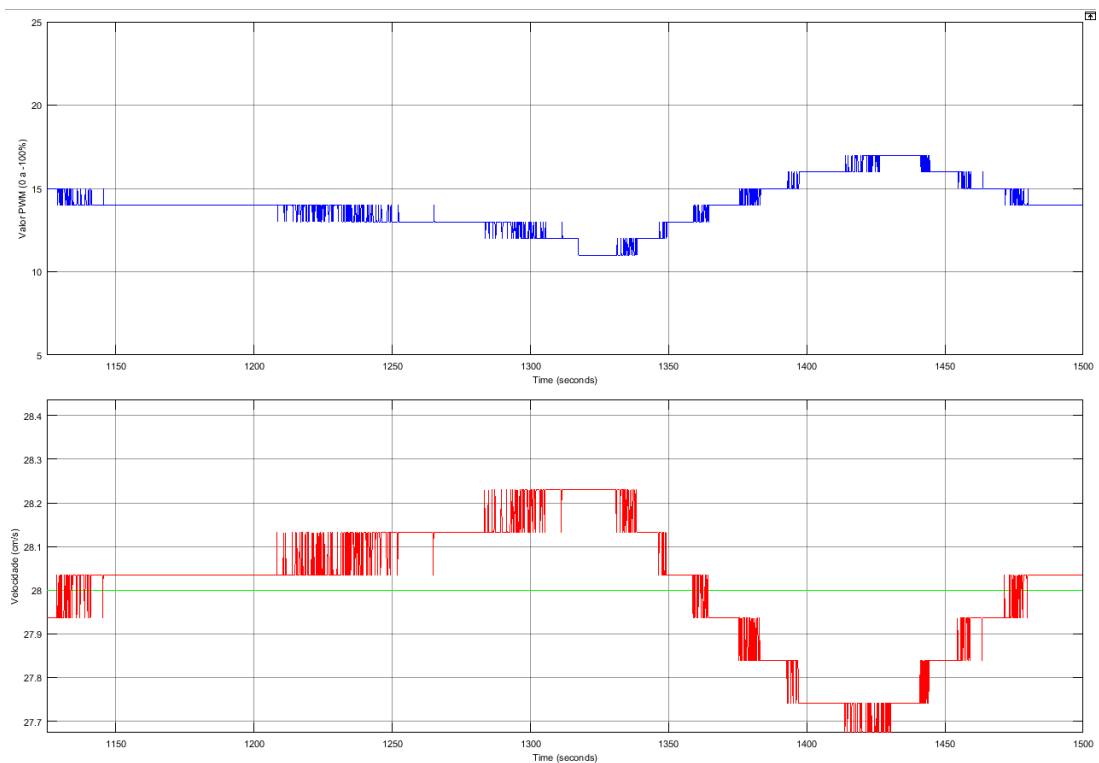


Figura 120: Detalhes do ponto onde o controle atinge o regime permanente.
Fonte: [23].

7 CONCLUSÃO

Realizar este trabalho possibilitou experimentar em um sistema real, os conhecimentos adquiridos não apenas na área de controle e automação como também em outras áreas de atuação da engenharia elétrica. Conhecimentos relacionados a outras áreas como, por exemplo, mecânica e termodinâmica, contribuíram para obtenção dos resultados desejados, fortalecendo a idéia de que a interdisciplinaridade ainda continua sendo um ponto chave para a solução de problemas da engenharia.

Os objetivos deste trabalho que eram a obtenção do conhecimento da construção instrumentação, identificação e controle da esteira transportadora foram alcançados com êxito, para realizar esses objetivos alguns problemas que não foram previstos precisaram ser resolvidos, como por exemplo, a identificação do módulo de aquecimento que possui uma dinâmica lenta se comparada à dinâmica do motor CC.

Para realizar a identificação de sistemas de forma satisfatória foi necessário aprender como realizar o tratamento dos dados e como realizar os experimentos para obter os dados mais significativos para identificação do sistema. Ainda na parte da identificação de sistemas foi necessário inspecionar os modelos matemáticos estimados pelo software, *System Identification Toolbox*, que obtiveram a melhor correlação com o modelo real. Esta investigação é importante, pois o software deve ser usado como uma ferramenta que auxilia na identificação de sistemas e não como o único método de identificação. Além disso, para se realizar uma boa identificação é preciso ter uma boa experiência na área.

Neste trabalho foram utilizados os controladores que foram estimados a partir dos modelos matemáticos identificados pelo software, pois como os controles funcionaram de forma satisfatória, não foi necessária a utilização de outras técnicas de identificação.

Apesar de terem sido apresentadas neste trabalho apenas uma sintonia para cada malha de controle, várias sintonias foram testadas e apenas apresentadas as que obtiveram o melhor resultado. Os testes dos controladores foram realizados na plataforma da esteira transportadora com parte dos controladores sendo implementada diretamente no CLP, com suas variáveis de processo e de controle monitoradas com uso da ferramenta Simulink e a outra parte implementada no próprio Simulink controlando

as variáveis de processo e de controle através do CLP. Em ambos os casos, o CLP atuou na interface de controle dos sistemas, estas interações das ferramentas do MATLAB com o CLP foram possíveis graças ao padrão OPC. O conhecimento deste padrão de comunicação foi mais um aprendizado que este trabalho proporcionou.

8 SUGESTÕES PARA TRABALHOS FUTUROS

Instrumentação:

- Substituir os sensores e transdutores construídos, por sensores e transdutores industriais;
- Realizar a conexão destes sensores e transdutores com o CLP utilizando redes industriais;
- Alimentar o motor CC com a tensão nominal recomendada;
- Utilizar uma caixa de redução para conexão mecânica do motor com a esteira, melhorando a relação de torque;
- Utilizar um *encoder* com uma resolução maior ou utilizar outro dispositivo para medição da velocidade;
- Substituir os módulos de resfriamento e aquecimento por módulos de aquecimento e resfriamento industriais.

Comunicação:

- Substituir a conexão serial do CLP com o PC pela conexão *ethernet*.

Modelagem:

- Obter um maior número de modelos matemáticos e compará-los;
- Utilizar outros softwares para sintonia dos controladores e técnicas de sintonização conhecidas para comparar seus resultados.

BIBLIOGRAFIA

- [1] ASTRÖM, K. J.; HÄGGLUND, **Advanced PID Control**, ISA, 2002.
- [2] ASTRÖM, K. J.; HÄGGLUND, **Advanced PID Control**, ISA, 2006.
- [3] BRYAN, L. A.; BRYAN, E., A. **Programmable Controllers - Theory and Implementation**, 2ª ed. Atlanta: Industrial *Text Company*, 1997.
- [4] FITZGERALD, A. E.; KINGSLEY JUNIOR, C.; UMANS, S. D. **Máquinas elétricas: com introdução à eletrônica de potência**. 6. ed. Porto Alegre: Bookman, 2006.
- [5] ISERMANN, R.; MUNCHHOF, M. **Identification of Dynamic Systems: An Introduction with Applications**. 2. ed. Germany: Springer, 2011.
- [6] IWANITZ, F.; LANGE, J. **OPC Fundamentals, Implementation and Application**. 2 rev..ed. [S.l.]: Huthig Verlag Heidelberg, 2002.
- [7] OVERSCHEE, P. V.; MOOR, B. **Subspace Identification for Linear System**. Dordrecht, The Netherlands: Kluwer Academic Publishers, 1996.
- [8] OGATA, K. **Engenharia de Controle Moderno**. 2ªed. Rio de Janeiro: LTC Editora S.A., 1993.
- [9] SODERSTROM, T.; STOICA, P. **System Identification**. 1. ed. London: Prentice-Hall, 1989.
- [10] S&E - Encoders Incrementais. Disponível em:
<<http://siteimages.radarindustrial.com.br/SiteImages/Client/29767/Product/93782/Document/pag17-encoders-956.pdf>>. Acesso em: 26/09/2015.
- [11] Encoders - HOME. Disponível em:
<<http://ljsautomacao.com/?p=1635>>. Acesso em 26/09/2015.
- [12] Nubotics WheelWatcher - WW-12. Disponível em:
<http://nubotics.com/products/ww12/ww12_manual.pdf>. Acesso em 26/09/2015.
- [13] MecaWeb - Sensores Ópticos. Disponível em:
<http://www.mecaweb.com.br/electronica/content/e_sensor_optico>. Acesso em 30/09/2015.
- [14] Sensores Ópticos - Mecatrônica Atua. Disponível em:
<<http://www.mecatronicaatual.com.br/educacao/1179-sensores-ticos>>. Acesso em 30/09/2015.
- [15] LM35 - Sensor de Temperatura - Webtronico. Disponível em:
<<http://www.webtronico.com/lm35-sensor-de-temperatura.html>>. Acesso em 06/10/2015.
- [16] LM35 - Texas Instruments. Disponível em:
<<http://www.ti.com/lit/ds/symlink/lm35.pdf>>. Acesso em 06/10/2015.
- [17] EAT00001- Conhecendo o kit de treinamento ZTK900. Disponível em:
<<http://www.hitecnologia.com.br/repositorio/atividades-de-treinamento>>, Acesso em 06/10/2015.
- [18] ENA0003300- HS1-OPC. Disponível em: <<http://www.hitecnologia.com.br/software/software-obsolotos/hs1-opc>>, Acesso em 28/10/2015.
- [19] UPS1095-5A - Portal da HI Tecnologia. Disponível em:
<<http://www.hitecnologia.com.br/produtos/hardware/fontes/fontes-chaveadas/ups1095-5a>>, Acesso em 18/10/2015.

[20] PET10800200-eZAP900-Controlador Lógico Programável eZAP900/901. Disponível em: <<http://www.hitecnologia.com.br/repositorio/manuais-pet/familia-zap900/PET10800200.pdf>>, Acesso em 20/10/2015.

[21] SPDSW- Portal da HI Tecnologia. Disponível em: <http://www.hitecnologia.com.br/software/control_e_de_processos/spdsw>, Acesso em 23/10/2015.

[22] EAT0100300 - Criando meu primeiro programa no SPDSW. Disponível em: <<http://www.hitecnologia.com.br/repositorio/atividades-de-treinamento>>, Acesso em 23/10/2015.

[23] MATHWORKS, **Simulink**, Disponível em: <<http://www.mathworks.com/products/simulink/>>, Acesso em 20/09/2015

[24] MATHWORKS, **System Identification Tool**, Disponível em: <<http://www.mathworks.com/products/sysid/>>, Acesso em 20/09/2015

[25] MATHWORKS, **PID Tuner**, Disponível em: <<http://www.mathworks.com/discovery/pid-tuning.html>>, Acesso em 20/09/2015